paper:72

Towards Supporting Systematic Mappings Studies: An Automatic Snowballing Approach

Fábio Bezerra[1], **Carlos H. Favacho**[1], **Rafael Souza**[4], **Cleidson de Souza**[2,3]

[1] ICIBE-UFRA – Belém-Pará-Brasil

[2]ITV-DS – Belém-Pará-Brasil

[3]Faculdade de Computação-UFPA – Belém-Pará-Brasil

[4]IC-UNICAMP – Campinas-São Paulo-Brasil

fabio.bezerra@ufra.edu.br

cleidson.desouza@acm.org

Abstract. *Systematic mapping is a secondary research method that aims to summarize and synthesize the current state of an area, providing a general map of the field. It is a technique that requires the execution of a series of steps, many of them repetitive, which makes this technique time-consuming and error-prone. To address these issues, this paper presents an algorithm for automatic selection of references based on both backward snowballing (from the list of references) and forward snowballing (finding citations to the papers). Our algorithm is especially useful for supporting the selection phase of a systematic mapping study, therefore it represents an effort towards a tool for facilitating systematic mapping research. In order to assess its efficacy and efficiency, we evaluated the algorithm in a set of experiments using data collected from a semester-long graduate course about Computer-Supported Cooperative Work (CSCW).*

1. Introduction

As a research area evolves through years, the number of studies in such area often increases. This can be noticed by the number of papers published in conferences and/or journals and even with the creation of new conferences focusing on that particular area. At some point, it then becomes important to summarize the current state of the area. Such an overview is helpful to guide new researchers as well as to help the field itself to assess its evolution, providing then new directions for future research.

An overview of a research area can be provided by the so-called secondary studies [Kitchenham and Charters 2007]. A secondary study aims to review all primary studies relating to a specific research question in order to integrate and synthesize evidences about this question [Kitchenham and Charters 2007]. Among the existing secondary studies, two stand out for having a well-defined methodology: systematic review and systematic mapping. These methods adopt a rigorously defined process in order to reduce the bias of their conclusions [Petersen et al. 2008, Kitchenham and Charters 2007, Scannavino 2012] and, thus, they are known as **systematic studies**. A systematic mapping is a method of secondary research that aims to show the state of the art of the analyzed area through a **general map**, usually presented as diagrams, charts and statistics [Petersen et al. 2008, da Silva et al. 2012]. On the other hand, a systematic review is a secondary research method used to provide a comprehensive and clear assessment of the state of a research area, relevant to a particular topic of interest [Felizardo et al. 2012].

Systematic reviews and systematic mappings differ in terms of goals, breadth and depth, and their usage have different implications for the classification of the topic being investigated and the research approach. According to [Kitchenham and Charters 2007] systematic mapping studies "are designed to provide a wide overview of a research area, to establish if research evidence exists on a topic and provide an indication of the quantity of the evidence" whereas the systematic review is used "to identify, analyze and interpret all available evidence related to a specific research question in a way that is unbiased and repeatable".

1.1. Motivation

The execution of a systematic study is a process quite costly and error-prone. First, it employs activities like studying, reading and sorting large amounts of articles. Second, it requires the cooperation of many researchers during the process. For that reason, it is essential that the search of papers return only the most relevant ones. Then, to obtain the desired quality in the activity of identifying articles, methods of strictly planned search are commonly used.

In this context, it is worth to notice that it is a common practice among researchers to select a set of the most important papers in a certain area of research and, from these papers, identify the relevant related work. This survey practice is the basis of the method of selection known as snowballing[Jalali and Wohlin 2012]. The most common snowballing approach works in the following way: from a relevant paper, references from this paper are selected, and then from these selected papers, new references are selected in a iterative process.

