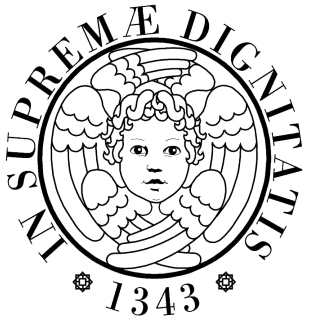
UNIVERSITÁ DEGLI STUDI DI PISA



FACOLTÁ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

MASTER DEGREE IN COMPUTER SCIENCE
Data Science & Technologies curriculum

ReviewerNet.org: Visualizing Citation and Authorship Relations for Finding Reviewers

Candidate
Mario Leonardo Salinas

Supervisors
Daniela Giorgi
Paolo Cignoni

Academic Year 2019 / 2020

Contents

CHAPTER 1

Introduction

This thesis focuses on the analysis and visualization of bibliographic data to provide decisional support in a complex task: the reviewer selection in the academic domain.

The number of digital academic documents, either newly published papers or documents resulting from digitization efforts, grows at a very fast pace: the Scopus digital repository counts more than 70 million documents and 16 million author profiles [1]; the Web of Science platform has more than 155 million records from over 34,000 journals [2]; Microsoft Academic collects about 210 million publications [3]. In 2018, over four thousand new records were added to DBLP [4], and bibliometric analysts estimated a doubling of global scientific output roughly every nine years [5]. Therefore, the volume, variety and velocity of scholarly documents generated satisfies the big data definition, so that we can now talk of *big scholarly data* [6].

Sensemaking in this huge reservoir of data calls for platforms adding an element of automation to standard procedures – such as literature search, expert finding, or collaborators discovery – to reduce the time and effort spent by scholars and researchers. In particular, there has been an increase in the number of visual approaches supporting the analysis of scholarly data. Visualization techniques were proposed to help stakeholders to get a general understanding of sets of documents, to navigate them, and to find patterns in publications and citations. Federico et al. [7] survey about 109 visual approaches for analysing scientific literature and patents published in-between 1991 and 2016. Most of the works focused on the the visualization of document collections and citation networks. A more ambitious goal for visualization platforms would be to enable users get enough understanding to make decisions.

In this thesis, we focus on the problem of reviewer finding by journal editors or International Program Committee (IPC) members, who are required to search for reviewers who know well a subject, yet are not conflicted with the authors of the paper under scrutiny. Finding good candidate reviewers requires to analyse topic coverage (possibly during time), stage of career, and past and ongoing collaborations. Every member of the community has its own approach to reviewer finding, which usually involves bibliographic research, and frequent visits to public repositories like DBLP [8] and researchers' home pages. In any case, one has to confront possibly large collections of data to make decisions, and a user may easily get lost after following a few links.

We propose ReviewerNet, a visualization platform which facilitates the selection of reviewers. The intuition behind ReviewerNet is that the authors of relevant papers are good candidate reviewers. ReviewerNet offers an interactive visualization of multiple, coordinated views about papers and researchers that help assessing the expertise and conflict of interest of candidate reviewers.

1.1 ReviewerNet in a nutshell

ReviewerNet supports the various actions that journal editors and IPC members perform while choosing reviewers, namely, searching the literature about the submission topic, looking for active experts in the field, and checking their conflict of interest. ReviewerNet does so by integrating an overview visualization of the literature with a visualization of the career of potential reviewers, their conflict of interests, and their nets of collaborators. This combined visualization helps to make sense of scholarly data, and rapidly get enough understanding to make a sensible decision, as shown in our user study (see Chapter 6).

ReviewerNet integrates the visualization of three main classes of data in a single window (see Figure 1.1):

- **Paper Network** (PN): a chronologically ordered visualization of the literature citation network related with the submission topic. The nodes represent papers, while arcs represent in- and out-citation relations between papers. The horizontal dimension represents time. By means of interactive graph expansion functionalities, the PN supports the rapid exploration of key papers in the literature with respect to the topic of the submitted paper. The authors of the key papers identified will define the set of the candidate reviewers. The PN is built by the users, starting from a small number of seed papers of their choice;
- **Researcher Timeline** (RT): a time-based visualization of the academic career of researchers, through horizontal lines and bars. The

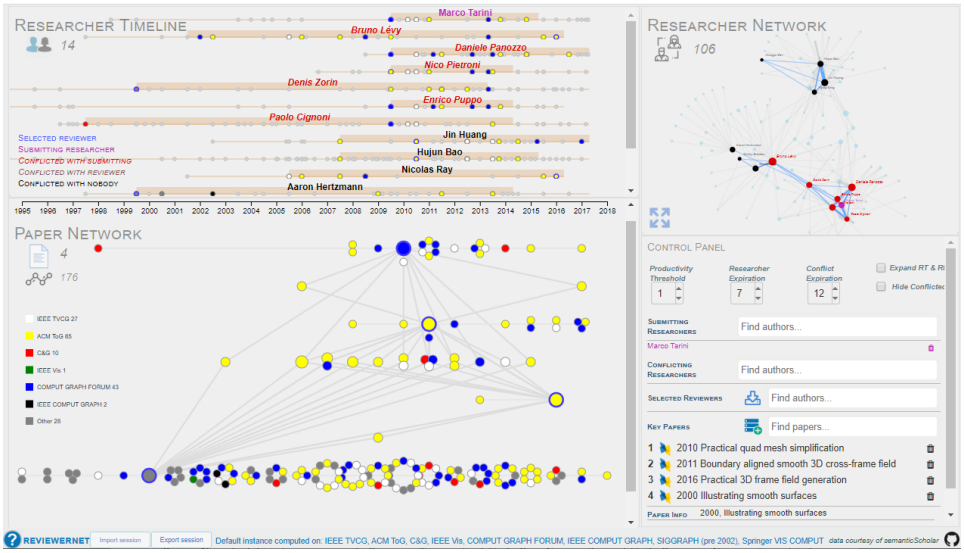


Figure 1.1: The main interface of ReviewerNet, divided into four main areas: the Researcher Timeline (top left); the Paper Network (bottom left); the Researcher Network (top right); the Control Panel (bottom right). The interaction with these areas allows the users to identify researchers working on the topic defined by a network of papers, to analyse the researchers' contributions through time, and to get aware of co-authorship relations and conflicts.

RT helps assessing the suitability of potential reviewers, showing thier topic coverage, productivity over years, and stage of career. Also, visual cues help the user to tell apart candidate reviewers from conflicting researchers. The RT is built automatically by ReviewerNet while the user builds the PN;

- **Researcher Network** (RN), a graph visualization of co-authorship relations: the nodes represent the authors in the PN and their collaborators in the dataset; the arcs connect authors who have publications in common. The aim of the RN is to visualize the research communities: indeed, the identification of network of collaborators helps looking for sets of independent, non-conflicting reviewers. As with the RT, the RN is built online by ReviewerNet.

The basic pipeline for finding reviewers with ReviewerNet involves: building the Paper Network starting from a small set of seed documents, whose titles are either manually typed, or imported from a list, and automatically parsed and matched in the dataset; evaluating possible choices of reviewers, by navigating the Reviewer Timeline and the Reviewer Network; and finally obtaining a justified list of chosen reviewers, along with possible substitutes

suggested by ReviewerNet in case of declined invitation. The user can navigate the different views and interact with the system through simple actions, to drive his/her investigation. Each view in ReviewerNet is linked to the other views, so that so that any action in a view is reflected in the others. Visual cues are used to improve the comprehension during interactive sessions: the position, colour, size, boundary, and style of visual elements visually represent important characteristics of the entities they stand for. To better explain how ReviewerNet works, Chapter 3 presents an example user scenario.

The user can navigate the different views and interact with the system through simple actions, to drive his/her investigation. Each view in ReviewerNet is linked to the other views, so that so that any action in a view is reflected in the others. Visual cues are used to improve the comprehension during interactive sessions: the colour, size, boundary, and style of visual elements visually represent important characteristics of the entities they stand for. Moreover, the coherence of visual cues across different views enforces their meaningfulness, and makes it easy for the user to switch between different views without losing focus.

ReviewerNet builds on a reference database including papers, authors and citations from selected sources (journal articles and conference papers) taken from the Semantic Scholar Research Corpus [9]. ReviewerNet can be built over any dataset, according to the domain of interest.

1.2 Summary of contribution

My contribution to this thesis is the development and validation of the visualization system, starting from the data gathering and processing. The challenges faced in this process were: dealing with both the size and quality of the corpus and having usable and meaningfull user interface and interactions. I downloaded and processed a Big-Scholarly-Data corpus with simple and reusable scripts, allowing the user to instantiate the platform over any subset of the original corpus with minimum effort (see Chapter 5.3).

The main contribution in deployment of the user interface is the resulting layout that allows the user to visualize and interact with all the classes of data visualized at once, thus providing the user with a high number of information about papers, researchers and relationships between them.

The platform has also been validated by a pool of senior researchers in Computer Graphics, who returned positive feedbacks about the ease-of-use and the distribution and quality of the candidate reviewers found using ReviewerNet, along with suggestion for further improvements (Chapter 6).

The tool is open, and the source code is available at `https://github.com/cnr-isti-vclab/ReviewerNet`, while the demonstration platform is available at `https://reviewernet.org`.

