Institute of Software Technology

University of Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Bachelorarbeit

Tool Support for Systematic Literature Reviews: Analyzing Existing Solutions and the Potential for Automation

Burak Karakan

Course of Study:	B.Sc. Softwaretechnik
Examiner:	Prof. Dr. Stefan Wagner
Supervisor:	Dr. Justus Bogner
Commenced:	September 5, 2020
Completed:	March 5, 2021

Kurzfassung

Die steigende Popularität des evidenz-basierten und empirischen Software-Engineerings hat dazu geführt, dass systematische Reviews sich in diesem Bereich als eine wichtige Forschungsmethode etabliert haben. Der systematische Charakter dieses Forschungsansatzes ermöglicht es, alle relevanten Beweise in Bezug auf ein bestimmtes Forschungsthema zu identifizieren, zu sammeln und zu interpretieren, wobei auch neue Forschungsaktivitäten zur weiteren Untersuchung durchgeführt werden können. Die Entwicklung eines umfassenden und detaillierten Review-Protokolls, das alle relevanten Methoden, Verfahren und Strategien für den Review dokumentiert, muss daher befolgt werden, um eine mögliche Verzerrung in der Bewertung durch den Forscher zu vermeiden und den wissenschaftlichen Wert der Ergebnisse zu erhöhen. Folglich sind die manuellen Tätigkeiten von systematischen Reviews sehr zeitaufwendig und erfordern einen hohen Arbeitsaufwand.

Angesichts dieser Herausforderungen gibt es ein großes Potenzial für die Automatisierung mit Tool-Unterstützung, um die Effizienz und Zuverlässigkeit bestimmter Schritte zu erhöhen. Bislang gibt es viele Tools, die zur Unterstützung des systematischen Review-Prozesses entwickelt wurden. Allerdings bieten diese Tools in der Regel nur eine begrenzte Unterstützung für den gesamten Prozess an. Da die Menge der existierenden Tools groß ist, ist es für Forscher schwierig, den Überblick über die existierenden Tools zu behalten um einen passenden Kandidaten auszuwählen. Das Ziel dieser Arbeit ist es, existierende Tools zur Unterstützung des systematischen Review-Prozesses zu identifizieren, zu sammeln und zu analysieren, um einen effizienten Überblick über evaluierte Tools zu erhalten, die bestimmte Features abdecken. Da eine ganzheitliche Tool-Unterstützung für den gesamten Prozess zu fehlen scheint, werden typische Praktiken für systematische Reviews im weiteren Kontext der Softwareentwicklung und Herausforderungen, denen Forscher bei der Durchführung eines systematischen Reviews gegenüberstehen, untersucht, um die wichtigsten Bereiche für die Automatisierung zu identifizieren und einen Vorschlag für ein ganzheitliches Tool-Design aus der Kombination bestehender Werkzeuge zu präsentieren.

Um die Ziele zu erreichen, wurden eine Reihe von sinnvollen Forschungsfragen und ein striktes Forschungsprotokoll entwickelt. Um bestehende Tools zu identifizieren und zu sammeln, wurde ein Rapid Review durchgeführt. Die Erfahrungen und Herausforderungen von Software-Engineering-Forschern, die bei der Durchführung von systematischen Übersichtsarbeiten auftreten, wurden durch halbstrukturierte Interviews untersucht. Um eine Reihe von identifizierten Tools zu analysieren, zu vergleichen und zu bewerten, wurde eine Feature Analysis durchgeführt.

Die Ergebnisse des Rapid Reviews zeigen eine zunehmende Verbreitung von Tools zur Unterstützung von systematischen Reviews in der Softwareentwicklung und eine begrenzte Evaluierung dieser Tools. Die Beobachtungen, die von den Teilnehmern durch halbstrukturierte Interviews gemacht wurden, zeigen eine Reihe von herausfordernden und problembezogenen Aktivitäten, die während des systematischen Review-Prozesses auftreten. In Bezug auf diese Aktivitäten wurden von den Teilnehmern verschiedene Anforderungen an Werkzeuge identifiziert, die für die Unterstützung von systematischen Literaturreviews wichtig sind.

Die Ergebnisse der Feature-Analyse zeigen, dass die Unterstützung durch bestehende Tools noch begrenzt ist. Auch wenn die Unterstützung durch bestehende Tools begrenzt ist, gibt es ein großes Potenzial für die Automatisierung bestimmter Aufgaben. Ein besseres Tool zur Unterstützung der systematischen Literaturrecherche wird immer noch benötigt.

Abstract

The rising popularity of evidence-based and empirical software engineering established systematic reviews as an important research method in this field. The systematic nature of this research approach allows to identify, collect and interpret all relevant evidence regarding a specific research topic where also new research activities can be performed for further investigation. Therefore, the development of a comprehensive and detailed review protocol that documents all relevant methods, procedures and strategies for the review has to be followed to reduce the possibility of researcher bias and to increase the scientific rigor of the results. In consequence, the manual activities of systematic reviews are very time-consuming and require a lot of effort.

Regarding these challenges, there is a great potential for automation with tool support to increase the efficiency and reliability of certain steps. To date, there are many tools that have been developed to support the systematic review process. However, these tools generally offer a limited support for the entire process. As the set of existing tools is large, it is difficult for researchers to keep an overview of existing tools and to select a fitting candidate. The aim of this thesis is to identify, collect and analyze existing tools supporting the systematic review process to get an efficient overview of evaluated tools which cover certain features. As a holistic tool support for the complete process seems to be lacking, typical practices for systematic reviews in the broader context of software engineering and challenges that researchers are facing while performing a systematic review are investigated to identify the most important areas for automation and to present a proposal of a holistic tool design suggestion from combination of existing tools.

To achieve these aims, a set of meaningful research questions and a rigid research protocol were developed. In order to identify and collect existing tools, a rapid review was conducted. To investigate the experiences and challenges of software engineering researchers while performing systematic reviews, semi-structured interviews were performed. To analyze, compare and evaluate a set of identified tools, a feature analysis was executed.

The findings of the rapid review show an increasing growth of tools to support systematic reviews in software engineering and a limited evaluation of these tools. The observations made by the participants through semi-structured interviews show a series of challenging and problem related activities that appear during the systematic review process. In relation to these activities, different tool requirements have been identified by the participants which are important for systematic literature review tool support. The results of the feature analysis show that the support of existing tools is still limited.

Even if the tool support of existing tools are limited, there is a great potential for automation of certain tasks and a better tool to support systematic literature review is still needed.

Contents

List of Figures

List of Tables

Acronyms

DASyR Document Analysis System for System Reviews. 59

EBM Evidence-based Medicine. 15

EBSE Evicende-based Software Engineering. 15

GL Grey Literature. 7

MCDA Multi-Criteria Decision Analyses. 40

MDD Model-Driven Development. 74

MLR Multivocal Literature Review. 21

MVC Model View Control. 79

RR Rapid Review. 7

SE Software Engineering. 13

SLR Systematic Literature Review. 7

SMS Systematic Mapping Study. 7

SVM Support Vector Machine. 71

TF-IDF Term Frequency-Inverse Document Frequency. 71

VTM Visual Text Mining. 99

1 Introduction

1.1 Motivation

The Systematic Literature Review (SLR) and the Sysematic Mapping Study (SMS) are evidence-based research methodologies in empirical software engineering which follow a predefined systematic approach to identify, collect and interpret all relevant evidence regarding a specific research topic to answer the prespecified research questions. Due to the systematic nature of these research methodologies that differ from traditional literature reviews, gaps and conflicts can be identified more easily. Furthermore, generalizations can be made in specific research areas where new research activities can be performed for further investigation. Additionally, the inclusion of grey literature in systematic reviews for certain software engineering topics gains more and more attention to remove publication bias and to close the gap between academic research and professional practise. However, most of these systematic approaches are manual activities and therefore require much more effort and time than traditional literature reviews. The reason for this is that the SLR has to follow a well-defined process for each step of the SLR, as it is required to develop a detailed research protocol which documents all important artefacts of the research.

While systematic reviews are predominantly manual activities and require a lot of effort, there is a great potential for automation with tool support to make certain steps more efficient and reliable. To reduce the effort, required to undertake a SLR without reducing the benefits, there are many existing tools that support certain steps of the SLR process. However, SLR specific tools only offer a limited support for the whole process and a holistic tool support for the complete process seems to be lacking. Moreover, as the set of existing tools is large, it is difficult for researchers to keep an overview of these tools and to select the fitting candidate.

1.2 Objective

The main objectives of this thesis are to identify, collect and analyze existing tools supporting the SLR process to get an efficient overview of evaluated tools which cover certain features of the SLR process. In addition, to investigate typical practices for SLR in the broader context of Software Engineering (SE), experiences and challenges that SE researchers are facing while performing a SLR will be analyzed to identify the most important areas for automation. The final artefact of this thesis presents a proposal of a holistic tool design combining existing tools and design suggestions for a tool to support SLR.

1.3 Structure

The thesis starts with an explanation of background information and therefore provides an overview of important terms and methodologies to understand the context of this study.

Afterwards, works related to this thesis are introduced. Consequently, studies associated with challenges and problems encountered during the SLR process and identified tool requirements for SLR tooling are discussed. In addition, a brief overview of publications that analyze and review existing tools to support SLR, are presented.

In Chapter 4, the overall research protocol is described which also includes the used research methods are described. First, a Rapid Review is conducted to identify existing tools to support SLR in SE and to detect the supported SLR stages and important features of these identified tools. The second qualitative research methodology used are semi-structured interviews to comprhend the experiences and challenges that researchers are facing during SLR process. To evaluate existing tools, a feature analysis was performed.

In Chapter 5, the results of the conducted rapid review, the semi-structured interviews and the feature analysis are presented.

Subsequently, the most important areas for the automation of SLR are discussed in Chapter 6 in order to suggest a holistic tool design combining existing tools that can make certain steps more efficient and reliable.

In Chapter 7 the thesis closes with a summary, recommendations and an outlook on possible future work.

2 Background and Fundamentals

This chapter provides an overview of important terms and methodologies which are required as a basis to fully understand this study.

2.1 Evidence-Based Software Engineering

The basis for most research projects is a thorough and appropriate literature review of some sort. Many researchers will follow this process to investigate related and important research within a particular area. Furthermore, a literature review assists to identify gaps in research areas where new research activities can be performed for further investigation. However, unless a literature review is duly performed and the process is not framed by clear systematic procedures for ensuring that the literature is surveyed in an objective manner [BB06], the literature review can fail to provide any scientific value [Kee07].This can lead to research bias and can have the effect that the researcher misses relevant studies for the research. In addition to that, the approach for identifying relevant research suffers from a methodology that is not adequately described in detail and the existing studies mostly have a limited objective assessment [HP14]. For this reason, Kitchenham, Dybå and Jørgensen recommended the adoption of evidence in software engineering and the usage of systematic literature reviews of software engineering literature to support Evicende-based Software Engineering (EBSE) [KDJ04]. Initially, evidence-based research and systematic literature reviews were developed in the medical field, because researchers experienced that expert opinion based on medical advice was not as reliable as advice based on aggregation of results from scientific experiments and research [KPB+09].

Based on Evidence-based Medicine (EBM) to integrate research evidence with clinical expertise and patient values, the goal of EBSE was defined as follows:

> *"to provide a means by which current best evidence from research can be integrated with practical experience and human values in the decision making process regarding the development and maintenance of software"* [KDJ04].

Fundamentally, EBSE aims to fill the gaps between research and practice and provides a process for solving practical problems based on a specific topic or research question. To achieve these goals, EBSE uses five steps to obtain evidence [KDJ04]:

1. Converting a relevant problem and the need for information into an answerable question
2. Searching and tracking existing literature for the best available evidence to answer the question
3. Critically appraising the evidence found for its validity, impact, and applicability
4. Integrating the appraised evidence with practical experience and software engineering expertise

5. Evaluating effectiveness and efficiency and seeking ways to improve it

The first three steps of EBSE as presented above define the synthesis of scientific studies on a research approach or research questions, which can be especially achieved by conducting a SLR. The affinity of performing a SLR has increased significantly and have become increasingly popular and important in empirical software engineering [KPB+09]. Besides, many researchers use this process to collect, summarize and assess existing publications for a certain research topic. Furthermore, it helps to identify gaps in specific research areas where new research activities can be performed and assists to have a better understanding of various tools or techniques.

2.2 Systematic Literature Review

A SLR is an evidence-based research methodology which follows a predefined approach. The process allows to identify, collect, evaluate and interpret all relevant evidence regarding a specific topic to answer the prespecified and formulated research questions in an unbiased and repeatable manner [Kee07]. The well-defined process increases the scientific value of the results in a reliable and objective way. Before a SLR is conducted, the researcher must develop a comprehensive and detailed protocol that documents all relevant methods, procedures and strategies for the review. This approach reduces the possibility of researcher bias that can occur without a precise predefined protocol [Kee07]. If the research methodology is followed in a proper way, the researcher can conduct reliably a SLR to identify relevant gaps in existing research areas, draw generalizations and explore conflicting results of primary studies. [Kee07]. Most of the SLRs can include qualitative as well as quantitative primary studies, while the review itself is a secondary study [Kee07].

Since this increasing improvement of the research methodology in the software engineering field numerous of SLRs have been performed in order to obtain sufficient evidence and benefit from this evidence approach [Kee07]. However, the biggest disadvantage of a SLR is that it requires more effort and time than traditional literature reviews [Kee07]. For conducting a SLR, there are several guidelines. However, the most commonly used and mentioned guideline in literature is authored by Kitchenham [Kee07]. This guideline for performing a SLR involves several discrete activities which can be categorized into three main phases: Planning the review, conducting the review and reporting the review [BKB+07].

Figure 2.1 illustrates the process of performing a SLR. It is important to note that many of the phases involve iteration, even if it looks sequential at the first view. This allows the researcher to adapt and refine the review protocol during all phases in the process, especially in the phases where primary studies have to be selected, data extracted or data synthesized. These elements are prespecified in the review protocol but can be refined after quality criteria are defined [Kee07].

Before researchers start with planning the phase, it is necessary to identify the need for a SLR. The need may depend on the requirements of the researchers, who either want to satisfy any gaps in research or summarize existing literature reviews which are of insufficient quality with the final goal to obtain the required evidence for a specific topic in an objective and scientific way [Kee07]. After identifying the need for a SLR, the process can begin with defining the necessary research question (see **1st** step in Figure 2.1). The research questions drive the entire literature review process and are therefore very important. These research questions design the objectives of the SLR and should be answered in an empirical manner. A common mistake here is to select research questions that

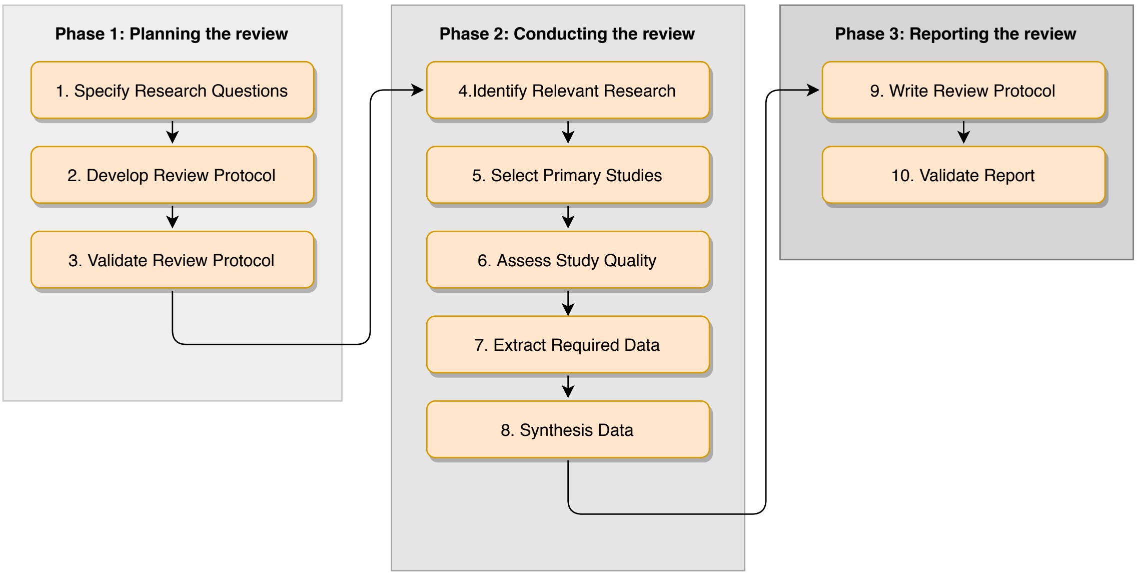

Phase 1: Planning the review
1. Specify Research Questions
2. Develop Review Protocol
3. Validate Review Protocol
Phase 2: Conducting the review
4.Identify Relevant Research
5. Select Primary Studies
6. Assess Study Quality
7. Extract Required Data
8. Synthesis Data
Phase 3: Reporting the review
9. Write Review Protocol
10. Validate Report

Figure 2.1: SLR process suggested by Brereton et al. [BKB+07].

are too broad. This can lead to a significant amount of results where the data is only superficially examined or the topic is not quite specifically covered [XW19]. For this reason, it is important to formulate the research questions as precisely as possible in order to obtain the desired results. Furthermore, the research questions should not be modified after the end of the planning phase unless you want to remove a question because you do not find enough evidence to answer it.

The **2nd** step (Figure 2.1) deals with the development of the research protocol and is comparable to a classic research design in which the research methods are defined and specified to conduct the review systematically. This step is also required and urgently necessary for increasing the quality of the review because it reduces the possibility of researcher bias [Kee07]. The review protocol should describe following elements of the review (defined by Kitchenham) [Kee07]:

- The background which is the rationale for the survey.
- The research questions that the review intends to answer.
- The strategy that will be used to search for primary studies including search terms and resources to be searched. Resources include digital libraries, specific journals, and conference proceedings. An initial mapping study can help to determine an appropriate strategy.
- Study selection criteria. Study selection criteria are used to determine which studies are included in or excluded from a systematic review. They are usually helpful to pilot the selection criteria on a subset of primary studies.
- Study selection procedures. The protocol should describe how the selection criteria will be applied e.g. how many assessors will evaluate each prospective primary study, and how disagreements among assessors will be resolved.
- Study quality assessment checklists and procedures. The researchers should develop quality checklists to assess the individual studies. The purpose of the quality assessment will guide the development of checklists.

- Data extraction strategy. This strategy defines how the information required from each primary study will be obtained. If the data require manipulation or assumptions and inferences to be made, the protocol should specify an appropriate validation process.
- The synthesis of the extracted data. This synthesis defines the synthesis strategy and it should clarify whether or not a formal meta-analysis is intended and if so, what techniques will be used.
- A dissemination strategy (if it is not already included in a commissioning document).
- A project timetable which should define the review schedule.

The **3rd** step (Figure 2.1) in the planning phase is the validation of the review protocol before the execution begins. The validation part also plays an important role, because the protocol is a censorious element and should be evaluated carefully and criticized by experts to spot any gaps in the procedure. If all the necessary steps are taken, the rigor of the research study can be increased tremendously [Kee07; XW19].

Once the protocol has been validated, the review can be conducted. The conducting phase starts with the **4th** step (Figure 2.1) of research identification and generation of the search strategy [Kee07]. The goal of the search phase is to find as many studies as possible which are driven by the research questions to achieve the objectives of the SLR in an unbiased manner [AC19; Kee07]. While searching, it is important to follow a search strategy. The search strategy includes the used electronic databases and digital libraries to receive the important literatures for the study. It is also recommended to use multiple databases, because most of the electronic databases and digital libraries cannot completely fulfil the requirements which are required to get the relevant papers for the study [XW19]. The search strings for the search are formed from keywords which should be derived from the research questions. However, when selecting the correct keywords for the search, it should be considered to choose the right and precise keywords, because keywords that are defined too broadly can retrieve too many irrelevant papers. Using more precise keywords can narrow down the search and improve the precision [XW19]. The documentation of the search strategy is also necessary to make the search and the results transparent and as replicable as possible [Kee07].

In the **5th** step (Figure 2.1), after composition of the search, the researchers can apply the inclusion/exclusion criteria whether the paper should be included or excluded for data extraction and analysis. This step is also called the selection of primary studies and follows a multistage process. The selection of studies should initially be applied generously by reviewing the titles and abstracts. Afterwards the review can be refined by examining the full texts [XW19]. The criteria for inclusion and exclusion should be applied based on the research questions, studies unrelated to the research questions should be excluded [XW19]. In the field of SE the abstracts of research papers often have poor quality and therefore, the conclusion should additionally be reviewed in combination with the abstract [BKB+07]. To reduce researcher bias and to resolve disagreements, the selection process should be performed by multiple researchers [Kee07].

In the **6th** step (Figure 2.1) the selected studies are assessed for their quality. By refining the full-text studies, the quality assessment can identify that these studies are not of sufficient quality. As a result, detailed inclusion and exclusion criteria could be drawn from the quality assessment and can be used to guide the interpretation of the results and determine their strengths [Kee07]. The quality assessments are based on “quality instruments” which can be used in the form of checklists to evaluate each study for quality. It is important to highlight that if a systematic review includes qualitative and quantitative studies, separate quality instruments are required because not all quality questions within the checklist can be applied to both types [Kee07].

The **7th** step (Figure 2.1) describes the data extraction of the selected studies gained from the previous step. As already mentioned in the beginning, data extraction forms should be defined and piloted when the study protocol is defined [Kita]. It is necessary to design the data extraction forms to collect all the needed information and data to answer the research questions. Usually, the extraction form delivers information including [Kee07]:

- The name of the reviewer
- The date of the data extraction
- The title, the authors, the journal and other publication details
- Some space for additional notes

When extracting data, it is important to maintain consistency. therefore it should be performed independently by multiple researchers. After the data extraction of all researchers has been completed, the results can be compared and possible disagreements can be resolved, which can provide more consistent results in an unbiased way. However, if there are a large number of papers to review it is not possible to carry out the data extraction with more than one researcher due to time constraints. For this reason, Brereton et al. [BKB+07] suggest that one reader acts as data extractor and one acts as data checker, assuming that all participants understand the data extraction process [BKB+07].

Once the collection and extraction of data from the included primary studies is completed, the data synthesis follows in the **8th** step (Figure 2.1). The acquired data is assimilated and analyzed in this step to synthesize answers for the defined research questions [Kee07]. Here, several methods can be used to analyze the gained knowledge and mostly depends on the type of primary studies used. The synthesis of qualitative data is called descriptive synthesis. In this case, the data extracted from the primary studies should be visualized in a tabular manner to show heterogeneity or homogeneity of results. The synthesis of quantitative data is called an interpretive review. In this case, the synthesis is referred to as a meta-analysis and has to be statistically analyzed and appropriately visualized [Kee07].

After the data synthesis, the final main phase, the reporting phase will be conducted. In the **9th** step (Figure 2.1), the research team develops the final report. The final report of academic systematic reviews are reported usually as journals and/or conference papers and have a size restriction. To keep the reliability for the SLR, all the processes and steps must be reported in adequate detail. The final report should follow a clear structure that connects the results in key themes, characteristics or subgroups [XW19] to answer the research questions in correct order. In the **10th** step (Figure 2.1), the reported paper will be peer-reviewed by external researchers to assess the quality and validity of the SLR [Kee07].

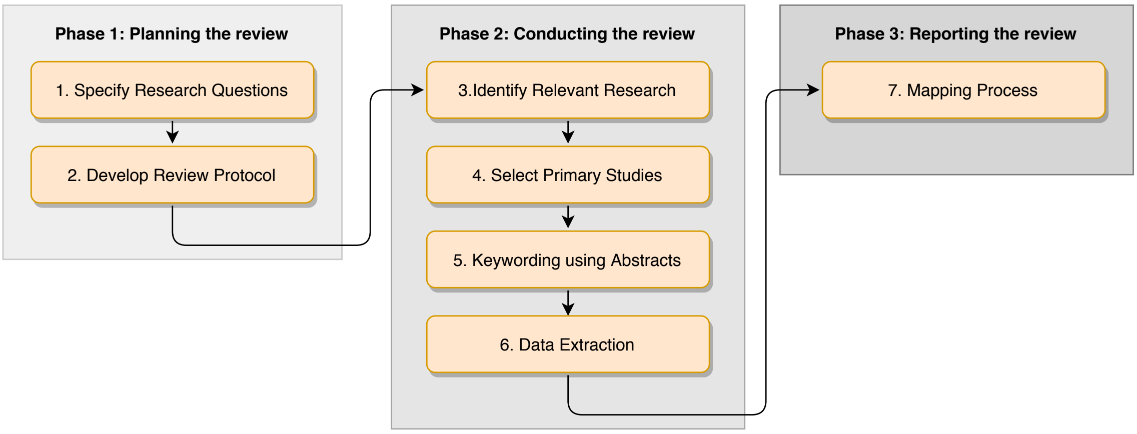

Phase 1: Planning the review
1. Specify Research Questions
2. Develop Review Protocol
Phase 2: Conducting the review
3.Identify Relevant Research
4. Select Primary Studies
5. Keywording using Abstracts
6. Data Extraction
Phase 3: Reporting the review
7. Mapping Process

Figure 2.2: The SMS process [PFMM08].

