Experiences from using snowballing and database searches in systematic literature studies

Deepika Badampudi
Department of Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
deepika.badampudi@bth.se

Claes Wohlin
Department of Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
claes.wohlin@bth.se

Kai Petersen
Department of Software Engineering
Blekinge Institute of Technology
Karlskrona, Sweden
kai.petersen@bth.se

ABSTRACT

Background: Systematic literature studies are commonly used in software engineering.There are two main ways of conducting the searches for these type of studies; they are snowballing and database searches. In snowballing, the reference list (backward snowballing - BSB) and citations (forward snowballing - FSB) of relevant papers are reviewed to identify new papers whereas in a database search, different databases are searched using predefined search strings to identify new papers. **Objective:** Snowballing has not been in use as extensively as database search. Hence it is important to evaluate its efficiency and reliability when being used as a search strategy in literature studies. Moreover, it is important to compare it to database searches. **Method:** In this paper, we applied snowballing in a literature study, and reflected on the outcome. We also compared database search with backward and forward snowballing. Database search and snowballing were conducted independently by different researchers. The searches of our literature study were compared with respect to the efficiency and reliability of the findings. **Results:** Out of the total number of papers found, snowballing identified 83% of the papers in comparison to 46% of the papers for the database search. Snowballing failed to identify a few relevant papers, which potentially could have been addressed by identifying a more comprehensive start set. **Conclusion:** The efficiency of snowballing is comparable to database search. It can potentially be more reliable than a database search however, the reliability is highly dependent on the creation of a suitable start set.

Categories and Subject Descriptors

D.2 [**Software Engineering**]: Management; G.3 [**Probability and Statistics**]: Experimental design

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
EASE '15 April 27 - 29, 2015, Nanjing, China
Copyright 2015 ACM 978-1-4503-3350-4/15/04 ...$15.00.
http://dx.doi.org/10.1145/2745802.2745818.

General Terms

DB- Database, SB- Snowballing, BSB- Backward snowballing, FSB- Forward snowballing

Keywords

Snowballing, Database search, Efficiency, Reliability

1. INTRODUCTION

Evidence based software engineering (EBSE) was introduced in 2004 by Kitchenham et al.[5]. The objective of EBSE is to synthesize the evidence from multiple primary studies. A means to synthesize the evidence are systematic literature reviews. Guidelines have been proposed for conducting literature studies. Kitchenham and Charters [4] recommended a search strategy using well-defined search strings in databases (DBs) and also the review of reference lists from relevant primary studies and review articles. In the guidelines [4] it is recommended to use backward snowballing (BSB), i.e. the use of the reference lists of studies to identify additional relevant papers, as a complement to DB searches. Webster and Watson [10] have recommended to use SB as the main method for conducting the search in literature studies.

As different search strategies exist, studies [3], [12] have been conducted to evaluate if the search strategy impacts the actual outcomes of the systematic literature studies. Jalali [3] compared DB search and BSB and found that the results are not dependent on the search strategy. More recently, Wohlin [12] has proposed guidelines for conducting SB in systematic literature reviews and performed a replication study using those guidelines. He concludes that using SB as a search strategy might be a good alternative to DB searches. However there are few studies comparing the search strategies and more studies are needed to investigate under which circumstances SB is preferred over DB search.

This study complements the studies [3] and [12] by evaluating the efficiency and reliability of SB and DB search. For the research presented in this paper a systematic mapping study using different search strategies was performed by different authors independently during the same time period. Snowballing (forward - FSB and backward - BSB) and DB search were the two different search strategies used in the mapping study. The same research questions and inclusion/exclusion criteria is used in both search strategies. The following research questions are answered in this evaluation

study:

- *RQ1 How are the relevant papers identified through the evolution of the SB process?*
  The main contribution of this research question is to recognize the identification pattern of the accepted papers.
- *RQ2 How efficient is the SB?*
  Efficiency is the noise vs. relevance ratio. In order to answer this research question, the efficiency of SB is evaluated and also compared with DB.
- *RQ3 How reliable is SB in capturing all relevant papers?*
  In this context reliability is regarded as the ability to identify all relevant papers. We contribute to this research question by evaluating the common and unique results identified by SB and DB search.

The remainder of the paper is outlined as follows. Related work is presented in Section 2. Section 3 describes the research method. The result of this study are presented in Section 4 and discussed in Section 5. Section 6 concludes the paper.

2. RELATED WORK

Only a few studies have focused on the reliability and efficiency of systematic literature studies. MacDonell et al. [6] evaluated the reliability of systematic reviews. They compared the results of two systematic reviews, which had common research questions performed by two independent groups of researchers. The conclusions of this study was that the reviews were robust to difference in the review process and people involved.

An audit was conducted to find how the primary sources were originally identified by Greenhalgh and Peacock [2]. The audit result was that only 30% of the studies were found by the DB search, whereas 51% of papers were found through SB. The conclusion of the audit was that a systematic review of complex evidence cannot completely rely solely on DB searches (predefined, protocol driven), instead browsing library shelves, asking colleagues, pursuing interesting references (SB) may have better efficiency.

Skoglund and Runeson [9] evaluated the reference based search with the objective to reduce the large number of papers to be reviewed. The results for the technically focused reviews were satisfactory with only a few missing relevant papers. However, for reviews where the search area is wide or included general terms, the results were not satisfactory, i.e. large number of papers were missing. Hence they conclude that the performance in terms of precision is context dependent.

In another study conducted by Jalali and Wohlin [3], they investigated the reliability by comparing the results of two systematic reviews on the same topic. Both reviews were conducted by the same authors, however, these reviews were not conducted in the same time period. The first study was conducted using DB search, and the second study was conducted after a couple of months using BSB as a search strategy. In Jalali and Wohlin's study they found that the actual results are not highly dependent on the search strategy. Another conclusion was that the efficiency of SB might be higher when the keywords for searching include general terms.