Actually, the snowballing is a more general method of selection of papers, but it also requires a set of initial articles, called seeds, to start the process. In this case, from the seeds, there are two approaches to selection: backward snowballing, which selects papers referenced by the seeds. This is the most common and intuitive adopted approach. The other approach is forward snowballing, which selects papers that do cite the seeds. The selected papers from an iteration of the snowballing algorithm compose the group of articles, which will be the seeds for the following iteration. The process continues iteratively until a stopping criterion is satisfied.

Given this description, it is clear that the snowballing approach requires tool support, because, otherwise, it would demand an effort that would make the approach infeasible, if manually conducted. For instance, just the selection of a large number of articles, in both snowballing approaches (backward and forward), can be a large, complex and and time-consuming task. Therefore, a tool that automates all, or part, of the snowballing process presents itself as an interesting alternative support to the selection of papers, because it would decrease the time needed for the implementation of the process, facilitate the work of the researchers involved, and, therefore, provide benefits for the entire research community.

1.2. Objective and Outline

This paper presents an algorithm that automates the snowballing method for selecting papers in a systematic study. The algorithm presented in this paper was implemented in a web system called Ramani [de Souza et al. 2013], developed for supporting a systematic mapping project. [de Souza et al. 2013] focus on the collaborative aspects of the tool, while this paper focuses on the algorithm description, implementation, and evaluation.

The rest of this paper is organized as follows: we present in Section 2 the theoretical background that informed the creation of the automatic snowballing algorithm and guided its implementation in the Ramani tool. Details of this algorithm are described in Section 3, while its evaluation is described in Section 4. Finally, in Section 5 we present our final comments and plans for future work.

2. Background and Related Work

In this section, we first describe the tasks necessary for supporting the systematic mapping process (Section 2.1). This overview is important at this point so that the reader can understand the complexity and rigor of a systematic study. In the context of this paper, the search step is especially important, for it is the core of snowballing algorithm. Moreover, searching is a step that requires a lot of effort from the researchers. Finally, in Section 2.2 we present some related work.

2.1. The Systematic Mapping Process

Figure 1 presents the phases of a systematic mapping study proposed in [Petersen et al. 2008]. In the first phase, the **definition of the research question**, questions should be formulated based on the objective of the research, always focusing in the ultimate goal of a systematic mapping: to produce an overview of a research area. As an outcome, the scope of the review is defined. This scope is used as an input for the search, the next step of the systematic mapping process.

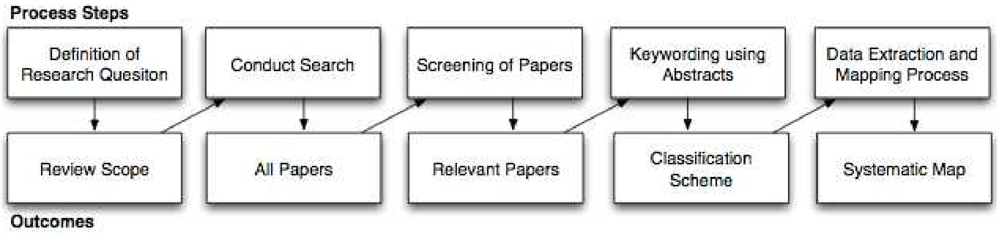


Figure 1. Systematic Mapping Process

The second step is to **conduct a search for primary studies**. These studies are identified by using search strings on scientific libraries / databases or browsing manually through relevant conference proceeding or journal publications. After obtaining the initial set of papers, they should be **screened to select the relevant papers** that help to answer the research questions, applying inclusion and exclusion criteria, i.e., criteria that define whether a paper should be included or excluded from the list of relevant papers. The following phase, **keywording of abstracts**, is often done in two steps. First, the reviewers read abstracts from the papers and look for keywords and concepts that reflect the contribution of the paper. When the final set of keywords has been chosen, they can be clustered and used to form the categories for the map.