2.3 Systematic Mapping Study

The other secondary study method that has gotten much attention in SE is the Systematic Mapping Study (SMS). In this section, the process of a SMS will be introduced and the main differences between a SLR and SMS will be shown.

The SMS is a more “open” form of SLR with the aim of categorizing and mapping published research reports and results [PFMM08]. While the main focus of a SMS is on classification, performing thematic analysis and identifying publication flora, the SLR mainly focuses on establishing the state of empirical evidence. In general, the SMS is much broader and the identified papers are not analyzed in detail and for this reason the evidence cannot be enriched. But both study types share the aim of identifying research gaps even though the SMS does not provide evidence [PFMM08]. The SMS process is relatively similar compared to the SLR process and differs merely in conducting the review in the **2nd** phase (Figure 2.2).

After selecting the relevant studies with consideration of the inclusion/exclusion criteria , the classification process follows in the **5th** step (Figure 2.2). This step is also called the keywording of abstracts. When the abstracts of included papers are of poor quality, the introduction or conclusion can also be reviewed for the keywording. Here, the important elements and keywords from the papers, which are relevant in the context of answering the research questions are classified to develop a comprehensive understanding about the conducted research. This classification and categorization of the keywords can then be used for clustering and mapping after the data extraction in the **6th** step (Figure 2.2) [PFMM08]. Analysis of the extracted data through categorization can then be served to identify gaps and opportunities for future research.

Because the goal of a SMS is not to establish the state of evidence, the papers are not evaluated and validated based on their quality, whereas the assessment of quality in SLR is a major focus, which increases the depth and the effort [PFMM08]. The phases in the SLR process, such as the **3rd** step validating the protocol (Figure 2.1), the **6th** step assessing the quality (Figure 2.1) and the **10th** step validating the report (Figure 2.1) are not considered in the SMS process. However, this can lead to judgment errors, because the papers are not evaluated in such detail. Additionally, a SMS is often conducted only based on the abstract which can pilot lacking important information. If more parts of papers are considered the more effort is required. A SMS that intensely examines selected studies can become more like a SLR [PFMM08].

Both review methods differ in terms of goals, depth and in some steps of the process but the structure is essentially the same. Therefore, a combination of both methods is also possible. A SMS can be performed first, to get an overview of the topic area and then a SLR can be conducted to get the state of evidence [PFMM08].

2.4 Snowballing

Snowballing is a procedure which can additionally be adapted to the search strategy for all types of systematic reviews to identify more new papers that can be relevant for the research. There are two different types of snowballing. Forward snowballing identifies new papers based on those papers citing the paper being examined [Woh14]. This involves examining each candidate who cites the paper under investigation. To investigate the cited papers, Google Scholar is often used because Google Scholar directly provides the most important information of the cited papers. If the information is not sufficient for the selection, the paper can be studied in detail to make a decision. Backward snowballing uses the reference list of the inspected paper to identify new papers that could be included [Woh14]. The papers listed in the reference list are examined one after the other and according to the defined criteria it is decided whether the paper is included or not. A more detailed investigation is required here too, if not enough information has been obtained from the reference list. For snowballing, it is also important to select a suitable starting set of papers to be examined in order to add as many relevant new papers as possible. Snowballing is performed iteratively and can include several iterations depending on the requirements [Woh14]. It is important to note that only papers which have been already included after the initial search, are examined in the snowballing procedure. The snowballing procedure is a good opportunity to find missing studies that are relevant. Sometimes it is difficult to define the best possible search strings and when the search strings for the initial search are formulated too broadly, important papers for the research can be missing. [KPB+10]. The main disadvantage of snowballing is that it can be very time-consuming to examine the cited papers and the reference lists, especially when multiple iterations are considered. The snowballing procedure is principally used to ensure the best possible coverage of literature and is not an alternative to database searches [Woh14].

2.5 Grey Literature Reviews

GL (Grey Literature) is a non-peer reviewed informative material that is not available using commercial academic libraries [YFW+20]. It includes preprints, white papers, technical reports, lectures, data sets, audio/video media, blogs, and forum entries that can be found on the internet. The inclusion of grey literature in systematic reviews receive more and more attention in the field of SE. It helps to remove publication bias while it gathers available evidence regarding a certain topic [YFW+20]. A systematic review which includes academic as well as GL is also called as Multivocal Literature Review (MLR) [GFM19]. Inclusion of GL can give essential benefits in certain areas of SE. One key benefit is to close the gap between academic research and professional practice, especially in terms of real world needs in industrial environments [GFM19], because several important types of knowledge and experiences can be missed out without the inclusion of GL, e.g., software testing, automation approaches used in the industry and challenges that developer's are facing. Failure to include such important information can have profound implications and steer research in the wrong direction [GFM16].

However, the inclusion of GL in SE is strongly controversial because studies from GL can be of poor quality which can lead to the fact that the holistic research itself also ends up being of a poorer quality [KBT+09]. Due to this fact, the inclusion of grey literature should be viewed with caution to provide reliability of the results and to avoid bias. This can become challenging with studies from GL as opposed to studies from scientific literature, especially during the quality assessment [NSEF19]. Therefore, the approaches to quality assessment for GL should be defined in a meaningful way. Some strategies to this approach are suggested by A.Yasin et al. [YFW+20].

3 Related Work

This chapter presents an overview of work related to this thesis. The focus is set on scientific publications that are relevant in the context of tool support within the SLR methodology. The chapter is divided into two sections. In Section 3.1, scientific publications are introduced which are associated with challenges and problems confronted during the SLR process. Thereby, publications by Kitchenham and Brereton [KB13], Carver et al. [CHHK13] and Brereton et al. [BKB+07] will be introduced in greater detail. Those publications discuss the challenges, problems of performing SLR, the most difficult and time-consuming phases of the SLR process and their proposals to improve the process, are discussed. Regarding the search process of SLR, the most recent SLR and survey by Al-Zubidy and Carver [AC19] will be presented, where the problems and solutions reported by SLR authors are extracted to deliver tool requirements related to the search process of the SLR process. From the observations made in regard to the challenges and problems during the SLR process, one can say that most SLR authors discuss the same general problems, and because of this, they deliver redundant information. Due to that, the SLR problems will be discussed and summarized. Supplementary, in Section 3.1, the identified tool requirements for SLR tooling will be outlined. Therefore, not only the SLR and survey by Al-Zubidy and Carver [AC19], but also the scientific publication by Hassler et al. [HCHA16] determining high priority SLR features through a community workshop as well as the article by Al-Zubidy et al. [Al-17], detailing desired SLR tool support requirements from the SLR community that are not covered well by existing tools, will be reviewed.

Section 3.2 presents a brief overview of publications that analyze, review and evaluate existing tools to support SLR. Accordingly, the research by Marshall et al. [MBK14] and the review by Al-Zubidy and Carver [AC14] are introduced. The aim of the research by Marshall et al. was to evaluate SLR tools that provide support for the whole process through a feature analysis. The publication by Al-Zubidy and Carver reviews a set of tools and outlines the features and limitations of these tools.

3.1 Problems During SLR Process and Tool Requirements

The paper by Brereton et al. [BKB+07] reports the experiences and practices of applying a SLR, by reviewing a number of performed SLR's to identify challenges and problems during the SLR process. One of the observations made by Brereton et al. was that the number of false positive papers (papers that seemed relevant first but on detailed reviewing have been excluded from the set of papers) in the initial search stage were extremely high [BKB+07]. In addition, Brereton et al. found out that specifying the research questions is the most critical element of a SLR and has to be treated with attention. For this reason, Brereton et al. determined that the data extractors have to be familiar with the member who defined the data extraction forms. If this criterua is not respected, it can cause misunderstandings [BKB+07]. The last critical issue found by Brereton et al. was that different search engines are organized with completely different models, making it impossible to use the same search term for all databases [BKB+07].

Furthermore, Kitchenham and Brereton [KB13] undertook a SLR to report and summarize the experiences of performing SLR's made by SE researchers. The three most significant problems located by Kitchenham and Brereton [KB13] that appear during the SLR process are described as follows:

- The time and effort needed for SLR. Additionally, the time taken for protocol construction.
- Digital libraries in SE are not well-suited to complex automated searches and are lacking in standardized keywords.
- The problem of quality assessment of papers based on different research methods.

The first problem mentioned above is related to the overall protocol development, while the second problem is connected with the search process, which was also mentioned in the paper by Brereton et al. [BKB+07]. The third problem is related to the study selection process. Moreover, Kitchenham and Brereton [KB13] identified several more issues, that were mentioned in publications between 2005 and 2007 and that differ from the problems mentioned in publications in or after 2008. The issues found are the following:

- Difficulty of defining research questions
- The need for domain knowledge
- The need for tools to support SLR

These issues identified by Kitchenham and Brereton [KB13] can be confirmed with the observations made by Brereton et al. [BKB+07].

The objectives of the performed survey by Carver et al. [CHHK13] were to identify the most difficult and time-consuming phases of the SLR process. The most difficult aspects of the SLR process that are identified in the survey are:

- Extracting data
- Assessing quality
- Study selection

The survey showed that the data extracting stage was the most difficult, followed by assessing the quality and the study selection [CHHK13]. Regarding the most time-consuming phases of the SLR process, Carver et al. identified the following results:

- Extracting data
- Study delection
- Searching in databases
- Assessing quality

The results showed that the data extraction and the study selection were the most time-consuming phases, followed by the search in multiple databases and the assessment of the quality. Concerning aspects that are most needed for tool support, Figure 3.1 shows the identified results:

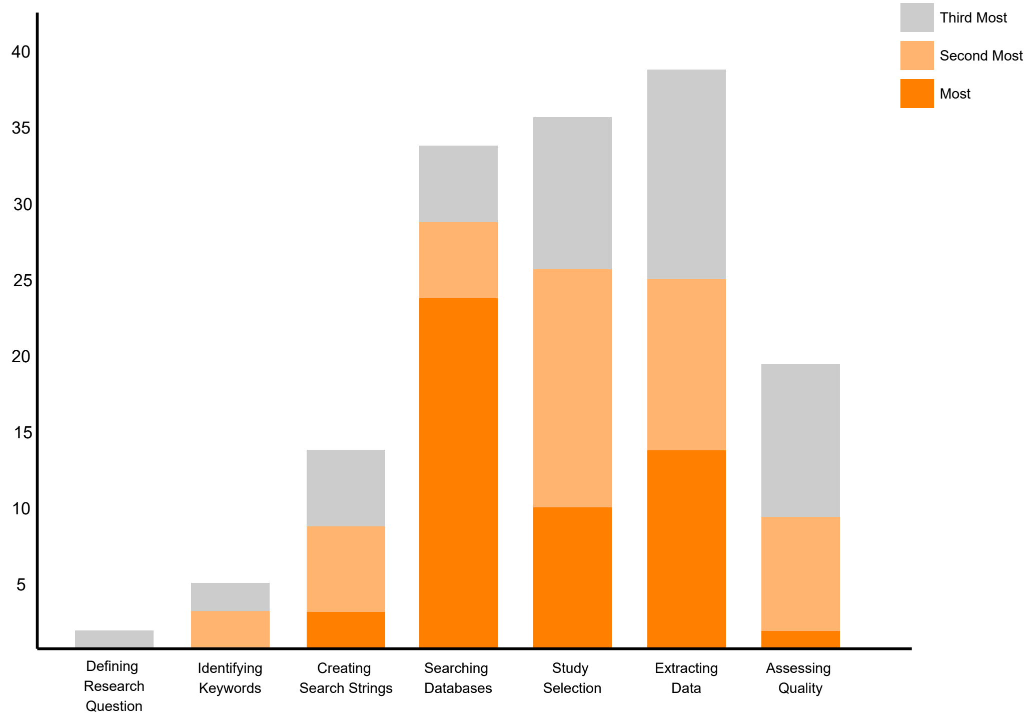


Figure 3.1: Most needed aspects for tool support identified by Carver et al. [CHHK13].

Searching in multiple databases was the most needed for tool support, followed by study selection and data extraction. Moreover, the respondents of the performed survey affirmed that the main problems were the inconsistencies in input format, behavior and usability of the several databases [CHHK13]. In addition, the paper by Carver et al. reports about suggestions for novices, which are listed below:

- Research questions must be scoped properly and expert feedback can guide improvements.
- Tailor the search strategy and plan to manipulate the search strings for each used database.
- Collaboration and expert review for each stage.

The performed SLR and survey by Al-Zubidy and Carver [AC19] determines the most important facets of the search process with respect to the problems, possible solutions and suggestions, search strategies, identified tool requirements and tools that were used by SLR authors during the SLR search process. Due to this, a SLR was performed to identify problems and solutions were reported by SLR researchers during the search process. The conducted survey also served to identify the problems and solutions from the respondents and to investigate tool requirements, which SLR researchers would like to have in a SLR search tool.

Based on the SLR, 84 studies were analyzed to identify 29 unique problems. The results of the SLR showed that 72% of the problems are encountered during the conducting phase of the SLR process. Moreover, 85% of the problems can be traced back to the digital libraries [AC19]. This limitation of the digital libraries was also mentioned in the scientific publications by Kitchenham and Brereton [KB13] and Brereton et al. [BKB+07].

Depending on the results from the SLR and survey, Al-Zubidy and Carver [AC19] conducted a mapping to compare the top 10 problems identified by the SLR and by the survey. The mapping of the problems is represented in Figure 3.2.

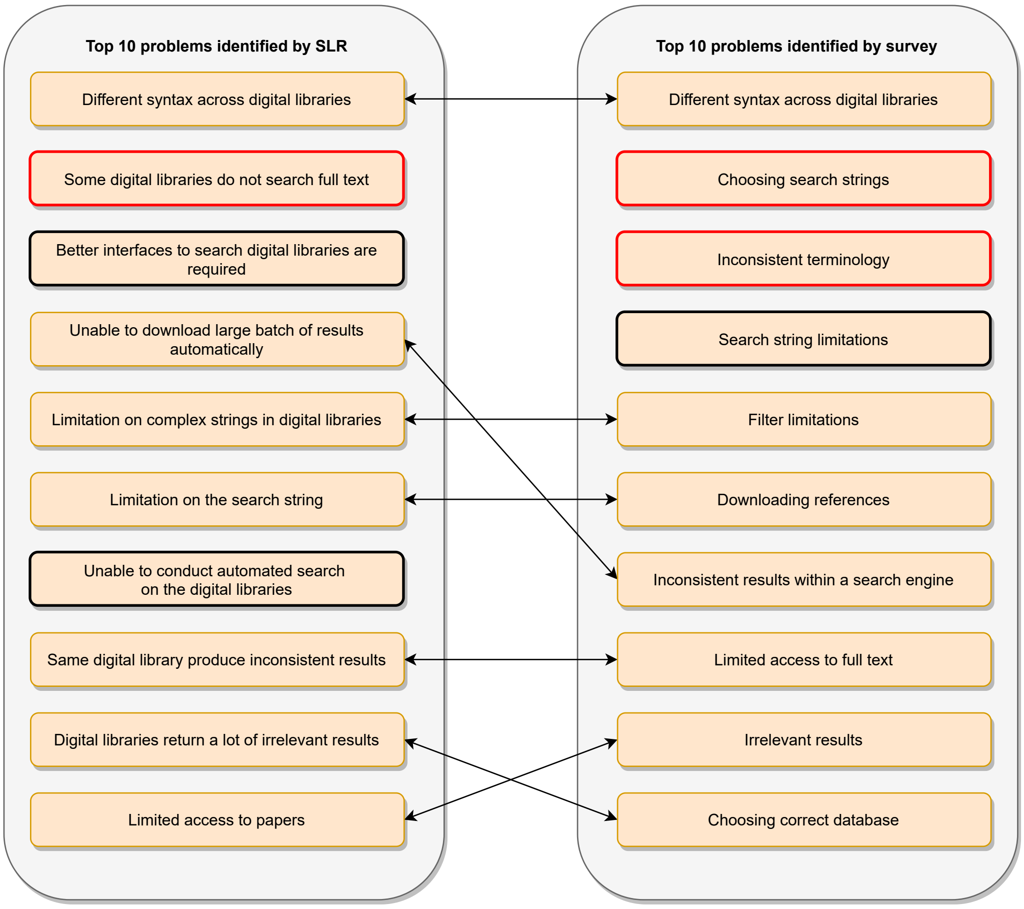

Top 10 problems identified by SLR
Different syntax across digital libraries
Some digital libraries do not search full text
Better interfaces to search digital libraries are required
Unable to download large batch of results automatically
Limitation on complex strings in digital libraries
Limitation on the search string
Unable to conduct automated search on the digital libraries
Same digital library produce inconsistent results
Digital libraries return a lot of irrelevant results
Limited access to papers
Top 10 problems identified by survey
Different syntax across digital libraries
Choosing search strings
Inconsistent terminology
Search string limitations
Filter limitations
Downloading references
Inconsistent results within a search engine
Limited access to full text
Irrelevant results
Choosing correct database

Figure 3.2: Mapping of the problems by Al-Zubidy and Carver [AC19].

This mapping of problems serves to determine whether the problems reported in the SLR do still occur and whether the respondents of the survey are confronted with new problems which were not stated in the literature. In Figure 3.2, the left column shows the ten most mentioned problems identified from the SLR and the right column shows the ten most mentioned problems gained from the survey. Problems that are connected from the problems identified in the SLR with the problems reported in the survey are mapped onto each other. However, the mapping of the last three problems does not make any sense. Unfortunately, Al-Zubidy and Carver do not explain how these mappings arose. Therefore, a mapping of the problem “same digital library produces inconsistent results” identified by the SLR with the problem “inconsistent results within a search engine” identified by the survey, a mapping of “digital libraries return a lot of irrelevant results” with “irrelevant results” and a mapping of “limited access to papers” with “limited access to full text” would be more logical. Problems that are black-rimmed are problems that could be matched to lower ranked problem onto literature problems or survey problems. The red-rimmed aspects are problems that do not match with any other problem and thereby are identified as new. The identified new problems which do not match with any of the other problems in the other set are “Some digital libraries do not search full text”, “Choosing search strings” and “Inconsistent terminology” [AC19].

Moreover, Al-Zubidy and Carver [AC19] identified methods to address them. The common approaches, which are formalized more as workarounds and suggestions to solve these problems, are described below [AC19] by also discussing the weaknesses of these methods.

- Iterative Search: Reform the search string iteratively until a sufficient search string is found. The problem of this method is the non-systematic nature which can lead to inadequate search results.
- Alternative Sources: The use of alternative digital libraries is suggested if the conventional source has limitations. However, this method can lead to relevant publications being excluded from the conventional digital library.
- Different Search Strings: Manipulate the search strings for each database if the databases use different formats and search rules. However, these suggestions is also non-systematic because different search terms were used for each database.
- Tool: The use of tools can help to bridge some of these problems. The problem with this is the fact that the documentation of the way the tool was used often misses.

Due to this, none of the described approaches can address the problems without reducing the benefits provided by the methodology.

Requirement
Search in multiple databases in standardized query
Removing duplicates
Filtering capabilities
Merging results from the digital libraries
Synonym recommendation
Repository of papers
Standardized export
Downloading papers
Automatic determination of relevance
Support collaboration
Automatic generation of search terms
Papers categorization

Table 3.1: SLR search phase tool requirements [AC19].

Based on the survey results and the addressed problems, Al-Zubidy and Carver [AC19] asked the respondents for desired tool requirements within the search phase. In total, 167 potential requirements were identified which were then grouped into 38 categories. The top identified requirements are depicted in Table 3.1.

The publication by Hassler et al. [HCHA16] determines high priority SLR features for individual steps as well as also the whole SLR process through a community workshop. The participants of the workshop rated the importance of each feature. The results showed that the highest ranked tool features were integrated search, followed by collaboration and traceability [HCHA16]. Moreover, Hassler et al. made the observation that these features are not well-supported by current tools. With regard to the identified requirements by Al-Zubidy and Carver [AC19] which contain equal tool requirements as well as reported in the paper by Hasser et al., it can be said that the search process still suffers from the lacking tool support.

Besides, the publication by Al-Zubidy et al. [Al-17] used interactions with the SLR community to identify and prioritize SLR tool requirements. The top ten requirements identified by Al-Zubidy et al. [Al-17] and Hassler et al. [HCHA16] can be summarized, as depicted in Table 3.2.

Top 10 Requirements
Integrated search
Collaboration support
Traceability
Support inclusion and exclusion
Data maintenance
Automated analysis
Support quality assessment
Support text-mining
Visualization
Coding of methods and data
Storage of studies

Table 3.2: Tool requirements identified by Al-Zubidy et al. [Al-17] and Hassler et al. [HCHA16].

Collaboration is one of the most important aspects for the SLR process, as Al-Zubidy and Carver [AC19] and Hassler et al. [HCHA16] obtained in their studies. Therefore, the use of collaborative work in SLR tools can help to avoid researcher bias.

3.2 Tools to Support SLR in SE

Marshall and Brereton [MB13] have conducted a mapping study to identify tools that automate specific parts of the SLR process. Therefore, tools with text-mining and visualization techniques were located and the most targeted stage of the identified tools was the study selection stage of the SLR process.

Based on the identified requirements in the conducted survey by Al-Zubidy and Carver [AC19], another aspect to investigate was to identify tools that were used by the respondents. The most reported tools were: Spreadsheet (Excel), Digital libraries, Jabref, Mendeley, StArt, Endnote and RefWorks [AC19].

Additionally, Al-Zubidy and Carver evaluated existing SLR tools with regard to the identified requirements. An evaluation of SE specific SLR tools was also conducted by Marshall et al. [MBK14]. While Marshall et al. evaluated the support for the complete SLR process, Al-Zubidy and Carver only evaluated the tools in regard to the search requirements which were identified in Section 3.1. Al-Zubidy and Carver reported on a paper where additional SLR tools were reviewed with regards to supported features and limitations of each tool [AC14]. All studies combined, these following tools supporting SLR were reviewed and evaluated: StArt [12], SLuRp [BHB12], SLR-Tool [10], SESRA [MB15], DBPedia [BLK+09], PEx [FNF+10], ReVis [FNMM14].

However, SLR-specific tools only offer very limited support for the whole SLR process and due to this such specifically designed SLR tools were used less than other general tools like, Excel [AC19]. Furthermore, there are also many other tools which were developed specifically for the SLR approach but an evaluation for these specific tools is needed. As a last point, the existing tools seem to be evaluated and reviewed largely against the existing literature, which makes the potential of the tools and their capabilities relatively speculative. Due to this, an accurate analysis of already existing and the latest tools to support SLR are still lacking.

4 Research Methodology

The following chapter will focus on the developed overall research protocol and the chosen methods. Moreover, the study designs of the chosen methodologies are described.

4.1 Research Protocol

In this section, the overall research context is presented with a compact overview of the overall research process and the research questions are defined.

4.1.1 Research Context

As already mentioned in Section 1.2, the objectives of the overall study are:

1. Identify, collect, analyze and categorize existing tool support for SLR, that support a part or all of the process steps.
2. Analyze typical practices for SLR in the broader context of SE and the challenges and problems that researchers are facing while conducting a SLR manually or with tool support.
3. Identify the most important areas for automation and the potential for automation to make certain steps more efficient and reliable.
4. Proposal of a holistic tool chain, from combination of existing tools and design suggestion for holistic tool support.

In a compact form, the goals can be summarized as follows:

Analyze the ***SLR*** *process and existing* ***tool support***
for the purpose of identifying common practices, open challenges, and potential solutions
with respect to reducing manual efforts via ***automation***
from the perspective of ***SE researchers***

4.1.2 Research Questions

To achieve these goals and objectives, the following research questions will be answered:

RQ 1) What tools are available to support the systematic review process in SE research?

RQ 1.1) Which stages of the systematic review process do the tools support?

RQ 1.2) What are the most important features of these tools?

RQ 1.3) To what degree are current SLR practices in SE research supported by existing tools?

RQ 2) What are experiences of SE researchers with respect to tool support in the SLR process?

RQ 2.1) What effort-related challenges do SE researchers face for the SLR process?

RQ 2.2) In which challenges do SE researchers see the most potential for automation?

RQ 2.3) Which tools do SE researchers use in the SLR process?

RQ 2.4) What requirements are important for SE researchers in general SLR tool support?

RQ 3) Based on the results of **RQ 1)** and **RQ 2)**, how could a holistic tool chain combining existing tools for the whole SLR process look like?

