INFLUENCES AMONG RESEARCH INSTITUTES

Iulian Radu, Ricardo Garcia, Hina Shah
iulian.radu@gatech.edu, rgarcia8@gatech.edu, hina.shah@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia 30332

Abstract

This paper presents a system for visualizing influences in the academic research community in the domain of Software Engineering, between the years of 1990 and 2007. Students, researchers, and industry professionals are able to use this system to perform tasks such as determining which institutions are influential in specific conferences or on other institutions. The application provides an interactive environment for visualizing publications in five Software Engineering conferences, collected from the ACM Digital Library. The Model-View-Controller architecture of our system uses a variety of information visualization techniques, developed as a result of iterative design process which included user evaluations.

1 Introduction

Over the years, many research areas have been evolving and research institutes located all across the world have been making active research contributions to these areas. In addition, new conferences get established, providing platforms for researchers to recognize and publish their work. However, some research institutes have greater influences in some research areas as compared to others. In this paper we measure the 'influence' of a university U. based on the number of papers U. publishes in conferences, and based on the number of references made from other universities to papers published in U.

The task of visualizing academic research has been approached in the past. In 2004, the IEEE InfoVis conference held the contest "The History of InfoVis"[1], about creating visualizations which support the discovery and identification of major research topics, relationships between members of the community, and trends over time. Some of the tasks that the contest focussed were: characterizing the research areas and their evolution, showing where a particular author fits within the research areas, and showing the relationships between two or more or all authors. Interesting visualizations were created to visualize major research topics and how individual researchers in Information Visualization domain influence its research. However, the focus was on understanding the history of the Information Visualization domain and how individual researchers influence the domain research.

In this project, we present a visualization tool that supports the understanding of research influences at a level above the individual researcher - at the research institutions level. The purpose is to provide users with facilities for realizing insights as to which research institutes are most influential in a specific research domain. Such information can be used for identifying top research institutes by students applying for graduate studies, by graduating students hunting for jobs in research institutes/universities, by companies wanting to recruit persons with specific skills and so on.

This project is currently limited to the domain of Software Engineering, due to constraints in project duration and development team size; however, we expect the prototype to be scalable to various disciplines of computer science, as well as other research domains, and thus be applied to finding answers to similar questions on a larger scale.

The main contributions of this paper are:

- A list of user tasks addressed through our visualization
- The description of the tool which allows visualization of research influences in software engineering
- The presentation of preliminary evaluation results of this visualization tool

The next section discusses various user tasks, section 3 discusses the design alternatives, the implementation details are discussed in the section 4, the section 5 presents the results of the informal evaluation of the visualization, and finally section 6 states the conclusion and future work.

[1] http://infovis.org/infovis2004/

2 Users and Tasks

Most of the existing resources such as U.S. News provide with information about ranking of the top research universities in textual format. Also, most of the times, rankings are done at the department level (i.e., computer science), and popular resources rank only the universities. We thought it to be interesting to visualize influences of research institutes (academic and industrial) among each other to get insights as to which research institutes are most influential in a research domain, where are the most influential programs in this domain, which institutions are innovators and which are followers, which are the most important conferences and how actively research institutes are involved in research in a particular domain.

We identified many user groups that can benefit by using research influences information to perform different tasks as listed below:

- New Students: searching for the university to go for further studies where top research is being conducted in research area of one's interest.
- Graduated Students: comparing between industrial and academic organizations to determine opportunities the do the most influential research.
- Professors: determining which university/research lab to collaborate with, accept students from, or go for a sabbatical to.
- Researchers within a Community: noticing trends of influences among different research institutes.
- Employers: identifying organizations (institutes and/or academics) to search for candidates with desired skills and profile.
- Funding agencies: determining what institutes to fund for conducting research. The most influential places as well as the fledging insitutions are the ones that should be targeted since they could make the best use of resources.

3 Design Approaches

The users and the tasks listed in the previous section influenced the design process of our visualization system. The design process started with in-class discussions of possible ideas. We explored options of visualizing the entire computer science research area. However, due to the foreseen scalability issues and taking into consideration the short time duration, we restricted the scope of our visualization system to support the domain of software engineering only. In addition, our current visualization system currently provides information about publications in four conferences over the period of 1990 to 2007 only.