More recently, in 2014, Wohlin conducted a replication study [12] to compare SB results including BSB and FSB and DB search. The outcome of the replication study using SB was similar to the original study conducted using DB search, also SB is considered to be a good alternative to DB search.

The reliability of mapping studies was discussed in a study [13], it includes potential areas of improvement to increase the reliability. One of the improvement suggested is to use SB as search strategy as SB based on researchers expertise and knowledge in an area is more efficient than finding optimal search strings. However the study indicates the need of more studies comparing the search strategies to understand which search strategy is better in which circumstances. Hence this study is conducted as a complement to studies [3] and [12]. In particular, More studies are needed to be able to determine when SB is better than DB search and in what circumstances.

3. RESEARCH METHOD

The main objective of this study was to reflect on the lessons learned from using SB as a search strategy in literature studies. Snowballing was used as a main method to find the existing literature related to the strategic decision-making process to select among different development options. The four main development options considered in the study were: in-house, outsource, COTS and open source. Hence the inclusion criteria for the study was to include papers if one of the following criteria was fulfilled: [1]Papers discussing decision-making process for selecting an option. [2] Papers comparing two or more of the options [3] Papers proposing solutions that support the decision-making process in the selection of an option in relation to other options. Papers were excluded if they were not related to component-based software or if papers discussed architectural and development aspects of components based system. Papers discussing adoption of software packages, IT services or operating system were excluded. Grey literature and non-English papers were also excluded from the study.

A DB search was also performed as a complementary step to validate the SB results. The two searches, i.e DB and SB were conducted by independent authors in the same time period. Deepika Badampudi and Claes Wohlin conducted the SB, while the DB search was conducted by Kai Petersen. The same inclusion and exclusion criteria were used by the authors to avoid the threats related to the judgment of inclusion and exclusion criteria, and hence making the results of the two searchers comparable. The treats related to the reliability of DB results as single researcher conducted it is under control as the researcher has vast experience in conducting DB search. In this study we focus on lessons learned from SB and compare the SB and DB search results. The details of the searches are described in the following sections.

3.1 Details of Snowballing

The SB search was conducted by first creating a start set, and thereafter conducting BSB and FSB of the start set in an iterative fashion.

Creation of the start set: The search strings to create the start set are shown in Table 1. They were applied on Google scholar. The keyword *"Software"* was added to three of the search strings as most of the papers retrieved were not related to software engineering.

Google scholar is not restricted to specific publishers, and hence it was selected as the index DB to create the start set. From the results returned by the Google scholar search, the first 10 results of each of the nine search strings (see Table 1), were reviewed using the inclusion and exclusion criteria. That is, a total of 90 search results were reviewed. The papers that fit the inclusion criteria were added to the start set in two phases.

In *Phase 1*, the papers were tentatively included based on title, abstract and introduction. In some cases more sections were reviewed. However an extensive full text review was not conducted in this Phase.

In *Phase 2* the inclusion or exclusion was done based on full text reading. The reviewing in Phase 1 and 2 was done by first two authors independently. At the end of each phase a review meeting was held to analyze the review process. The decision rules (cf. [7] for an overview) shown in Table 2 were applied for the final inclusion or exclusion.

Five papers were selected that were used as input for the SB activity (including BSB and FSB).

Snowballing activities: BSB and FSB were carried out in iterations. The reference list of the papers in the start set were reviewed during BSB and citations were reviewed during FSB. The citations were retrieved from Google scholar.The papers that were included in the iterations were added to the start set and SB of the newly added papers was done in the next iterations. This process was followed until no new papers were found. When all the papers were added to the start set it was considered as the final set, which included all primary studies. The same inclusion and exclusion criteria were used and the decision to include or exclude was made in two phases (see above) as in the start set. However the review process for Phase 1 was different from the start set, in BSB the following was reviewed in Phase 1:

1. Title of the referenced paper.
2. The point of reference (reference context) of the paper.
3. Abstract of the referenced paper.

Whereas in FBS the following order is followed:

1. Title of the paper citing.
2. Abstract of the paper citing.
3. The point of reference (reference context) to the paper being cited.

The reference context refers to the text surrounding the reference citation within the primary study. This was done as the text surrounding the reference citation allows to understand the context of the citation, for example the reason for citing the paper. Note that steps 2. and 3. was reversed in FSB as it was easier to read the title and abstract first when looking at the papers citing a relevant primary study.

Table 1: Nine search strings used by SB

In-house vs. outsourcing
In-house vs. COTS
In-house vs. OSS
COTS vs. OSS
COTS vs. outsourcing
Outsourcing vs. OSS
Additional search strings
In-house vs. outsourcing and software
Outsource vs. OSS and software
COTS vs. Outsourcing and software

Table 2: Inclusion and exclusion decision rules

Case	Action
Both authors accept a paper	Include the paper for next step
Both authors reject a paper	Exclude the paper
Either one of the authors accepts a paper	Include the paper for next step

3.2 Details of database search

The DB search was performed to increase the confidence in the selection of papers. The search terms were divided into population, interventions, comparison, and outcome (PICO). The search strings as shown in Table 3 were applied in Scopus and Inspec/Compendex. The papers were either included or excluded based on the same inclusion and exclusion criteria used for SB. Thereafter, the first author reviewed the included papers to check their relevance. In cases of disagreements the authors discussed the papers until they were resolved.