4.1.3 Research Process

A Rapid Review (RR) and semi-structured interviews are the two methodologies used in order to answer the research questions. Therefore, the research process is divided into two phases:

- State of art (RR)
- State of practice (Semi-Structured Interviews)

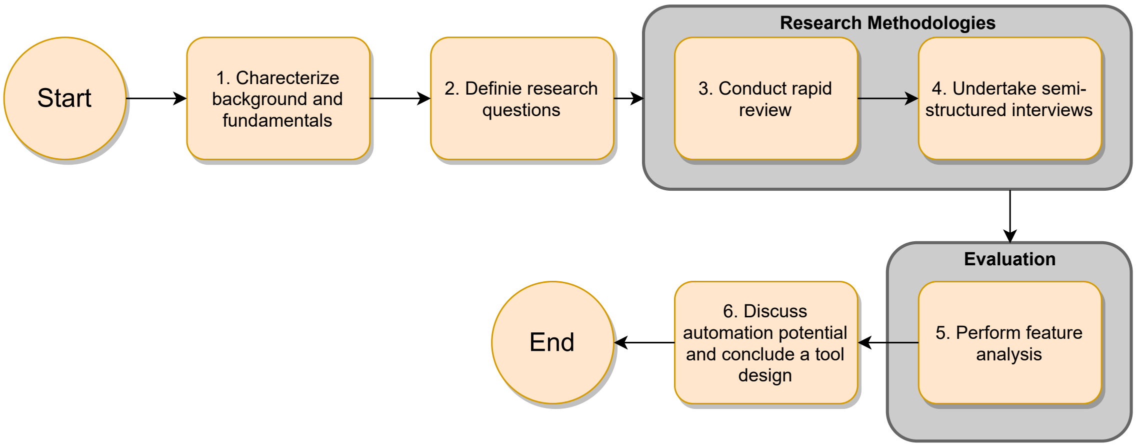

Start
1. Charecterize background and fundamentals
2. Definie research questions
Research Methodologies
3. Conduct rapid review
4. Undertake semi-structured interviews
Evaluation
5. Perform feature analysis
6. Discuss automation potential and conclude a tool design
End

Figure 4.1: Overall research process.

Based on the results of the RR and the interviews, the aim is subsequently to establish and evaluate the degree of SLR practices in SE research supported by existing tools through a feature analysis, which relates to **RQ 1.3**. Figure 4.1 shows the complete research process that is performed in this thesis.

4.2 Rapid Review

RR is a lightweight secondary study with the aim of providing evidence in a timelier manner [CPS18]. The RR especially focuses on decision-making processes towards solutions, challenges and problems that software engineers are facing in practice [CPS20]. The RR methodology is relatively similar to the SLR method. The main differences are that the RR does not consider the quality appraisals to reduce costs and to save time, and is reported through an alternative medium, like Evidence Briefings, which is mostly limited to one to three pages, instead of traditional research papers [CPS20]. It is important to point out that the RR also follows a systematic protocol. To develop this protocol the guideline by B. Cartaxo et al. was used [CPS20]. Therefore, the main goals of the conducted RR are:

- Identify, collect and categorize existing tools to support SLR in SE.
- Detect the supported SLR stages of the identified tools.
- Find out the most important features of these tools.

4.2.1 Study Design

The aim of RR is specially to answer the research questions RQ 1, RQ 1.1 and RQ 1.2, which were developed in the research protocol (see Section 4.1). Accordingly, to achieve these goals, a study design was developed. The study design includes the developed search strategy, containing consistent search terms used for several databases, the selection procedure with the defined inclusion and exclusion criteria and the extraction procedure. After completing these steps, the results are presented to answer the proposed research questions (see Section 5.1). (The categorized results can be inspected in appendix)

Search Strategy

For the search process, keywords and search strings were used. In order to find out the relevant studies for the RR, different digital libraries were explored which are the following:

- Google Scholar
- ACM Digital Library
- IEEE Xplore

To find further tool support that is not only available in scientific publications, but also are mentioned in grey literature, the following databases are used additionally:

- Google Search

- SLR Toolbox

The SLR Toolbox[1] provides access to a large set of SLR tools and allows reviewers to identify appropriate tools to support their systematic reviews based on their needs. For this purpose, it is a good opportunity to use the toolbox as an additional search engine.

Before conducting the search, the following search string was defined for Google Scholar, ACM Digital Library and IEEE Xplore:

> (tool OR tools OR "tool support" OR tooling OR support OR supporting OR approach) AND ("systematic literature review" OR "systematic review" OR "systematic literature reviews" OR "systematic reviews" OR "systematic mapping study" OR "mapping study" OR SLR) AND ("software engineering" OR SE)

As the number of results found is too large for Google Scholar and ACM Digital Library, the following limits are set:

- the Google Scholar search will be limited to the first 50 pages. Google Scholar shows 10 results per page, so 500 results are examined.
- the ACM search will be limited to the first 10 pages. ACM shows 50 results per page, so 500 results are examined.

For the search with Google, five search strings will be used. In order to refine the search and find literature that does not appear in the academic databases, the databases ACM Digital Library and IEEE Xplore are excluded from the Google search using the `- site` command. Duplicates that have already been added in the previous searches will be excluded and not considered in the Google search. The search will be limited to the first 10 pages for each search string. Google shows 10 results per page, so 500 results are examined. The five search strings used for Google are:

> **1.** tool support "literature review" "software engineering"
> -site:https://ieeexplore.ieee.org/ -site:https://dl.acm.org/
>
> **2.** tool support "systematic review" "software engineering"
> -site:https://ieeexplore.ieee.org/ -site:https://dl.acm.org/
>
> **3.** tool support "literature study" "software engineering"
> -site:https://ieeexplore.ieee.org/ -site:https://dl.acm.org/
>
> **4.** tool support "literature survey" "software engineering"
> -site:https://ieeexplore.ieee.org/ -site:https://dl.acm.org/
>
> **5.** approach "systematic literature review" "software engineering"
> -site:https://ieeexplore.ieee.org/ -site:https://dl.acm.org/

[1] http://systematicreviewtools.com/

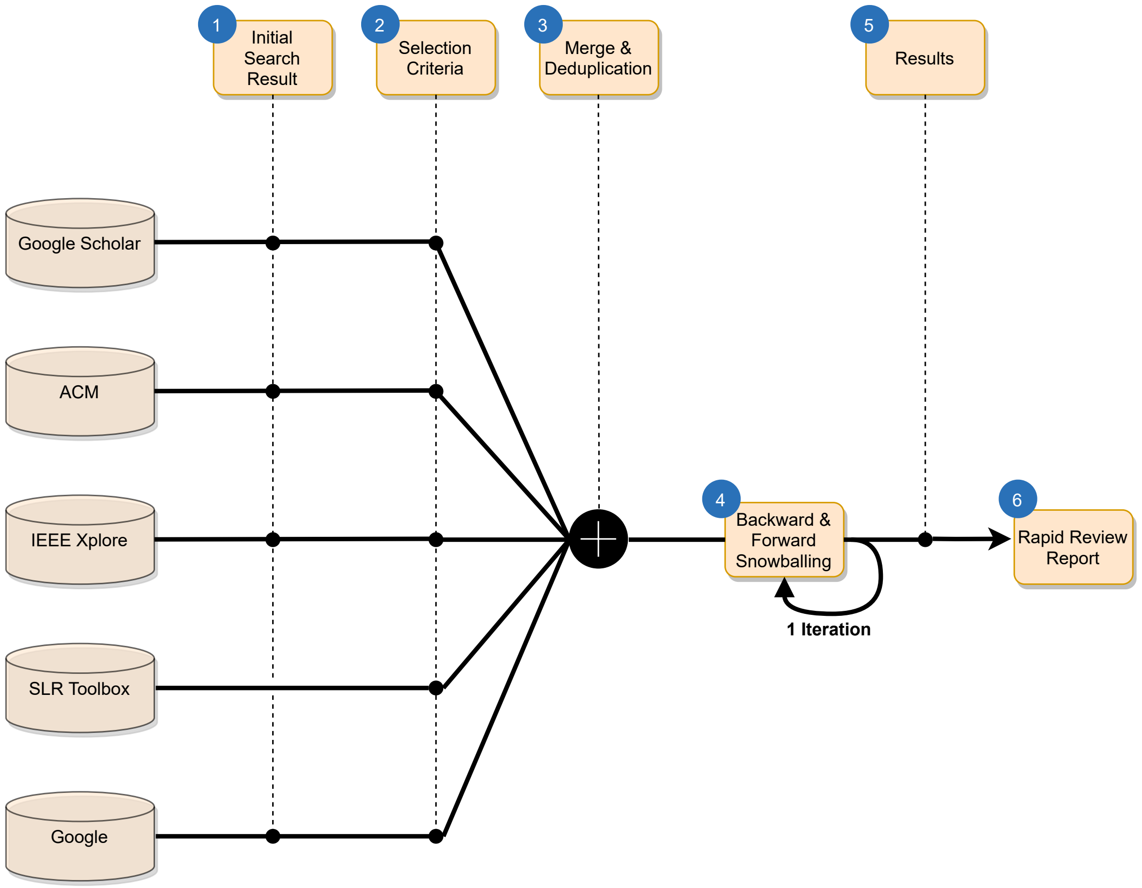


Figure 4.2: Selection procedure of RR.

Figure 4.2 shows the selection procedure of the undertaken RR. In the first and second phase, the results from the initial search were assessed for inclusion by investigating the title and abstract of the papers. Afterwards, papers were assessed against the inclusion/exclusion criteria using the full text. After completing the selection criteria, the papers were merged and the duplicates removed. The use of snowballing was also intended as a part of the search strategy. After identifying a set of papers and evaluating them for the selection criteria, the included papers were entered to the backward and forward snowballing procedure. To ensure that the RR is carried out in a timely manner, only one iteration was performed. For the citation search, Google Scholar was used. The snowballing procedure was only executed for the white literature.

Selection Procedure

The selection procedure was based on the following inclusion/exclusion criteria.

Inclusion Criteria:

- must discuss tool support or a specific tool for systematic literature studies (SLR, MS, etc.)
    - tools that can support at least one stage of the SLR/MS process
    - the paper can report on any stage of development of the tool (i.e. proposal, prototype, functional, tool demo etc.)
- not only primary studies but all papers in which tools are mentioned can be included (systematic literature reviews, mapping studies, grey literature etc.)
- only publications from 2007 to August 2020

Exclusion Criteria

- papers that are not written in English
- studies not related to the research questions
- tools not useable for the software engineering discipline
- reference and document manager tools are not considered

Extraction Procedure

The following data was extracted from each paper:

- tool name, short description of the tool and tool characteristics
- type of approach underlying the tool
- the particular stage of the SLR process that the tool supports

In the last step the results were reported through an Evidence Briefing. The briefing document presents the main findings of the research and the extracted data. The Evidence Briefing can be found in the Zenodo repository[2]. The results of the RR are presented in Section 5.1.

[2] https://doi.org/10.5281/zenodo.4584710

4.3 Semi-Structured Interviews

As a second qualitative research methodology, semi-structured interviews were undertaken. The main goals of the semi-structured interviews are:

1. General experiences, challenges and problems that researchers are facing while performing SLR in SE.
2. Report the strategies and best practices from conducting SLR.
3. Explore the main strengths and weaknesses of the used tools to support SLR.
4. Identify features and requirements that researchers would find most important for SLR tool support.

To achieve these goals, a less structured approach is the best way, to investigate the experiences of SE researchers qualitatively. The interview protocol and the design were established by using the guidelines from Carolyn B. Seaman [Sea99] and Shull et al. [SSS08]. The interview design contains the elementary aspects that are important for conducting the interviews and the overall interview procedure.

4.3.1 Interview Design

As the focus of the interviews is to get deeper knowledge about experiences, challenges and problems that researchers are facing while performing SLR's and to investigate quantitative, as well as qualitative information from the participants, the deployment of semi-structured interviews is most suitable. In contrast to traditional interviews, semi-structured interviews include a combination of open, closed questions and additional unexpected questions, which can be asked to the interviewee, whenever required [SSS08]. Closed questions are questions that may provide a predefined set of response from which the respondent can choose or can be answered with a single word or phrase [Int05]. With open questions, the participants are free to respond in a flexible and unlimited way. These types of questions enable a fluid conversation between the interviewer and the interviewee and therefore allows to get into more depth to gain qualitative data from the participants.

Research Questions

In Section 4.3, the objectives of the semi-structured interview are described. These objectives are related to the research questions of RQ 2 that were developed in the research protocol (see Section 4.1). With the use of semi-structured interviews, the following research questions will be answered:

- What experiences do SE researchers have with respect to tool support in the SLR process?
- In which challenges do SE researchers see the most potential for automation?
- Which tools do SE researchers use in the SLR process?
- What requirements are important for SE researchers in general SLR tool support?

Interview Instruments

After defining the main objectives for conducting semi-structured interviews as the initial step, the interview guide was developed (see Figure 4.3). The interview guide was used as a help to organize and direct the interviews. Therefore, it consists of a list of questions that can be asked during the interviews and contains notes to direct them under different circumstances. Questions driving the interviews were grouped into five main categories:

Group 1: Domain Context and Professional Background

Group 2: Personal Experiences Performing Systematic Literature Reviews

Group 3: Experiences with Tool Support

Group 4: Important Features of SLR Tools

The first group (Group 1) focused on the demographic and the professional background of the participants. Therefore, it contained a series of open questions related to the domain context and professional background. Moreover, questions were asked to discover the use and role of SLR's within their domain. Group 2 focused on the personal experiences of conducting a SLR. This group of questions included open as well as closed questions. So, the first step was to find out how many SLR's were performed by the participant, when it was last carried out, how many researchers were involved and which types of literature studies they have performed, has been experienced. To learn more about the experiences, main challenges and problems during specific phases of the SLR, most manual elaborating, error-prone activities and solutions for mentioned challenges, open questions were asked. In addition, experiences about including grey literature or using the snowballing procedure were also asked. Group 3 emphasized the use and experiences of the interviewee with tool support for SLR. Accordingly, the aim was to identify tools that were used by the participants to assist or perform the SLR. Furthermore, if a tool was used, the strengths and weaknesses of the tools were realized.

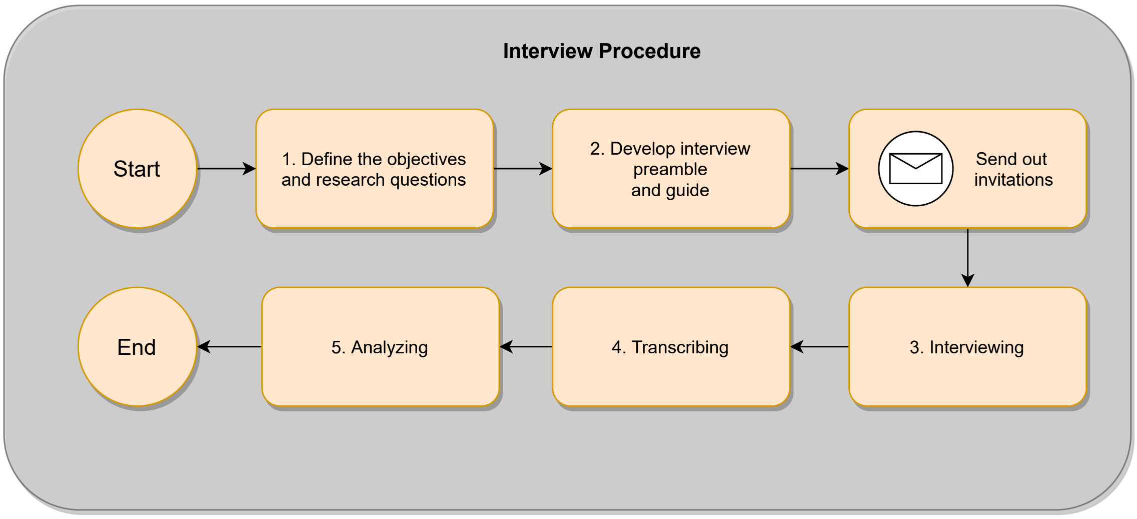


Figure 4.3: Interview procedure.

Questions driving the last part of the interviews are questions related to important features and requirements which a tool need to support SLR. Therefore, questions were asked to perceive important requirements and desired features for SLR. If the participant did not name enough requirements or did not know any, a pre-defined list with functional and non-functional requirements was shown to choose and order the most important requirements. The list of requirements were generated based on the requirements of SE researcher for specific phases of the SLR or for the overall tool support. Therefore, the studies by E. Hassler et al. and A. Al-Zubidy et al. were observed. The aim of these studies was to identify and prioritize the necessary SLR tool features and requirements based on the input from the SLR community [AC19; HCHA16]. Moreover, the second meeting of the International Collaboration for Automation of Systematic Reviews (ICASR) was used as a measure, where the challenges for the automation identified by participants and the proposed principles, which a tool needs to support SLR, were discussed [OTG+18]. The overall interview procedure is depicted in Figure 4.3.

The semi-structured interviews were conducted in December 2020. The selected participants to interview were software engineering researchers with knowledge and experiences with any kind of SLR's. An email invitation was sent to the potential participants. Prior to the interview, each participant was additionally sent an Interview Preamble. This document summarizes the main purposes of the study that were covered during the interview, the interview process with the expected duration, and measures to treat the interviews confidential and anonym. Additionally, the conducted interviews were recorded. After finishing the interviewing part, the recordings were transcribed and converted manually into text files. In accordance to the research questions, the required data were extracted from the transcripted files.

All relevant interview artefacts can be found in this repository[3].

Coding Process

The coding procedure was realized based on the qualitative analysis methods described by Lethbridge et al. [LSS05] and Seaman [Sea99]. Some of the collected data from the semi-structured interviews were coded and grouped into labels to answer the research questions. The attached codes were developed and categorized based on the grouped questions which were used to drive the interviews. In according to the collected data, the developed codes therefore were grouped into labels and sub-labels which were assigned to a specific coding theme. Consequently, these set of coding themes were used which then were grouped into constant labels and sub-labels:

- Main problems of conducting a SLR
- Activities that require most manual effort
- Activities that are most error-prone
- Experiences with SLR tools
- SLR tool features

[3] http://doi.org/10.5281/zenodo.4480371

First, specific text passages deduced from the transcripts were categorized into consistent sub-labels which specify the required content for a certain coding theme. After that, the sub-labels were generalized into matched labels. The coding process for the qualitative analysis was performed iteratively, i.e., a section of text was firstly read through it once, then specific text passages were assigned into labels and as a last step, the sections were read through it again to make sure that the codes and labels were used consistently. Figure 4.4 the overall thematic analysis related to RQ 2. The results from the analyzing part are presented in Section 5.2.

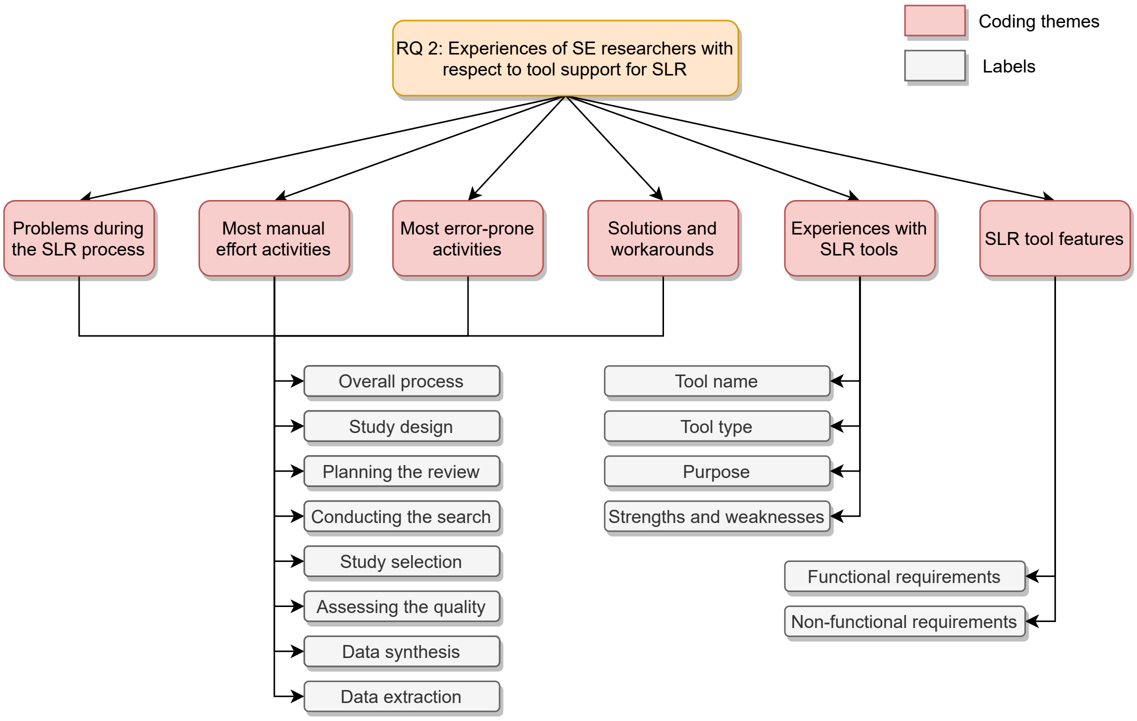


Figure 4.4: Thematic analysis related to RQ 2.

4.4 Feature Analysis

The feature analysis aims to analyze, compare and evaluate the selected tools that support specific steps or the entire process of the SLR. The undertaken feature analysis assumes a form of a Multi-Criteria Decision Analyses (MCDA) activity [TD12]. The first step of this study, is to develop a evaluation framework for the selected tools to support the SLR in SE. Therefore, a set of features will be created, which such a tool for SLR should include. After generating the features and weighting the importance for the defined features, the tools are evaluated against each feature and as a last step, the tools are discussed in terms of their strengths and limitations.

As a basis for the performed feature analysis, the work of C. Marshall et. al was related [MBK14]. The aim of the research was to evaluate and compare a set of candidate tools which is similar to this study. The differences of the research by C. Marshall et al. are that the research is evaluated to tools which only provide the whole systematic review process. In addition, only a small amount of tools were selected and the defined set of features are different.

4.4.1 Analysis Method

This subsection describes the analysis method that is used to evaluate the selected candidates tools for the feature analysis, which takes a form of a MCDA. To understand the evaluation and analysis approach, a explanation of the DESMET methodology [KLL] and the feature analysis method is followed.

Evaluation of Tools

The selection of available software tools is hardly manageable. The increasing size and complexity make it difficult for end users to find the best tool for their needs. In order to keep the quality and functionality of these tools manageable, software evaluations can be performed. Therefore, software evaluation is defined as the problem of determining the extent to which a software product satisfies a set of requirements [Duj]. To achieve software quality, software tools must satisfy a number of demanding requirements. Hence, it is important to develop a suitable software evaluation method and to define desired and effective combinations of features and requirements in order to achieve the desired software quality.

In case of this study, the software evaluation can be seen as a MCDA problem. The MCDA helps to make decisions about available alternatives that are characterized by certain attributes [JS09]. The additional goals of MCDA based on the evaluation of tools to support SLR are:

- to help researcher to find the best fitting tool for their needs
- to highlight candidates that are good
- to help rank the candidates in decreasing order of covered requirements

Moreover, the MCDA method promises a structured and transparent approach to get an overview of alternative tools through a clear consideration of the different criteria that need to be defined [TD12]. The main aspects of the MCDA method are:

- the candidate tools to be appraised
- the criteria, attributes or features against which the selected tools are appraised
- scores that reflect the value of a tool expected performance on the criteria
- criteria weights that measure the relative importance of each criteria

DESMET Methodology

DESMET is a comprehensive multi-component methodology for evaluating software engineering methods and tools [KLL]. It identifies a method of evaluation and a set of criteria to help evaluators in selecting the most appropriate method based on their needs. Moreover, it is very difficult for researchers to keep an overview of tools which support SLR and to make a decision which tool to support SLRs in SE is most fitted for their needs. Therefore, the DESMET methodology is best suited in this case. The first stage of DESMET methodology is to select an evaluation method. The nine different evaluation methods described by Kitchenham et al. [KLL] are the following:

Evaluation type	Description
Quantitative experiments	An investigation of the quantitative impact of methods/tools organized as a formal experiment.
Quantitative case studies	An investigation of the quantitative impact of methods/tools organized as a case study.
Quantitative survey	An investigation of the quantitative impact of methods/tools organized as a survey.
Feature Analysis- screening mode	A feature-based evaluation done by a single individual (or cohesive group) who not only determines the features to be assessed and their rating scale but also does the assessment. For initial screening, the evaluations are usually based on literature describing the software method/tools rather than actual use of the methods/tools.
Feature Analysis- case study	A feature-based evaluation done by a group of potential users who are expected to try out the method/tools on typical tasks before making their evaluations.
Feature Analysis- experiment	A feature-based evaluation performed by staff who have used the method/tool on a real project.
Feature Analysis- survey	A feature-based evaluation done by people who have had experience of using the method/tool or have studied the method/tool. The difference between a survey and an experiment is that participation in a survey is at the discretion of the subject
Quantitative effects analysis	A subjective assessment of the quantitative effect of methods and tools, based on expert opinion.
Benchmarking	A process of running a number of standard tests using alternative tools/methods (usually tools) and assessing the relative performance of the tools against those tests.