Different categories of papers can be used depending on the research question. We illustrate an interesting category proposed by Petersen et al. (2008) that focuses on the type of contribution reported in the paper: a tool, a process, a method, etc. This category is based on an existing classification of research types and described in [Wieringa et al. 2006]:

Solution Proposal papers the paper presents a new solution (method or tool), but does not evaluate it;

Validation Research papers the paper presents a new solution (method or tool) and evaluates it in a simulated or fictitious scenario;

Evaluation Research papers the paper presents a new solution (method or tool) and evaluates it in a real scenario;

Philosophical papers the paper proposes a taxonomy or a conceptual framework of the field;

Opinion papers the paper express a personal opinion of a solution (method or tool) that already exists, but it does not report an evaluation; and

Experience papers the paper reports the use of a solution (method or tool) that already exists and relates an experience assessment of the solution.

Finally, in the **data extraction and mapping** phase, once the classification scheme is developed, one must extract the data from each paper including year of the publication, authors, venue, categories, etc and document this in a format that can be later processed (e.g., a spreadsheet). With this information, the frequencies of publications in each category can be computed. The analysis of the results focuses on presenting the frequencies of publications for each category allowing one to find out which categories have been emphasized in past research and, as a consequence, to identify gaps and possibilities for future research.

2.2. Related Work

Sytematic studies are mainly based on search strings in databases [Kitchenham and Charters 2007, Dieste and Padua 2007, Petersen et al. 2008, Kitchenham et al. 2009], but there are some efforts based on the selection of the list of references and snowballing [Webster and Watson 2002, Runeson and Skoglund 2009, Jalali and Wohlin 2012]. Whatever the chosen approach, what it is really important in a systematic review is to find as many primary studies relating to the research question as possible [Kitchenham and Charters 2007]. In this context, even systematic reviews based on search strings recommend the selection of **relevant** primary studies as another source of selection.

[Jalali and Wohlin 2012] report a comparison between using snowballing and search strings as a way for conducting a systematic study. An important result of this work is that “despite the differences in the included papers, the conclusions and the patterns found in both studies are quite similar”. That is, for the context of this work, snowballing appears as a good automatic approach. For example, in [Webster and Watson 2002] the authors recommend snowballing as the main method to find relevant literature. More specifically, they suggest the use of relevant papers from leading journals in the beginning of the method as seeds. However, comparing to our approach, both of these works do not deal with snowballing in an automatic way.

The work of [Runeson and Skoglund 2009] is closer related to the snowballing approach presented here. These authors present a search strategy based on four components: (i) a "take-off paper"; (ii) papers referenced by the "take-off paper"; (iii) identification of "cardinal papers"; and (iv) papers from external sources referencing the "cardinal papers". The "take-off paper" is a paper regarded as very relevant on the topic – the authors argue that researchers conducting a systematic review should easily be able to select such a relevant paper, based on their pre-understanding of the research question. On the other hand, "cardinal papers" are those papers referenced more than others – the authors believe that those papers are more likely to be referenced also from the relevant papers available in external resources. However, different from the snowballing algorithm presented in Section 3, the first two components of this approach select just the list of references from the "take-off paper". As we will explain later, this is equivalent to only the first iteration of our snowballing algorithm. Similarly, the last two components of [Runeson and Skoglund 2009]'s approach adopt a procedure alike forward snowballing, but limited to one iteration. As we will describe in the following section, our algorithm conducts forward and backward snowballing during several iterations.

3. An Automatic Snowballing Algorithm

As mentioned before, the main contribution of this paper is an approach for selecting papers based on snowballing, which we call automatic snowballing. Such an approach is called automatic because it does not require the interaction of the researcher during the selection of papers. The automatic snowballing simultaneously uses both backward and forward snowballing during its execution, which ends when the algorithm is not able to find additional articles from the group of referenced or cited papers. Therefore, in order to minimize the stress of processing, this algorithm considers as an input parameter the definition of one or more conferences (or journals) to limit the search space or scope, i.e., this works as a filter for the selection of articles referenced or cited. This algorithm is described in the following section.