3.3 Research Questions

In order to reflect on SB and compare it with DB search, the following research questions are formulated -

- *RQ1 How were the relevant papers identified through the evolution of SB process?*
  As stated earlier, in SB the first step is to create the start set and keep adding papers through BSB and FSB. In this research question we analyze the pattern in which the papers were found. With additional studies reporting patterns, if we recognize repeating patterns we can improve the guidelines for the SB procedure. For example, this can lead to more concrete guidelines on how to construct the start set, and which papers to apply SB on. Overall, with a higher number of data points, in the future we could propose steps based on reliable evidence that lead to a more effective and efficient SB procedure.
- *RQ2 How efficient is SB?*
  Efficiency is the number of papers included in relation to the total number of papers reviewed. One of the

Table 3: Search strings used by database search

Database search
Search 1: (X OR Y) AND (I1 OR I2 OR I3) AND E
Search 2: (X OR Y) AND (I1 OR I2 OR I3) AND F
Search 3: (X OR Y) AND (I1 OR I2 OR I3) AND G
E: A AND (B OR C OR D)
F: B AND (C OR D)
G: C AND D
A: (COTS OR "components off the shelf" OR "component off the shelf")
B: (In-house OR inhouse)
C: (open-source OR "open source" OR OSS)
D: (outsource OR out-source OR outsourcing OR "third-party")
"software (X)" and *"component (Y)"*
"trade-off (I1)", *"decision (I2)"*, and *"selection (I3)"*

risks in mapping studies is to review a large number of non-relevant papers, i.e noise [12]. Hence to answer RQ2 the noise vs. relevance ratio is analyzed. As SB was carried out in iterations, the efficiency of all the iterations as well as BSB and FSB was analyzed independently in this research question. The efficiency was also compared to the efficiency of the DB search. Besides the title and abstract, the reference context was also used as a decision base in SB. Reviewing the reference context is an additional step which is not considered in the DB search. Hence, we evaluated the usefulness of the reviewing reference context in decisions to include or exclude papers. In this research question we also considered the total number of decisions made based on the reference context of the primary studies and how many of these decisions changed when the full text of the paper was reviewed. Hence the following sub-questions were answered in this research question

  - RQ2.1 What was the efficiency of start set, iterations, BSB and FSB?
  - RQ2.2 How useful was the reference context as a decision base?
  - RQ2.3 How efficient was SB in comparison to DB search?

- *RQ3 How reliable was SB in capturing all relevant papers?*
  In this context the reliability of SB is the ability to capture all relevant papers. A complimentary DB search was conducted to evaluate and compare the reliability of SB. If the papers found in DB search are mostly the same as papers found in SB, with only a few unique papers found in DB search, then we can claim that in this case the SB process was reliable. The DB search used additional keywords related to the dimension (selection, trade off and decision). Hence, we evaluated if the SB process is as reliable as the DB search in identifying papers addressing different dimensions, and also development options. Hence the research question was divided into the following sub questions:

  - RQ3.1 What were the common and unique *papers* identified in SB and DB searches?
  - RQ3.2 What were the common and unique *development options* identified in SB and DB searches?
  - RQ3.3 What were the common and unique *dimensions* identified in SB and DB searches?
  - RQ3.4 To what extent did we have the same conclusion using two different searches? Note that the same question has been asked in [3].

4. RESULTS

4.1 Evolution of the SB process (RQ1)

The first step in SB was to identify a start set from which the SB can start. The goal was to find papers comparing two or more development options. Based on the four options considered in the study we have six possible pairs to compare the development options. Each comparison pair is considered as a category in the start set. Hence we have six comparison categories as shown in Figure 1.

Thus, one search strings was formulated for each combination of development options as shown in Table 1.

The first ten results from Google Scholar for each search were reviewed. The evolution of identified papers is depicted in Figure 1.

Start set: In total, five papers were included into the start set. Three papers from the COTS vs. OSS search string, one from In-house vs. COTS and one from In-house vs. OSS search string. However, the paper found in the In-house vs. OSS search string was also comparing COTS hence, the paper was added to a new category (In-house vs. COTS vs. OSS) as shown in Figure 1. No papers were found for the In-house vs. Outsource, COTS vs. Outsource and Outsource vs. OSS comparisons. No papers were identified for any combination with outsource as one of the development option.

We found a high number of papers that were not software related for the outsource combination search strings hence, we included three additional search strings by adding software as a keyword. The inclusion of software in the search strings did not provide any additional papers. Although we were hopeful that the categories that had papers would contribute to the empty categories as they have some common options. Most papers (3/5) were added to the category COTS vs. OSS. In total 5 papers were added to start set represented by circles in Figure 1.

Iterations: Four iterations were needed to reach saturation.

1st Iteration: In the 1st iteration the reference list and citations of 5 papers were reviewed. Two papers in start set did not generate any papers. Snowballing papers in the COTS vs. OSS category added 3 papers in new category (only OSS), it also added papers in two existing categories (In-house vs. COTS and In-house vs. COTS vs. OSS). In total 10 papers were added in this iteration represented by triangles in Figure 1.

2nd Iteration: In this iteration a paper is added in a new category i.e Make vs. buy vs. share. One paper each is added to COTS vs. OSS and In-house vs. COTS categories. In total 3 papers were added in this iteration represented by squares in Figure 1.

3rd Iteration: 2 papers were added in this iteration represented by diamonds in Figure 1.

4th Iteration: No papers were added in this iteration as SB of the two papers identified in 3rd did not find any new

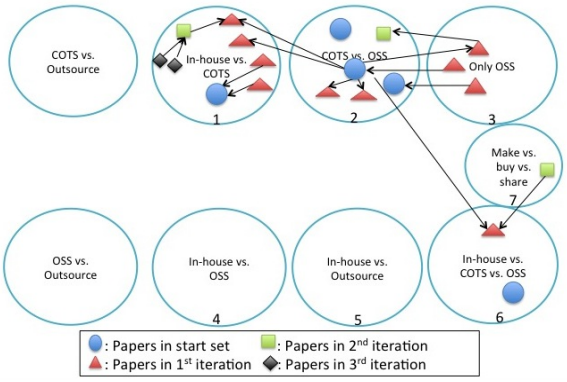

Figure 1: Evolution of identified papers

papers. Hence this is the final iteration in the SB process.