Table 4.1: DESMET methodology evaluation types by Kitchenham et al. [KLL].

As one of the objectives is to analyze existing tools to support SLR, an evaluation in a qualitative form is most suitable in this case.

Feature Analysis

As an evaluation in a qualitative form is most suitable in this case, a qualitative feature analysis including a scoring process will be performed. One of the advantages of the feature analysis is that it can be carried out fast and cost-effectively. However, a subjective evaluation of a single person can lead to subjective bias [JS09; Kitb]. The main objective of the evaluation is to compare tools which support a subset of features or steps for SLR, keep an overview of tools to support SLR and to provide input into a decision about whether a researcher should select the tool and which tool is most suitable for their needs. According to Kitchenham following areas should be covered in the feature analysis [Kitb]:

1. **Appropriateness** - Which stages of the SLR does the tool support and is the provided support appropriated?
2. **Functional requirements** - Does the tool offer reliable functional features?
3. **Non-functional requirements** - Does the tool offer reliable non-functional features?
4. **Strengths and weaknesses** - What are the main strengths and weaknesses of these tools?

Feature sets were generated based on the requirements that SE researcher have for specific phases of the SLR or for the overall tool support. These features have already been generated in Chapter 4. Therefore, as elemental indicators to create the features, of course the results of the rapid review and semi-structured interviews were used.

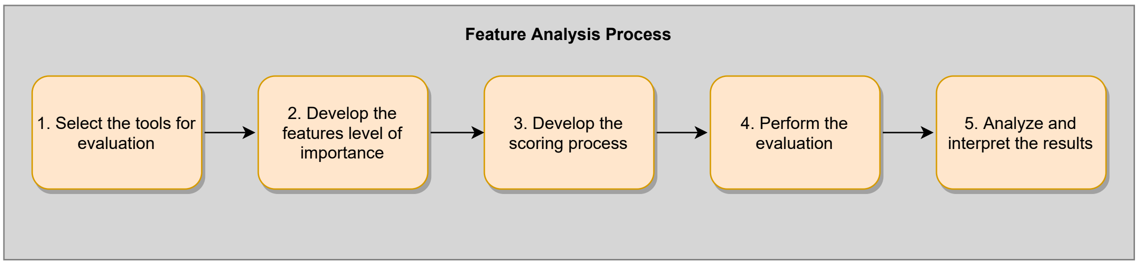

Feature Analysis Process
1. Select the tools for evaluation
2. Develop the features level of importance
3. Develop the scoring process
4. Perform the evaluation
5. Analyze and interpret the results

Figure 4.5: The feature analysis process.

The main stages of the feature analysis are defined as follows:

The **1.** step (Figure 4.5) was to select a set of candidate tools for the evaluation. The identification of existing tools to support SLR was done by the Rapid Review. However, only the tools that were executable were considered in the feature analysis. The **2.** step (Figure 4.5) was to generate a set of required features. Most of features were already generated in the Rapid Review and were also used in the semi-structured interviews to ask the participants, which features were most important. After generating the features, the participant ratings from the interviews for functional and non-functional requirements were used to prioritize these requirements. In **3. step** (Figure 4.5), the importance of the features, the interpretation of the evaluation and the overall weighted scoring system were developed to score each tool against all defined requirements. After defining the formal aspects, the assessment was carried out to determine how well the defined requirements are covered in the feature sets for each tool (**4.** step Figure 4.5). As a last step of the feature analysis in **5.** step (Figure 4.5), the analysis, interpretation and presentation of the results were completed. The selected candidates, developed features, the scoring system and the results of the evaluation are introduced in Section 5.3.

5 Results

In this chapter the results from RR, semi-structured interviews and feature analysis are presented.

5.1 Rapid Review (RQ 1)

While conducting the RR, strict attention was paid to which papers, tools and web resources (GL) were included or excluded during the specific steps. Figure 5.1 shows the exact number of papers, tools and GL that were left after each step of the research protocol. The initial stage of the search contains a total amount of 1391 results distributed over the four search engines, Google Scholar, ACM Digital Library, IEEE Xplore and Google. The additional used search engine SLR Toolbox registers a total amount of 209 tools that support various tasks within the literature review process for several disciplines. However, as the SLR Toolbox provides a good user interface with filter options to find appropriate tools that meet the defined requirements, it was not considered in the initial stage.

After the initial stage and assessing the selection criteria in the selection process, 11 papers, (containing 13 tools) were included from Google Scholar, eight papers from ACM Digital Library (containing 10 tools), six papers from IEEE Xplore (containing five tools), two papers and 5 GL's (containing three tools) from Google. From the additional search engine SLR Toolbox, 12 papers and eight GL's (containing 14 tools) were included. After completion of the selection criteria, the papers were merged, the duplicates removed, snowballing was performed for the included papers. Through one iteration of the snowballing procedure, eight additional papers were included. In total, 35 papers and 13 GL's were received that report about tools to support SLR in SE and 24 tools were extracted from the included results.

Figure 5.2 shows the distribution of included tools from each of the used search engines. Tools found through multiple sources have also been added into the set of included papers. An interesting finding here is that most of the tools were added from SLR Toolbox, followed by Google Scholar and ACM Digital Library. The lowest number of tools were provided by IEEE Xplore and Google. Figure 5.3 displays the distribution of all included papers over the years. The diagram shows that there had been varying release of papers over the years. The lowest point with zero release can be found in 2008. After 2011, there was an increase of released papers, peaking in 2012 and 2018. The rise of releases in these years might be due to the fact that new approaches and techniques were developed to support specific tasks for the SLR process, as especially the machine learning and active learning methods increased in these years.

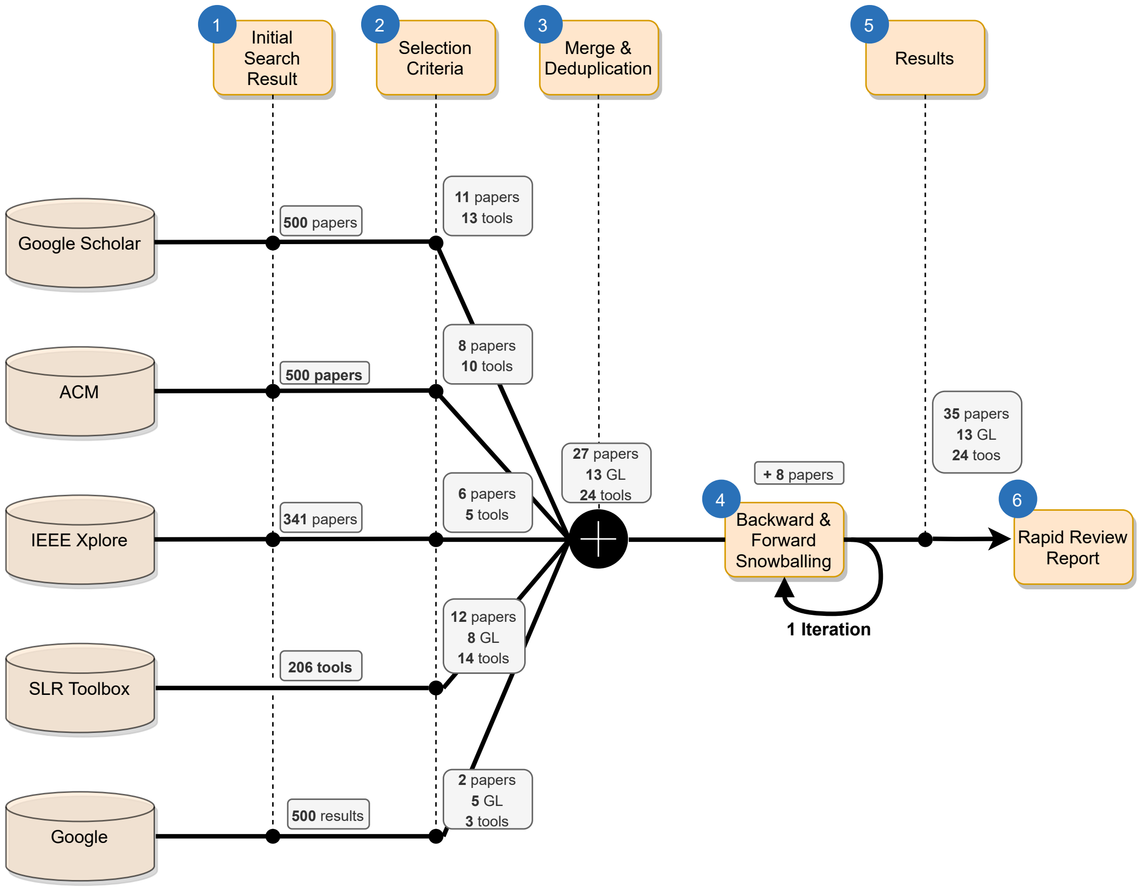


Figure 5.1: Selection procedure of RR with included papers, tools and GL.

5.1.1 Available SLR Tools (RQ 1)

In order to answer the RQ 1, Table 5.1 shows the identified tools to support SLR. The most mentioned tool in the reported publication is StArt which is reported in seven papers, followed by PEx, ReVis and SLuRp which are reported respectively in five papers. The tool DBPedia is reported in four papers and SLR-Tool in three papers. ReviewBuilder, REviewER and Sysrev are the only tools that are not reported in any papers and are found based on the grey literature or SLR Toolbox. Table 5.2 gives a brief overview of the tools with some tool characteristics, supported stages and underlying approaches. The full overview with more characteristics can be found in this repository[1]. Tools that support the whole process are Buhos, revtools, ReLis, SESRA, SLR-Tool, SLRTOOL, StArt and Thoth.

[1] http://doi.org/10.5281/zenodo.4475689

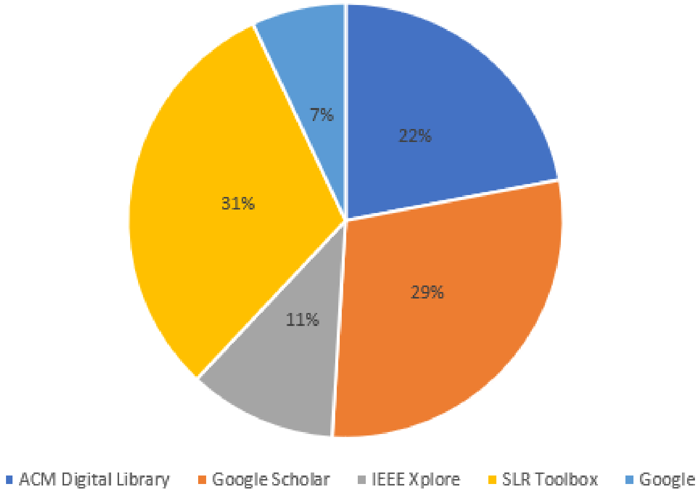


Figure 5.2: Distribution of the included tools from each search engine.

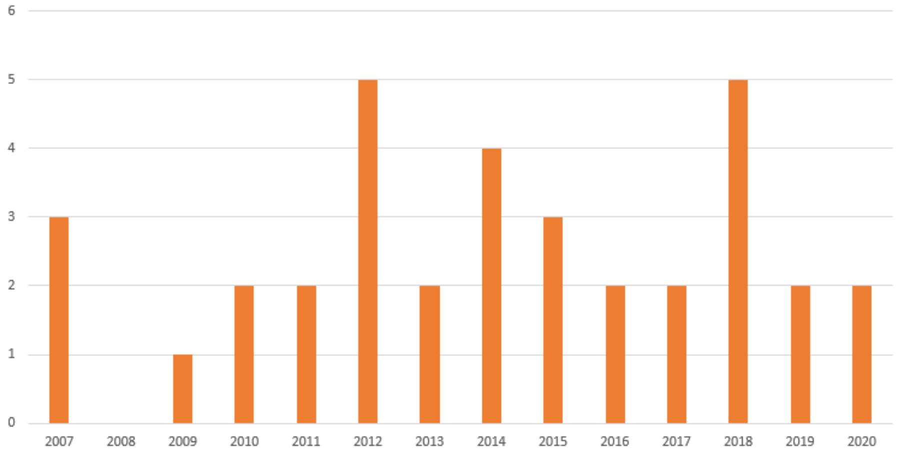


Figure 5.3: Overall distribution of included publications per year.

5.1.2 Supported Stages and Important Features (RQ 1.1 & RQ 1.2)

Figure 5.4 illustrates the distribution of the supported phases by the tools. More than half of the identified tools support the conducting phase of the SLR process. The distribution of the tools that support the planning and reporting phase is roughly the same and thus covers approximately the other half of the overall distribution.

Tools that are addressing the conducting phase, are the most commonly targeted on the selection process. Therefore, the features abstract/text screening and the inclusion and exclusion procedure of the selection process are supported by most of the identified tools. Moreover, 5 tools provide an integrated search, namely Parsifal, ReviewBuilder, REviewER SLRONT and SLuRp. Buhos, revtools and SESRA support de-duplication of papers in the selection stage. In addition, Buhos is

[2] Pl: Planning phase, Co: Conducting phase, Re: Reporting phase

#	Tool	Reported Papers
T21	StArt	7
T20	SLuRp	5
T12	ReVis	5
T8	PEx	5
T4	DBPedia	4
T17	SLR-Tool	3
T5	FASTREAD	2
T13	SESRA	2
T14	Site Content Analzyer	2
T15	SLRONT	2
T18	SLRTOOL	2
T1	ASReview	1
T2	Buhos	1
T3	DASyR	1
T6	revtools	1
T7	Parsif.al	1
T9	ReLis	1
T16	SLR-Tool by Andreas Hinderks	1
T19	SLR.qub	1
T22	SurVis	1
T24	Thoth	1
T10	ReviewBuilder	0
T11	REviewER	0
T23	SysRev	0

Table 5.1: Identified tools to support SLR.

also the only tool that supports the snowballing procedure. Data extraction and synthesis approaches are the second most commonly targeted stages in the conducting phase. Additionally, 14 tools were found which provide collaboration as an additional tool feature. A full overview of supported stages and important features of the tools is available in this repository[3].

Figure 5.5 addresses the underlying approaches supported by the tools. The biggest clusters of tools are based on methods for visualization and text-mining. 12 tools are based on the approach of data maintenance. This means that these tools additionally support storage of references and can be used as a supplemental reference manager. The third largest cluster, which is supported by four tools, presents tools with machine learning and active learning approaches.

The RR which is related to RQ 1 has provided insight into the current state of research regarding tools to support SLR in SE. In summary, the results show that there are some tools that support certain phases of SLR and a significant growth of tools to support the SLR process, which is recognizable over the years. Especially the study selection phase is supported by most of the tools.

[3] http://doi.org/10.5281/zenodo.4475701

Tool	**Development stack**	**Web-based**	**Supported stages**[2]	**Underlying approach**
ASReview	Python	No	Co	Active Learning, Data Maintenance
Buhos	Ruby	Both	Pl-Co-Re	Text-Mining, Visualization, Data Maintenance
DASyR	N/A	No	Co	Active Learning
DBPedia	Scala	No	Co	Text-Mining
FASTREAD	Python	No	Co	Active Learning
revtools	R package	No	Co-Re	Text-Mining, Visualization
Parsif.al	Python	Yes	Pl-Co-Re	Text-Mining, Visualization
PEx	Java	No	Co-Re	Text-Mining, Visualization, Data Maintenance
ReLis	PHP	Yes	Co-Re	Visualization, Data Maintenance
ReviewBuilder	Pyhton	N/A	Co	Visualization
REviewER	Java	No	Co	Text-Mining, Visualization
ReVis	N/A	No	Co	Text-Mining, Visualization
SESRA	Javascript	Yes	Pl-Co-Re	Data Maintenance, Visualization, Data Maintenance
Site Content Analyzer	N/A	No	Re	Text-Mining
SLRONT	N/A	No	Co	Text-Mining
SLR-Tool by A. Hinderks	Javascript	Yes	Co	MVC approach, Data Maintenance
SLR-Tool	Java	No	Pl-Co-Re	Text-Mining, Data Maintenance
SLRTOOL	PHP	Yes	Pl-Co-Re	MVC approach, Data Maintenance
SLR.qub	Javascript	Yes	Pl-Co	Text-Mining, Visualization
SLuRp	Java	No	Co-Re	Text-Mining, Visualization, Data Maintenance
StArt	N/A	No	Pl-Co-Re	Text-Mining, Visualization, Data Maintenance
SurVis	Javascript	Yes	Co	Text-Mining, Visualization
SysRev	N/A	Yes	Co	Active Learning, Visualization, Data Maintenance
Thoth	PHP	Yes	Pl-Co-Re	MVC approach, Data Maintenance

Table 5.2: Tools to support SLR with tool characteristics.

Moreover, the results indicate to a superiority of tools that support text-mining and visualization approaches. Tools to support SLR with machine learning techniques are increasing in the last few years as well.

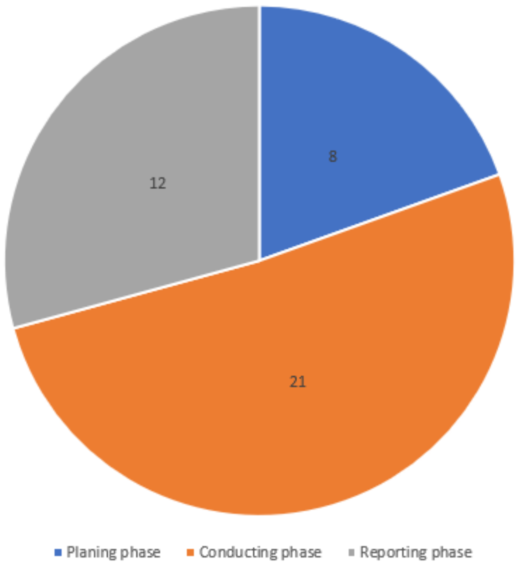


Figure 5.4: Distribution of the supported phases.

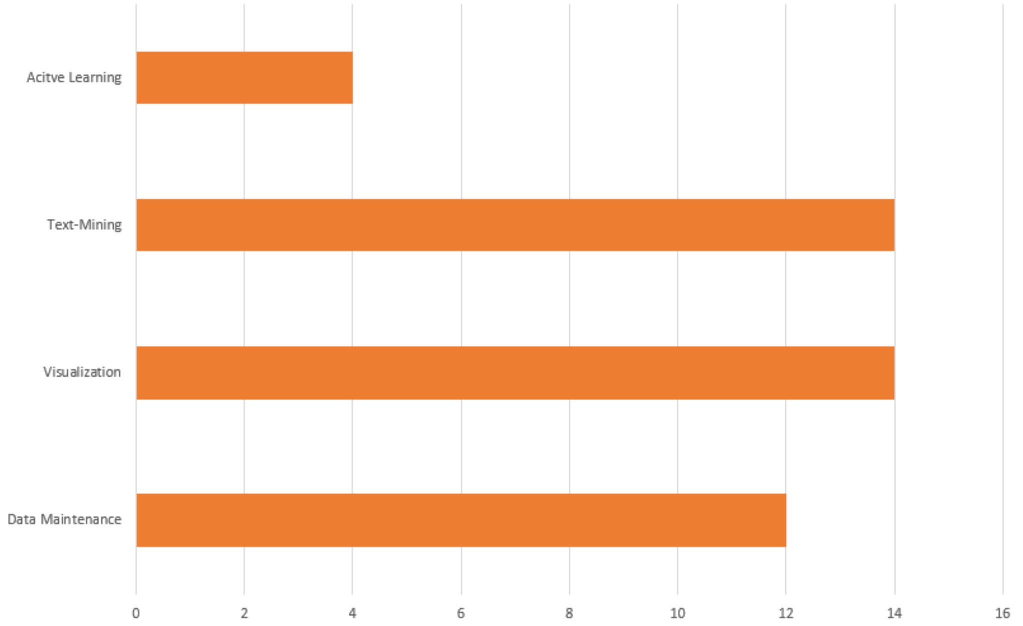


Figure 5.5: Underlying approach supported by tools.

5.2 Semi-Structured Interviews (RQ 2)

This section presents the results of the conducted semi-structured interviews.

5.2.1 Demographics

To get a sense of the respondent's experience with SLR, this subsection provides demographic information about the participants. (see Table 5.3). All of the participants were researchers in SE with knowledge of the SLR process and had at least performed one SLR. In total, seven researchers from four different institutions and from one company agreed to be interviewed.

Participant ID	Institute/Company	Role	No. of SLRs	Completed/Uncompleted	Last SLR
P01	Institute	Teaching and Research Assistant	6-8	Completed	2020
P02	Company	Postdoctoral Researcher	1	Uncompleted	2011
P03	Institute	Research Associate	5-6	Completed	currently performing
P04	Institute	Research Associate	3	Completed	2020
P05	Institute	Research Associate	1	Completed	2019
P06	Institute	Research Associate	6-7	Completed	currently performing
P07	Institute	Senior Researcher	2-12	2 Completed, 10 Uncompleted	2018

Table 5.3: Demographics of the invited participants.

Moreover, the type of the conducted reviews was mostly SLR or SMS. However, three participants had also experiences with GL. In addition, the number of involved researchers in the performed SLRs are varying. The average number of involved researchers was four. Mainly, the additional involved researchers were responsible for "*controlling the work (P05)*" and for "*assessing the quality (P01)*". Furthermore, additional researchers were involved to perform the search and study selection process "*independently (P06)*" with the goal "*to reduce personal bias in the results P(06)*" and especially "*to compare the results and determine the further steps based on several opinions P(06)*".

5.2.2 Challenges of Conducting a SLR (RQ 2.1)

In this subsection, the main challenges of the participants associated with SLR, are presented. The data obtained here was based on the responses to Question Group 2 (see Section 4.3). The main problems are shown in Table 5.4.

One of the biggest complaints was the time and effort consuming nature of the overall SLR process. By the participants, SLR's "*are very intense and require a lot of resources (P01; P02)*" and therefore "*take a lot of manual effort to get a representative set of papers (P05)*". Developing the review protocol was also considered as a challenging part, as it is important "*to find the correct start (P06)*" and "*to follow the strict guidelines (P06)*" for the SLR methodology to avoid research bias and to conduct the SLR in a systematic way. Moreover, most of the problems occur during the search stage. For three participants, the number of returned results from the search were too high which make it also "*difficult to obtain and access the full texts for each returned result (P02)*" and then "*to analyze the large corpus of papers (P05; P07))*". Moreover, two participants had the problem that the "*false positive papers (P05; P03)*" returned from the databases were also too high. For example, searching in databases using acronyms can be difficult, as there are many ambiguous acronyms with

SLR Phase	Main problems	Researchers
Overall process	Time/effort consuming	5
Planning the review		1
Conducting the search	High number of returned results	3
	Irrelevant results	2
	Limited access to the papers	1
	Choosing incorrect database	2
	Search string limitations	1
	Different export formats of the digital libraries	2
	Database results changes over time	1
	Variation of query languages	1
	Incomplete results	1
Study selection	Subjective results	2
Data extraction	Inconsistent extraction of data	2

Table 5.4: Main problems by the participants.

different meanings which can lead to a high number of results that are irrelevant for the research. One respondent who also had experiences with GL had the problem to choose the correct database for the GL, because search engines like Google or Bing often display "*ad-financed results which are pushed up by advertising, even though they are not important at all (P04)*". Another problem with search engines like Google or Bing is the limitation of the search strings. "*Such search engines mostly have other influencing factors that are hard to understand (P04)*", e.g., why a certain result for a re-used search string is displayed on a different page. This problem rarely occurs with digital libraries, as the results are ordered in descending order of relevance. A further problem is that the digital libraries often use different export formats. In consequence to this, you have "*to deal with different API's (P05)*" and firstly "*you have to find ways to aggregate the data (P05)*" which can also be time-consuming. Even though a SLR is time- consuming, a participant made the observation that the conducted SLR "*has to be finished faster rather than later because otherwise it would be kind of obsolete (P05)*" which can also be considered as a problem. In other words, if the SLR takes too much time, the obtained results "*probably are already obsolete when it is published (P05)*". For the search strategy development, one participant mentioned "*the variation of query languages (P05)*" of databases which means that the search strings for different databases have to be adapted because of format restrictions forced by the digital libraries. Based on the snowballing procedure, another important point that was mentioned is "*the selection of incorrect set of papers for the snowballing procedure (P06)*" which can lead to incomplete results. The study selection process was also considered as a challenging part which can lead to subjective bias if the studies were selected by only one researcher. Therefore, at this stage, it is important to involve at least one additional researcher to ensure that the study selection process is identical and consistent. Another problem discovered from the participants is the extraction of different data. One participant had the issue that "*sometimes different authors had similar names and different authors had the same name P07)*" which can be very challenging and can cause the extraction of wrong data. Another participant shared this challenge and mentioned that the "*data extraction procedure has to be proceeded consistently (P06)*" to avoid this problem.