3.1. The Algorithm

Our algorithm requires four input parameters: (i) **seeds**, which represent the list of known relevant papers; (ii) **conferences**, which are used to filter the papers selected during the process, limiting the search space and contextualizing the topic of interest; and finally, the last two input data, (iii) **project** and (iv) **collaborator**, which are used to save the selection (in a given project), and associate the collaborators or researchers in the given project that can have access to the returned papers. The result is saved in a variable **groupsOfSeeds**, and it is a "list of list of papers", that is, each iteration of the algorithm returns a *list of papers*, which is added in the *groupsOfSeeds* and contains the seeds of next iteration.

The algorithm is described on Algorithm 1. The seeds used as input are added in the result set (**groupsOfSeeds**) at the very beginning of algorithm, then they are saved as a partial selection for the collaborator in the project. The snowballing process ends when there are not seeds to be processed anymore, that is, in Line 4 when the list of papers (**seeds**) is empty. Otherwise, the lists of references and citations of each paper in the

Algorithm 1: Automatic Snowballing

```
   Input: seeds
   Input: conferences
   Input: project
   Input: collaborator
   Output: groupsOfSeeds
 1 groupsOfSeeds ← {};
 2 groupsOfSeeds.add(seeds);
 3 select(seeds, project, collaborator);
 4 while not seeds.isEmpty() do
 5     seeds ← createSeeds(seeds);
 6     if seeds not null then
 7         seeds ← removeDuplication(groupsOfSeeds, seeds);
 8         seeds ← filterByConferences(seeds, conferences);
 9         groupsOfSeeds.add(seeds);
10         select(seeds, project, collaborator);
```

seeds list are selected, creating the seeds to be used in the next iteration (**createSeeds** in Line 5).

It is worth to note that **createSeeds** is the function responsible for extracting the data from the digital library, that is, in an implementation of this algorithm it should include a mechanism for collecting this data, like a web-crawler. The papers found by that function may have already been selected before, so it is important to remove the duplications (Line 7). Moreover, a filtering is applied over the resulted list of papers (Line 8), removing those articles that are not published in the given conference list (**conferences**).

3.2. Implementation issues

The automatic snowballing algorithm is available as a selection function in the Ramani, a collaborative software tool [de Souza et al. 2013], which was designed over a set of free and popular technologies like Java, JSF, Primefaces, JPA and MySQL. To use Ramani, the user should upload a file with the initial seeds and the conference list used to define the scope of the review. This file is added within a specific project, which can be accessed by a specific set of collaborators.

Furthermore, as mentioned before, Ramani implements a crawler that reads information from specific sites, namely the ACM or IEEE digital libraries, since the current implementation of the crawler supports only these two digital libraries. During the execution of the crawler, when an article is found in the list of references or citations from a seed, this article is first searched in the local database. If this article is not in the local database, then its data is extracted from the digital libraries. By conducting queries in the local database first, we aim to optimize the processing time of our tool. In addition, to avoid being blocked by the digital libraries, we randomly set a delay in seconds for the next query.

4. Automatic Snowballing Algorithm: Assessment

For the assessment of the proposed algorithm, we have used the data described in Section 4.1, while the results of test are presented in Section 4.2.

4.1. Materials and Methods

The materials used for the assessment of the automatic snowballing algorithm were collected in a graduate class on 2012. This class focused on the topic of Computer-Supported Collaborative Work, or simply CSCW. To be more specific, during this class, a review study was conducted by 10 participants: seven students, a faculty, and the first two authors. The faculty is the last author of this paper, while collaborators are from UFRA.

The objective of that study was to create a mapping of the studies about collaborative software engineering in the context of the CSCW area, i.e., using the papers available in the ACM CSCW Conference, through the systematic mapping methodology (see Section 2.1). For that, all papers published in the previous 16 editions of CSCW conferences were analyzed, i.e., the period from 1986 to 2012. This included an universe of 639 full and short papers, which were filtered by the participants to select those which satisfy the following pre-requisites: (i) papers that focused on the development of a particular software collaboratively; (ii) papers that described empirical studies on how software engineers worked; or (iii) papers that described software tools or methods that allowed software development to be performed collaboratively, or that allowed the construction of tools that supported collaborative software development. We present in Table 1 the 31 papers selected in that study. Due to space limitations, we present only the DOI (Document Object Identify) number for each paper.