In the start set the COTS vs. OSS category had most papers, but at the end of the first iteration in-house vs. COTS category had the highest number of papers. Also 5/8 papers in the in-house vs. COTS category were from the COTS vs. OSS category. The 2 papers in the start set did not generate any papers. Interestingly, even though some categories were not covered in the start set, the SB procedure allowed us to find papers in these categories (here Only OSS and Make vs. buy vs. share).

Charateristics of start set The creation of the start set was an important step as the papers identified in the SB process depend on the papers in the start set. The characteristics of a good start set are discussed in the SB guidelines [12]. The characteristics are stated in Table 4, along with the compliance to the guidelines in the SB procedure conducted for this paper.

The papers in the start set were organized in clusters. Each cluster contained papers that had no citation relations to each other, i.e. they did not refer to each other and/or did not have common authors. One paper was referring to another paper in the start set, this also was the same paper with common authors. Hence, they belonged to the same cluster. In this way the start set represented to the best of the authors knowledge at this stage different clusters. It may be fewer cluster, since other papers found later may bridge between two clusters identified at this stage in the process. Even though not all papers in the start set were adhering to the guidelines, each cluster had the characteristics of a good start set as the papers were referring to each other and did not have same authors. The papers found in start set were not published recently. Hence, based on this there is a good chance of finding papers in both FSB and BSB.

Citation Matrix: The referencing and citing between the papers is shown in Figure 2. The referencing is denoted as (×), for example it can be seen that P5 was referencing papers P6 and P11. Similarly P5 was cited by papers P12 and P15. It was not possible to cite the papers that had not been published yet, this is denoted as (−). There was one exception, i.e. P12 (2007) is referencing P17 (2008) even though P17 was published later. This is because the reference was to an unpublished version of the same paper. The blank cells indicate that the papers were not referencing or citing other papers. The papers shown in Figure 2 are grouped according to the iterations:

- P1 to P5: start set
- P6 to P15: 1st iteration
- P16 to P18: 2nd iteration
- P19 to P20 3rd iteration.

Furthermore, the papers in each iteration are arranged according to the publishing year.

P4 was referencing six papers, which was the highest number of references to other included papers in this study. P11 had the highest number of citations, which was five papers. Both P4 and P11 were published in 2006. As the year 2006 was halfway in the publishing time-line (2000 to 2014), it gave the papers published in 2006 a good chance to be both cited and referring to papers included. P5 was also published in 2006 although it had two citations and two references to included papers. This may be due to the fact that P4 and P11 belonged to a cluster which had a higher number of papers identified compared to the cluster that contained P5.

Here, it is interesting to observe that, even though papers belong to different categories, as a common topic was studied (development options), cross-referencing between categories could be found (see also Figure 1). Though, relying on this assumption is a risk, one should aim to cover categories as well as possible with at least one paper (see Section 5 for further elaboration).

Table 4: Compliance with the SB guidelines

Guideline	Compliance
The papers in start set should not refer each other.	4 out of 5 papers are not referring to each other.
The number of papers must be reasonable. Focused (specific) research areas requires fewer papers than broader research area.	The number of papers depend on the research topic. Considering that the research conducted on this topic is not perceived to be extensive, we believe that the papers in start set were of a good size for SB, i.e. five papers.
The start set should cover several different publishers, years and authors.	The start set covers papers from three different years, two papers have common authors.
The start set ought to be formulated from keywords in the research questions.	The search strings were formulated using keywords in the research questions.

4.2 Efficiency of Snowballing (RQ2)

4.2.1 *Efficiency of identifying start set (RQ2.1)*

In total 90 papers were reviewed, out of which five papers were included in the start set, which resulted in the efficiency of finding a start set as **5.6 %** (5/90). However, a closer examination is required to correctly determine the efficiency. A paper is excluded based of title if:
a: The title is not related to the research topic/questions,
b: The paper is in the grey literature or it is not in English,
c: The paper has already been reviewed.
Exclusion based on title requires substantially less effort compared to abstract, introduction or full text. Hence efficiency is recalculated using the total number of abstracts

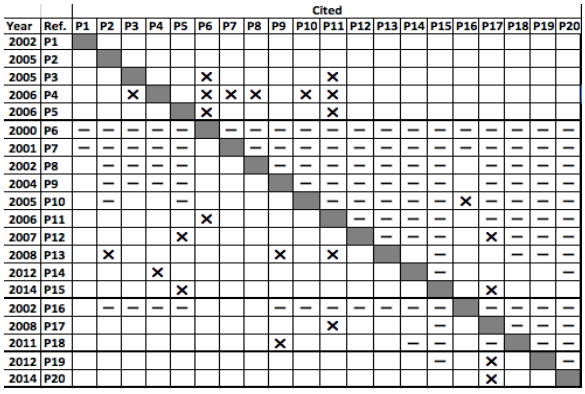

Figure 2: Citation Matrix

and introductions reviewed in Phase 1 and full text reviewed in Phase 2 which is **6.4 %** (5/78).

Overall, the total number of papers to be reviewed was relatively low for the start set when being compared to traditional search-based literature reviews (e.g. [8, 1] both selected from an initial set of over 600 papers). Though, this changes when taking the iterations also into consideration (see the following section).

4.2.2 Efficiency of iterations (RQ2.1)

The efficiency of each iteration is calculated and shown in Table 5.