We explored interesting designs for visualizing research influences. Firstly, we investigated options of representing them as network graphs where the nodes of the graph represent each institute and connection between nodes represent the influences. However, with only the network graph, the location of each node would not have conveyed any information and hence there would be a need to label and identify each institute independently. Therefore, we adopted an alternative approach of super-imposing the network graph onto a geographic map where the nodes of the graph still represent each institute and the connection represent the influences. However, now the location of the nodes maps to the relative locations of the institute on the globe. Such an alternative representation, providing contextual information along with details about research influences seemed more useful and beneficial.

Secondly, we were debating on whether the user should be provided with the facility to select their main view (that is, the ability to swap between which view should be displayed in the center of the application). However, we eliminated the option of selections of main view and decided on making the map-based influence network view as the main view because this view provided the most relevant and extensive information. We added the other views as the peripheral views - providing information that may be useful but not necessary. All the views and the application architecture are discussed in detail in the next section.

Another important source of ideas and inspiration was the poster session. Here the team presented some alternatives to the original design such as using dynamic queries for the time selection. Some of the ideas inspired by the poster session in the final design were the ability to indicate the most important topics which an institution focuses on, and the idea of showing a historical graph on the time slider. The poster session also allowed us to prioritize ideas, causing us to give high priority to panning and zooming in the map, and low priority to allowing the user switch between views. Some of other ideas were also popular with the poster audience, such as having a more artistic representation of links between institutions; however, because of time constraints, this we had decided not to implement.

In all, the current design abstracts some features from the original version. Papers are still included, but the main focus now is on the institution. Nevertheless, the current design still manages to display interesting emergent features, which are the insights we are interested in obtaining.

4 Implementation

In this section we describe the implementation details of our system. We begin with an overview of data sources and existing tools, follow with a description of our application, then outline information visualization techniques used in the application, and conclude with a description of our architecture.

4.1 Data

Activity about academic publications can be found on a variety of online databases, such as CiteSeer [10], ACM [11] and IEEE Xplore[12]. All databases provide access to paper abstracts, authors, title, and publication date. However, we have preferred the ACM Digital Library as our data source for several reasons: (1) it groups publications by their publication venue and publication year, thus allowing us to easily aggregate information by conferences, (2) it provides the affiliation of paper authors, allowing us to aggregate data by institutions, and (3) it provides links to the references made by each paper, allowing us to build an influence network. The ACM database could only be accessed through a web browser, thus we were forced to build our own HTML scraper. Our dataset is focused on Software Engineering from 1990 to 2007, and we have collected data in 5 conferences: PASTE, SIGSOFT, AOSD, IWSSD, and ISSTA. The total number of collected papers in these conferences was 885, and the total number of collected institutions was 558. Several issues occurred during data collection, which caused our visualization to work with a subset of this data: (1) when authors specify affiliation, they do not follow a consistent format, therefore "MIT Media Lab" and "Media Lab, Massachusetts Institute of Technology" are treated as two distinct institutions in our dataset; (2) due to Optical-Character Recognition errors in ACM recordings, reference links from one paper to another are sometimes not available.

4.2 Existing Tools

Prior to choosing our implementation approach, we did consider three alternatives: the Processing language [13], Java Swing [14], and Prefuse for Java [15]. Our interface would include a dynamic interactive map, a time slider, and modules for drawing stack charts and bar charts, all requiring a simple Model-View-Controller framework; thus we biased our evaluation toward supporting these issues.

Processing is a Java-based language suited for graphic designers. Facilities are built for easy drawing and interaction with graphical elements, thus we would be able to easily construct the necessary views. The downside of the language is that it provides no framework for easily encapsulating graphical entities (such as Java provides for Swing containers), and that no existing information visualization modules are available.

Java Swing provides an object-oriented approach to managing window graphics. Graphical components are contained within a framework which handles layout and interaction. Due to the constraints imposed by the framework, however, creating highly interactive visual components is quite difficult and time-consuming.

Prefuse is an information-visualization library for Java. It is a framework for creating information visualization applications, and allows the user to create and manipulate data sources, and create views on the data through custom components. Unfortunately, the examples provided were too simplistic for the application we had in mind; thus we observed that we would have to create our own map module and custom interactions, after becoming familiar with the framework.

After evaluating these alternatives, we decided that our interface would be best suited by an environment where graphical views and interactions can be easily constructed, and settled on the Processing language.

4.3 Visualization Application

As outlined in the previous section, our application is focused on providing interactive views on the collected data of academic publications in Software Engineering.