1.3 Outline

In Chapter 2, we review state-of-the-art applications and methods built on top of scholarly data.

Then, we show how ReviewerNet works and supports the reviewer selection process. We present an example user scenario, in the field of Computer Graphics, in which we succesfully extract a pool of reviewers who are expert on a certain topic, are at a certain career stage, who have a certain track of publishing records, who are not conflicting with neither the submitters nor other reviewers, and who are well-distributed in the scientific community (Chapter 3).

In the first part of Chapter 4, we define our goals and use-case along with the notation that is used in the document, then, in the second part of the Chapter, we describe how the user interface is composed and the possible interactions the user can exploit during the reviewer selection (Section 4.2).

Chapter 5 describes all our implementative choices: the reference corpus, languages and external libraries used to develop and deploy the system, and additional ReviewerNet features.

We evaluate the platform through a user study involving 15 real end-users from the Computer Graphics community and show how they were able to get acquainted with ReviewerNet even with very limited training, and how they rated very positively ReviewerNet functionalities (Chapter 6).

We then conclude with a discussion on the main contributions of this thesis and further imporovements to the system (Chapter 7).

1.4 Publications

A preliminary version of the platform developed for this thesis was presented at the 2019 edition of the Smart Tools and Applications in Graphics conference [10], where the paper earned the "*Best Paper*" nominee and was selected for publication in a *Special Issue* of the Elsevier Computers & Graphics journal. The newly submitted paper is currently under review.

Acknowledgments

The authors would like to thank the Allen Institute for Artificial Intelligence for providing the Semantic Scholar corpus of bibliographic data.

CHAPTER 2

Related work

Concerning the reviewer selection process, the literature mostly focused on the automatic reviewer *assignment* task, which is a different problem than ours. Indeed, the reviewer assignment problem requires finding the best assignment between a finite set of reviewers (e.g., the members of the Programme Committee of a conference) and a finite set of papers (the papers submitted to the conference); this is usually done using bi-partite graph matching and taking into account pertinence of the reviewers with the papers and fair distribution of loads; an overview of this problem is presented in [11].

In what follows, we briefly review the state-of-the art about the search, analysis and recommendation services offered by scholarly data platforms (Section 2.1), and the visualization of bibliometric networks (Section 2.2).

2.1 Scholarly data platforms

Many applications have been developed on top of the big scholarly data platforms to search for authors, documents, venues, and analyse statistics about for example distribution per research area, citations, and other bibliometric indices. Most academic search engines also provide research paper recommendations according to one's research interests.

Microsoft Academic provides a semantic search engine that employs natural language processing and semantic inference to retrieve the documents of interest. It also provides related information about the most relevant authors, institutions, and research areas [12]. Scopus enables one to search for authors or documents, track citations over time for authors or documents, view statistics about an author's publishing output, and compare journals

according to different bibliometric indices [1].

These and similar applications offer basic functionalities and static visualizations which researchers do use while looking for reviewers. Though, none of them offers an integrated service to support the higher level tasks of fine-tuned reviewer selection, where both expertise and conflicts of interest have to be taken into account.

2.2 Visualization of bibliometric networks

The visualization of bibliometric networks is an active area of research [7,13]. Bibliometric networks include citation, co-citation, co-authorship, bibliographic coupling and keyword co-occurrence networks.

Concerning visualization of citations, most part of the literature focused on co-citation and bibliographic coupling networks, rather than on direct citations. One of the first visualization of citation networks is Garfield's historiography [14], a node-link diagram where citation links are directed backwards in time. Garfield and colleagues underline how citation networks enable one to analyse the history and development of research fields. CiteNetExplorer [15] is a software tool to visualize citation networks which builds on Garfield and colleagues' work: it improves the graph layout optimization to handle a larger number of papers, and offers network drill-down and expansion functionalities. PaperVis [16] is an exploration tool for literature review, which adopts modified Radial Space Filling and Bullseye View techniques to arrange papers as a node-link graph while saving the screen space, and categorizes papers into semantically meaningful hierarchies.

In [17], is described a visual analytics system for exploring and understanding document collections, based on computational text analysis; it supports document summarization, similarity, clustering and sentiment analysis, and offers recommendations on related entities for further examination. Rexplore [18] is a web-based system for search and faceted browsing of publication. Rexplore also includes a graph connecting similar authors, where similarity depends on research topics as extracted from document text. At any rate, using keywords as proxies for research topics can be noisy. Therefore, in ReviewerNet we only rely on co-authorship relations.

2.3 Beyond the state-of-the-art

Many of the approaches for bibliographic network visualization make limited use of user interaction, and often use a loose coupling of views [7]. With ReviewerNet, we propose an integrated environment which facilitates a high-level task (reviewer discovery and selection) by means of coordinated, interactive views. Also, only a few works include an in-depth evaluation of the techniques proposed through user studies. We report a user study in-

volving real end-users, namely 15 experts in Computer Graphics, who tested ReviewerNet and filled in an anonymous questionnaire (Chapter 6).

One of the main advantages of ReviewerNet is that it only relies on citations, to analyse the literature, and on co-authorship relations, to analyse conflicts. Citations are an essential part of research: they represent a credible source of information about topic similarity and intellectual influence. Moreover, since citations have author-chosen reliability, they are a very robust cue to relatedness. Similar reasonings hold for co-authorship relations. Therefore, an important contribution is the demonstration that a well-combined visualization based on citation and co-authorship relations only can support the reviewer search process, without the need for more complicated content analysis techniques.

CHAPTER 3

The platform

To better explain how ReviewerNet works and supports the reviewer selection process, we firts present an example user scenario, then for the technical and implementative details we refer respectively to Chapters 4 and 5. We introduce Robert, a fictitious academic researcher. Robert is in the IPC of a conference in the field of Computer Graphics; he is the primary reviewer for a paper, and he is in charge of finding three additional reviewers, plus alternative reviewers in case of decline. Below we describe Robert's interaction with ReviewerNet

Starting ReviewerNet

Robert is in charge of finding reviewers for a paper about polycube maps, authored by Marco Tarini and Daniele Panozzo. In the Control Panel area, he inputs their names in the *Submitting Researchers* field, also with the help of a drop-down menu. The authors are now shown in the Researcher Timeline and the Reviewer Network, marked as purple.

Building the Paper Network

The first step is to build the Paper Network (Figure 3.1), that is, a set of key papers which are relevant to the submission topic. Later on, Robert will chose his reviewers among the authors of those key papers. Robert thinks of a first pair of documents about polycube maps, which serve as seeds for building the network (*PolyCube-Maps*, 2004; *Polycube simplification for coarse layouts of surfaces and volumes*, 2016. He inputs their titles in the *Key papers* field. His knowledge of the domain helps him in this initial step, though he can also take advantage of title- and author- based suggestions,

Figure 3.1: When the user inputs the seed papers (bottom right), ReviewerNet starts building the Paper Network (bottom left), the Researcher Timeline (top left), and the Researcher Network (top right). The dots representing papers in the Paper Network and the Researcher Timeline are coloured according either to their citation count – from green (few citations) to yellow (many citations) – or to the venue they were published in. Grey dots in the Researcher Timeline are papers in the reference database, but not included in the current Paper Network. Selected papers are circled in blue.

which are shown in a drop-down menu, listed by publication year. Also, the list of references in the submitted paper or in previously identified seed papers can provide a convenient strategy to initialize the network. Robert copies (a subset of) citations in the *Import from bibliography* window, and the system returns the list of matching papers in the dataset (Figure 3.2). Robert can now approve or discard the suggestions. He ends up adding three more papers (*A divide-and-conquer approach for automatic polycube maps construction*, 2009; *L_1-based construction of polycube maps from complex shapes*, 2014; *All-hex mesh generation via volumetric polycube deformation*, 2011). The five papers are now included in the Paper Network, along with their in- and out-citations.

While Robert builds his Paper Network, ReviewerNet automatically adds the authors of selected papers in the Researcher Timeline and the Researcher Network, as candidate reviewers.

Robert can now expand the Paper Network, to discover additional documents and therefore additional candidate reviewers. Papers which cite (are cited) by a single key paper are displayed along a common line, and clustered by year. Papers which cite (are cited) by more than one key paper are good candidates for being added, therefore, they are positioned in-between key paper lines, so that they can be easily identified as interesting nodes.

With a double click, Robert selects papers he deems relevant to polycube maps. The Paper Network then updates with the in- and out-citations of the selected papers, so that Robert can further explore the literature. Robert navigates the network, and decides to reduce its size by deselecting a paper he realizes he is no longer interested in, because its citations suggest it addresses a different topic than the submission. Robert continues until he feels the selected papers and their citations offer a good coverage of the literature about the topic at hand. Robert checks the paper details, including the link to the respective DBLP page, shown in the bottom right corner of the interface. A quick keyword search with *polycube maps* in the *Key papers* field let him notice that there is an important paper he was missing (*Interactive applications for sketch-based editable polycube maps, 2016*); the paper can be easily told apart from papers already in the network, thanks to visual cues in the drop-down menu. Finally, the selection of 6 papers produces a list of 23 candidate reviewers.