As SB was continued until no new papers were found, the efficiency of the last iteration will always be zero. The efficiency of the second and third iteration was very low. In total 1044 papers were reviewed in all four iterations out of which 785 decision were based on titles. Thus, similar to the start set, many decision were based on titles. Moreover, the number of papers that were already reviewed increased in the iterations. Therefore, there was a need to revise the efficiency calculation by considering only the number of abstracts and reference contexts reviewed. The revised efficiency based on the number of abstract and reference context reviewed in the iterations is shown in Table 6.

As seen in Table 5 and Table 6, the efficiency drastically improved when papers, where decisions were made based on the title, were removed from the calculations.

4.2.3 Comparison of BSB and FSB (RQ2.1)

In this section BSB and FSB were analyzed to find the number of papers identified in each process, and hence identify the most efficient process. In total 15 papers were identified in the iterations out of which 7 papers were identified through BSB and 8 were identified through FSB, which is quite similar. Also the difference in the total number of papers reviewed in BSB (470) and FSB (574) SB was not large. Hence, in this particular case BSB **1.5%** (7/470) and FSB **1.4%** (8/574) SB were equally efficient.

It is also of interest to study the relevance of papers, where relevance refers to papers that should be considered for inclusion based on the relevance to the topic. In BSB, references not related to the research topic were quite often identified, such as references on research methods or tools used in the research. Also there was a chance of finding a lot of grey literature in the FSB as citations could be any document, such as master theses or project reports. Hence, we looked at the number of duplicates, grey literature and non English papers in BSB and FSB, respectively. The results are shown in Table 7.

The duplication percentage and grey literature count were nearly same. Surprisingly, less grey literature was found in FSB than expected. Hence, in total the noise in BSB was 463 (470-7) and in FBS 566 (574-8). The noise includes grey literature, non-English papers, duplicates and non-relevant papers. Among the considered noise, it is fairly easy to exclude grey literature, non-English papers and duplicates, in comparison to non-relevant papers.

Therefore, we compare how many non-relevant papers (excluding other types of noise as described above) were identified in both BSB and FSB. We already know the number of grey literature, non-English papers and duplicates in our study. The total number of non-relevant papers in BSB was 265 (463-(104+93+1)) and in FSB 309 (566-(119-92-46)). Hence, the percentages of non-relevant papers in relation to the total number of papers reviewed were **56.3%** (265/470) for BSB and **53.8%** (309/574) for FSB. In this case FSB has slightly lower number of non-relevant papers in comparison to BSB.

Hence, neither BSB and FSB is considered to be more or less efficient in this case, both are equally important in finding relevant papers. Given the strategy to find the start set, there was a risk in missing a substantial number of papers if only choosing one of the two SB processes (BSB or FSB). This situation may vary depending on how the start set is chosen, for example if having a highly cited seminal paper the situation may be better than in the case described here.

4.2.4 Reference context as decision base (RQ2.2)

Inclusion and exclusion based on title and abstract was straightforward. In BSB the reference context was reviewed before reviewing the abstract of the paper being evaluated for the simple reason that it is better to use as much information as possible from the current paper before going to the new paper. However, reviewing the reference context alone did not lead to decisions in that many cases. For example, only 16 decisions were based on the reference context. Out of these papers, only one paper was decided to be included, and one was tentatively included, although the latter got excluded based on reading the full text. The remaining 14 papers were excluded based on reference context. The following examples of reference contexts have been formulated in a way that they led to a decision.

- It is easier to exclude than to include using the reference context - For example, in the following reference context *"Detailed discussions on sample selection in this study are reported in..."* it is clear that the paper is a method paper. Hence, it was easy to exclude the paper based on the reference context.

Table 5: Efficiency of each iteration

Iteration	Efficiency
First iteration	5.5 % (10/181)
Second iteration	0.5 % (3/627)
Third iteration	1.0 % (2/188)
Fourth iteration	0.0 % (0/48)

Table 6: Revised efficiency of each iteration

Iteration	Efficiency
First iteration	14.5 % (10/69)
Second iteration	2.1 % (3/140)
Third iteration	4.4 % (2/46)
Fourth iteration	0.0 % (0/4)

Table 7: Comparison on BSB and FSB

Snowballing	Already reviewed	Grey literature	Not English
Backward	22.1% (104/470)	19.8% (93/470)	0.2% (1/470)
Forward	20.7% (119/574)	16.0% (92/574)	8.0% (46/574)

- Only one paper was finally included based on the reference context, which was as follows *"Some studies compared the differences between COTS and OSS products per se and concluded that there is still no empirical evidence that OSS fosters faster system growth and that OSS is more modular than closed source software."*. It was clear from the reference context that OSS and COTS were compared, which was one of the inclusion criteria of the study.

- However, sometimes the reference context can be deceiving. For example, consider the following reference context: *"Pizka reported experience on building the same system with three different strategies, such as wrapping an OSS component, adapting/changing the source code of the OSS component, or building the same component from scratch"*. This looks very promising as it compares OSS and in-house (inclusion criterion). However, the paper is related to Operating System development, which is not within the scope of the study. Thus, the paper was excluded based on reading the full text.

Though, in most cases the reference context did not prove as useful as anticipated, i.e. no decision could be made. This means that the decision could not be made based on the reference context because of the following reasons:

- The reference context might not reflect the goal of the paper - Sometimes the reference context might be something that is of no interest, although the paper actually might have other parts that are interesting. For example, in the following reference context "*Other argued that it is the quality, not the number, of the eyes looking at code that count*". The reference context is about quality, but the papers also compares OSS with COTS, which cannot be seen in the reference context.
- The reference context is vague or unclear - "*Previous studies have looked at using COTS and OSS components in software development.*". In this reference context, the reference is vague. It is not clear what "looked at"means. Sometimes the reference context includes just a keyword which is not very useful to understand. Whenever the reference context is not understandable, the enclosing paragraph was read to understand the context.
- Difficulty in finding reference context - Often different reference styles are used. Some use numbering while some other use author names as a reference index. In both cases it becomes difficult to track the reference context if no ordering is used. For example, the first reference context could be 15 instead of starting with 1 which makes navigation difficult.