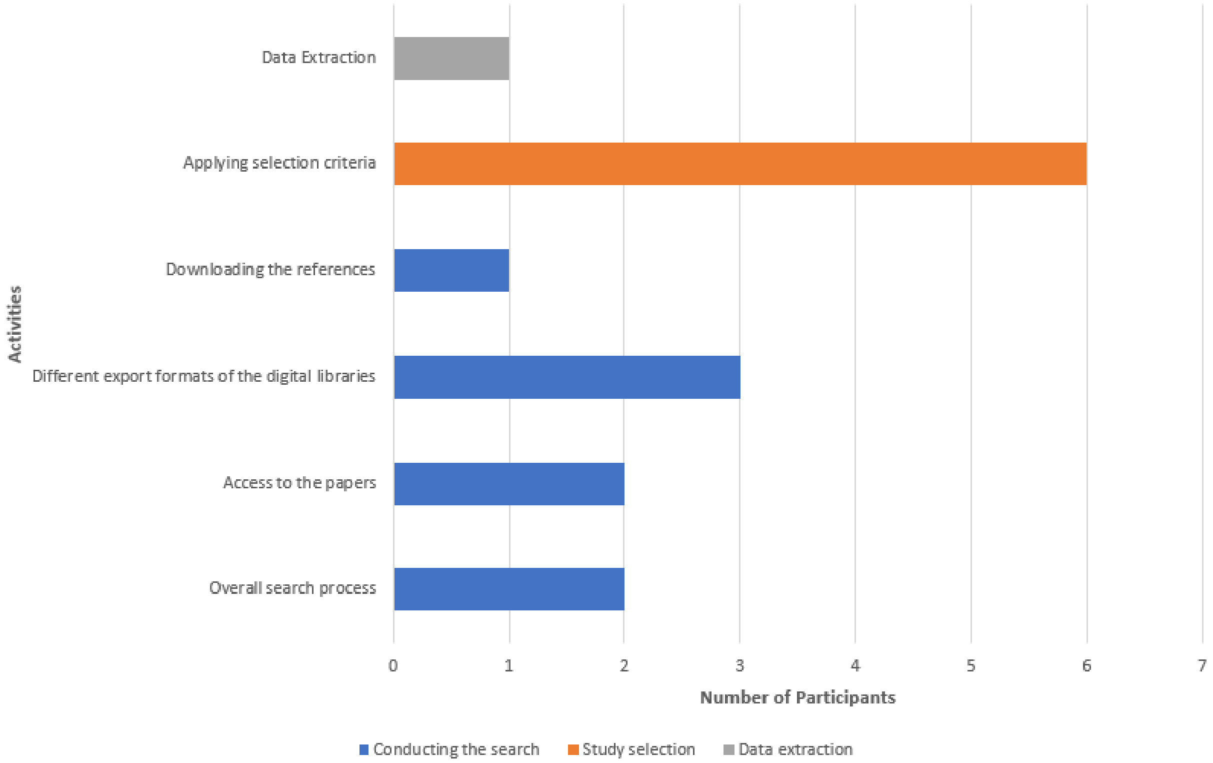


Figure 5.6: Reported activities with most manual effort.

Regarding the most manual activities of the SLR process, Figure 5.6 represents the responses indicating which activities of the SLR process the participants found most efforting. As the Figure 5.6 shows, the most manual and efforting activities occur during study selection, followed by conducting the search and data extraction. For most of the participants, applying the selection criteria to exclude or include the large amount of papers is the activity that require the most manual effort. It is difficult "*to read through all abstracts and texts and to find fitting information and keywords (P02)*" and then "*to decide if the paper is relevant or irrelevant for the research (P03)*".

Moreover, during the search stage, adapting the search strings for each search engine and then "*to normalize the export formats of these used databases (P05)*" was also considered as an efforting task by many participants. One participant also indicated that "*there are search engines that allow exporting the results in BibTex format (P05)*" but there are also search engines that do not provide this feature, so "*you have to modify manually all results to one format (P05)*". In addition, some respondents found the data extraction also as an elaborating task, as "*it is important to define the extraction form precise and perform the data extraction as well as possible to facilitate the analysis (P06)*".

Regarding the most error-prone activities of the SLR process, Figure 5.7 illustrates the responses indicating which activities of the SLR process participants found most error-prone. Here, the figure shows that extracting data and conducting the search, followed by study selection and data synthesis were seen as the most error-prone activities. One participant mentioned that "*data extraction is not an objective process (P03)*", consequently different researchers do it differently and as a "*subjective process (P03)*", the data extraction may be error-prone. Another participant mentioned

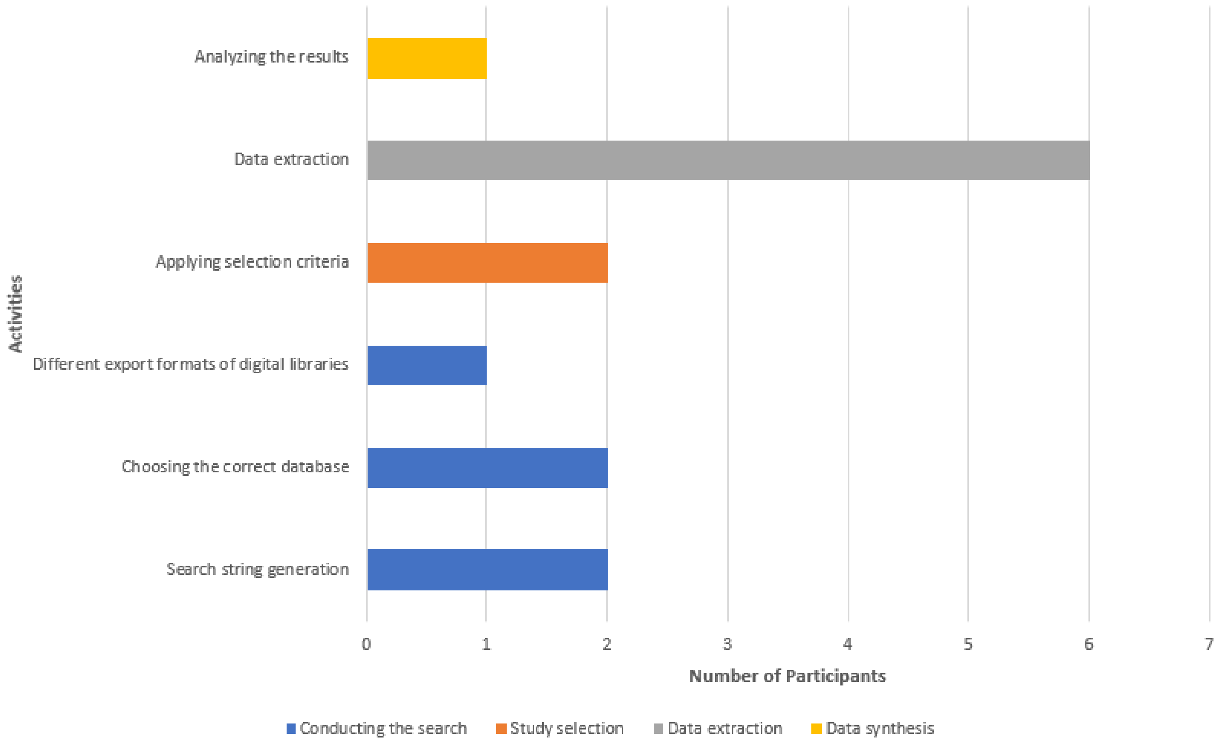


Figure 5.7: Most error-prone activities.

that "*the probability to extract incorrect data is high (P07)*". Other participants shared this view and mentioned that "*there is always a risk to extract the wrong data (P07)*", especially when two persons are extracting data and the results are different. Error-prone activities during the search stage are mostly related to the defined search terms which can lead to an "*overlap with other domains (P01)*" or "*to forgot important synonyms (P04)*". These can be important for the search and cause to "*missing relevant papers (P04)*" or that the "*expected knowledge could not be found (P02)*" which is in relation with the study selection. Moreover, the data synthesis procedure was also indicated as an error-prone activity, as it is required "*to choose the right method for analyzing (P06)*".

Table 5.5 summarizes the solutions and workarounds that are mentioned by the participants to address some of the challenges. Moreover, some of the participants stated that a collaborative work is one of the key elements to perform the SLR successfully. Due to this, the ideal number of involved researchers is important. If the team size is too small, it is "*more difficult to avoid research bias (P05)*". However, the team size should also not be too big. A higher number of involved researchers can lead to more effort in communication, coordination and getting agreement in particular tasks [BZ09].

Due to the time-consuming manual effort of the study selection, one participant had also made the experience to find no solution for this problem. Accordingly, the SLR was discontinued due to the high effort involved as "*the expected profit from the SLR was too low (P02)*" and "*the dynamism in the group was missing (P02)*".

SLR Phase	Solutions/Workarounds
Overall process	Follow the guidelines
Planning the review	Precise definition of the protocol
Conducting the search	Iteratively manipulate search strings and perform multiple searches Trial search queries Choose the correct set of papers for snowballing Search in multiple databases Use of a document and reference manager
Study selection	Expert judgement
Data extraction and synthesis	Review of extracted data Template creation Manual work and analysis

Table 5.5: Solutions and workarounds suggested by the participants.

5.2.3 Experiences with SLR Tools (RQ 2.3)

This subsection provides information about tools that are referenced by the participants. Table 5.6 shows the tools that are mentioned by the participants.

Tool type	Tool name	Total mentions
Reference management tool	Mendeley	3
	JabRef	3
	Zotero	1
	Citavi	2
Productivity tool	Microsoft Excel	5
	Google Sheets	5
Analysis tool	Statistical Analysis Software	1
	Publish or Perish	1
	Nvivo	1
	Atlas.Ti	1
Custom-Built tool	Proprietary Script	3
Other	Google Apps Script (Development Plattform)	1
	Zenodo (Open-Access Repository)	1

Table 5.6: Tools used by the participants.

The majority of tools used by participants were reference management and productivity tools. In particular, Mendeley, JabRef, Zotero and Citavi were mentioned most often. The purposes of these tools are relatively equal. Therefore, reference management tools were used to organize and manage the included papers. Such tools also offer good features like "*marking relevant places in the paper (P06)*", "*adding notes and tags (P06)*" which afterwards can be categorized or grouped to sort and filter the entries. In addition, reference management tools were used by the participants to remove duplications and to create bibliographies in which the entries are stored in a consistent format. The second most used tools by the participants were productivity tools such as Excel and

Google Sheets. These tools were used with the purpose of structuring and documenting results to get a better overview. Moreover, Google Sheets provides collaboration. The created sheets can be "*shared and multiple users can edit the Google Sheets at the same time (P03)*". In addition, advanced analysis tools were identified by the participants, which provide data or statistical analysis features with the purpose of "*aggregating extracted data (P07)*", "*perform statistical analysis of numeric data (P06)*" or "*for qualitative analysis of the papers (P06)*". The identified tools were Statistical Analysis Software, Publish or Perish, Nvivo, and Atlas.Ti. Further, three participants have used custom-built scripts to assist the search, specifically to generate search queries and to automatically gather the results from the databases. Table 5.7 presents the strengths and weaknesses of the top five identified tools as summarized by the participants.

Tool name	**Strengths**	**Weaknesses**
Mendeley	- Managing the papers - Annotations of papers - Sharing folders	- No versioning - Annotations are not visible for other operating systems
JabRef	- Managing the papers - Automated search with keywords - De-duplication - Minifying bibliographies	- No confidence in the quality of the automated search - Usability issues
Zotero	- Helps exporting results from Google Scholar	- Extraction of BibTex of each reference is not possible - Banning from google scholar is possible
Microsoft Excel	- Better overview of the results - Documentation of the results	- not efficient enough for SLR support
Google Sheets	- Collaboration support	- not efficient enough for SLR support

Table 5.7: Main strengths and weaknesses of the top 5 identified tools by the participants.

5.2.4 SLR Tool Features and Potential for Automation (RQ 2.4 & RQ 2.2)

In this subsection, answers received by the respondents related to important features and requirements which a tool needs to support are presented. Therefore, Table 5.8 shows the participant ratings for functional requirements of the SLR activity. As you can see, the requirements inclusion/exclusion and automated search were rated as the most important features which a tool to support SLR needs to have, followed by abstract/text screening and snowballing procedure. Many participants felt that an automated search would "*save a lot of time (P05)*", especially if the search is performed "*from one place (P05)*" "*to pilot the search terms (P04)*" for the development of the search strategy. However, for one participant, such an automation should function more as a "*suggestion system (P01)*" where best fitting search queries were suggested and which can be customized anytime. Moreover, two participants pointed out that "*not the whole search process can be automated (P03)*" as the results of false positive papers "*would be higher than the manual search (P03)*". Another participant agreed and remained "*skeptical (P04)*" towards automated search because the process can be "*very error-prone (P04)*" and due to this "*you have to trust the tool firstly (P04)*". The snowballing procedure was also rated by many participants as a requirement that has a lot of potential for automation because it is an "*exhausting task (P04)*" and requires "*a lot of effort (P04)*" and therefore, it would be a "*very helpful (P04)*" feature. In addition, three participants felt that "*storing all information into a uniform format and making it accessible would be interesting (P05)*". A feature that prepares extracted data which are demonstrated in plots or charts or, if the tool "*offers clustering methods for meta-data (P07)*" which can be directly used for the analysis was also praised.

Furthermore, two participants rated the support for the review protocol as "*nice to have (P01)*" where the support for the review protocol functions as a "*guide (P04)*" to ensure that all relevant steps of the SLR process are followed.

Functional requirements	Total participants
Inclusion/Exclusion	5
Automated search	5
Abstract/Text screening	4
Snowballing	4
Data extraction	3
Automated Analysis	3
Searching in multiple databases	2
Protocol development and validation	2
Report of findings and validation	1

Table 5.8: Participant ratings for functional requirements of the SLR activity.

Table 5.9 shows the participant ratings for functional requirements of the process management. As depicted, the requirements for multiple user support, traceability and storage of studies were rated as the most important features. Many participants felt that collaboration is "*very important (P07)*" and agreed that a SLR done by a single person has a "*lower reliability (P07)*" and due to this, performing the SLR in collaborative way was rated as important, especially to reduce overlaps in results and for resolution of disagreements among the SLR team. With respect to the collaboration support, traceability was also highly rated as important. The tool should provide "*history of flow (P02)*" where certain tasks or activities of all involved researchers are tracked "*to keep an overview of which tasks are done or which tasks still need to be completed (P06)*".

Functional requirements	Total participants
Support for multiple users	7
Traceability	5
Storage of documents and references	4
Data Sharing	2
Management of roles	2

Table 5.9: Participant ratings for functional requirements of process management.

Table 5.10 shows the participant ratings for non-functional requirements. As shown, maintainability was rated by most of the respondents as very important. One participant mentioned that a "*good tool has to be well maintained (P05)*" because "*time flies and nothing stands on the same place (P05)*" and due to this, the tool needs to be evolved with new features and updates, especially for "*bug fixing (P06)*". In addition, the financial payment for a SLR tool is important for three participants the sense of that the tools are provided for free. One participant stated, that SLR tools "*have a quite niche (P04)*", as the trust in such tools is usually lacking and due to this "*not everyone would be willing to pay money for a SLR tool (P04)*". Nevertheless, there are also respondents who would pay for a tool, as long as the tool supports "reasonable features (P07)" and the tool "*really helps and simplifies the process (P05)*".

Non-functional requirements	**Total participants**
Maintainability	5
Reasonable system requirements	3
Financial payment for the tool	3
Simple installation and setup	2
Extendibility	2
Installation guide/Tutorial	1

Table 5.10: Participant ratings for non-functional requirements.

All participants agreed that better tool support for SLR is needed. However, opinions differ as to what exactly the tool should support and in which challenges the automation potential lies. There are participants who would like to use a SLR tool "*to have the process as little overhead as possible (P03)*" and are not interested in a SLR tool which supports the whole process. On the other hand, there are also participants that would like to have a tool that "*does everything (P07)*", mainly a tool which also "*enables collaborative work (P01; P05)*" and supports all phases. Moreover, the tool should provide a "*continuous support function (P06)*" and not only a "*step-by-step guide (P06)*" which goes through all steps and establishes checklists for all these steps. Another participant shared this view and also stated that such SLR tools need "*a vibrant community (P04)*" so that they can always be evolved. The participants reported a number of challenges that require a lot of manual work. However, there are participants that see a potential for automation in some of the reported challenges to reduce the SLR effort. In regard to the inconsistent terminology of the databases, some of the participants prefer a tool that "*does the search from one place (P05)*" which combines the different interfaces of the databases and extracts the entries to the "*same format (P05)*". Moreover, the snowballing procedure is also "*very difficult and requires a lot of manual work (P04)*" and most of the participants would like to have this feature supported by a SLR tool. In addition, the participants would like to have a tool that supports automated generation of visualizations for simple meta information to have a brief overview of the collected data for the research, like "*in which year are the papers published (P01)*" or "*how many papers were included or excluded (P01)*", etc..

Furthermore, most tools used by the participants are general productivity and reference management tools and are not well-adapted for the SLR support. For example, as one participant stated that "*Excel or Google Sheets are not efficient enough for the SLR support (P02)*".

5.3 Feature Analysis (RQ 1.3)

With the conducted Rapid Review, many tools to support the SLR process were found. There are tools that usually support certain steps of the SLR process and tools that cover the whole process. However, the current tool support for SLR in SE had received limited evaluation, especially with regard to the automation potential. This makes it difficult for researchers to keep an overview and to select fitting candidates for their needs. Moreover, the existing evaluations for tool support are generally speculations about the automation potential and were only evaluated on the basis of the given literature and not properly tested. This section reports the most important elements of the feature analysis activity. The aim of this activity is to analyze and compare a set of selected tools that support specific steps of the SLR process. The selected tools are evaluated with a predefined framework that includes a set of functional and non-functional requirements and a defined scoring process for these tool requirements. This feature analysis is intended to provide an overview of the evaluated tools and to show, how well these selected tools cover the defined features for SLR support.

The results from the Rapid Review showed that a variety of tools exist that support the SLR process in SE. Especially tools were identified, which predominantly support the conducting stage of the SLR process with predominance of text-mining techniques, visualization techniques, active learning and machine learning approaches. Based on the performed Rapid Review and the found literature, the corresponding websites of the tools and the source code availability could be identified, which are important as the basis of the feature analysis to execute the selected tools.

5.3.1 Selected Tools, Features and Scoring Process

This section gives an overview of the non-selected and selected candidate tools for the evaluation, which were identified by the rapid review. Moreover, the feature sets, scoring and rating process, for which the tools are evaluated are presented in this section.

5.3.2 Selecting the Candidate Tools

As the results of the rapid review, 24 tools to support SLR was identified. The tools that are examined are those that were found through the rapid review and are used therefore for the feature analysis. Since the number of tools is too large for the evaluation and not all of the tools are executable for the analysis, because there were also tools which do not function properly. Here is a list of excluded tools from the rapid review with brief explanations, why the tool was excluded for the feature analysis:

1. **Document Analysis System for System Reviews (DASyR)** is a semi-automatic document analysis system and can help to annotate published papers for the purpose of ontology population. The tool is based on a bootstrapping annotation method and is followed by machine learning algorithms to enlarge the annotation set. In the course of this, DASyR can be helpful to support the data analysis part of the SLR [PLLH15]. But the tool is not selected for the feature analysis process because it is only mentioned and presented in the paper and no source code availability or a website of the tool was found.
2. **DBPedia** is a text-mining based tool and assists to extract structured information from Wikipedia [BLK+09]. DBPedia is a where the extracted data from Wikipedia is illustrated in structured data. The framework can be helpful for the selection part but only accesses the data from Wikipedia [AC14]. As we know, the SLR approach is based on the use of electronic databases, where academic literatures can be found but Wikipedia is an open encyclopedia with free content, where everyone can contribute. This approach cannot provide evidence for the SLR and therefore the tool was not selected for the feature analysis.
3. **ReviewBuilder** is a tool for automating parts of the SLR and is still in progress. As described in the GitHub Repository, the tool currently supports only creating automatic BibTex files with the results from Google Scholar and augmenting the metadata for each result by retrieving its

abstract but running the tool does not work. The automated search with entered search query gives an attribute error: `module 'scholarly' has no attribute 'search_pubs_query'`. For this reason, the tool is not selected for feature the analysis.

4. **SLRONT** is an ontology-based tool which is developed based on the researcher experiences of conducting a SLR [YYH+12]. The tool is not selected for the feature analysis process because it is only mentioned and presented in the paper. No source code availability or a website of the tool was found.

5. **SLRTOOL** is an open-source web-based collaborative tool for supporting the whole SLR process. The tool is developed using a model-driven approach and it is able to use the tool for different disciplines [14]. To use the tool registration is required but does not work. The web-based tool constantly indicates that the mail already exists and the registration cannot be completed. For this reason, there is no access to the tool and the functionalities cannot be tested.

6. **SLR-Tool** is a free tool for conducting a SLR and is not only created for the software engineering discipline. SLR-Tool supports functionalities like refining searches within the studies by applying text-mining techniques, defining classification forms in order to facilitate data synthesis and generates the obtained results to tables and charts. Furthermore, the ability to export the references from the primary studies to the used bibliographic formats like EndNote, BibTex or RIS is also provided. In short, it basically supports the whole process but the reason why this tool is not selected for further feature analysis is, because the free tool is no longer available on their website [10].

7. **PEx** is an open-source visualization tool which allows the collection of document on a map. It is developed with different projection techniques and text handling facilities to determine similarity among documents and can cluster related documents [FNF+10]. But the usage of the visualization works not well because only PDFs can be imported and the text of the imported PDFs are not clearly formatted. The tool also requires experiences with projection techniques and clustering algorithms to understand the output. Because of these restrictions the tool was not selected for the feature analysis.

8. **ReVis** is a visual text-mining tool that supports the conducting phase in the SLR process. It offers a framework of different projection techniques to visualize citations and co-citations relationships among imported documents and to create content maps to investigate content similarity relationships of certain data sets [FNMM14]. Although ReVis is an open-source tool, unfortunately there is no access to the tool. Therefore, the tool is not considered for the evaluation.

9. **Site Content Analyzer** is a text-mining tool that allows to extract information from HTML documents [CBSJ07], so there is no possibility to import BibTex, CSV or similar formats of files, which would be important to get the references directly from the academic libraries.

10. **SLR.qub** is a query builder that uses text-mining techniques and algorithms to gather the abstracts from the primary studies from IEEE Xplore with the used search strings [MSS15]. Besides, the tool was not developed for commercial use, so there is no access to the tool.

11. **StArt** was developed to provide automated support for the whole SLR process [12]. Therefore, the tool provides support for the planning phase, conducting phase, where the selection procedure takes place and reporting phase, where the extracted data from the accepted

papers can be analyzed. To meet all this features, the tool uses visualization and text-mining techniques. However, to-date, the tool is currently not available and therefore cannot be evaluated.

12. **SLuRp** is open-source tool which enables the support for the entire SLR process. The tool is developed in Java with a SQL database and uses a client side Java component which allows to search in digital libraries and semi-automatically extracts information with specific E-Library queries [BHB12]. Up-to-date, the provided download website for the tool, is not accessible. Accordingly, SLuRp was not selected for the feature analysis.

13. **revtools** is a R package that supports article screening for SLRs. The tool runs in a graphical user interface in ther browser. The main functionalities of revtools are the identification of duplicates and the screening of titles and abstracts of the imported papers [Wes19]. Moreover the tool supports visualization of the search results using text-mining to identify the main topics in the corpus. However, this feature does not work orderly and the results cannot be indicated. In addition the screening of titles and abstracts of the imported papers is very buggy. In consequence to this, revtools was not selected for the feature analysis.

14. **SESRA** is a web-based tool that supports the whole SLR process, therefore it provides support for the planning phase, conducting phase and reporting phase. In the planning phase researchers involved in the research can specify and define the relevant information, that are needed to continue with the next phase. Accordingly to this, the research questions, search strategy which includes study selection criteria, data extraction forms and dissemination strategy in case of disaggrements of the involved reviewers, can be defined. However, after completing the planning phase, the web-based tool breaks down and a error orccures. In consequence to this, the conducting phase and the reporting phase with tool cannot be performed. Regarding to this error, SESRA was not selected for the evaluation , because most of the feature cannot be tested [MB15].

The selected tools for the evaluation are:

1. ASReview
2. FASTREAD
3. Parsif.al
4. ReLis
5. SLR-Tool by Andreas Hinderks
6. SurVis
7. SysRev
8. Thoth
9. Buhos
10. REvieER

Feature Sets

The features and the importance of features are based on the results from the rapid review and the conducted semi-structured interviews with researchers. In Table 5.11 an overview of the generated features with assessment of importance, interpretation for the evaluation and the assigned weightings are depicted.