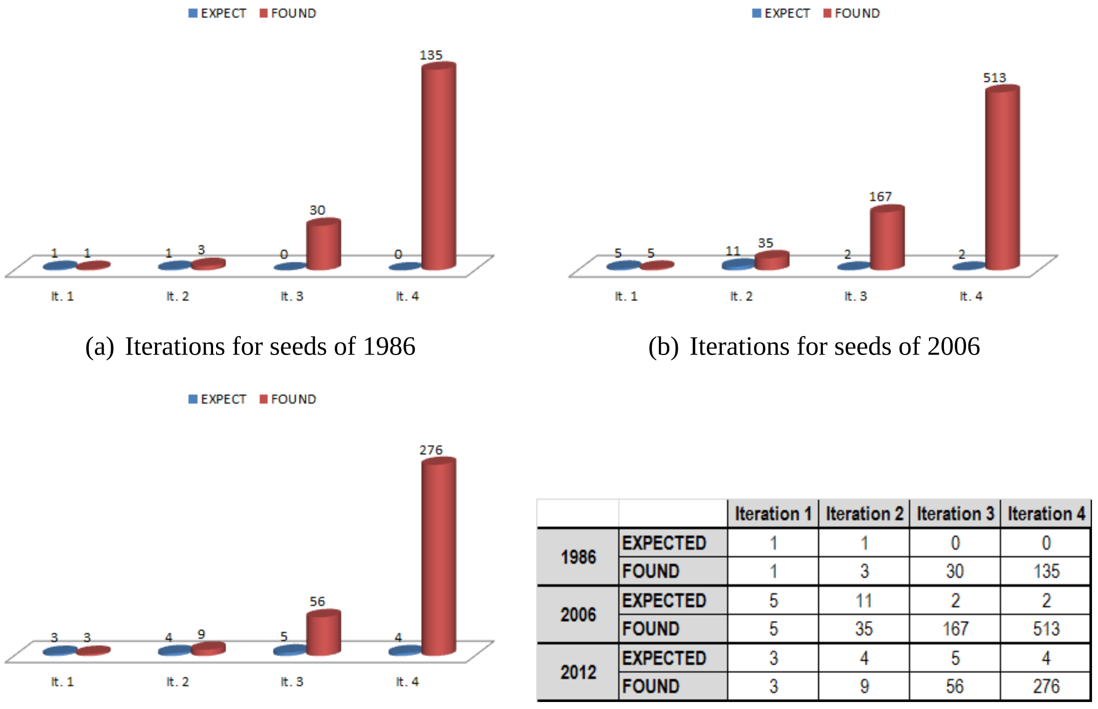
Table 1. Selected papers about collaborative software development

Year	DOI	Year	DOI	Year	DOI
2012	10.1145/2145204.2145401	2008	10.1145/1460563.1460654	2004	10.1145/1031607.1031612
2012	10.1145/2145204.2145403	2008	10.1145/1460563.1460581	2004	10.1145/1031607.1031620
2012	10.1145/2145204.2145345	2008	10.1145/1460563.1460565	2004	10.1145/1031607.1031622
2011	10.1145/1958824.1958851	2006	10.1145/1180875.1180883	2002	10.1145/587078.587080
2011	10.1145/1958824.1958923	2006	10.1145/1180875.1180882	2000	10.1145/358916.359004
2011	10.1145/1958824.1958889	2006	10.1145/1180875.1180884	1990	10.1145/99332.99352
2010	10.1145/1718918.1718972	2006	10.1145/1180875.1180906	1990	10.1145/99332.99356
2010	10.1145/1718918.1718958	2006	10.1145/1180875.1180929	1990	10.1145/99332.99347
2010	10.1145/1718918.1718973	2004	10.1145/10316071031704	1986	10.1145/637069.637071
2010	10.1145/1753326.1753677	2004	10.1145/10316071031621		
2010	10.1145/1718918.1718971	2004	10.1145/1031607.1031611		