To make the reference context more useful for systematic literature studies, it is required that authors describe the references more clearly.

4.2.5 Total efficiency of SB (start set and iteration)

Based on the total number of papers reviewed in start set and iterations, the efficiency becomes **1.8 %** $(5 + 10 + 3 + 2 + 0)/(90 + 181 + 627 + 188 + 48)$

As efficiency is the ratio of noise vs. relevance, we calculate the total amount of noise comprised of grey literature, duplicates, non-English papers and non-relevant papers. In our study the amount of noise in SB was 1114 (1134-20) papers, including the total of grey literature, duplicates and non-Eglish papers, which was 483(195+239+49). Hence, the number of non-relevant papers is calculated as noise - (grey literature + duplicates + non-English papers), which were 631 (1114-483) papers. Hence, the percentage of total noise was **98.23 %** (1114/1134) out of which the percentage of non-relevant papers was **55.64%** (631/1134) and **42.59%** (483/1134) of the total papers consisted of grey literature, duplicates and non-English papers.

It was easy to exclude papers based on grey literature, duplicates and non-English papers in comparison to the exclusion of papers based on non-relevance of the paper. Hence, we recalculate the efficiency based on the number of relevant papers in comparison to the total number of non-relevant papers, which was: **3.17%** (20/631)

As progress was made, more decision were based on titles, the percentages of decisions made based on titles for each iteration was as follows: Iteration 1 = 61.9 %, Iteration 2 = 77.7 %, Iteration 3 = 75.5 %, and Iteration 4 = 91.7 %. Thus in this case, the percentages for exclusion based on titles increased for each iteration.

There were some factors that affected the efficiency of SB such as inclusion/exclusion criteria and data extraction. Clear inclusion and exclusion criteria were a must in SB, otherwise there was a risk to include unwanted papers. The latter will result in conducting SB on unwanted papers that later get excluded. Thus, all papers potentially identified from a paper that later was excluded must also be excluded. Including papers that later were excluded will result in a waste of time and effort.

It was beneficial to extract the data of the included papers before conducting the SB on the new papers identified. Not only did it assure a valid inclusion as detailed reading was done, but also it gave a good idea of the reference context. The inclusion or exclusion based on reference context can be done during extraction.

4.2.6 Efficiency of SB vs. DB search (RQ2.3)

Finally we compared the efficiency of the two searches. The efficiency indicates the amount of noise and relevance in DB search and SB. To calculate, first the total number of papers were considered in the calculation, which was the total efficiency. Then the revised efficiency was calculated based on the total number of abstracts reviewed.

As seen in Table 8 the total efficiency of DB search was more than SB, i.e. in SB more papers were reviewed. The efficiency was also calculated by considering the total number of abstracts reviewed. The percentages shown in Table 8 indicate that, based on the total abstract reviewed SB was more efficient that DB search. Hence, in this study the efficiency of SB is comparable to the efficiency of DB search.

Table 8: Efficiency comparison

Search approach	Total efficiency	Only abstracts
Snowball	1.76% (20/1134)	6.23% (20/321)
Database	3.21% (13/404)	5.70% (13/228)

4.3 Reliability of Snowballing (RQ3)

In this section the two searches are compared on the following aspects -

- Papers identified by both studies
- Development options identified by both studies
- Dimension of research identified by both studies
- Conclusion derived from both studies

In each comparison the number of papers uniquely identified by the SB and DB searches and papers commonly identified by both are presented. This presentation allows us to compare SB and DB search in three ways. 1:all papers identified by each search; 2:unique papers identified by each search; 3:papers commonly identified by both searches.

4.3.1 Common and unique papers (RQ3.1)

As seen in Figure 3, in total 24 papers were identified, out of which 7 were commonly identified by both SB and DB search. 13 papers were uniquely identified by SB and 4 were uniquely identified by DB. In total SB found 20 papers and DB search found 11 papers. The overlap between two searches was more than the papers identified only through DB search (7>4). Snowballing found more papers than DB search, this indicates that SB is more reliable. However we need to investigate how the identified papers support the different development options, which is discussed in the next section.

4.3.2 Common and unique dev. options (RQ3.2)

The main objective of the literature mapping was to identify studies that compare different development options. For this reason it was important to compare the searches based on development options identified. No papers were identified for COTS vs. Outsource and Outsource vs. OSS by both searches, hence they are not mentioned in the comparison.

As shown in Figure 4, considering the unique and common papers identified by both searches, we can see that in total both SB and DB were comparing 5 options each.

Among the 5 options, 3 were commonly identified by both searches, which were In-house vs. COTS, COTS vs. OSS and Make vs. buy vs. share (OSS). This means that two options were uniquely identified by each search. The options In-house vs. OSS and In-house vs. Outsourcing were only identified by DB. It should be noted that for the comparison In-house vs. OSS, SB identified one paper. Though, it was also comparing COTS in addition, hence it was mentioned as a separate comparison i.e In-house vs. COTS vs. OSS. Hence it can be said that In-house vs. Outsourcing was the only option that was not identified by SB. Half (2/4) of the papers found only through the DB search contributed to the In-house vs. Outsource category, for which SB process did not find any papers.

As shown in table 1 three additional search strings with the inclusion of software keyword were used to identify papers comparing outsource option. However, the additional searches did not result in finding new papers. Since DB search was successful in finding the papers comparing the outsource option, it implies that a more extensive search should have been used in finding the papers in start set. All the unique paper found through the DB search were found in Google scholar using the search strings in Table 1, although not in the first 10 search results. Hence, it is clear that more search results should have been reviewed, specially for categories where there were no papers identified.