ID	Functional/ Non-Functional Features	Feature Set	Feature	Feature ID	Importance of Feature	Interpretation of Evaluation
FS1	Non-Functional Features	Economic	Financial Payment	NF-FS1-01	Significant (S)	IE[4]
FS2		Ease of Introduction and Setup	Reasonable system requirements	NF-F2-01	Significant (S)	
			Simple installation and setup	NF-F2-02	Significant (S)	
			Installation guide/ Tutorial	NF-F2-03	Nice to have (N)	
FS3	Functional Features	SLR Activity Support	Development and validation of protocol	F-F3-01	Significant (S)	
			Automated search	F-FS3-02	Highly Significant (HS)	
			Searching in multiple databases	F-FS3-03	Significant (S)	
			Abstract/Text screening	F-FS3-04	Highly Significant (HS)	
			De-duplication	F-FS3-05	Nice to have (N)	
			Inclusion/Exclusion	F-FS3-06	Highly Significant (HS)	
			Snowballing	F-FS3-07	Highly Significant (HS)	
			Inclusion of grey literature	F-FS3-08	Nice to have (N)	
			Quality assessment	F-FS3-09	Nice to have (N)	
			Data extraction	F-FS3-10	Significant (S)	
			Automated analysis	F-FS3-11	Significant (S)	
			Report findings and validation	F-FS3-12	Nice to have (N)	
FS4		Process Management	Support for multiple users	F-FS4-01	Highly Significant (HS)	
			Management of roles	F-FS4-02	Significant (S)	
			Support for multiple projects	F-FS4-03	Nice to have (N)	
			Data sharing	F-FS4-04	Significant (S)	
			Traceability	F-FS4-05	Highly Significant (HS)	
FS5	Non-Functional Features	Other Non-Functional Features	Maintainability	NF-FS5-01	Highly Significant (HS)	
			Extendibility	NF-FS5-02	Significant (S)	
			Security	NF-FS5-03	Nice to have (N)	

Table 5.11: Feature sets, assigned importance and interpretation of evaluation used for the feature analysis.

Firstly, the features are separated into functional and non-functional features. The functional features describe the SLR activities and the features which are important for the process management to perform a SLR with tool support. Therefore, the functional features are subdivided into the features sets, “SLR Activity Support” (FS3) and “Process Management” (FS4). To cover the technical, economic and quality aspects, the non-functional features are separated into the feature sets, “Economic” (FS1), “Ease of Introduction and Setup” (FS2) and “Other Non-Functional Features” (FS5).

The generated features which a tool to support SLR has to cover, are based on the guidelines and experiences of performing SLRs reported in the literature and the conducted semi-structured interviews. The suggested features are characterized as follows:

Feature Set 1: Economic

- Financial payment (NF-FS1-01)
  - The tool is free to obtain

Feature Set 2: Ease of Introduction and Setup

- Reasonable system requirements (NF-FS2-01)
  - The tool runs with reasonable system requirements and additional specification or complex configurations to run the tool correctly are not required.
- Simple installation and setup (NF-FS2-02)
  - The installation and setup is easy for the user.
- Installation guide/ Tutorial (NF-FS2-03)
  - The tool includes an installation/introduction guide or a tutorial is available

Feature Set 3: SLR Activity Support

- Development and validation of protocol (F-FS3-01)
  - A template for the protocol development is provided with the ability to define relevant information (e.g. objectives, research questions, search strategy, inclusion/exclusion criteria, quality assessment criteria, data extraction forms, etc.).
  - supports automatically generated checklists to keep the process in mind.
- Automated search (F-FS3-02)
  - The ability to perform an automated search using search queries.
  - The ability to convert different search string formats depending on the requirements of the databases.

[4] Interpretation of Evaluation

- Searching in multiple databases (F-FS3-03)
  - The tool can search in multiple databases.
- Abstract/Text screening (F-FS3-04)
  - The tool supports screening of titles, abstracts, meta-information or full-text of the papers.
- De-duplication (F-FS3-05)
  - The tool automatically identifies duplicates and removes them.
- Inclusion/Exclusion (F-FS3-06)
  - The ability to assess the studies for inclusion or exclusion.
  - The studies can semi-automatically assessed for inclusion or exclusion
- Snowballing (F-FS3-07)
  - The ability to map references of the snowballing procedure.
  - The ability to perform automated backward or forward snowballing.
- Inclusion of grey literature (F-FS3-08)
  - The tool provides a facility to include grey literature.
- Quality assessment and validation (F-FS3-09)
  - The tool allows to use and define suitable quality assessment criteria and to evaluate the quality of the studies.
  - The tool provides a feature to solve disagreements among the SLR team.
- Data extraction and validation (F-FS3-10)
  - The tool supports the extraction of quantitative data using classification, mapping or labeling approaches and manages numerical data that are reported in studies.
- Automated analysis (F-FS3-11)
  - The tool supports automated analysis of extracted data (i.e. text analysis, keyword analysis, topic analysis, statistical analysis).
- Report findings and validation (F-FS3-12)
  - The tool provides an organized and structured template to assist the report.
  - The tool supports evaluation checklists to validate the review.

Feature Set 4: Process Management

- Support for multiple users (F-FS4-01)
  - The tool allows to invite further users and to work together on a SLR.
- Management of roles (F-FS4-02)
  - The tool allows to allocate specific tasks or roles to different users (e.g. analyst, reviewer, evaluator, etc.).
- Support for multiple projects (F-FS4-03)
  - Using the tool, the users are allowed to perform multiple SLRs.
- Data sharing (F-FS4-04)
  - The ability to share data between processes and tasks, and among the SLR team.
- Traceability (F-FS4-05)
  - The ability to map relationship between papers or tracking of extracted data back to source study.
  - The ability to track the activities of the invited users.

Feature Set 5: Other Non-Functional Features

- Maintainability (NF-FS5-01)
  - The tool is well maintained with regular updates (e.g. bug fixing, new features).
  - The ability to contact if support is needed.
- Security (NF-FS5-02)
  - The tool guarantees security in relation to stored data in the tool and provides therefore a secure registation and log-in system.
- Extendibility (NF-FS5-03)
  - The source code of the tool is open-source and available.
  - The ability to extent the tool for new functionalities or features.

Scoring Process

In this section, the elementary ingredients of the scoring process are described. Each selected tool for the feature analysis will be scored against each feature. Thereby, the importance of the features plays a decisive role, which are used as multiplier to level the importance of the features. For this study, a feature can be categorized as "*Highly Significant (HS)*" , "*Significant (S)*" or "*Nice to Have (N)*". Table 5.12 shows the multiplier related to each importance level. The assigned importance level to each feature is shown in Table 5.11.

Importance	Multiplier
Highly Significant (HS)	·3
Significant (S)	·2
Nice to Have (N)	·1

Table 5.12: Importance of features.

The interpretation of evaluation helps to determine, whether a feature is fully, partially or not at all supported. If a feature or requirement is fully present, it was assigned with the score 1. If a feature or requirement is partially supported, the feature was assigned with the score 0.5. For features or requirements that are absent, the feature was awarded with the score 0. The interpretation is shown in Table 5.13.

Is the feature/requirement supported?	Score
Yes	1
Partially	0.5
No	0

Table 5.13: IE - Interpretation of the evaluation.

By multiplying the importance of the features with regard to the interpretation score, the weighted scores for each feature are calculated. These weighted scores are combined to calculate a percentage score for each feature set. The percentage score for each i-feature set (FS_i) is calculated as follows:

$$Percentage\ Score(FS_i) = \frac{\sum_{F \in (FS_i)} (WeightedScore(F))}{MaximumScore(FS_i)}$$

For example, the Feature Set 5 (FS5) has three features, Maintainability (NF-FS5-01), Extendibility (NF-FS5-02) and Security (NF-FS5-03). (NF-FS5-01) has been classified as "*Highly Significant (HS)*", therefore the maximum score for this feature is three. Feature (NF-FS5-02) has been classified as "*Significant*", therefore the maximum score for this feature is two. The maximum score for (NF-FS5-03) is one, because it is classified as "*Nice to Have*". Accordingly, the maximum score for Feature Set 5 is six. The interpretation of evaluation declares, whether the feature is fully, partially or not all supported. If the (NF-FS5-01) is for example partially supported, it can score from 0.5 to 2.5. When the feature is fully supported, the full reachable score will be assigned.

The overall percentage score is the sum of all weighted feature scores divided by the total number of achievable scores.

5.3.3 Feature Analysis Results

In this section, the results of the feature analysis for each tool are presented.

ASReview

Table 5.14 shows the scores for ASReview.

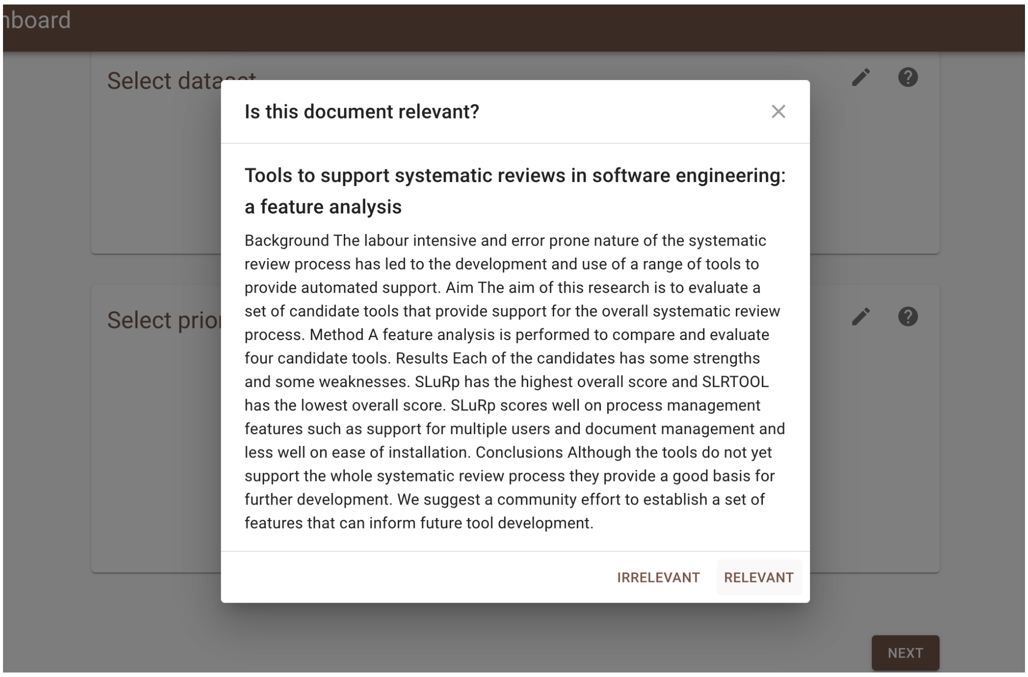
Feature Set 1: Economic ASReview is an active learning tool for SLR and was developed for a research project to explore the capability of active learning in reviewing large amounts of textual papers. The entire project is open-source, therefore the tool is free and the source code is openly available via GitHub [5].

Feature Set 2: Ease of Introduction and Setup The software and all other documentations for the use are available on the developer's website. ASReview requires Python 3.6 or higher and therefore the tool has reasonable system requirements and is very easy to install. The available step-by-step guide is detailed and well described for the operating systems Windows, MacOS and Linux. The installation is carried out with the command `pip` by running the terminal and is successfully installed in a few minutes. Even if you have problems with the installation, in the guide are many structured suggestions and solutions for the troubleshooting's and therefore relatively quickly to handle.

Feature Set 3: SLR Activity Support ASReview does not support protocol development or automated search. In addition, since the automatic search is not possible, the tool does not support searching in multiple databases. The tool is especially designed to perform the screening phase of the SLR. To perform the screening phase, ASReview requires to import a set of data, which includes the abstracts of the scientific papers, which can be obtained from the digital libraries or from reference management tools where several scientific papers can firstly be collected from different databases. Thereby, ASReview allows the formats RIS and CSV which are supported by most of the digital libraries or reference management tools. After importing a set of data, the tool requires a prior set of documents to activate the active learning approach. This is the pre-screening phase of the tool. To label a prior set of documents as irrelevant or relevant, specific documents can be found by keyword searching or by clicking on "random" to show a couple of random documents.

The tool screens the titles and the abstracts of the documents, where the documents can be labeled as irrelevant or relevant, which is shown in Figure 5.8. After a document has been labeled, the next document is displayed directly, which is a real advantage because you can save time and do not have to open the documents separately each time for the selection process. It is important to note that the tool requires a couple of documents to be labeled, to use the machine learning model. For the automatic labeling process, the tool provides various classifiers to calculate the relevant values of each document. The supported classifiers are Naive Bayes, Support Vector Machine, Logistic Regression, and Random Forest [Tea20b]. The default classifier is Naive Bayes, which works relatively well especially for large datasets. The Naive Bayes approach actually works relatively simply, where the documents in this case can be divided into two classes "irrelevant" or "relevant". By pre-labeling the documents, the classifier learns to assign the documents automatically to

[5] https://github.com/asreview/asreview



Is this document relevant?

Tools to support systematic reviews in software engineering: a feature analysis

Background The labour intensive and error prone nature of the systematic review process has led to the development and use of a range of tools to provide automated support. Aim The aim of this research is to evaluate a set of candidate tools that provide support for the overall systematic review process. Method A feature analysis is performed to compare and evaluate four candidate tools. Results Each of the candidates has some strengths and some weaknesses. SLuRp has the highest overall score and SLRTOOL has the lowest overall score. SLuRp scores well on process management features such as support for multiple users and document management and less well on ease of installation. Conclusions Although the tools do not yet support the whole systematic review process they provide a good basis for further development. We suggest a community effort to establish a set of features that can inform future tool development.

IRRELEVANT RELEVANT

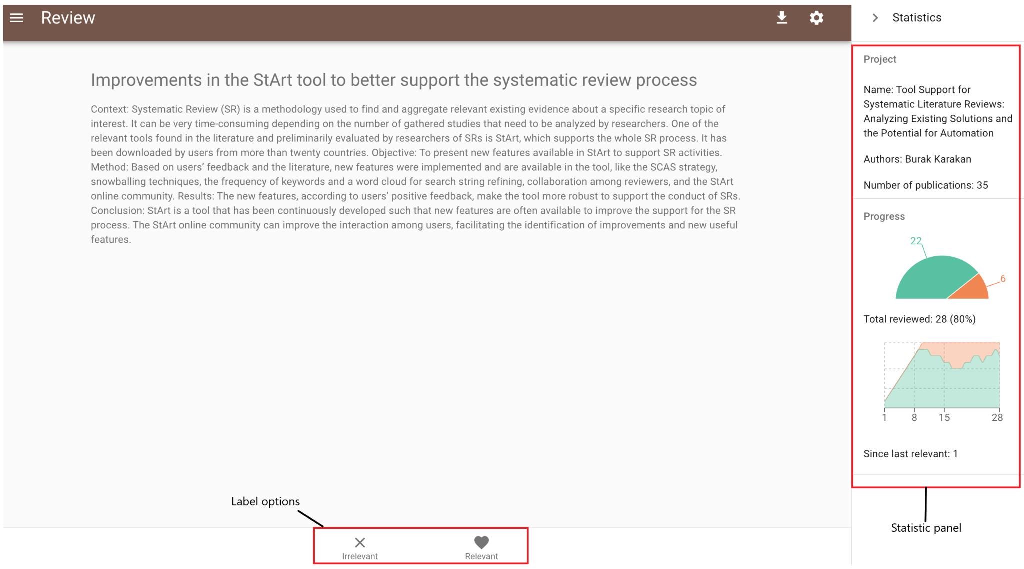
Figure 5.8: Pre-screening process of ASReview.

the correct class. The query strategy determines which document is shown after the model has computed the relevant scores and the feature extraction decides the method, how text is translated into a vector that can be used by the classifier [Tea20b].

In the main screening process, the automatically labeled documents can be changed and be manually modified at any time. The user interface of the screening process has been kept as simple as possible and is designed very user-friendly. This makes it easy to concentrate on the displayed texts of documents in order to make a decision, as shown in Figure 5.9. During the screening process, a statistics panel is also provided on the right hand side (see Figure 5.9, on which some diagrams are displayed to give an overview of the so far undertaken screening process. The pie chart shows an overview of how many relevant (green) and irrelevant (orange) records have been screened. Additionally, the total number of records screened, as well as the percentage screened, is displayed relative to the total number of records in the dataset. The second plot is a progress plot. The x-axis tracks the number of records screened and the y-axis displays the average between relevant and irrelevant records.

After the process is completed, the results can be downloaded as Excel or CSV. By allowing selected documents to be marked as irrelevant or relevant, the tool supports as well a part of the inclusion/exclusion process. A part of the data extraction and automated analysis is also provided, where some diagrams give automatically an overview of the screening process, which can contain some quantitative numerical data.

Feature Set 4: Process Management Unfortunately ASReview does not support collaboration, therefore the work of multiple users on a SLR or the support of performing specific tasks among different users with conflict resolution support, are not possible. However, the tool supports to perform multiple SLR projects. The created projects are stored locally and are displayed in the



Improvements in the StArt tool to better support the systematic review process

Context: Systematic Review (SR) is a methodology used to find and aggregate relevant existing evidence about a specific research topic of interest. It can be very time-consuming depending on the number of gathered studies that need to be analyzed by researchers. One of the relevant tools found in the literature and preliminarily evaluated by researchers of SRs is StArt, which supports the whole SR process. It has been downloaded by users from more than twenty countries. Objective: To present new features available in StArt to support SR activities. Method: Based on users' feedback and the literature, new features were implemented and are available in the tool, like the SCAS strategy, snowballing techniques, the frequency of keywords and a word cloud for search string refining, collaboration among reviewers, and the StArt online community. Results: The new features, according to users' positive feedback, make the tool more robust to support the conduct of SRs. Conclusion: StArt is a tool that has been continuously developed such that new features are often available to improve the support for the SR process. The StArt online community can improve the interaction among users, facilitating the identification of improvements and new useful features.

Project

Name: Tool Support for Systematic Literature Reviews: Analyzing Existing Solutions and the Potential for Automation

Authors: Burak Karakan

Number of publications: 35

Total reviewed: 28 (80%)

Since last relevant: 1

Figure 5.9: Screening process of ASReview.

user-friendly project dashboard. Moreover, the import and export of created projects is possible, so, these exported projects can be easily imported on other devices or can be shared to other users, who also use the tool.

Feature Set 5: Other Non-functional features ASReview is well and freely maintained by its developers and regularly updated for bug fixing and performance improvements. The latest release was at the end of the year 2020 and contains a number of internal changes to resolve various bugs in the web application. As the web-based tool can only be used locally, it does not include a log-in system. Additionally, ASReview supports extensions, which enable to integrate additional packages or features for the ASReview framework. This can extend the tool, e.g. with new classifiers, query strategies, balancing strategies and other feature extraction techniques. Other extensions or plugins are already offered by ASReview. For example, there is a visualization extension that can be adapted to suit the needs and can be integrated into the framework [Tea20a].

FASTREAD

Table 5.15 shows the scores for FASTREAD.

Feature Set 1: Economic FASTREAD is an open-source tool, which can be obtained for free via GitHub repository [6]. The tool supports the conducting phase of the SLR, especially to reduce the effort for study selection in SLR's [YKM18].

[6] https://github.com/fastread/src

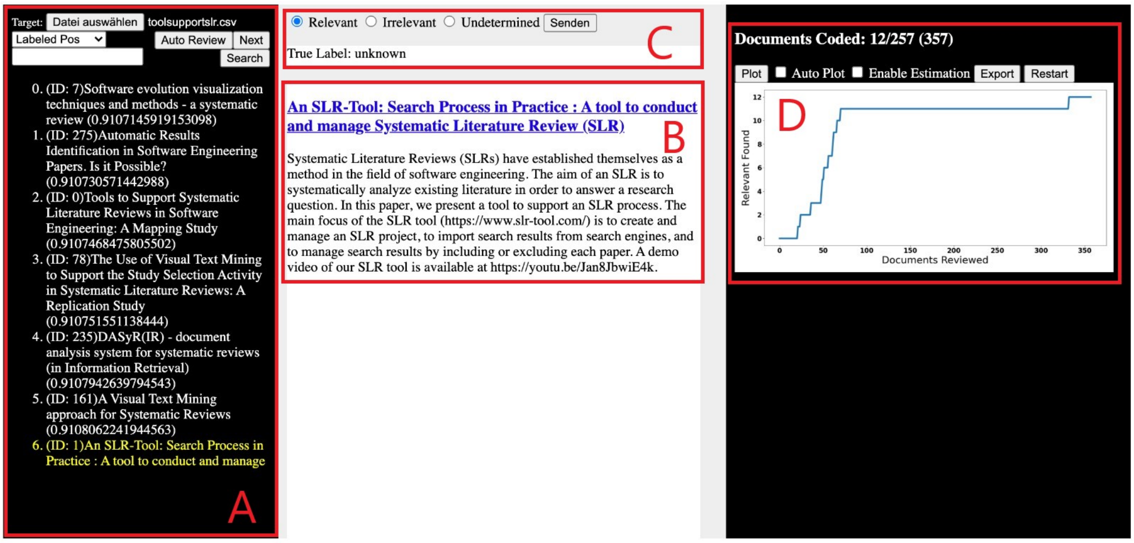
Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	5/5	100%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	1	1/1		
FS3	F-FS3-01	S	0	0/2	7/24	29%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	0.5	2.5/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	0/1		
	F-FS3-10	S	0.5	0.5/2		
	F-FS3-11	S	0.5	1/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	0	0/3	1/11	9%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	0	0/1		
	F-FS4-04	S	0.5	1/2		
	F-FS4-05	HS	0	0/3		
FS5	NF-FS5-01	HS	1	3/3	4/6	66%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	1	1/1		
Total Score			**Overall % Score**			
19/48			39.6%			

Table 5.14: Scores for ASReview.

Feature Set 2: Ease of Introduction and Setup The tool does not require any advanced hardware or software for performing the functionalities. What you have to do is, to set up Python and to install the requirements to run the script. After installing the required packages and running the script, the tool opens your browser and runs the tool on localhost. Therefore, installation and the setup was simple and a quick installation guide is also provided by the developers.

Feature Set 3: SLR Activity Support FASTREAD is a tool that only supports the screening of titles and abstracts in the conducting phase of the SLR, which can be used in the study selection process for the inclusion/exclusion criteria. The tool is structured very similar to ASReview and uses also active learning and machine learning approaches to automatically review the data sets. To use the tool, firstly you have to import the study list, which contains the title, abstract, year and the PDF link of the listed documents. However, only the input of CSV format is provided and the import works only for CSV files, which can be obtained from IEEE Xplore. Therefore, collecting studies in reference managers and export them as CSV format does not work with FASTREAD.

The interface of FASTREAD is kept very simple, as you can see in Figure 5.10. After the data set has been imported, the candidate studies are randomly displayed on the left hand side (marked as A in Figure 5.10). To review the studies, the titles and the abstracts are displayed in the center (see marking B in Figure 5.10). Above the screening, the selection bar is located (marked with C in



0. (ID: 7)Software evolution visualization techniques and methods - a systematic review (0.9107145919153098)
1. (ID: 275)Automatic Results Identification in Software Engineering Papers. Is it Possible? (0.910730571442988)
2. (ID: 0)Tools to Support Systematic Literature Reviews in Software Engineering: A Mapping Study (0.9107468475805502)
3. (ID: 78)The Use of Visual Text Mining to Support the Study Selection Activity in Systematic Literature Reviews: A Replication Study (0.910751551138444)
4. (ID: 235)DASyR(IR) - document analysis system for systematic reviews (in Information Retrieval) (0.9107942639794543)
5. (ID: 161)A Visual Text Mining approach for Systematic Reviews (0.9108062241944563)
6. (ID: 1)An SLR-Tool: Search Process in Practice : A tool to conduct and manage

True Label: unknown

An SLR-Tool: Search Process in Practice : A tool to conduct and manage Systematic Literature Review (SLR)

Systematic Literature Reviews (SLRs) have established themselves as a method in the field of software engineering. The aim of an SLR is to systematically analyze existing literature in order to answer a research question. In this paper, we present a tool to support an SLR process. The main focus of the SLR tool (https://www.slr-tool.com/) is to create and manage an SLR project, to import search results from search engines, and to manage search results by including or excluding each paper. A demo video of our SLR tool is available at https://youtu.be/Jan8JbwiE4k.

Documents Coded: 12/257 (357)

Figure 5.10: Screening process of FASTREAD.