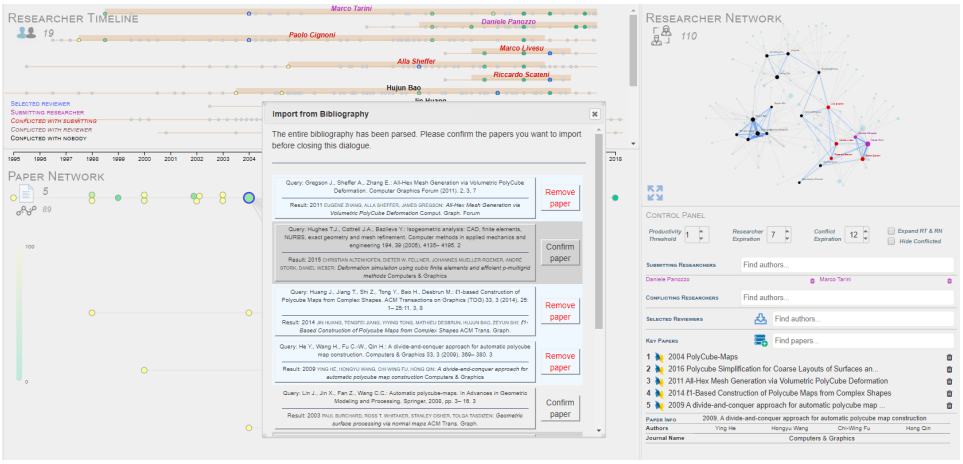


Figure 3.2: The seed papers can be imported from the bibliography of a paper. The user can copy and paste the references, which are parsed by the system; matching titles in the dataset are returned and submitted to the user for either approval or rejection.

Exploring the Researcher Timeline and the Researcher Network

Robert now explores the Researcher Timeline to assess the suitability of candidate reviewers. In the Researcher Timeline, researchers are represented as horizontal lines, spanning their academic career. Robert checks the expertise of candidate reviewers by looking at their stage of career, and production over years. Since each view is linked to the other views, Robert checks topic coverage by looking at who published what, by hovering the mouse over pa-

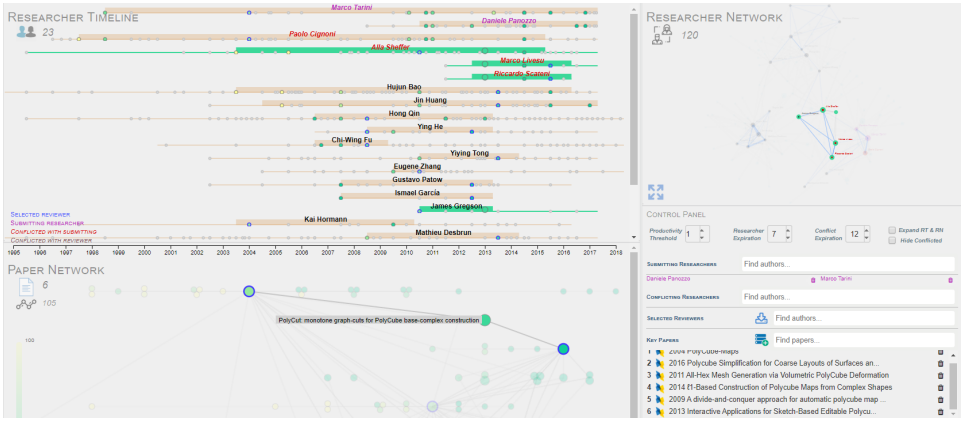


Figure 3.3: In the Researcher Timeline, researchers are represented as horizontal lines, spanning their academic career; the bars over the lines indicate the years in which the authors published about the submission topic, namely, the years for which they have papers in the Paper Network. When hovering over an entity representing a paper, the authors of that paper are highlighted in the other views.

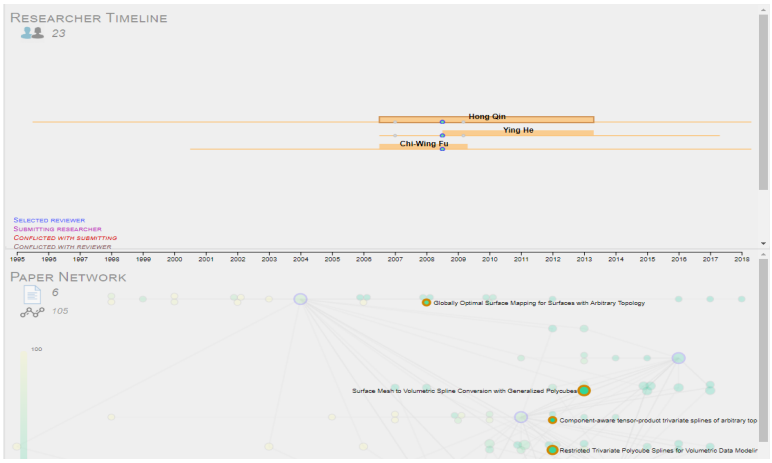


Figure 3.4: Focusing on a researcher by clicking on her/his name in the Researcher Timeline allows one to highlight her/his co-authors and production in the Paper Network.

pers to highlight their authors in all the views (Figure 3.3). With a mouse click on a researcher, ReviewerNet highlights both his/her co-authors and papers (Figure 3.4). While looking for candidate reviewers, Robert can always checks conflicts of interests, thanks to colours and font style (Figure 3.5). The visualization also helps Robert analysing the network of collaborators of candidate reviewers. This is fundamental to find sets of independent, well distributed reviewers. Robert can navigate the Researcher Network, a

graph visualization of co-authorship relations among the candidate reviewers and their collaborators in the dataset. He switches to full-screen mode, then pans and zooms and uses the different handlers available to discover conflicts-of-interest of different degrees, as well as to identify network cliques that represent communities of collaborators.

He finds out that there are three distinct groups of collaborators dealing with the topic at hand (Figure 3.6).

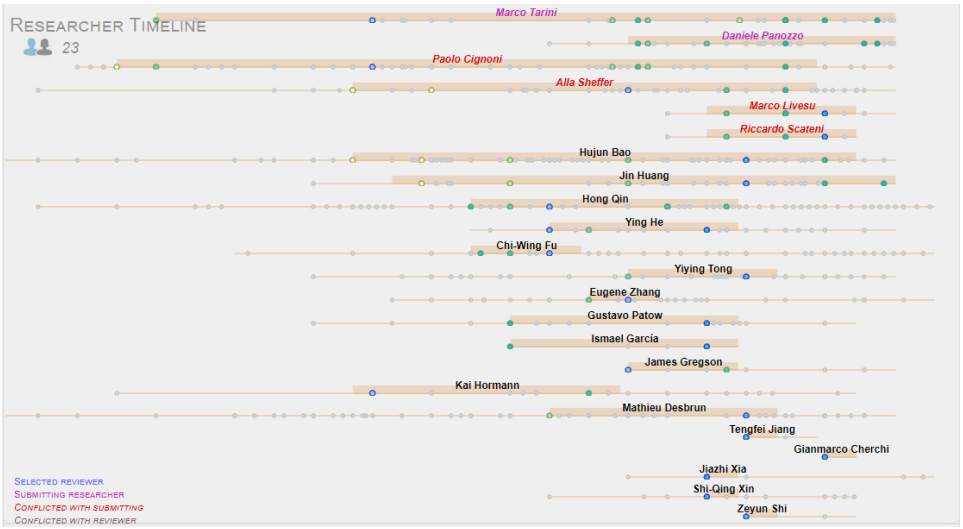


Figure 3.5: The Researcher Timeline lists the names of potential reviewers. The name colouring emphasizes the distinction among roles: submitting authors (marked as purple), their co-authors (red), selected reviewers (blue), their co-authors (brown), and non-conflicting, candidate reviewers (black). The font style of names further helps to tell apart conflicting researchers (italic) from non-conflicting candidate reviewers (normal). The candidate reviewers are ordered vertically according to their relevance (cf. Section 4.1 for details).

Selecting reviewers Once Robert identifies one or more candidate reviewers, he inputs their names in the *Selected Reviewers* field (also with the help of the drop-down menu). He decides to chose Gustavo Patrow, whose expertise fits with his requirements. The colouring of the selected reviewer switches to blue both in the Researcher Timeline and the Researcher Network, and the colouring of his co-authors switches to grey, to identify them as conflicting potential reviewers, and tell them apart from the remaining available candidates. Then, Robert selects Hujun Bao, a senior researcher, and Gianmarco Cherchi, a younger researcher who belongs to a different community than the previous two, and has been working very recently on the subject at hand. The icon beside the reviewers name links to their