Options Only OSS and In-house vs. COTS vs. OSS were uniquely identified by SB. It is interesting to note that the search strings used to identify the papers were comparing only two options. However, in SB a paper comparing three options was also identified. Also papers discussing only one option were only found through SB.

Most of the common papers identified (5/7) belong to the In-house vs. COTS category, and 5/6 papers in COTS vs. OSS category were found through SB.

4.3.3 Combined and unique dimensions (RQ3.3)

As seen in Figure 5, both searches have been successful in finding papers in all dimension. Database search used additional keywords such “decision”, “trade-off” and “selection” in the search strings. Snowballing was successful in finding a good number of papers in each dimension, even though additional keywords were not included.

4.3.4 Conclusions consistency (RQ3.4)

In the mapping study for which our SB and DB search procedures were used, conclusions were formed from collective results of the DB search and SB. We investigate whether the same conclusions could be drawn when only SB and only DB search would have been used.

Table 9 presents the results of the investigation, a tick (✓) mark indicates that the results still hold true, whereas a cross mark (×) indicates same conclusion cannot be drawn.

Interestingly papers identified through the SB process had some conflicting results, which were not reported through

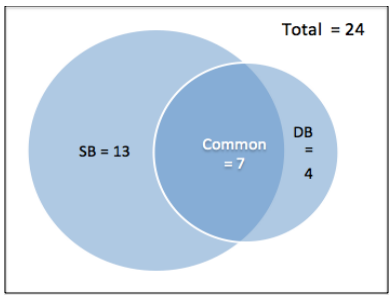

Figure 3: Venn diagram for the overlapping papers

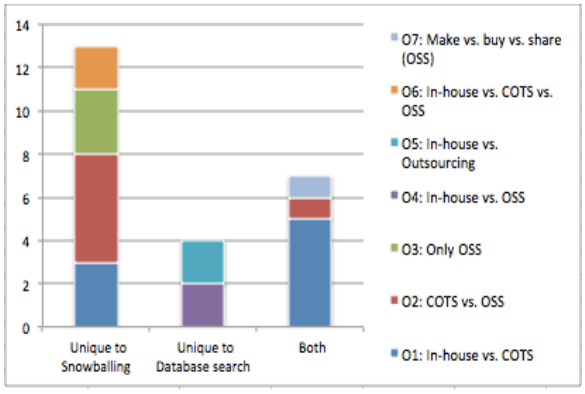

Figure 4: Options Coverage

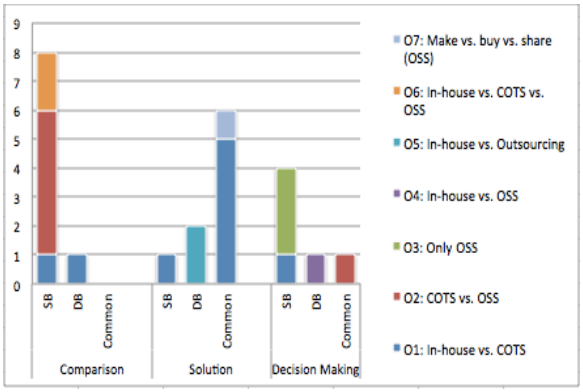

Figure 5: Dimension Coverage

papers identified through DB. The entries 9, 11 and 12 in Table 9 are such conflicting results.

Only a subset of research types proposed by Wieringa [11] were identified by DB, whereas papers identified through SB had covered all research types.

Overall only one conclusion would not hold true if only SB was used and nine conclusions would not hold true if only DB search was used. If more care was taken in the creation of start, then SB seems to be highly reliable in this case.

5. DISCUSSION

The results of this study help to understand some potential factors affecting the efficiency and reliability of SB. Such improvement areas or factors are discussed in this section. Organizing papers in start set into different categories based on the different concepts of the study can help in finding the relevant papers. It is important that each category has at least one paper in the start set. Even though papers in the categories contribute to finding papers in other categories through SB, relying on it to find papers in empty categories did not work. Hence proceeding with empty categories in the start set is considered as a risk of missing some relevant papers. Also clear inclusion/exclusion criteria and extracting data from included papers before the papers are used in SB can improve efficiency.

The efficiencies of SB and DB search techniques do not differ a lot when the total number of abstracts reviewed are considered. However, when the total number of papers reviewed are considered, SB seems less efficient. As the search results might include duplicates, grey literature or non-English papers, reviewing titles does not take much time and effort. Hence, comparing efficiencies based on abstracts seems more reasonable. The BSB and FSB are both equally efficient in this study. Hence, the results of this study support the recommendation suggested in [3] that both BSB and FSB process should be implemented. The efficiency might also be affected by the effort required to review the reference context. In this study only 16 decisions were based on reference context. Therefore, in this case the reference context did not proof to be useful in including or excluding papers. However, the usefulness of the reference context highly depends on how well the reference context is described in the primary studies.

In this study 45.9% of the studies were identified through DB search and 83% of the studies were identified through

Table 9: Conclusion Analysis

No.	Conclusions	Only SB	Only DB
	Options		
1	Majority of primary studies considered in-house development, followed by OSS and COTS.	✓	×
2	Some studies only consider one option in decision making.	✓	×
3	Only two studies consider outsource.	×	✓
	Dimensions		
4	Solutions mainly concentrate on In-house vs. COTS.	✓	✓
5	Comparative studies are mainly focused on COTS vs. OSS.	✓	×
6	In-house vs. COTS vs. OSS is the only option that does not provide any decision criteria.	✓	×
	Decision Criteria Factors		
	Software quality:		
7	5 quality criteria were identified.	✓	✓
	Project metrics:		
8	In-house vs. COTS considered all project metrics with most studies considering cost.	✓	✓
9	Conflicting results were reported with regard to cost.	✓	×
	Context factors		
10	Source code reliability has highest number of studies in COTS vs OSS category.	✓	×
11	Studies disagree on market evolution, ease of use and vendor support is being an issue.	✓	×
12	Studies disagree on whether OSS affects maintainability negatively or positively.	✓	×
	Solutions (5/8 solution papers were commonly found)		
13	Most studies define optimization models.	✓	✓
	Research Types		
14	Most studies are empirical.	✓	✓
15	All research types proposed by Wieringa[11] could be identified in the set of primary studies.	✓	×

SB, out of which 29.1% were overlapping papers. In total SB identified more papers in relation to DB search which is similar to the findings of [2]. However, the SB procedure failed to identify a few important papers. The impact of additional and missing papers identified is stated below.