A view of the interface is shown in Figure 1. The interface contains several views of the data: a map view, a stack charter and a bar charter view; furthermore, filters on the data can be controlled through selection components: time slider, chart dropdown lists, and the "Institution / Conference" tabs. Details of the different components and their interactions are described below.

Data Views: The data view consists of the following views:

1. Map View

   The Map module shows geographic location of academic institutions in the dataset, denoting institutions by circular marks. If the user has not selected any university, the size of each university mark is an aggregate of the the number of papers published in the selected year. If the user selects a specific institution, the application will focus all its views on the activities of that institution. In this case, the selected university will be linked to all universities that have been influenced by it (ie: that have made references to it) during the year, and the current university's size

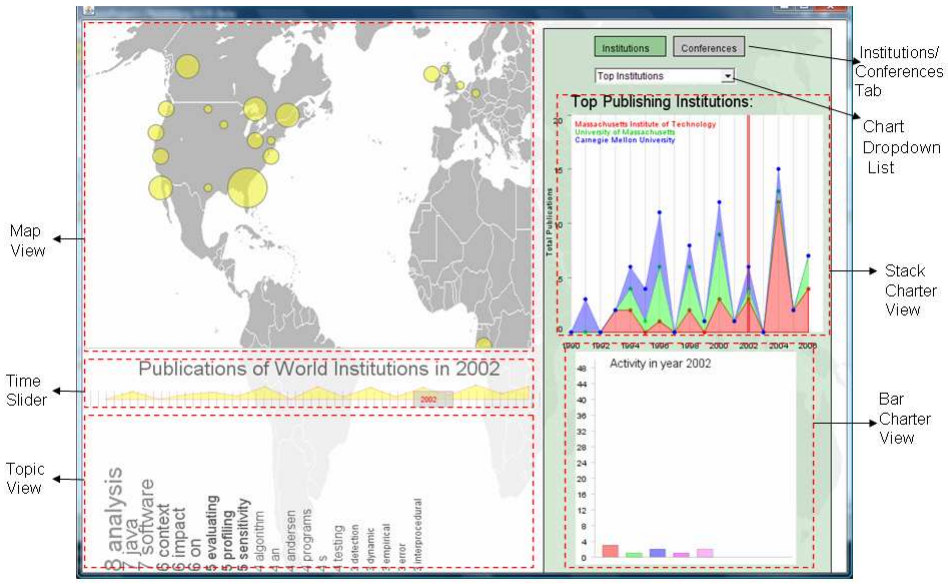

Figure 1. Snapshot of the tool

will be proportional to the number of references made by all other universities.

2. Stack Charter View

   The Stack Charter component shows historical information from the year 1990 to 2007, with the currently selected year being highlighted. The information graphed may be about institutions or conferences, and the selection is determined by the "Institutions/Conferences Tab" selector. In the Institutions mode, the default view shows the top publishing institutions, stacked on top of each other. When a user chooses to focus on an institution (by selecting it on the map), the stack charter shows the institution's influence on its neighbors. In the Conferences mode, the default view shows the history in the top conferences with most publications. When an institution is selected through the map view, the stack charter displays the top conferences in which the current institution publishes. Stack charts are useful for displaying historical information for multiple entities; however, this may be difficult to read, therefore in both modes, the user can chose to focus on only one chart using the Chart Dropdown List filter, described below.

3. Bar Charter View

   The Bar Charter provides a more detailed view of the current year, by displaying bar charts of the information shown in the Stack Charter. This visualization allows one to more easily compare between institutions or conferences within a given year.

4. Topic View

   The final view of the data allows the user to see what topics are discussed in academic publications. This component shows a list of words sorted by their frequency. The frequency of a word increases when the institution publishes papers whose title contains the word, and also when another institution refers to a paper whose title contains this word. Thus, the component shows what topics an institution is influential in.

Data Filters: The data filters include the following:

1. Time Slider

   This component allows the year which a user wishes to focus on. When this component is dragged, the application views are updated to show data for the specific year.

2. Chart Dropdown List

   This list focuses the display of the stack and bar chart modules. The user can choose to view

"Top" institutions / conferences in the charter views, or he/she can select a specific entity to focus on. This allows the user to have a clearer view of the historical information for their selection, by filtering extraneous entities.

3. Institutions / Conferences Tab

   This component directs the stack and bar charts to display information either about institutions or conferences, as described above. The switching functionality allows the interface to maintain its simple design, at the same time that it provides users with these different aspects of the data.