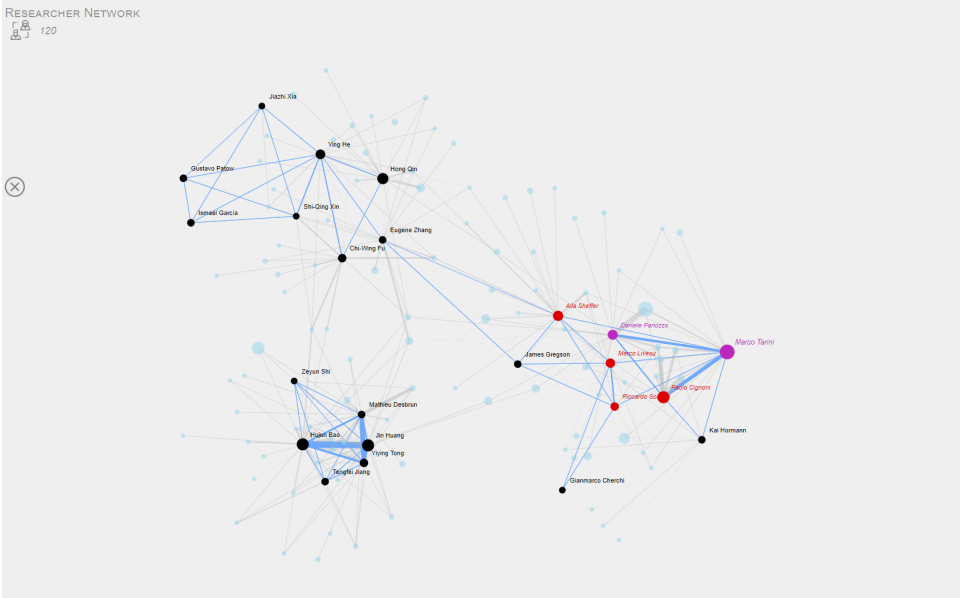


Figure 3.6: In the Researcher Network, arcs connect authors who have publications in common. Arcs are blue when the co-authored papers include a selected paper. The colouring of nodes and names emphasizes roles as in the Researcher Timeline, while the relevance of authors is rendered through the dimension of nodes and fonts. The thickness of arcs renders the number of co-authored papers. The visualization can be fine-tuned by adjusting a set of parameters defining the criteria on productivity to be included in the visualization, or the criteria that define conflicts (cf. the end of this Section for their definition).

respective DBLP pages, so that Robert can further check about conflicts of interest possibly deriving from the co-authorship of papers published on venues not included in the dataset.

Robert downloads his list of three reviewers with a click on the download button. The list reports reviewers' names and bibliographic references to their papers (Figure 3.7). After contacting the reviewers, Robert finds that one of them declines his invitation. Fortunately, for each reviewer selected by Robert, ReviewerNet has automatically added a list of potential alternative reviewers, in case of a negative answer by the original reviewer. Alternative reviewers are chosen from the candidate ones, so that they only conflict with the declining reviewer. Robert evaluates possible substitutes, again taking advantage of ReviewerNet functionalities, and finds his best replacement.

Discussion This abbreviated scenario shows how ReviewerNet can support investigating the literature, learning who are the experts in a field, and exploring relationships among them. The description above necessarily sim-

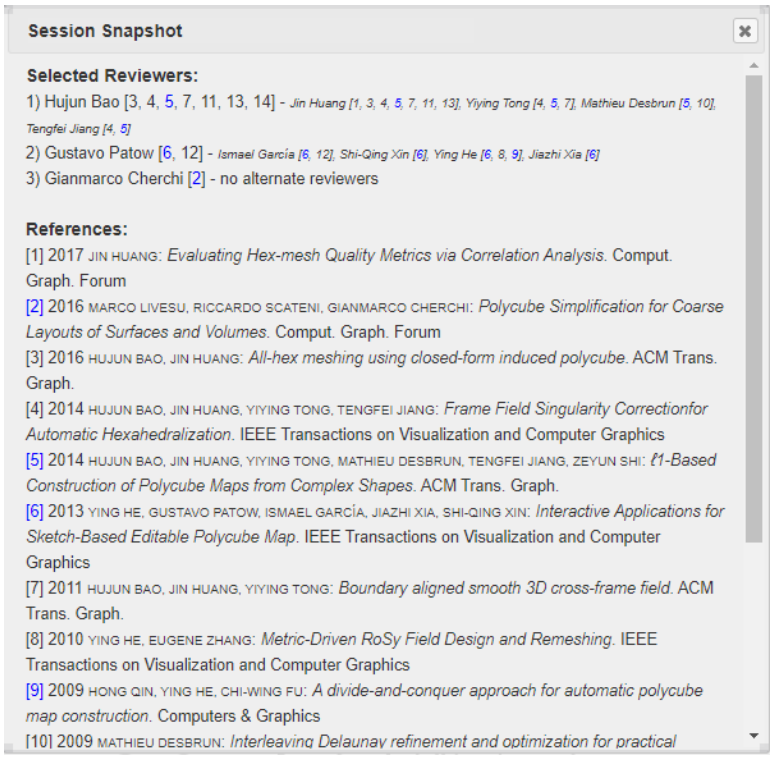


Figure 3.7: The list of selected reviewers, together with substitutes in case of decline, and a bibliography. A substitute reviewer is a researcher who has authored a similar set of publications and has the same conflicts as the original reviewer. The bibliography motivates the reviewer selection, since it lists, for each selected reviewer, the papers he/she has authored.

plified a typical interaction process: Robert could of course switch back and forth between different tasks; as the coherence of visual cues across different views enforces their meaningfulness, it is easy for him to switch between different views without losing focus. Robert could have also refined the Paper Network after having examined the list of candidate reviewers. He could have adjusted the size of the list by fine tuning the optional parameters. The process is iterative in nature, and the desiderata may evolve as the search proceeds. Thanks to the user-friendly interface which leaves the user control over the process, ReviewerNet enables the user to narrow down as well as widen the scope of analysis. In turn, the combined visualization of different aspects of the problem at hand well supports the decision making process.

CHAPTER 4

Technical details

The following section details the notation used in the rest of the paper, and the formal definition of paper and researcher attributes (Section 4.1). Moreover we describe the composition of the user interface and all the possibile interactions and visual cues (Section 4.2).

4.1 Data and notation

The aim of ReviewerNet is to facilitate the reviewer selection process in the academic domain. While designing ReviewerNet, we took into account the characteristics of the *users*, the users' *tasks*, and the *data* the users are working with [19].

The *users* of ReviewerNet are researchers, and in particular those playing the role of journal editors, associate editors, and members of IPCs of big conferences, in any field of research. Their *task* is searching for reviewers for a submitted paper: this involves searching the literature for key papers and authors in the field; evaluating the candidates' research interests and their evolution over time; and assessing the candidates' conflict of interest with respect to the submitting authors and other reviewers. ReviewerNet supports all these subtasks, by visualizing the literature related with a topic, the career of relevant researchers in the field, and the relationships among researchers.

The data pertain to three types of entities: *papers*, *researchers*, and *citations*. The data attributes are both quantitative and qualitative, and the time dimension is central.

In ReviewerNet, the attributes of a paper which are visualized are its *citation count* – the number of papers citing it – as well as standard *bib-*

liographic attributes – title, authors, publication year, venue. Papers are related through *direct citations*.

Researchers have two attributes in ReviewerNet: relevance and conflict of interest. We define a researcher's *relevance* as a reviewer according to the authorship of relevant papers. The concept of relevance can be tuned according to the user needs (e.g., looking for highly-specialized reviewers, as opposed to generalists). The second attribute of researchers is their *conflict of interest*, with either the submitting authors or other reviewers. We model the conflict of interest after *co-authorship* relations: two researchers have a conflict of interest if they have papers in common. We let the degree of conflict, and hence the availability as a reviewer, be modulated according to the number of papers in common, and the years passed since the last co-authored paper, again according to the user intent.

Let $\mathcal{P}$ denote the set of papers in a reference dataset, and let $\mathcal{P}_V \subseteq \mathcal{P}$ be the set of papers relevant to a submission. $\mathcal{P}_V$ is built by the users starting from a small number of seed papers of their choice (cf. Chapter 3).

A paper $p \in \mathcal{P}_V$ is marked as *selected*, if it is considered as a key paper by the user; we denote by $\mathcal{P}_S$ the set of selected papers, with $\mathcal{P}_S \subseteq \mathcal{P}_V \subseteq \mathcal{P}$.

If $\mathcal{C}(p)$ is the set of papers citing p, the *citation count* $c(p)$ is its cardinality: $c(p) = |\mathcal{C}(p)|$.

Let $\mathcal{A}(p)$ be the set of authors of a given paper p, and $\mathcal{R}$ the set of authors of papers in $\mathcal{P}$. Then, the set $\mathcal{R}_C \subseteq \mathcal{R}$ of *candidate reviewers* is given by the set of researchers who authored a selected paper:

$$\mathcal{R}_C = \{r \in \mathcal{R} \; s.t. \; \exists \; p \in \mathcal{P}_S : r \in \mathcal{A}(p)\}$$

For a candidate reviewer r, let $\mathcal{P}_S|_r$ be the set of papers in $\mathcal{P}_S$ authored by r. Then, the *relevance score* $s(r)$ of the candidate reviewer r is defined as a weighted sum of the number of selected and non-selected papers in $\mathcal{P}_V$ authored by r:

$$s(r) = \alpha|\mathcal{P}_S|_r| + \beta|\{\mathcal{P}_V - \mathcal{P}_S\}|_r|$$

with α and β real-valued coefficients summing up to one. We set $\alpha = 0.7$ and $\beta = 0.3$ as default parameters. The set of candidate reviewers will be visualized in the Researcher Timeline in order of their relevance; relevance will also define the dimension of nodes in the Researcher Network.

Finally, $\mathcal{CA}(r)$ denotes the set of co-authors of a researcher r, or, in other words, the set of researchers who have a conflict with him/her.