Impact of additional papers: The papers found in SB provided a deeper investigation on the factors influencing the adoption decision. For example, some studies suggest that OSS affects maintenance positively, while some other studies suggest that OSS affects maintenance negatively. Such conflicting results were not found in papers identified only through the DB search. The DB search strings were more detailed and specific in comparison to SB search strings, yet SB found more relevant papers than the DB search. How-

ever, more general search strings (as those used in the SB procedure, see Table 1) resulted in reviewing a lot of noise , which was 98.23% and the noise in the DB search was 95.17%. This contradicts with the findings of [3] where the noise in SB was relatively less in-spite of the more general search strings.

Impact of missing papers: Snowballing identified more relevant papers and contributed in better analysis. However it failed to identify papers comparing in-house vs. outsource which were identified through DB papers.

6. CONCLUSION

In this study, we evaluated the efficiency and reliability of SB by comparing it with DB search used in a mapping study on choosing among different development options. The results of this evaluation study are summarized in the following research questions.

- *RQ1 How were the relevant papers identified through the evolution of SB process?*
  The start set consisted of 5 papers, out of which 2 papers did not generate any papers on SB them. Half (10/20) of the total papers were identified in the first iteration. The start set was organized in different categories representing different concepts. We found that SB of one paper in one category added papers in other categories. 5/8 papers in one category were added by a paper in other category. Also new categories emerged as the SB progressed.
- *RQ2 How efficient is SB?*
  Although the total number of papers reviewed by SB were more compared to DB, 42.59% of the total papers were either grey literature, duplicates or non-English papers, which did not take much time and effort to exclude. The total efficiency of SB is **3.17%** based on the papers included in relation to the total non-relevant papers (excluding grey literature, duplicates or non-English papers). The difference of efficiency based on the total number of abstracts reviewed by DB search and SB is not too drastic hence, we conclude that in our case the efficiency of SB and DB search is comparable. Also we found that, within SB, BSB and FSB are equally efficient and first iteration was most efficient.
- *RQ3 How reliable is SB in capturing all relevant papers?*
  The DB search identified 45.9% of the papers, SB found 83% of the papers. Snowballing provided a richer analysis, included papers representing all dimension of decision making and all types of research types classified by Wieringa [11] were identified. However, SB failed to find papers in in-house vs. outsource comparison category, whereas DB search was successful. This is a result of leaving the categories in the start set empty. The missing papers were found in Google scholar using the same search strings. However, since only first 10 results from each search string were used, they were not identified. Hence, if more care is taken to find at least one paper in each category then SB would be highly reliable and be considered as a good alternative to DB search.

7. REFERENCES

[1] L. Chen, M. Ali Babar, and N. Ali. Variability management in software product lines: a systematic review. In *Proceedings of the 13th International Software Product Line Conference*, pages 81–90. Carnegie Mellon University, 2009.

[2] T. Greenhalgh and R. Peacock. Effectiveness and efficiency of search methods in systematic reviews of complex evidence: audit of primary sources. *Bmj*, 331(7524):1064–1065, 2005.

[3] S. Jalali and C. Wohlin. Systematic literature studies: database searches vs. backward snowballing. In *Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement*, pages 29–38. ACM, 2012.

[4] B. A. Kitchenham and S. Charters. Guidelines for performing systematic literature reviews in software engineering. Technical report, Technical report, EBSE Technical Report EBSE-2007-01, 2007.

[5] B. A. Kitchenham, T. Dyba, and M. Jorgensen. Evidence-based software engineering. In *Software Engineering, 2004. ICSE 2004. Proceedings. 26th International Conference on*, pages 273–281. IEEE, 2004.

[6] S. MacDonell, M. Shepperd, B. Kitchenham, and E. Mendes. How reliable are systematic reviews in empirical software engineering? *Software Engineering, IEEE Transactions on*, 36(5):676–687, 2010.

[7] K. Petersen and N. B. Ali. Identifying strategies for study selection in systematic reviews and maps. In *Proceedings of the 5th International Symposium on Empirical Software Engineering and Measurement, ESEM 2011, Banff, AB, Canada, September 22-23, 2011*, pages 351–354, 2011.

[8] M. Riaz, E. Mendes, and E. Tempero. A systematic review of software maintainability prediction and metrics. In *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement*, pages 367–377. IEEE, 2009.

[9] M. Skoglund and P. Runeson. Reference-based search strategies in systematic reviews. In *the Proceedings of the 13th International Conference on Evaluation and Assessment in Software Engineering, Durham, England*, 2009.

[10] J. Webster and R. T. Watson. Analyzing the past to prepare for the future: Writing a literature review. *Management Information Systems Quarterly*, 26(2):3, 2002.

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

[12] C. Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *8th International Conference on Evaluation and Assessment in Software Engineering (EASE 2014)*, pages 321–330. ACM, 2014.

[13] C. Wohlin, P. Runeson, P. A. da Mota Silveira Neto, E. Engström, I. do Carmo Machado, and E. S. de Almeida. On the reliability of mapping studies in software engineering. *Journal of Systems and Software*, 86(10):2594–2610, 2013.