4.4 Interaction Techniques

In order to appropriately convey information to users, a variety of information visualization techniques are employed by our application.

Information on institutions is multivariate, as each entity contains a name, geo-position coordinates, a time-series of publication counts, time-series of references from other universities, and time series of participation in conferences. Our application splits this data by providing multiple views, and allowing Brushing interactions between views (for example highlighting one item in the stack chart and seeing its position on the map), allowing users to collect information about one entity from multiple visual representations simultaneously. This process is enhanced by the tools functionality of aggregating data, for example showing institution sizes on the map as being proportional to the number of references received in a specific year, a facility which permits users to get a summary view of information. However, at the same time, some of the aggregated information can be dissociated into its individual components, as users can see which institutions reference a selected institution, or determine at which conferences an institution has its publications.

Zooming and panning are implemented in the geographical view, as well as linking between institutions. Focus and context is also implemented in several components, as the Bar Chart view shows details of the historical information shown in the Stack Chart, and the Time Slider provides a contextual view of the institutions publications. Details about an institutions publication topics are shown in the Topic View module. More details on demand are provided through the use of tooltips, and the applications switching views when an institution is selected.

Filtering and dynamic queries are enabled through interaction with the data filtering modules. A user primarily controls the institution to focus on by interacting with the map marks. Additionally, he/she may dynamically select a year to focus on by dragging the Time Slider, or focus the

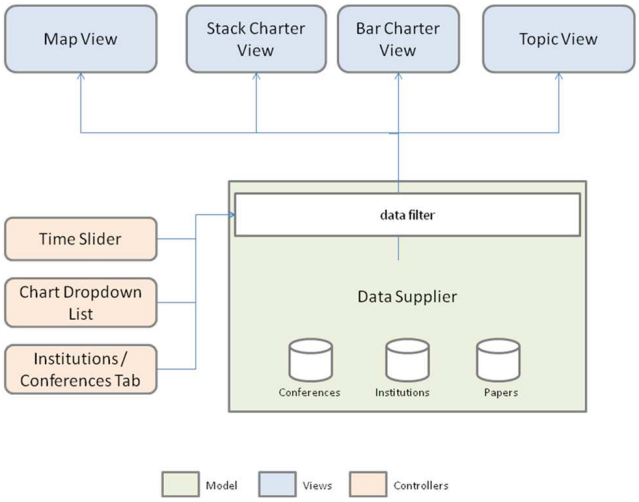

Figure 2. shows the Model-View-Controller architecture of our visualization system.

charter modules on a specific institution or conference by controlling the dropdown list.

4.5 Application Architecture

The application components described above are either views or filters on the data. Because of this degree of interconnection between components, we have designed the architecture around a Model-View-Controller model, as shown in Figure 2

The View components are the specified map view, stack charter view, bar charter view, and topic view. All components are dynamically updated from the data model. The Model is held by the internal DataSupplier component, which contains a database of collected Conferences, Institutions, and Papers; these entities are loaded from an external database file produced by a separate scraper program. Internal to the DataSupplier module is a filtering functionality, which extracts data for the views as described by the controller components. Finally, Controller components are the time slider, chart dropdown list, and institution/conference tab selector. Some components may take both Controller and View functionality, for example the Map module has both data viewing capabilities (since it displays institutions on a map), as well as control abilities, since it can focus the application on a specific institution.

Through this architecture, we have enabled modules to be dynamically influenced by each other as the user navigates the dataset, while at the same time the modules remain decoupled, allowing easy addition and removal of components.

Three students have built this application. In the final

version, the ACM scraper codebase was 1832 lines of Java code (of which 453 lines were imported from an existing scraper), and the application codebase was 3319 lines of Java code (of which 935 lines are common with the Scraper, and 344 lines were taken from existing examples).