4.2 User interface

The entry page to the platform enables users to either select a pre-defined academic domain (Computer Graphics in our demonstration), or to load a customized dataset in any field of interest. The customized dataset can be

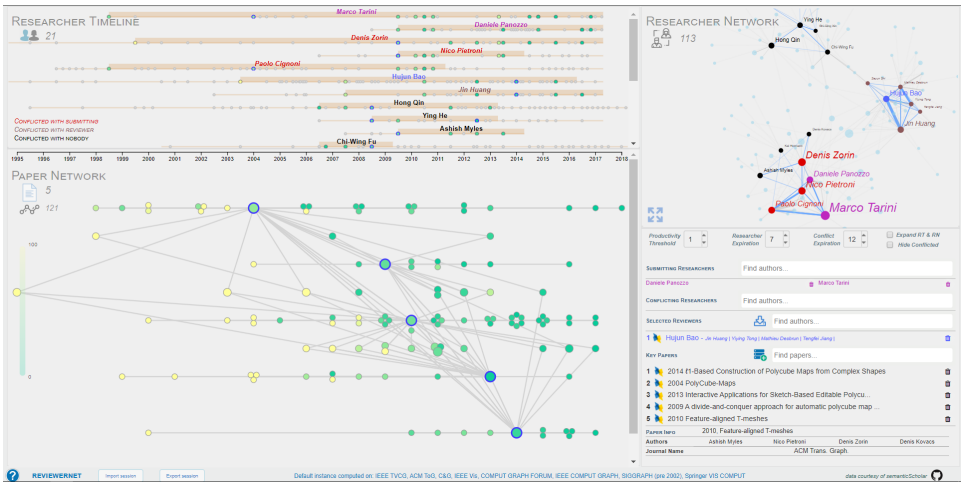


Figure 4.1: ReviewerNet user-interface.

created through a script that downloads and filters the Semantic Scholar corpus, according to a given list of venues.

Then, the visual composition of the four regions in the interface (Figure 4.1) helps the user to gain different perspectives on the problem at hand, within a single visualization. Each region can be resized in height; the Researcher Network can be made full-screen, as in Figure 3.6, for better visualization and interaction.

The nodes in the Paper Network (PN), at the bottom-left hand side of the screen, represent papers in $\mathcal{P}_V$, while the arcs represent in- and out-citation relations between them. Apart from pathological cases, the network is by definition a DAG (Directed Acyclic Graph). The visualization of the DAG is constrained along the horizontal direction, since papers are ordered horizontally according to their publication year. While in the previous platform version [10] a force-directed graph drawing algorithm determined the layout in the vertical direction [20], in the new version we preferred a visualization which puts focus on nodes with degree higher than one, that is, on papers which cite/are cited by more than one selected paper. Those papers are are likely to represent relevant papers in the field, and therefore good candidates for being selected as key papers and expanded in the network. Therefore, each key paper is assigned a rectangular region of predefined vertical height; all its citing/cited papers with node degree equal to one are arranged inside the rectangle, whereas nodes with higher degree are positioned in-between the rectangular regions. Their positions minimize the sum of distances from the citing/cited selected papers. This visualization of the network enables one to easily detect higher-degree nodes and network cliques, and tell apart relevant papers, which are good candidates for node selection and expansion.

Each line in the Researcher Timeline (RT), at the upper-left side of the

screen, represents a candidate reviewer r in $\mathcal{R}_C$, that is, the author of a selected paper in $\mathcal{P}_S$. The dots over the line represent the set $\mathcal{P}|_r$ of papers authored by r in the reference database $\mathcal{P}$.

The nodes in the Researcher Network (RN), at the upper-right hand side of the screen, are the researchers in $\mathcal{R}_V$ along with their collaborators in $\mathcal{R}$. The arcs connect authors who have publications in common: for each node representing a researcher r, the node degree is the cardinality $|\mathcal{CA}(r)|$. A force-directed graph drawing algorithm determines the graph layout, so that authors who have a large number of publications in common tend to be close to each other. Both the Researcher Timeline and the Researcher Network are built automatically by ReviewerNet while the user builds the Paper Network.

The Control Panel (CP), at the bottom-right hand side of the screen, allows the user to input and manage the names of submitting authors, the names of selected reviewers, and the titles of key papers. The CP area also displays information about papers, upon request. The DBLP icon beside reviewers' names and paper titles links to their respective DBLP page. Moreover, the CP includes parameters boxes and check-boxes to fine-tune the visualization:

Size of data visualized: To limit the number of candidate reviewers visualized in the RT and the RN, the user can set two thresholds a researcher has to meet to be considered as a candidate reviewer:

- *Productivity threshold*: the minimum number of authored selected papers in $\mathcal{P}_S$ (i.e., $|\mathcal{P}_S|_r|$ has to be greater than the threshold, for a researcher r to be included in the set $\mathcal{R}_C$ of candidate reviewers);
- *Researcher expiration*: the maximum number of years since the last authored paper in the reference dataset $\mathcal{P}$ (i.e., the number of years has to be lower than the threshold for a researcher to be considered active and included in $\mathcal{R}_C$).

The user can also remove conflicting authors and their co-authors from the visualization, by ticking the *Hide Conflicted* checkbox. To augment instead the number of potential reviewers visualized, the user can tick the *Expand RT & RN* checkbox: the visualization will include all the researchers in $\mathcal{R}_V$ (all the authors of relevant papers) instead of the researchers in $\mathcal{R}_C$ only (the authors of selected papers only). Note that visualizing a large number of researchers can slow down the interface.

Conflict-of-interest: Finally, to modulate the conflict of interest, the user can set a threshold for two researchers to be considered as co-authors, namely

- *Conflict expiration*: the maximum number of years since the last co-authored paper in $\mathcal{P}$.

A larger threshold will increase the number of candidates marked as conflicted. Conversely, a smaller threshold will increase the number of available reviewers.

4.2.1 Visual consistency

Visual cues include the position, colour, size, boundary, and style of visual elements representing papers, researchers and their relations across the different views.

Visual cues for papers For a paper $p \in \mathcal{P}_V$, the color may correspond either to the citation count $c(p)$ – from yellow (few citations) to green (many citations) – or to the venue where it is published – according to the eight-value color scale in [21] for the most relevant venues, plus grey for the others; the relevance of a venue is given by its number of incoming citations in the dataset. The colormap applies to both nodes in the PN and dots in the RT. Dots corresponding to papers in $\mathcal{P} - \mathcal{P}_V$ (papers in the reference database, but not included the PN) are marked as grey. Selected papers in $\mathcal{P}_S$ are circled in blue, both in the PN and the RT. Arcs are blue in the RN when the co-authored papers include a selected paper.

Visual cues for researchers For researchers in the RT, the name colouring emphasizes the distinction between roles: submitting authors (marked as purple), their co-authors (red), selected reviewers (blue), their co-authors (brown), and non-conflicting, candidate reviewers (black). The nodes in the RN corresponding to researchers in the RT follow the same rule, whereas nodes representing their co-authors in $\mathcal{R}$ are light blue and have no name labels attached. For researchers in the RT, the font style of names further helps to tell apart conflicting researchers (italic) from non-conflicting candidate reviewers (normal). The same colour/font rules apply to the names suggested in the selected reviewers' drop-down menu in the CP. The candidate reviewers in the RT are ordered vertically according to their relevance score (Figure 3.5). The same score is rendered in the RN through the dimension of nodes.

4.2.2 Actions

Each view (PN, RT, RN, CP) is linked to the other views, so that any action in a view is reflected in the others. The shortcut key $Ctrl + Z$ enables the user to undo an action.

Actions on Papers The user initialises the Paper Network with small set of seed papers. The user can either type the titles of the seed papers in the *Key papers* field, with the help of title-based suggestions, or press the *Import*

from bibliography button and paste a list of references. The references are parsed to identify matching titles in the dataset through a fuzzy search; the candidate titles with higher scores are shown to the user, who can select or discard them. Then, the seed papers are visualized in the PN, along with their in- and out-citations. The user can now expand the network, to discover additional documents. With a double click, he selects interesting nodes, i.e., papers he/she deems relevant to the submission topic. The PN then updates with the in- and out-citations of the selected papers.

Papers can be deselected either with a double click or through the trash bin icon in the Control Panel.

When the users focuses on a paper in one of the views by mouse hovering, the same paper is highlighted in the other views. For example, when hovering the mouse over a node in the PN, the corresponding dot in the Researcher Timeline is highlighted, and vice-versa. Also, the paper details (title, publication year, venue) are shown in the Control Panel on a mouse click. Likewise, by hovering over or clicking on the title in the CP, the corresponding node and dot are highlighted in the PN and the RT. When hovering the mouse over an entity representing a paper (a node in the PN, a dot in the RT bars, the title in the CP), the paper authors are highlighted in the RT and RN, if present (Figure 3.3). A mouse click on the focused paper lets the user navigate the visualization with highlighted items. A single click restores the previous visualization. The icon beside paper titles in the Control Panel links to DBLP pages.