Figure 5.10), where the studies can be labeled as relevant, irrelevant or undetermined [YKM18]. However, the selection of papers, whether to decide if it is relevant or irrelevant, does not work fluently, as you have to click on the "submit" button after each selection in order to appear for the next study. In addition, after a number of studies have been trained manually, the tool uses a Support Vector Machine (SVM) classifier to automatically classify the studies. FASTREAD transforms the relevant textual components into numerical vectors using the feature extraction Term Frequency-Inverse Document Frequency (TF-IDF), to predict the relevance of the studies [YKM18]. Unfortunately, the support of automatic reviewing did not work during the execution of the tool. On the right area a plot is automatically generated after reviewing the documents, where the x-axis shows the number of relevant selected studies and the y-axis the number of studies reviewed. After completion of the selection process, the reviewed studies can exported as CSV format, which contains an additional column with the reviewer-decided label. Because of this, the tool supports only a small subset of the data extraction and analysis feature. So, the generated plot can be used for quantitative analysis in SLR's or SMS's, to see how many relevant studies are selected.

Feature Set 4: Process Management Since the tool focuses specially on the selection part of studies, none of the process management features are provided.

Feature Set 5: Other Non-functional features The basic version of the tool was firstly released in 2016. Since then, there were six more version logs of the tool with bug fixing and newly developed features. The last big release was in 2017, which is some time ago and the tool has still bugs and some features do not work. For that reason, FASTREAD covers maintainability not very well. Moreover, the tool does not provide any security or expandability functionalities. As I mentioned before, the interface of FASTREAD is kept very simple and does not offer any additional features on the UI. On the whole, the selection of studies works relatively well, except that only CSV files from IEEE Xplore can be imported and the automatic reviewing does not work.

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	5/5	100%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	1	1/1		
FS3	F-FS3-01	S	0	0/2	6/24	25%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	0.5	2/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	0/1		
	F-FS3-10	S	0.5	0.5/2		
	F-FS3-11	S	0.5	0.5/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	0	0/3	0/11	0%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	0	0/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0	0/3		
FS5	NF-FS5-01	HS	0.5	0.5/3	0.5/6	8%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	0	0/1		
Total Score			**Overall %**			
13.5/48			28.1%			

Table 5.15: Scores for FASTREAD.

Parsifal

Table 5.16 shows the scores for Parsifal.

Feature Set 1: Economic Parsifal is a tool to support researchers to perform the whole SLR within the context of SE and was designed to help through all those steps. The tool is a web-based tool and is online available for free [7]. The only requirement is that you have to register to use the functionalities.

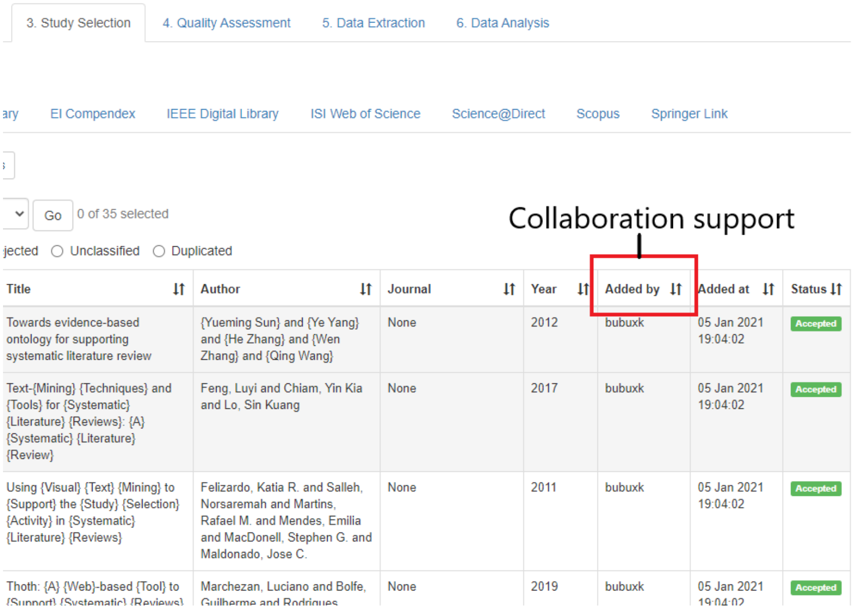
Feature Set 2: Ease of Introduction and Setup Since the tool is web-based, there are no further system requirements or installations needed. Besides, the online tool does not provide any tutorial or a guide. On the web page there is only a small description of what the tool does and some features are described.

Feature Set 3: SLR Activity Support As I mentioned before, Parsifal is a tool which helps through the whole SLR and is conceptualized for the SE context. It provides a step-by-step guide and helps to remind, the important elements during a SLR. During the planning phase, Parsifal

[7] https://parsif.al/

assists the researcher to define the important components that are required in SLR's to enrich the preferenced evidence. Therefore, it includes input areas to define the objectives, research questions, search strings, keywords, selection of sources, where the studies can be obtained, and the inclusion/exclusion criteria. Moreover, the planning phase of Parsifal provides a mechanism, where the quality assessment questions, the quality assessment scores and the data extraction forms can be defined. This can be a real help after the study selection in the conducting phase is done, because it provides a good overview of which studies still need to be assessed for the quality and what data is extracted from each study.

During the conducting phase the tool offers to import BibTex files from multiple databases and select the relevant studies with regard to the inclusion and exclusion criteria. In addition, Parsifal includes a semi-automated search, where the defined search strings access automatically to the databases. Parsifal is integrated with Scopus and ScienceDirect databases but unfortunately it shows only the results from these databases and cannot be used for the study selection. So if you want to conduct the study selection, you have to import BibTex files. There is also a possibility to categorize the imported data to the respective databases, where the data comes from. The study selection process works relatively well but sometimes the abstracts are not displayed correctly and there are some bugs of screening the important information from the papers. Automatic duplicate detection is not supported by the tool, however, what is supported is that if duplicates exist, they can be manually labeled as duplicates.



Collaboration support

Title	Author	Journal	Year	Added by	Added at	Status
Towards evidence-based ontology for supporting systematic literature review	{Yueming Sun} and {Ye Yang} and {He Zhang} and {Wen Zhang} and {Qing Wang}	None	2012	bubuxk	05 Jan 2021 19:04:02	Accepted
Text-{Mining} {Techniques} and {Tools} for {Systematic} {Literature} {Reviews}: {A} {Systematic} {Literature} {Review}	Feng, Luyi and Chiam, Yin Kia and Lo, Sin Kuang	None	2017	bubuxk	05 Jan 2021 19:04:02	Accepted
Using {Visual} {Text} {Mining} to {Support} the {Study} {Selection} {Activity} in {Systematic} {Literature} {Reviews}	Felizardo, Katia R. and Salleh, Norsaremah and Martins, Rafael M. and Mendes, Emilia and MacDonell, Stephen G. and Maldonado, Jose C.	None	2011	bubuxk	05 Jan 2021 19:04:02	Accepted
Thoth: {A} {Web}-based {Tool} to {Support} {Systematic} {Reviews}	Marchezan, Luciano and Bolfe, Guilherme and Rodrigues	None	2019	bubuxk	05 Jan 2021 19:04:02	Accepted

Figure 5.11: Study selection process and collaboration support of Parsifal.

So, the use and development of suitable quality assessment criteria and the ability that multiple users are allowed to evaluate the quality, is provided by the tool. Furthermore, Parsifal supports some statistical analysis, which are generated automatically. Hence, the tool presents plots, where the articles are categorized per source, the accepted and reviewed studies per source and a statistic about the years of the final artefacts. As the last part of the SLR, the tool supports the reporting stage as well. After the conducting stage of the research is completed, researchers are allowed to export a final report. This final report contains the most important components that were defined during the planning phase, the source search strings that were used and the number of finally included studies for the data extraction.

Feature Set 4: Process Management One of the biggest advantages of Parsifal is that the tool supports collaboration for its users. The tool allows to assign multiple users to work on a SLR. Unfortunately, Parsifal does not support management of roles, where different users can be allocated to perform specific tasks. Nonetheless, the researchers can perform multiple SLR's with Parsifal. Moreover, as you can see in Figure 5.11, the tool supports traceability of users in the study selection stage. So, every time when new studies are added or assigned as included or excluded by different users, the activities of the different users can be tracked.

Feature Set 5: Other Non-functional features Since the web-based tool was released in 2014, there have been regular updates until 2019, where new features were added and bugs were fixed. Therefore the tool is well maintained by its developers and is easy to conduct if support is needed. The tool includes a log-in system and the data are stored online, therefore the tool offers security and the conducted researches can only be seen by the assigned users. In general, Parsifal offers good support for the whole SLR process. It helps to remember what is important during a SLR and provides a way to document the whole process. The user interface is therefore very structured and organized and there are no problems to manage well through the online tool.

ReLis

Table 5.17 shows the scores for ReLis.

Feature Set 1: Economic ReLis is a web-based tool for conducting SLR's and is freely and publicly available on the official website [8]. The tool is open-source and the openly available source code on GitHub [9] can also be installed locally on the servers.

Feature Set 2: Ease of Introduction and Setup ReLis runs online, therefore no specific system requirements are needed to use the tool. Although the tool provides tutorials and setup guidelines. The setup of ReLis was complex, because the tool has to be set up by defining a configuration model that automatically installs the project on the cloud [BS18]. ReLis follows a Model-Driven Development (MDD) with the aim of ensuring flexibility to the researchers to configure projects by themselves without any system administration skills [BS18]. So, before starting with the project, the semantic of the project has to be set up with custom and editable configurations which accompany during the entire SLR process. First of all, the SLR `PROJECT` identifier and the `name` has to be defined. For the `SCREENING` step, the researcher can configure, how the papers are assigned to the researchers. In Listing 5.1 is a configuration example of the ReLis tool. Here, the paper assignment is defined as `Automatic`, which means that the papers are assigned automatically to the researchers for reviewing. For the screening process, the inclusion and exclusion criteria can also be defined. There is also the opportunity to define `conflict_resolution`. The tool supports two types of conflict resolutions: either if one reviewer excludes a paper that another has included, or if the exclusion criteria do not conflict [BS18]. The conflict is resolved when the decision is unanimous or if the majority agree [BS18]. If questions and possible answers for the quality assessment are

[8] http://relis.iro.umontreal.ca/auth.html

[9] https://github.com/geodes-sms/relis

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	4.5/5	90%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	0.5	0.5/1		
FS3	F-FS3-01	S	1	2/2	12/24	50%
	F-FS3-02	HS	0.5	0.5/3		
	F-FS3-03	S	0.5	1/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	0.5	2/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	1	1/1		
	F-FS3-10	S	0.5	1/2		
	F-FS3-11	S	0.5	1/2		
	F-FS3-12	N	0.5	0.5/1		
FS4	F-FS4-01	HS	1	3/3	6/11	55%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0.5	2/3		
FS5	NF-FS5-01	HS	1	3/3	4/6	67%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	1	1/1		
Total Score			**Overall %**			
28.5/48			59.4%			

Table 5.16: Scores for Parsifal.

desired, the researcher has to define `QA` with the questions and the possible response, which are associated with scores (see Listing 5.1). The `CLASSIFICATION` part defines extracted data which can be categorized and collected from each paper (see Listing 5.1). The example in Listing 5.1 shows only a small subset of configuration opportunities. There are many possibilities to extend the configuration.

Feature Set 3: SLR Activity Supptor To-date, ReLis does not support protocol development, automated search or searching in multiple databases, however most other stages are supported by the tool. The online tool supports the screening of papers, which can be included in BibTex, Endnote or CSV format. The meta information are store in cloud and each imported study can be tagged with its source and search strategy [BS18]. During the screening process the users can decide to include or exclude the studies. The inclusion and exclusion criteria can be defined in the configuration mode. Moreover, to assist quality assessment, ReLis allows researchers to define quality statements and in case of disagreements between quality scores from different users, it supports conflict resolutions. In addition, ReLis supports data extraction forms, which are automatically generated from the configuration model. The researcher can classify, categorize or highlight relevant information from the included studies, as depicted in Figure 5.12. From this classified information, reviewers can

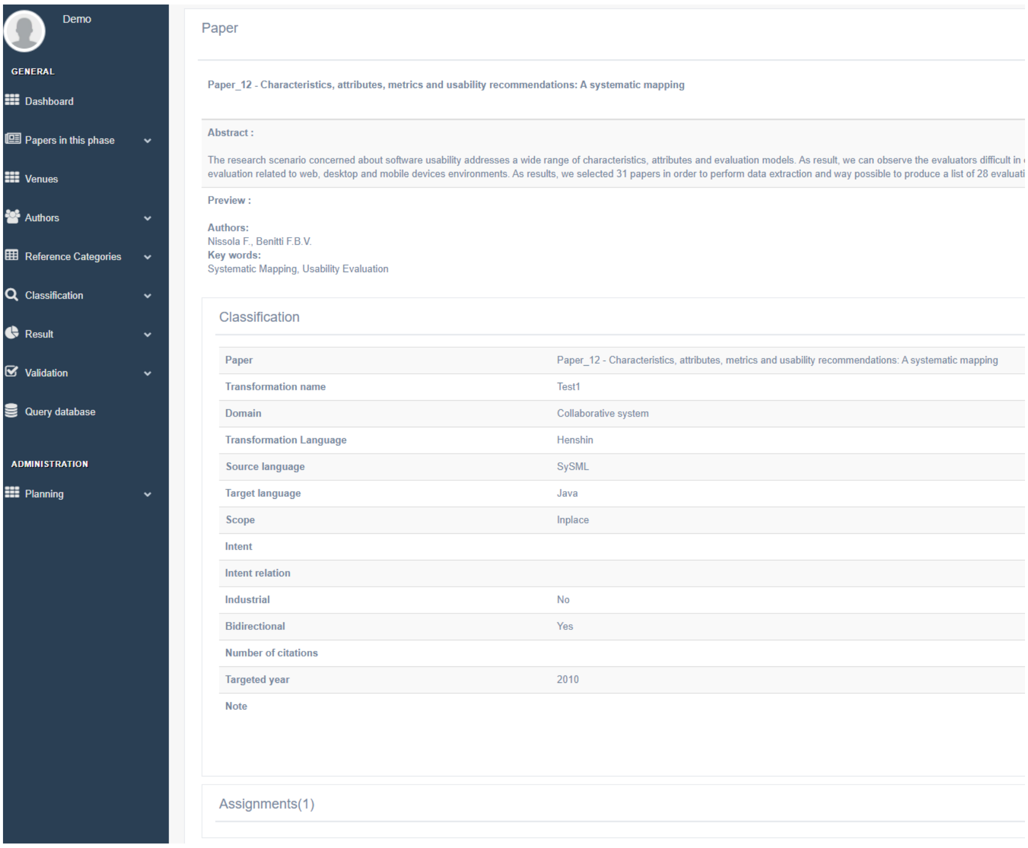
```
PROJECT

ts "Tool support"

SCREENING
    paper_assignment = Automatic
    conflict_type = IncludeExclude
    conflict_resolution = Unanimous
    exclusion_criteria = ["Article is not in english",
        "tools not useable for the software engineering discipline",
        "studies not related to the research questions",
        "reference and document manager tools are not considered"]
QA

 Questions =["Question 1","Question 2","Question 3","Question 4"]
 Response=["Yes":2,"Partially":1,"No":0]
 Min_score 1

CLASSIFICATION

Simple tool_name "Tool name" * [10] :string(100)
DynamicList underlying_approach "Underlying approach"  [10] "Underlying approach" =["Text-
Mining","Collaboration","Visualization","Machine Learning","Active Learning", "Whole Process"]
DynamicList slr_stage "SLR Stage"  [10] "SLR Stage " =["Planning Phase","Conducting Phase","
Reporting Phase"]
```

Listing 5.1: Configuration of ReLis.

explore the quantitative results in form of graphs, charts or tables [BS18]. All extracted data can also be exported as CSV file. Furthermore, ReLis supports validation for results of the screening process. It randomly assigns the processed studies to another reviewer for validation [BS18].

Feature Set 4: Process Management One of the high-level features of ReLis is, that it supports collaboration. Also, a support for management of roles, where different users can perform specific tasks, is given. Moreover, users can perform multiple SLR's using the tool. However, sharing data between processes and tasks among the SLR team is not supported. What supported additionally for the process management is traceability, where the tool can track the progress and the activities of the assigned reviewers.

Feature Set 5: Other Non-functional features Any information about the maintainability could not founded. The web-based tool was released in 2018 and since then there have been no further updates or releases for the tool. Even if the tool provides security through a login system, the configurated model for the conducted research is public. Therefore, the security is guaranteed partly. However, the tool supports good extension possibilities because of the configuration model. When specific parts, like definition of the quality assessment is not needed, it can be simply omitted. The configuration model of the tool can be modified at any time.



Paper

Paper_12 - Characteristics, attributes, metrics and usability recommendations: A systematic mapping

Abstract :

The research scenario concerned about software usability addresses a wide range of characteristics, attributes and evaluation models. As result, we can observe the evaluators difficult in
evaluation related to web, desktop and mobile devices environments. As results, we selected 31 papers in order to perform data extraction and way possible to produce a list of 28 evaluati

Preview :

Authors:
Nissola F., Benitti F.B.V.
Key words:
Systematic Mapping, Usability Evaluation

Classification

Paper	Paper_12 - Characteristics, attributes, metrics and usability recommendations: A systematic mapping
Transformation name	Test1
Domain	Collaborative system
Transformation Language	Henshin
Source language	SySML
Target language	Java
Scope	Inplace
Intent	
Intent relation	
Industrial	No
Bidirectional	Yes
Number of citations	
Targeted year	2010
Note	

Assignments(1)

Figure 5.12: Generated data extraction forms in ReLis.

SLR-Tool by Andreas Hinderks

Table 5.18 shows the scores for SLR-Tool.

Feature Set 1: Economic SLR-Tool by A. Hinderks is a free web-based tool and can be accessed from the developer's website [10]. The main focus of the tool is to manage the lifecycle of the SLR process. It also helps to organize and qualify the search results from the search engines.

Feature Set 2: Ease of Introduction and Setup The SLR-Tool follows a registration process and therefore the tool does not require any specific system requirements and can be used online, at the developers website. Unfortunately, the source code of the tool is not available. Brief introduction guides could not be found at the website. But in the reported conference paper, where the tool is presented, a demo video of the SLR-Tool is linked [HMTE20]. The video demonstrates a brief tutorial about the main functionalities and a quick introduction about the usage of the tool.

Feature Set 3: SLR Activity Support The SLR-Tool does not provide protocol development and validation, automated search or the ability to search in multiple search engines. The main focus of the tool lies on the study selection process, where the search results from the search engines can be checked for their quality to assess the inclusion and exclusion criteria. Moreover, it provides and

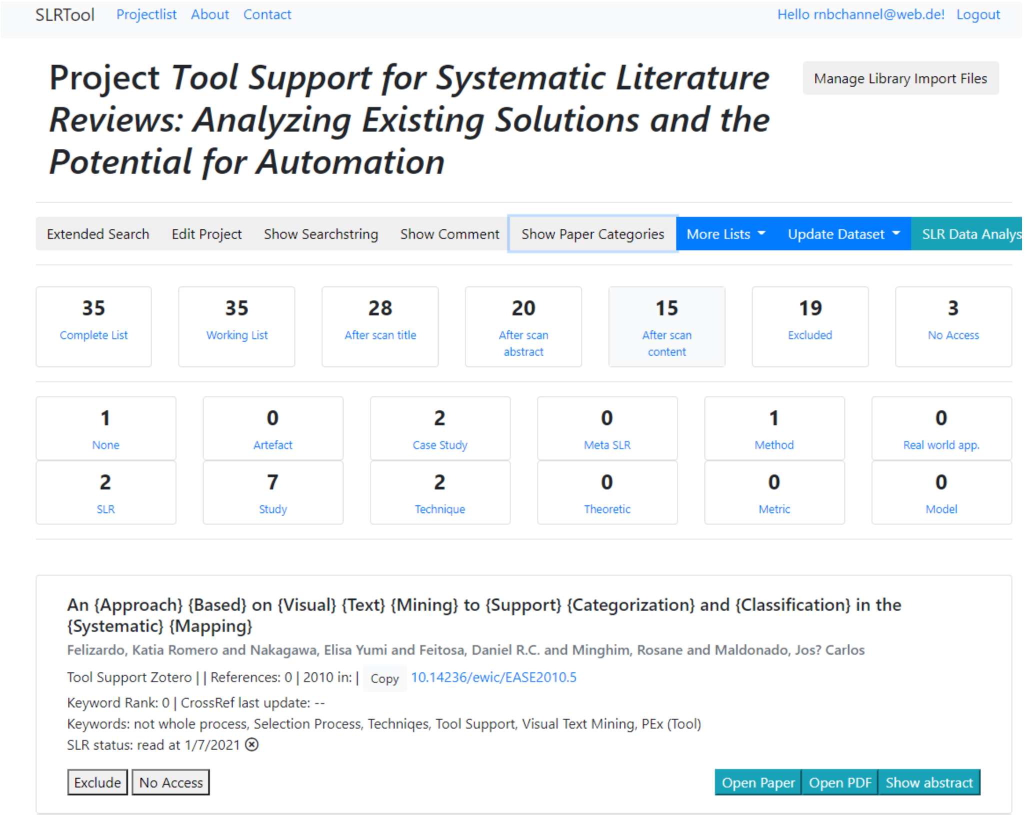
[10] https://www.slr-tool.com/

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	3.5/5	70%
	NF-FS2-02	S	0.5	0.5/2		
	NF-FS2-03	N	1	1/1		
FS3	F-FS3-01	S	0	0/2	9/24	38%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	1	3/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	1	1/1		
	F-FS3-10	S	0.5	1/2		
	F-FS3-11	S	0.5	1/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	1	3/3	9/11	81%
	F-FS4-02	S	1	2/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	1	3/3		
FS5	NF-FS5-01	HS	0	0/3	1.5/6	25%
	NF-FS5-02	S	1	1/2		
	NF-FS5-03	N	0.5	0.5/1		
Total Score			**Overall %**			
25/48			52.1%			

Table 5.17: Scores for ReLis.

good facility, to manage the search results of the SLR process. To conduct the selection process, it is required to import the search results from the search engines. For this, the tool accepts the import of BibTex. Additionally, the import of CSV files, which are gained from ACM Digital Library or SpringerLink is also allowed. Moreover, the SLR-Tool uses the CrossRef API to update the imported studies, if the abstract does not exist [HMTE20]. Furthermore the tool provides an extended search function to filter the previously imported search results using keywords. Based on the keyword search, a ranking will be calculated with regard of the number of keywords found. After calculation of the ranking, the papers with the most frequently found keywords are displayed at the beginning of the working list [HMTE20].

In the working list, the papers can be assessed for the inclusion and exclusion criteria. For this, the SLR-Tool supports the inclusion and exclusion functions for specific screening steps. Therefore, the steps, scanning after title, scanning after abstract and scanning after content is supported (see Figure 5.13). The paper lists in each step are manageable and can be used for further proceeding. Besides, the excluded papers, or papers, which cannot be accessed, are listed in separate lists. Furthermore, additional information can be stored for each paper and can be categorized for specific attributes [HMTE20]. However, the tool does not support de-duplication, snowballing, the quality assessment procedure, data extraction, automated analysis or reporting functionalities.



SLRTool Projectlist About Contact Hello rnbchannel@web.de! Logout

Project *Tool Support for Systematic Literature Reviews: Analyzing Existing Solutions and the Potential for Automation*

Manage Library Import Files

Extended Search | Edit Project | Show Searchstring | Show Comment | Show Paper Categories | More Lists ▾ | Update Dataset ▾ | SLR Data Analys

35	35	28	20	15	19	3
Complete List	Working List	After scan title	After scan abstract	After scan content	Excluded	No Access

1	0	2	0	1	0
None	Artefact	Case Study	Meta SLR	Method	Real world app.
2	7	2	0	0	0
SLR	Study	Technique	Theoretic	Metric	Model

An {Approach} {Based} on {Visual} {Text} {Mining} to {Support} {Categorization} and {Classification} in the {Systematic} {Mapping}

Felizardo, Katia Romero and Nakagawa, Elisa Yumi and Feitosa, Daniel R.C. and Minghim, Rosane and Maldonado, Jos? Carlos

Tool Support Zotero | | References: 0 | 2010 in: | Copy 10.14236/ewic/EASE2010.5

Keyword Rank: 0 | CrossRef last update: --

Keywords: not whole process, Selection Process, Techniqes, Tool Support, Visual Text Mining, PEx (Tool)

SLR status: read at 1/7/2021 ⊗

Exclude | No Access | Open Paper | Open PDF | Show abstract

Figure 5.13: Management of the search results in SLR-Tool by A. Hinderks [HMTE20].

Feature Set 4: Process Management In the conference paper in which the tool is reported, it is mentioned that the SLR-Tool by A. Hinderks allows to invite researchers to existing projects [HMTE20] but the collaboration support in the tool is not supported. There is no possibility to invite users for reviewing the research. Furthermore, sharing data or traceability is also not supported. Nevertheless, the tool has support for multiple projects, so the users are allowed to create multiple SLR's using the tool.