This list of papers allowed us to identify the initial seeds used in the assessment of our automatic snowballing algorithm, as well as the efficacy and efficiency of the algorithm, i.e., the parameters for evaluation. For evaluating the efficacy, we expected as the result of our algorithm to obtain the same set of papers presented in Table 1. On the other hand, for evaluating efficiency or performance, we expected to obtain such papers in a short number of iterations – but at most four iterations for the context of this assessment.

Because our automatic snowballing algorithm applies both backward and forward snowballing, we decided to assess three kinds of initial seeds, namely: (i) seeds from the beginning of the whole period used in the search (1986); (ii) seeds from the middle period under analysis, i.e., papers from 2006; and (iii) seeds from the final of the period (papers from 2012[1]). These seeds were chosen because testing the papers of 1986 help to assess the forward snowballing component, while the papers of 2012 help to assess the backward snowballing component. Finally, seeds from the middle help us to assess both backward and forward aspects of the snowballing algorithm.

4.2. Results and Discussion



(a) Iterations for seeds of 1986

(b) Iterations for seeds of 2006

(c) Iterations for seeds of 2012

		Iteration 1	Iteration 2	Iteration 3	Iteration 4
1986	EXPECTED	1	1	0	0
	FOUND	1	3	30	135
2006	EXPECTED	5	11	2	2
	FOUND	5	35	167	513
2012	EXPECTED	3	4	5	4
	FOUND	3	9	56	276

(d) Data from three different set of seeds

Figure 2. Results for automatic snowballing using three different sets of seeds

We present in Figure 2 the results of the algorithm assessment. In Figure 2(d) we report the data used for plotting the three graphics, which report, for each iteration, both the number of found papers and the number of expected papers among the found ones. The first iteration represents the initial set of seeds, and because they were composed from the papers of Table 1, the number of found papers is equal to the number of expected ones.

Among the three sets of seeds, the one that helped find more new expected papers in four iterations was the set of papers of 2006 (see Table 2). Such a set of seeds helped find 15 new expected papers, while the set of seeds of 1986 helped find just one new expected paper, and the set of seeds of 2012 helped find 13 new expected papers. Moreover, at the end of the forth iteration the seeds of 2006 helped select 720 papers of the ACM CSCW Conference, while the seeds of 1986 helped select 169 papers and the seeds of 2012 helped select 276 papers. Therefore, because it intensively explored both backward and forward search, since this set of seeds was in the middle of the search space, the seeds of 2006 helped select more papers of the given search context (CSCW

[1] When this paper was being written, the last edition of the CSCW Conference happened on March, 2014, while in the data available for the evaluation of the algorithm the last edition is 2012, since that is when the class who conducted the systematic mapping took place.

Conference), so this helped find more new expected papers. As reported in Table 2, the seed of 2006 has bigger recall (it helped find more papers than were expected).

Table 2. Data after four iterations

	Initial Seeds	Found	Expected	New Expected	Recall	Precision
1986	1	169	2	1	6.45%	1.18%
2006	5	720	20	15	64.52%	2.78%
2012	3	344	16	13	51.61%	4.65%

Combining the results of each set of seeds, we identified that 814 papers of CSCW Conference were selected. Among these, 22 papers are part of the set of expected papers presented in Table 1. It is worth to notice that the number of available papers in the CSCW Conference increased on 2014, so the universe of papers is bigger than the 639 availables on 2012.

A problem of the current implementation of this algorithm is its processing time. At each iteration, because more seeds are found, the processing time increases in a polynomial way. For that reason, the results presented herein are limited to four iterations, which required, for each assessed set, approximately 12h of processing. Therefore, future extensions of this algorithm should consider a more accurate selection of papers that will compose the seeds of the next iteration.