Actions on Researchers In a similar fashion to papers, when the user focuses on a researcher in one of the views by mouse hovering, the same researcher is highlighted in the other views. When hovering the mouse over a node in the Researcher Network, the name of the corresponding researcher appears on the upper-right corner. When hovering the mouse over an entity representing a researcher (a bar in the RT, a dot in the RN, the name in the CP), the papers authored by the researcher are highlighted in the Paper Network view. In Figure 3.4, a mouse click on a researcher puts the focus on him/her, his/her production and his/her personal net of collaborators.

The user can navigate a visualization with selected items and additional functionalities. Only the set of co-authors is visualized in the Researcher Timeline and the Researcher Network. While hovering on one of the co-authors, the common publications are shown in the PN, and the arc representing the co-authorship relation is visualized in the RN. Another mouse click will get the user back the previous visualization. When hovering the mouse over an arc in the RT, like in Figure 4.2, a pop-up on the upper-right corner shows the pair of co-authors names, the number of common papers in the dataset $\mathcal{P}$, and the number of common relevant papers in $\mathcal{P}_V$. In turn, for blue arcs, the common papers are highlighted in the PN.

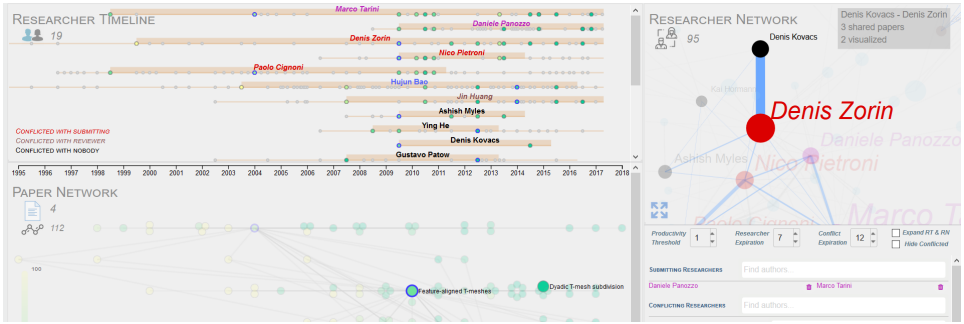


Figure 4.2: Hovering over a segment joining two researchers in the Researcher Network shows details about their co-authored papers and highlights them in the Paper Network.

The icon beside the researcher name in any of the fields in the Control Panel links to the DBLP page of that researcher. A researcher can be removed from the list of selected reviewers either with a double click or through the trash bin icon in the Control Panel. The user can exchange a reviewer with one of his/her substitutes by clicking on the name of the substitute. The export button enables the user to download the list of reviewers and their potential substitutes. Work sessions can be saved for later re-use and re-assessment.

CHAPTER 5

Implementation deatils

Reviewernet is a fully client-side application. It builds on a bibliographic dataset extracted from a reference corpus containing more than 180 million records.

In this chapter, we describe our implementation choices: the data and the preprocessing (Section 5.1), languages, frameworks and external libraries used to deploy the user interface (Section 5.2); we conclude with an analysis of the code that implements additional features of our interactive visualization system (Section 5.3).

5.1 The data

To construct the reference dataset, we collected papers, authors and citations from eight selected sources in the field of Computer Graphics, taken from the Semantic Scholar Research Corpus [9].

The original corpus currently contains more than 180 millions research papers published in all fields, provided as JSON objects, one per line. Archives are partitioned in batches and shared as a collection of gzipped files. In Figure 5.1 there is the full list of the attributes of a generic record that represents a publication.

To demonstrate the application in the field of Computer Graphics, We filtered the corpus extracting only publications from journals and conference proceedings listed in Table 5.1, spanning the years in-between 1995 and 2019.

The final reference dataset contains 22.887 papers, 145.900 citations, and 29.549 authors.

id string
S2 generated research paper ID.

title string
Research paper title.

paperAbstract string
Extracted abstract of the paper.

entities list
Extracted list of relevant entities or topics.

s2Url string
URL to S2 research paper details page.

s2PdfUrl string
URL to PDF on S2 if available.

pdfUrls list
URLs related to this PDF scraped from the web.

authors list
List of authors with an S2 generated author ID and name.

inCitations list
List of S2 paper IDs which cited this paper.

outCitations list
List of S2 paper IDs which this paper cited.

year int
Year this paper was published as integer.

venue string
Extracted publication venue for this paper.

journalName string
Name of the journal that published this paper.

journalVolume string
The volume of the journal where this paper was published.

journalPages string
The pages of the journal where this paper was published.

sources list
Identifies papers sourced from DBLP or Medline.

doi string
Digital Object Identifier registered at doi.org.

doiUrl string
DOI link for registered objects.

pmid string
Unique identifier used by PubMed.

fieldsOfStudy list
Zero or more fields of study this paper addresses.

Figure 5.1: Definition of attributes of the Semantic Scholar research corpus, from http://api.semanticscholar.org/corpus/.

ACM Transactions on Graphics	2594
Computer Graphics and Applications	1697
Computer Graphics Forum	3521
Computers & Graphics	2092
IEEE Transactions on Visualization and Computer Graphics	3638
Visual Computer	2107
Proceedings of IEEE Conference Visualization (pre 2006)	474
Proceedings of ACM SIGGRAPH (pre 2003)	6718

Table 5.1: The selected sources from the Semantic Scholar Research Corpus used in our demonstration scenario. The final reference dataset contains 22.887 papers, 145.900 citations, and 29.549 authors.

Pre-processing

The total size of the corpus is about 180 GB and the data quality is fairly good, but there's ambiguity in how journals and venues are referenced (acronyms, multiple abbreviations, ecc...) and there might be consistency

problems in citations as extracted from JSON files. Hence the pre-processing phase is crucial to lower data complexity while maintaining coherence and topic coverage to properly support the reviewer selection process.

After non-paper (such as acknowledgements to reviewers, prefaces, etc.) and useless attributes deletion, each single JSON has been parsed and filtered separately with a python script. In this filtering step, we use a string-matching algorithm to correctly assign papers to venues and journals (see Sections 5.2 and 5.3). Then we have the final consolidation step in which we:

- check citations consistency, to ensure we only extract citations published in one of the venues in our reference list (Table 5.1);
- precompute co-authorship relathionships among researchers;
- merge the filtered data obtaining 3 JSON files (authors, papers, journals) representing the Computer Graphic reference corpus.

The *papers' file* is a JSON representation of the PN. It contains two lists: *nodes* and *links*, representing respectively papers and citation relathionships.

id string	paper ID.
value string	Paper title.
year int	Publication year of this paper.
authsId list	List of S2 generated author IDs.
jn string	Name of the journal that published this paper.
j_id string	ID of the journal.
venue string	Extracted publication venue.
v_id string	Venue ID.
color int	Number of in-citations.
nOc int	Number of out-citations.

Table 5.2: Definition of attributes of the generic node in the PN.

The node entity is essentially the same as in the reference corpus with less attributes, as described in Table 5.2. The link object is simpler: for each citation in the dataset we have the source and target ID.

The *authors' file* is a list of JSON, one for each researcher. Table 5.3 describes the author entity in ReviewerNet.

The `coAuthList` field is a fondamental one: it allows one to rapidly discover conflicts among researchers, easing the real-time computation load.

The *journals' file*, eventually, contains information about the venue names consolidation and useful statistics. For each entity we store:

- venue unique ID;

id string	researcher ID.
value string	Researcher name.
lastPub list	List containing the researcher last publication's year and ID, in the extracted dataset.
paperList list	List of papers' IDs authored by this researcher in the extracted dataset.
coAuthList dict	A dictionary with one entry for each co-author of the researcher. Each dictionary element contains pre-computed information about the shared works between the two researchers in the dataset.

Table 5.3: List and description of researchers' attributes in the ReviewerNet representaion.

- a list of names and achronims that refers to the journal/venue; this list is used in the approximate string matching routine;
- the number of papers, authors and citations, collected with the reference list 5.1;
- a pre-computed *venue score*, that is the total number of in-citations, over all the venue publications, coming from a different venue; this score is used to decide in which order the venues will be showed to the user and in the corresponding color-map.

This file is loaded first once the user chooses the instance upon which to run a ReviewerNet session and is used to quickly print statistics about the instance, without loading the entire dataset.

5.2 Languages & external libraries

Our visualization system is fully client-side, meaning it is a web page and the bibliographic database is fully loaded and queried on the local client.

For downloading and pre-processing the data we run a bash script that downloads the gzipped partitions and executes python scripts in parallel to process them. The only external dependency in this phase is the string matching python library *fuzzywuzzy*[1] that performs the approximate string matching of venues names. We chose this approach to overcome ambiguity problems in the data (see Section 5.1).

The static part of the website is mainly written in HTML5 and CSS, while for the dynamic one we use Javascript. In the user interface, we also use Bootstrap v3.3.7 responsive and adaptive components and JQueryUi v1.12.1 search-bars and widgets.

[1] `https://github.com/seatgeek/fuzzywuzzy`

Finally, the core external library is D3.js v4.13.0: a JavaScript library for visualizing data using web standards that combines powerful visualization and interaction techniques with a data-driven approach to DOM manipulation [20]. We use it for graph drawing (PN and RN) and for managing the data-driven visualization in general - e.g. the code lines that implement nearly all the available user interactions, described in Section 4.2.2, are calls to this library.