5 System Evaluation

5.1 Tool Evaluation

To gather some feedback of our visualization we conducted preliminary evaluation studies. The evaluation consisted of two parts: internal evaluation and external evaluation as described below:

Internal Evaluation: The internal evaluation covered self evaluation of the of the system. The approach we adopted to self-evaluate the system was that each group member selected couple of user tasks listed in section 2 and tried to perform those tasks using the visualization system. We faced issues and identified some bugs in the system such as the nodes on the map were initially opaque due to which the underlying information was hidden; we solved this issue by increasing the transparency of the nodes to view the underlying details. We observed that the size of the nodes provided with quick quick idea of the count of the publications, and made the comparison tasks simpler and easier. In addition, we even gained some interesting insights about the data which is as listed below:

- Massachusetts Institute of Technology has consistent publications since 1994 in ISSTA
- ISSTA conference is held once in every two years, and this was then confirmed by visiting the official website of ISSTA conference [2]
- The publications in each conference have increase over years (i.e., the number and the size of nodes). For instance, in 1992 the publications in ISSTA were less and then it increases gradually in 2004
- In the year 2004 there has been many publications in the domain of software engineering
- Massachusetts has lot of papers on test,testing related publications, University of Massachusetts has many analysis related publications, which may indicate that they are specialized in the testing and analysis research areas within software engineering respectively.
- It was easy to figure out the number of publications Georgia Tech has in ISSTA in the year 2002 (i.e., 2 publications). We even verified this by looking at the websites of software engineering faculty in Georgia Tech.

External Evaluation: Using the tool made some of the task simple and easy to perform. However, we believe that an external evaluation would help to get deeper insights into our visualization system's usefulness and usability aspects. Hence we conducted an informal evaluation study with one graduate student working in the software engineering group. The evaluation process was very informal and structured as : introducing the system, allowing them to play around with it, asking them how they would perform some of the user tasks, asking if they would use the system to perform any other tasks and if so then which, and finally asking about the probable improvements. The consolidated replies are as follows:

- The layout is very appealing and it was a good attempt to show a lot of information
- The connection lines are colored gray, which makes it difficult to identify. Changing it to some contrasting color will serve the purpose better
- The horizontal listing of the text at the bottom does not look appealing. "Why do I have to twist my head to look at the relevant data?". However, one of the group members were of the opinion that the horizontal layout seemed impressive

We addressed some of the feedback comments, which were doable considering the time constraints. The rest of the comments are addressed as future work in section 6

6 Conclusion and Future work

In this paper we have presented an application for visualizing influence within the academic research community. The system aggregates data from collected papers in the domain of Software Engineering, permitting users to focus on the level of research institutions and conferences. Multiple visualization techniques, ranging from brushing, to filtering, to overview and details, are employed to permit intuitive navigation through influences between research institutions. Data collected from the ACM Digital Library is presented through the Model-View-Controller architecture of our system which was developed in the Processing language.

In terms of future work for the system there is room for the improvement and addition of several features. Performance is a problem for our application - currently some

[2] http://issta08.rutgers.edu/

caching is performed at the module level, however more caching is necessary in order to run the application on slower machines. Additionally, the application needs to better parse institution names, an issue which has been discussed in the data section, so as to display the whole set of papers in the selected conferences. In the future, we expect more data to be added to the system by way of more conferences. On the other hand, there are also several additional features that would contribute to the ease of use and power of the system. Primarily we would like to see more details about how an institution influences another; thus, a third mode could be added to the application, allowing one to select two (or more) universities and see the influences between them. The application can also benefit from more filtering abilities such as dynamic time sliders and sorting. Currently, the time selector can choose single years, but with a range query, it would be possible to see trends through the years more easily.

References

[1] Acm digital library: http://www.acm.org/dl.

[2] Citeseer research index: http://citeseer.ist.psu.edu/.

[3] http://prefuse.org.

[4] http://www.cs.umd.edu/hcil/iv04contest/, 2004.

[5] http://www.gapminder.org/.

[6] Ieee xplore: http://ieeexplore.ieee.org/.

[7] R. A. Amar and J. T. Stasko. BEST PAPER: A knowledge task-based framework for design and evaluation of information visualizations. In *INFOVIS*, pages 143–150. IEEE Computer Society, 2004.

[8] Fry, Ben. Investigating exhibits. In *Proceedings of DIS'02: Designing Interactive Systems: Processes, Practices, Methods, & Techniques*, pages 32–36, 2002.

[9] J. Heer and M. Agrawala. Software design patterns for information visualization. *IEEE Trans. Vis. Comput. Graph*, 12(5):853–860, 2006.

[10] J. Heer, Card, S. K., Landay, and J. A. prefuse: a toolkit for interactive information visualization. In *Proceedings of ACM CHI 2005 Conference on Human Factors in Computing Systems*, volume 1 of *Interactive information visualization*, pages 421–430, 2005.

[11] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL*, pages 336–343, 1996.