5. Conclusions and Future Work

Because a systematic study demands the cooperation of many researchers during the process and it is quite costly and error-prone,a tool that supports part or the whole process of conducting systematic studies seems to be interesting. In this context, Ramani, which is a collaborative web system tool, appears as a solution for such a demand. This work presented an automatic snowballing algorithm helpful for supporting the selection phase in Ramani. The goal of the automatic snowballing algorithm is to return only the most relevant papers in the context of a systematic review.

The evaluation of the proposed algorithm was based on data collected during a graduate class about CSCW that took place on the Spring of 2012. In that class, 10 participants selected, in a peer review process, 31 papers about collaborative software engineering, published until 2012 in the ACM CSCW Conference. From these papers, we selected three sets of seeds for assessment, the papers published in 1986 (the beginning of the series), 2006 (the middle of the series) and 2012 (the end of the series). After four iterations of executions for each set of seeds, the algorithm did not achieve a 100% of accuracy. Moreover, it returned better results with the seeds of the middle of the series, which helped find 15 new papers.

In the future, in order to get a broader assessment of algorithm, we plan to evaluate other seeds. Such an assessment may help us find what we call *golden seeds* for a given context (research topic or area). These are the seeds with better accuracy (precision and recall) and efficiency (performance). Moreover, others extensions for the automatic (or semi-automatic) snowballing approach may be developed, for instance, an interative snowballing approach or a filter based on an ontology to better select the references and citations. Finally, we plan to extend the support of crawler to other digital libraries.

References

da Silva, F. Q. B., Suassuna, M., França, A. C. C., Grubb, A. M., Gouveia, T. B., Monteiro, C. V. F., and dos Santos, I. E. (2012). Replication of empirical studies in software engineering research: a systematic mapping study. *Empirical Software Engineering*, 19(3):501–557.

de Souza, R., Lopes, C., Bezerra, F., and de Souza, C. R. B. (2013). Ramani: Uma ferramenta de apoio à colaboração durante a execução de estudos sistemáticos. In *Proceedings of the X Brazilian Symposium in Collaborative Systems*, SBSC '13, pages 144–147, Manaus, Amazonas, Brazil. Sociedade Brasileira de Computação.

Dieste, O. and Padua, O. (2007). Developing search strategies for detecting relevant experiments for systematic reviews. In *Empirical Software Engineering and Measurement, 2007.*, ESEM 2007, pages 215–224, Madrid. IEEE.

Felizardo, K. R., MacDonell, S. G., Mendes, E., and Maldonado, J. C. (2012). A systematic mapping on the use of visual data mining to support the conduct of systematic literature reviews. *Journal of Software*, 7(2):450–461.

Jalali, S. and Wohlin, C. (2012). Systematic literature studies: database searches vs. backward snowballing. In *Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement*, ESEM '12, pages 29–38, Lund, Sweden. ACM-IEEE.

Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., and Linkman, S. (2009). Systematic literature reviews in software engineering: a systematic literature review. *Information and Software Technology*, 51(1):7 – 15.

Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, Keele University and Durham University Joint Report. Tech. Rep. EBSE 2007-001.

Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M. (2008). Systematic mapping studies in software engineering. In *International Conference on Evaluation and Assessment in Software Engineering*.

Runeson, P. and Skoglund, M. (2009). Reference-based search strategies in systematic reviews. In *The Proceedings of the 13th International Conference on Evaluation and Assessment in Software Engineering*, Durham, England. British Computer Society.

Scannavino, K. R. F. (2012). *Evidence-based software engineering: systematic literature review process based on visual text mining*. PhD thesis, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.

Webster, J. and Watson, R. T. (2002). Analyzing the past to prepare for the future: writing a literature review. *Management Information Systems Quarterly*, 26(2):13–23.

Wieringa, R., Maiden, N., Mead, N., and Rolland, C. (2006). Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements Engineering*, 11(1):102–107.