Concerning graph drawing, the PN is a fixed-layout force simulation, while nodes in the RN are free to move and their position is computed with a many-body (or n-body) force simulation. It can be used to simulate gravity (attraction) if the strength is positive, or electrostatic charge (repulsion) if the strength is negative. In order to highlight research communities in the RN, we chose to charge the nodes with a repulsive force, while links attract them proportionally to the number of shared papers. This configuration of the force simulation enhances clustering of "close" researchers and communities.

The D3 implementation of the n-body simulation uses quadtrees and the Barnes-Hut Approximation [22] to improve performance.

5.3 Additional features

ReviewerNet is a visualization platform, but it integrates additional functionalities that the user can exploit to adapt the application to his/her needs and also to speed up the process in general.

Export and load a ReviewerNet session As soon as the user starts to populate one of the networks or views, it is possible to download a plain-text file representing the current ReviewerNet session. The session file contains information about the parameter values (see Chapter 4.2), along with submitting, conflicting and candidate reviewer researchers selected and key papers. For each researcher/paper we store both id and name/title because IDs referring to the same entity may vary across different corpus snapshots; in this way, if, for example, the ID of a researcher in the sessions file is no longer valid, ReviewerNet shows an infromative pop-up that reports the name/title of missing researchers/papers, allowing the user to manually find it with the searchbars.

The user can download the session file with a mouse click either on the "Export Session" button in the CP or in the page footer. If the button in the CP is clicked, ReviewerNet will also report reviewers' names and bibliographic references to their papers (see Chapter 3 and Figure 3.7).

Now, if a user reloads the page and still wants to work on one of his/her previous sessions, with a mouse click on the "Import Session" button in the page footer it is possible to load any session file generated by the platform.

Automated key papers insertion Manually initializing the PN with a set of seed documents can sometimes be time consuming, hence we introduced in ReviewerNet the possibility to automatically add papers to the visualization.

The intuition is that the list of references in the submitted paper or in previously identified seed papers can provide a convenient strategy to initialize the network.

A mouse click on the "Import from bibliography" button in the CP opens a textbox in which the user can copy and paste a set of reference directly from a PDF file. Then, with a mouse click on the "Parse" button in the same window, the system returns the list of matching papers in the dataset, and the user can choose which one of the matching papers to include in the visualization (See Figure 3.2).

Given a reference pasted from a PDF, our parsing and matching routine initially performs string cleaning and then extracts the 3 best matching papers in the dataset, if any, with a score.

The score depends on the number occurrence of words in the paper's title. The main data structures that enable the scoring and matching are two dictionaries:

1. the first stores, for each word in each papers title, the paper in which it occurred. This dictionary is precomputed when the data is loaded;

2. the second is a dictionary of counters, with one entry for each paper that matches one ore more word in the reference title; this dictionary instead is dynamically computed during the import procedure.

So when parsing a reference, for each word, we check if a dictionary entry exists, if so we increment the counters of the papers listed in the word's dictionary entry. When all the words in the reference have been processed, the 3 best matching papers are returned and the best matching one in shown.

Generating a custom ReviewerNet instance For our demonstration, we extracted the publications in the reference dataset from 8 selected sources, all in the field of Computer Graphics. However, if the user feels there are missing venues or simply wants to instantiate the application over a completely different field, ReviewerNet can be built over *any* subset of the Semantic Scholar corpus and customized to include the venues of interest.

We developed a user-friendly procedure to build a customized instance of ReviewerNet that is basically the implementation of the pre-processing steps described in Section 5.1. The code is available on `https://github.com/cnr-isti-vclab/ReviewerNet/tree/master/parser`.

Our solution consists of 5 files:

1. `download_and_parse.sh`, the main file: a bash scripts that downloads each single JSON and runs in parallel python scripts to process the data;

2. `parse_and_filter.py`, a python script that filters a single JSON with the approximate string matching algorithm (see Section 5.2);

3. `merge_data.py` , a python script that consolidates and merges the data as filtered from the corpus;

4. `journals.txt`, a plain-text JSON file that contains the reference list of venues for which we want to extract the publications; for each venue we store an ID and a list of possible names with which the venue is referred in the corpus - we need this information to solve the ambiguity problem with the venue names (see Section 5.1);

5. `util.py`, a python library containing all functions and methods that implement parsing, filtering and merging routines.

To get a personalized instance of Reviewernet, the user has to open `journals.txt` with a text editor and change the content of the JSON object with the names of the venues that will be used to build the topic-based datasets; then, he/she can run `download_and_parse.sh`.

The provided script downloads and parses the partitions in parallel, allowing a maximum number of 10 concurrent downloads and parsers. The output is:

- 180 corpus partitions as gzip files;
- at most 180 parsed files ending with "*-filtered*" suffix;

When all partitions have been downloaded and filtered the merging phase can start: filtered corpus partitions are merged together and the personalized dataset is built. Eventually the script asks the user whether to delete or not the interemediate files (gzipped/filtered partitions).

At this point the `datasets` folder will contain the files needed to run ReviewerNet - that are the authors, papers and journals files (cf. Section 5.1 for details)- ending with "*_pers*" suffix. To use the your customized dataset just start a local or remote ReviewerNet session and, in the starting page, click on the "*Upload instance*" button and select the three files just created.

CHAPTER 6

Evaluation

We evaluated the preliminary version of ReviewerNet described in [10] on the dataset focused on Computer Graphics described in Chapter 3. We decided to ask the scientific community directly, and involve real end-users instead of in-house testers. We sent an email to the 60 members of the IPC of Eurographics Conference 2019, and to additional experts with a record of publications in the top venues of the sector. None of the subjects were involved in the work on ReviewerNet, and none of them knew the system prior to the evaluation test. The participation was on a volunteer basis, with no reward.

We collected 7 responses from the IPC members (10% of the IPC) and 8 responses from additional experts for a total of 15 users. The questionnaire was anonymous and the volunteers were asked to answer three questions about themselves: number of years from their PhD, reviews and reviewer selections per year; Table 6.1 shows the distribution of the results of this part of the questionnaire

The volunteers were asked to search three reviewers for a paper that they had to choose reviewers for in the recent past. This was done so that we could not only collect feedback on the system itself, but also enable the volunteers to comparatively evaluate the performance of the system.

For training, the volunteers were only provided with a 6-minutes video demonstrating the usage of ReviewerNet, namely the video recording a similar scenario to Chapter 3. We did not give any additional training. Also, we asked for a response within five days. This was done to evaluate whether it was easy to get acquainted with ReviewerNet, and whether the system was intuitive and quick to learn. Only one user out of 15 (6.7% of the sample) reported s/he was not able to figure out how to use the system.

Table 6.1: Information about the 15 participants in the user study.

	PhD	≤ 12	> 12
Years from PhD	0.0%	66.7%	33.3%

	< 10	10 - 20	> 20
Reviews per year	6.7%	40.0%	53.3%

	≤ 3	> 3
Reviewer selections in 2018	13.3%	86.7%

The other 14 (93.3%) were able to complete the task assigned with the little support offered. This confirms the user-friendliness of the instrument even if the tool offers many different interaction modalities.

The rest of the questionnaire was divided in two sections, whose questions and summary of answers are reported in Table 6.2 and Table 6.3, respectively. The first section asked the user's opinion about the different functionalities of ReviewerNet, namely: finding key papers (and hence key researchers); presenting the scientific career of candidate reviewers; avoiding conflicts of interest; and finding sets of well distributed reviewers:

73.3% of the testers evaluated ReviewerNet as either good or excellent in finding key papers and researchers.

80.0% of the testers evaluated as good or excellent the presentation of the scientific career of candidate reviewers. One of the testers found that *"[...] the timeline also is a great added value with respect to imagining whether an author is doing a similar research now or he did many years ago"*;

86.7% of the testers thought ReviewerNet was good or excellent to help avoiding conflicts of interest. One of the testers observed how *"[...] the tool actually follows my current practice, that is, look among authors of key papers"* but with the added value of the explicit labelling of conflicting reviewers. He/she also observed that *"[...] the labelling of conflicting reviewers helps also a lot. [...] the tool also helps in selecting reviewers from different areas, covering better the topic of a paper."*

66.7% evaluated as good or excellent ReviewerNet support to find sets of well distributed reviewers.

The second section of the questionnaire asked the users an overall opinion on ReviewerNet, in terms of improvement of the overall quality of the reviewing process, and reduction in the time spent to search for reviewers:

71.4% of the testers agreed or strongly agreed that ReviewerNet helps choosing good sets of reviewers, and hence improves the overall quality of the reviewing process;

71.4% agreed or strongly agreed that ReviewerNet reduces the time spent to look for good sets of reviewers.

In addition, the testers could insert additional comments about ReviewerNet strengths and weaknesses, and suggestions for improvement. We used their comments and suggestions to improve the preliminary version of the platform.

In particular, one of the testers observed that *"[...] inserting manually key papers, takes a little more time, but then the system helps a lot navigating trough related papers and authors"*. Therefore, to reduce the burden on users to initialize the Paper Network and the whole system, we added the *Import from bibliography* functionality, which enables the automatic parsing and matching of lists of references pasted from the References section of articles.

One of the testers found *"[...] a little difficult the interpretation of the researchers network on the top/right. The view is a bit complicated, not immediately clear which node corresponds to the clicked reviewer from the bars on the top/left, [...] but probably again it is just a matter of more practice*". The Researcher Network can be now made full-screen for better visualization and interaction. The nodes are labelled with names, while the node embedding better reflects the closeness of researchers in terms of collaborations (number of co-authored papers). The embedding can be also modified interactively by the user via drag-and-drop.

Concerning ReviewerNet ability to find key papers and researchers, one of the testers observed that vision-related venues (e.g., conferences such as CVPR and ICCV and journal such as IJCV and TPAMI) were missing from the list of sources for key papers and authors on which the demonstration tool was built. He/she observed that the tool would have been more useful if these were included, since many works overlap vision and graphics. Similarly, another tester observed that the homogeneous nature of the sources selected made so the proposed reviewers could show no enough divergence and could be scarce. In this respect, it is worth noticing that ReviewerNet can be built over *any* subset of the Semantic Scholar corpus and customized to include the venues of interest. Therefore, these comments mostly apply to the particular instance used for testing. The version of the platform presented in this thesis features the possibility to upload a personalized list of venues, pertaining to any academic domain, and on which to build a customized dataset.

Table 6.2: Distribution of answers to the first section of the questionnaire (14 participants) where the acronyms in the first row stand for: Very poor, Poor, Average, Good and Excellent.

	VP	P	A	G	E
How do you rate ReviewerNet in finding key papers and researchers?	0.0%	0.0%	26.7%	**46.6%**	26.7%
How do you rate ReviewerNet in presenting the scientific career of candidate reviewers?	0.0%	6.7%	13.3%	**46.6%**	33.3%
How do you rate ReviewerNet in avoiding conflicts of interest?	0.0%	6.7%	6.7%	40.0%	**46.7%**
How do you rate ReviewerNet in finding sets of well distributed reviewers?	0.0%	6.7%	26.7%	**40.0%**	26.7%

Table 6.3: Distribution of answers to the second section of the questionnaire (13 participants) where the acronyms in the first row stand for: Strongly Disagree, Disagree, Neutral, Agree and Strongly agree.

	SD	D	N	A	SA
I think that ReviewerNet helps choosing good sets of reviewers, and hence improves the overall quality of the reviewing process.	0.0%	0.0%	28.6%	**35.7%**	**35.7%**
I think that ReviewerNet reduces the time spent to look for good sets of reviewers.	0.0%	0.0%	28.6%	28.6	**42.9**

We believe the results from the evaluation study showed a very high value of user satisfaction, and also offered useful suggestions for improvement.

CHAPTER 7

Conclusions

In this thesis I presented ReviewerNet, a novel system for choosing reviewers by visually exploring scholarly data. ReviewerNet enables scientific journal editors and members of IPCs to search the literature about the topic of a submitted paper, to identify experts in the field and evaluate their stage of career, and to check possible connections with the submitting authors and among the reviewers themselves. This helps to avoid conflicts and to build a fairly distributed pool of reviewers. To do so, ReviewerNet features a combined visualization of the literature, the career of potential reviewers, their conflict of interests, and their nets of collaborators. Interestingly enough, the system is able to help the process even without exploiting any content-based analysis of the papers.

Throughout the implementation of the system I dealt with dimension and quality of the reference corpus. I solved ambiguity and inconsistency problems and output both a consolidated dataset upon which our default ReviewerNet instance runs, and a user-friendly procedure to build a well-formed dataset starting from a list of venues given by the user.

ReviewerNet web interface was developed with the lowerst possible number of external dependencies. The core library is D3.js. It allows one to bind arbitrary data to a Document Object Model and perform data-driven transformation to it. I have extensively used this library to set up and manage force simulations, and deploy informative visual interactions.

The evaluation of a preliminary version of the demonstration platform with both in-house testers and members from the Computer Graphics community confirmed that the users were able to get acquainted with the system even with a very limited training, and appreciated the different functionalities of ReviewerNet and its capability of improving the reviewer search

process. Some of the users also highlighted the potential of ReviewerNet as a tool to support bibliographic research, besides the reviewer selection process.

The evaluation also highlighted that there was room for improving the system, which we did, during the last months of the thesis project, by: improving on the effectiveness of the visualization and the user-friendliness of the interaction modes; improving the initial process of selecting key papers, whose manual insertion was signalled as a weakness by some users, by importing the bibliography of papers; defining a procedure to generate instances of the platform with customizable data coverage, by loading one's own reference dataset. Additional directions for improvement include: an automatic strategy to suggest key papers by computing network features (e.g., betweenness centrality); and a user-friendly visual interface for building the reference dataset, through the assisted selection a list of venues of interest.

Bibliography

[1] "Scopus fact sheet," 2019, accessed on February 14th, 2020. [Online]. Available: https://www.elsevier.com/__data/assets/pdf_file/0010/891397/Scopus_GlobalResearch_Factsheet2019_FINAL_WEB.pdf

[2] "Web of science platform: Summary of coverage," 2020, accessed on February 14th, 2020. [Online]. Available: https://clarivate.libguides.com/webofscienceplatform/coverage

[3] "Microsoft academic," 2020, accessed on February 14th, 2020. [Online]. Available: https://academic.microsoft.com/

[4] "Dblp statistics - new records per year," 2019, accessed on February 14th, 2020. [Online]. Available: https://dblp.uni-trier.de/statistics/newrecordsperyear.html

[5] L. Bornmann and R. Mutz, "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references," *Journal of the Association for Information Science and Technology*, vol. 66, no. 11, pp. 2215–2222, 2015.

[6] S. Khan, X. Liu, K. A. Shakil, and M. Alam, "A survey on scholarly data: From big data perspective," *Information Processing and Management*, vol. 53, pp. 923–944, 2017.

[7] P. Federico, F. Heimerl, S. Koch, and S. Miksch, "A survey on visual approaches for analyzing scientific literature and patents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 9, pp. 2179–2198, 2017.

[8] M. Ley, "The dblp computer science bibliography: Evolution, research issues, perspectives," in *International symposium on string processing and information retrieval*. Springer, 2002, pp. 1–10.

[9] W. Ammar, D. Groeneveld, C. Bhagavatula, I. Beltagy, M. Crawford, D. Downey, J. Dunkelberger, A. Elgohary, S. Feldman, V. Ha, R. Kinney, S. Kohlmeier, K. Lo, T. Murray, H.-H. Ooi, M. Peters, J. Power, S. Skjonsberg, L. L. Wang, C. Wilhelm, Z. Yuan, M. van Zuylen, and O. Etzioni, "Construction of the literature graph in semantic scholar," in *NAACL*, 2018.

[10] M. Salinas, D. Giorgi, F. Ponchio, and P. Cignoni, "A visualization tool for scholarly data," in *Eurographics Conference on Smart Tools and Applications in Graphicsl*. Eurographics Digital Library, 2019.

[11] F. Wang, N. Shi, and B. Chen, "A comprehensive survey of the reviewer assignment problem," *International Journal of Information Technology & Decision Making*, vol. 9, no. 4, pp. 645–668, 2010.

[12] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. P. Hsu, and K. Wang, "An overview of microsoft academic service (ma) and applications," in *Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion)*. ACM New York, USA, 2015.

[13] C. Chen, *Mapping Scientific Frontiers - The Quest for Knowledge Visualization*, revised 2nd edition ed., ser. Elementary Differential Geometry Series. Springer, 2013.

[14] E. Garfield, A. I. Pudovkin, and V. S. Istomin, "Why do we need algorithmic historiography?" *Journal of the American Society for Information Science and Technology*, vol. 54, pp. 400–412, 2003.

[15] N. J. van Eck and L. Waltman, "Citenetexplorer: A new software tool for analyzing and visualizing citation networks," *Journal of Informetrics*, vol. 8, pp. 802–823, 2014.

[16] J.-K. Chou and C.-K. Yang, "Papervis: Literature review made easy," *Computer Graphics Forum*, vol. 30, pp. 721–730, 2011.

[17] C. Görg, Z. Liu, J. Kihm, J. Choo, H. Park, and J. Stasko, "Combining computational analyses and interactive visualization for document exploration and sensemaking in jigsaw," *IEEE transactions on Visualization and Computer Graphics*, vol. 19, no. 10, pp. 1646–1663, 2013.

[18] F. Osborne, E. Motta, and P. Mulholland, "Exploring scholarly data with rexplore," in *The Semantic Web – ISWC 2013*, H. Alani, L. Kagal, A. Fokoue, P. Groth, C. Biemann, J. X. Parreira, L. Aroyo, N. Noy, C. Welty, and K. Janowicz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 460–477.

[19] S. Miksch and W. Aigner, "A matter of time: Applying a data–users–tasks design triangle to visual analytics of time-oriented data," *Computers & Graphics*, vol. 38, pp. 286–290, 2014.

[20] M. Bostock, V. Ogievetsky, and J. Heer, "D3 data-driven documents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2301–2309, Dec. 2011.

[21] C. Ware, *Information visualization. Perception for design.* Morgan Kaufmann, 2012.

[22] J. Barnes and P. Hut, "A hierarchical o(n log n) force-calculation algorithm," *Nature*, vol. 324, no. 6096, pp. 446–449, 1986. [Online]. Available: https://doi.org/10.1038/324446a0

Ringraziamenti