Feature Set 5: Other Non-functional features The web application tool was developed with ASP.NET Core 2.2, where the design pattern Model View Control (MVC) represents the data layer with the including data logic. MS SQL Server 2016 was used for the management of data storage [HMTE20]. However, the online tool, does not seem to be well maintained, because there is no information about the maintainability available. The tool has some bugs, which could be fixed. For example, after a short absence, the system logs off automatically and you have to re-log in all the time. Since no source code is available and no information about maintainability is provided, the extendibility of the tool seems also to be lacking.

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	4.5/5	90%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	0.5	0.5/1		
FS3	F-FS3-01	S	0	0/2	5/24	21%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	0.5	2/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	0/1		
	F-FS3-10	S	0	0/2		
	F-FS3-11	S	0	0/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	0	0/3	1/11	9%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0	0/3		
FS5	NF-FS5-01	HS	0	0/3	1/6	17%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	1	1/1		
Total Score			**Overall %**			
13.5/48			28.1%			

Table 5.18: Scores for SLR-Tool by A.Hinderks.

SurVis

Table 5.19 shows the scores for SurVis.

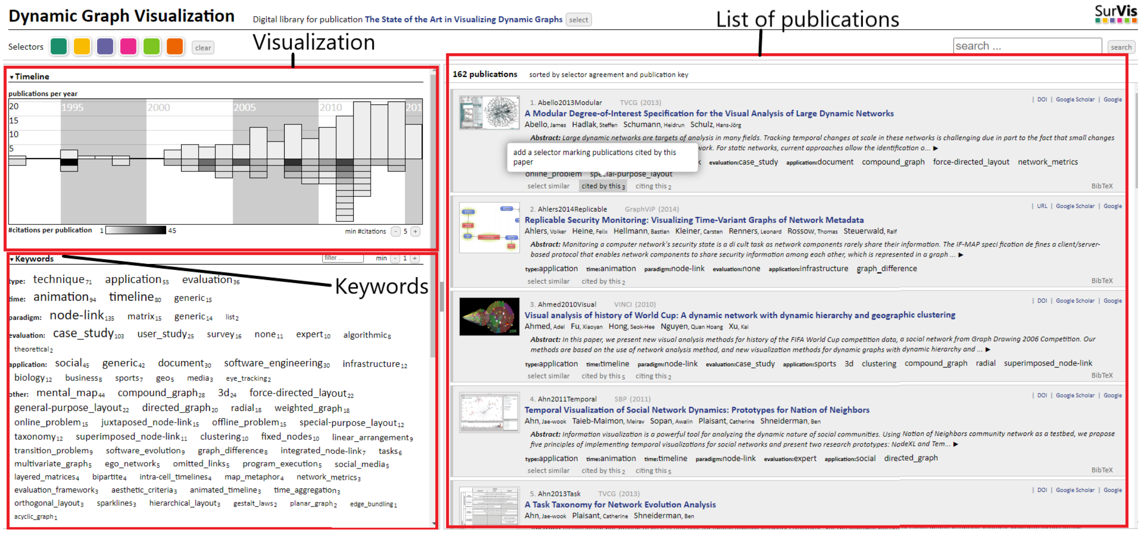
Feature Set 1: Economic SurVis is a web-based visual analysis tool with the aim of analyzing and structuring literature collections [BKW16]. The latest version of the SurVis tool can be downloaded on the developer's repository via GitHub [11] and is freely available.

Feature Set 2: Ease of Introduction and Setup To run the online tool, you have to open the downloaded HTML file in your browser and the setup does not require any specific requirements, therefore the installation and setup process was reasonable. Currently, there is no tutorial for the tool. For this reason, it was difficult to get to grips with the tool, because the interface is structured complex.

Feature Set 3: SLR Activity Support SurVis is especially conceived for the purpose to analyze literature collections. Therefore, most of the SLR steps are not supported by the tool. However, the tool supports the screening of titles, abstracts and meta information that can be used to create

[11] https://github.com/fabian-beck/survis

conclusive visualizations. To use the purpose, it is required to add the BibTex entries and the tool parses the results automatically. But before the entries are imported, it is recommended to punctuate the entries with keywords, if there are no keywords in the publications, so the tool can automatically structure the publications based on the keywords. The tool offers a wide range of analysis options. SurVis supports word-size visualizations, which is also called sparklines. The aim of this visualization approach is to display relationships between entities, selectors and publications [BKW16]. The integrated selectors identify all publications cited by or citing a specific publication and visually displays the list of publications. The Sparklines technique is also used to create adjacency matrices to represent dynamic graphs or word clouds to represent the temporal occurrence of concepts [BKW16].



Dynamic Graph Visualization

Digital library for publication The State of the Art in Visualizing Dynamic Graphs

type: technique$_{71}$ application$_{55}$ evaluation$_{36}$
time: animation$_{94}$ timeline$_{80}$ generic$_{15}$
paradigm: node-link$_{135}$ matrix$_{15}$ generic$_{14}$ list$_{2}$
evaluation: case_study$_{103}$ user_study$_{25}$ survey$_{16}$ none$_{11}$ expert$_{10}$ algorithmic$_{8}$ theoretical$_{2}$
application: social$_{45}$ generic$_{42}$ document$_{30}$ software_engineering$_{30}$ infrastructure$_{12}$ biology$_{12}$ business$_{8}$ sports$_{7}$ geo$_{5}$ media$_{3}$ eye_tracking$_{2}$
other: mental_map$_{44}$ compound_graph$_{28}$ 3d$_{24}$ force-directed_layout$_{22}$ general-purpose_layout$_{22}$ directed_graph$_{20}$ radial$_{18}$ weighted_graph$_{18}$ online_problem$_{15}$ juxtaposed_node-link$_{15}$ offline_problem$_{15}$ special-purpose_layout$_{12}$ taxonomy$_{12}$ superimposed_node-link$_{11}$ clustering$_{10}$ fixed_nodes$_{10}$ linear_arrangement$_{9}$ transition_problem$_{9}$ software_evolution$_{9}$ graph_difference$_{8}$ integrated_node-link$_{7}$ tasks$_{6}$ multivariate_graph$_{5}$ ego_network$_{5}$ omitted_links$_{5}$ program_execution$_{5}$ social_media$_{5}$ layered_matrices$_{4}$ bipartite$_{4}$ intra-cell_timelines$_{4}$ map_metaphor$_{4}$ network_metrics$_{3}$ evaluation_framework$_{3}$ aesthetic_criteria$_{3}$ animated_timeline$_{3}$ time_aggregation$_{3}$ orthogonal_layout$_{3}$ sparklines$_{3}$ hierarchical_layout$_{3}$ gestalt_laws$_{2}$ planar_graph$_{2}$ edge_bundling$_{1}$ acyclic_graph$_{1}$

162 publications sorted by selector agreement and publication key

1. Abello2013Modular TVCG (2013)
A Modular Degree-of-Interest Specification for the Visual Analysis of Large Dynamic Networks
Abello, James Hadlak, Steffen Schumann, Heidrun Schulz, Hans-Jörg
Abstract: Large dynamic networks are targets of analysis in many fields. Tracking temporal changes at scale in these networks is challenging due in part to the fact that small changes … work. For static networks, current approaches allow the identification o... ▶
add a selector marking publications cited by this paper
… k evaluation:case_study application:document compound_graph force-directed_layout network_metrics
online_problem special-purpose_layout
select similar cited by this$_{3}$ citing this$_{2}$

2. Ahlers2014Replicable GraphViP (2014)
Replicable Security Monitoring: Visualizing Time-Variant Graphs of Network Metadata
Ahlers, Volker Heine, Felix Hellmann, Bastian Kleiner, Carsten Renners, Leonard Rossow, Thomas Steuerwald, Ralf
Abstract: Monitoring a computer network's security state is a di cult task as network components rarely share their information. The IF-MAP speci fication de fines a client/server-based protocol that enables network components to share security information among each other, which is represented in a graph ... ▶
type:application time:animation paradigm:node-link evaluation:none application:infrastructure graph_difference
select similar cited by this$_{2}$

3. Ahmed2010Visual VINCI (2010)
Visual analysis of history of World Cup: A dynamic network with dynamic hierarchy and geographic clustering
Ahmed, Adel Fu, Xiaoyan Hong, Seok-Hee Nguyen, Quan Hoang Xu, Kai
Abstract: In this paper, we present new visual analysis methods for history of the FIFA World Cup competition data, a social network from Graph Drawing 2006 Competition. Our methods are based on the use of network analysis method, and new visualization methods for dynamic graphs with dynamic hierarchy and ... ▶
type:application time:timeline paradigm:node-link evaluation:case_study application:sports 3d clustering compound_graph radial superimposed_node-link
select similar cited by this$_{5}$ citing this$_{2}$

4. Ahn2011Temporal SBP (2011)
Temporal Visualization of Social Network Dynamics: Prototypes for Nation of Neighbors
Ahn, Jae-wook Taieb-Maimon, Meirav Sopan, Awalin Plaisant, Catherine Shneiderman, Ben
Abstract: Information visualization is a powerful tool for analyzing the dynamic nature of social communities. Using Nation of Neighbors community network as a testbed, we propose five principles of implementing temporal visualizations for social networks and present two research prototypes: NodeXL and Tem... ▶
type:application time:animation time:timeline paradigm:node-link evaluation:expert application:social directed_graph
select similar cited by this$_{2}$ citing this$_{5}$

5. Ahn2013Task TVCG (2013)
A Task Taxonomy for Network Evolution Analysis
Ahn, Jae-wook Plaisant, Catherine Shneiderman, Ben

Figure 5.14: The user interface of SurVis.

On the right in Figure 5.14, the added list of publications is indicated. Each publication is represented as a box and contains the important information for the screening. Therefore each publication includes the title, meta-information like publication key, publication series, year and keywords and as an important information source, the abstract. The timeline represented (see Figure 5.14) at the top provides an overview of publication years of the collection. Thereby, the bars at the top outline the number of publications per year. The boxes below envision the most cited publications which are found in the collection [BKW16]. The darkness of the boxes indicates the citation frequency. To condense the keywords and the meta-information of the collected publications the tool utilizes the concept of word clouds, which are represented below the timeline (see Figure 5.14) [BKW16]. Morveover, SurVis supports an interactive clustering method, to classify the selected publications [BKW16]. For this approach, the researcher has to choose between keywords and authors or both which are allocated as input for the clustering process. Then, the clustering algorithm generates a feature vector for each publication and displays the resulting clusters, which are similar to a word cloud [BKW16]. These visualization and clustering approaches are especially suitable for the quantitative analysis of SLR's or SMS's. However, the visualization and clustering results cannot be exported from the web-based tool.

Feature Set 4: Process Management Unfortunately, the tool does not support most of the process management features, therefore collaboration and data sharing is not supported. However, the tool supports a small subset of traceability, because SurVis allows to map relationship between papers and the individual keywords consisting of meta information which can be traced back to the initial paper.

Feature Set 5: Other Non-functional features The first release of SurVis was in 2015. Since then, there seem to have been no further updates for the tool, therefore the tool is not maintained regularly. However, the repository of the developers provide a contact form, if support is needed. As it is an open-source project, SurVis can be individually extended at any time. However, the developers do not provide any additional packages to extend.

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	4/5	80%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	0	0/1		
FS3	F-FS3-01	S	0	0/2	7/24	29%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0	0/1		
	F-FS3-06	HS	0	0/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	0/1		
	F-FS3-10	S	1	2/2		
	F-FS3-11	S	1	2/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	0	0/3	2/11	18%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	0	0/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0.5	2/3		
FS5	NF-FS5-01	HS	0	0/3	1/6	17%
	NF-FS5-02	S	0.5	1/2		
	NF-FS5-03	N	0	0/1		
Total Score			**Overall %**			
16/48			33.3%			

Table 5.19: Scores for SurVis.

Sysrev

Table 5.20 shows the scores for Sysrev.

Feature Set 1: Economic The web-based online tool Sysrev[12] is a multifaceted platform for reviewing and extracting data from a collective set of documents. The basic version of the tool is for free and allows you to create unlimited public projects. Moreover the free version supports unlimited project reviewers and project management. The pro version of Sysrev costs ten dollar per month and has additional advanced features, like the ability to create private projects, analytics features and group labels.

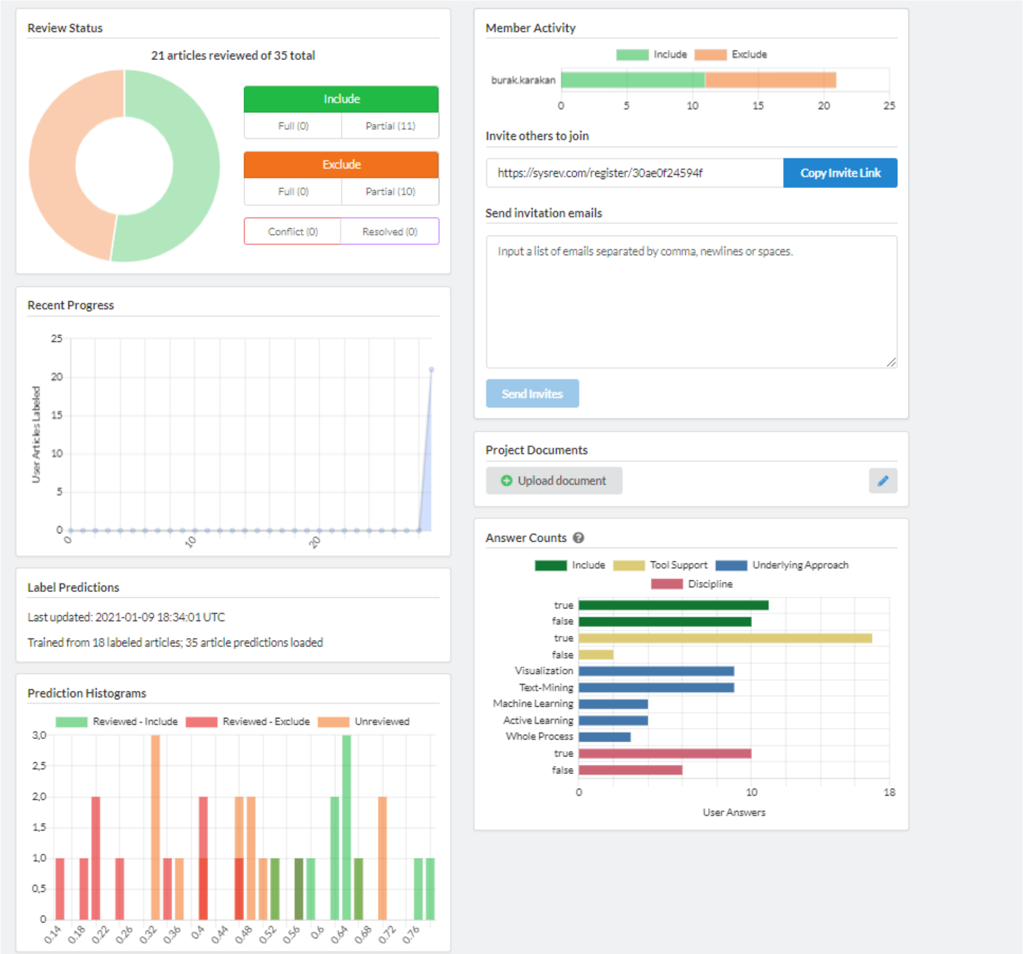
Feature Set 2: Ease of Introduction and Setup Sysrev can be accessed from the developers website following a registration process. Unfortunately the source code of the tool is not available. The setup does not require any specific installations, therefore the setup process was considered as reasonable. Additionally the developers website of the tool provides publicly accessible blogposts, where a brief introduction and a tutorial can be found.

Feature Set 3: SLR Activity Support Sysrev does not support protocol development and validation. The tool can be used for multiple disciplines. Especially for the healthcare discipline, the tool supports automated PubMed searches, where medical abstracts can be found through the entered search terms. However, this automated approach does only work with PubMed. For other disciplines, the documents can be imported customly. Thereby, the tool supports several different import formats. Sysrev automatically detects the duplicates. After the duplicates have been removed and only unique articles are left, the reviewing process can be activated and the labels for the inclusion and exclusion criteria can be defined. The tool supports several label options that can be used as well as for the data extraction forms. Therefore, adding boolean, categorical and string labels are allowed. After the review tasks and labels are defined, the included papers can be reviewed in the screening process.

As mentioned, the tool allows the definition of data extraction forms as well which can be used for quantitative data analysis. Accordingly to this, Sysrev supports automated analysis of the extracted data and gives an statistical overview of the research, as shown in Figure 5.15. The overview of the research includes several analytics. The tool supports a statistical analysis of the reviewed papers, which screens the included and excluded papers, tracks member activities, shows the recent progress of the research, counts the answers for the data extraction forms and displays the most important terms, which appear in the articles. After reviewing, all defined and analyzed data can be exported. However, the tool does not support the report of findings which can assist the write-up.

Feature Set 4: Process Management Sysrev supports collaboration. Once a project has been created, new members can be invited. The invited users can be grouped for different tasks and roles. Accordingly, the tool supports multiple users and management of roles. With Sysrev, the ability to perform multiple projects at the same time, is also supported. The tool does not allow to share data between process and tasks among the SLR team. However, the tool tracks the activities of the invited member, therefore, every reviewer action is recorded by Sysrev and is accessible for the reviewers [Lue19]. In consequence, the results of any kind of document review can be audited.

[12] https://sysrev.com/



Review Status

21 articles reviewed of 35 total

Recent Progress

Label Predictions

Last updated: 2021-01-09 18:34:01 UTC

Trained from 18 labeled articles; 35 article predictions loaded

Prediction Histograms

Member Activity

Invite others to join

https://sysrev.com/register/30ae0f24594f

Send invitation emails

Input a list of emails separated by comma, newlines or spaces.

Project Documents

Answer Counts

Figure 5.15: Basic automated analytic forms of Sysrev.

Feature Set 5: Other Non-functional features As it is reported in the blogposts, there are some releases with new features. However, more information on maintainability could not be found. Correspondingly, maintainability of Sysrev is partially supported. Moreover, the source code of Sysrev is not available and therefore there is no possibility to extend the tool for further functionalities. Additionally, the online tool follows a registration process and the data are stored in the cloud. However, in the basic freely version of the tool, the projects are open to the public and can only be switched to private with the pro version. Subsequently, the security is partially supported.

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	0.5	1/2	1/2	50%
FS2	NF-FS2-01	S	1	2/2	5/5	100%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	1	1/1		
FS3	F-FS3-01	S	0	0/2	13/24	54%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0.5	1/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	1	1/1		
	F-FS3-06	HS	1	3/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	1/1		
	F-FS3-10	S	1	2/2		
	F-FS3-11	S	1	2/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	1	3/3	8/11	73%
	F-FS4-02	S	1	2/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	2/2		
	F-FS4-05	HS	0.5	1/3		
FS5	NF-FS5-01	HS	0.5	2/3	2.5/6	42%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	0.5	0.5/1		
Total Score			**Overall %**			
29.5/48			61.5%			

Table 5.20: Scores for Sysrev.

Thoth

Table 5.21 shows the scores for Thoth.

Feature Set 1: Economic Thoth is web-based tool for SLR support and is freely available on the developers website[13]. The source code of the open-source project is also accessible via GitHub[14]. The incentive to develop such a tool, was the lack of important features in similar tools [MBR+19]. Therefore, based on the feature requirements for SLR, Thoth was developed. The tool provides many important features of the SLR process and was designed to help through all those SLR steps.

Feature Set 2: Ease of Introduction and Setup Following a registration process, the tool can be accessed online and does not need any additional requirements to install. Therefore, the introduction and setup was considerably easy. A tutorial of the tool is not available and the introduction guide about the main functionalities and setups is very briefly concepted. Hence, the introduction guide support is only partially provided.

[13] http://200.132.136.13/Thoth/

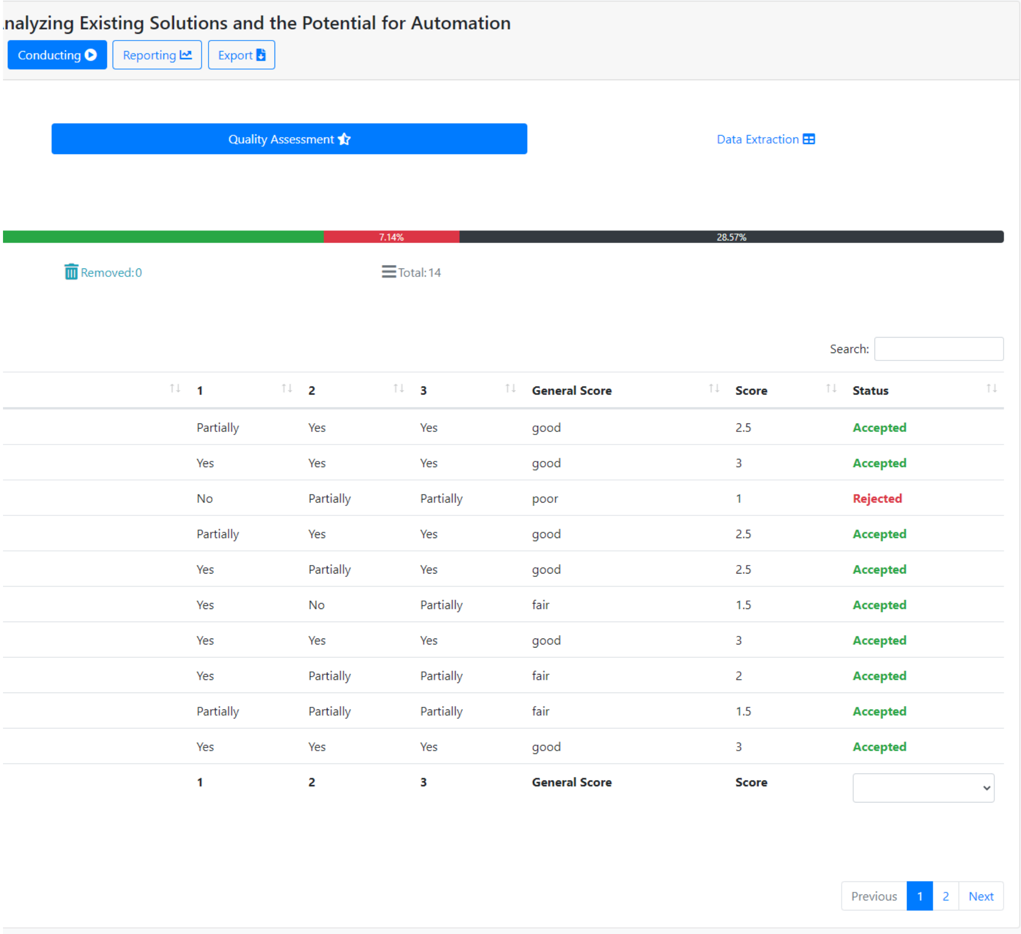
[14] https://github.com/ProjetoESE/Thoth

Feature Set 3: SLR Activity Support Thoth supports all main phases of the SLR process. The project creation starts with defining the important information for the planning phase. Accordingly, overall information, the research questions, used databases, search strings, the search strategy, inclusion/exclusion criteria, quality assessment questions and data extraction forms can be defined within the planning phase. So, Thoth does support protocol development. However, the validation part of the planning phase is not supported. Moreover, the online tool does not support automated search. The studies found from the various databases have to be imported. The imported studies from the respective databases can be categorized by database, providing an overview of the database from which the individual studies originate. Therefore, the tool provides partially the feature to search in multiple databases. For the import process, the tool supports BibTex formats. After the definition of the relevant information in the planning phase and the import of studies is finished, the conducting phase can be performed. In this phase, de-duplication is partly supported, because the tool does not identify duplicates automatically but can be labeled as duplicate. Furthermore, the screening process in the study selection with inclusion and exclusion criteria is also supported. If the paper meets the inclusion criteria, or if one of the defined exclusion criteria apply, Thoth automatically assigns the paper for accepted or rejected. The progress of the study selection process can be traced at any time. Moreover, the tool supports the quality assessment procedure. Each included paper after the study selection process, can be assessed for the quality. Here, the tool offers a wide range of options.

Therefore, to assess the relevance of the studies through the quality criteria, scores can be assigned to each quality assessment question. The minimum general score for the approvement can also be set. In Figure 5.16 the minimum general score to approve was set to 1.5. Accordingly, all papers that have scored minimum 1.5, are automatically accepted and studies that scored below 1.5, are automatically rejected. Besides, Thoth supports the extraction of quantitative data. For the extraction of data, the defined data extraction forms, which were specified in the planning phase, can be used.

Further, Thoth supports partially automated analysis. The tool illustrates several processes with graphs, which are generated from the data in the conduction of the project [Tea]. It supports a general overview of the project, which demonstrates the imported studies, duplications found, studies selected, studies that are assessed for the quality and the status of the data extraction process. As shown in Figure 5.17 the imported studies per database are depicted in a circle chart. However, the analysis of the data extraction forms does not work, therefore the automated analysis is partially supported. All of these generated analytic graphs can be exported. In addition, Thoth supports a collaborative template to assist the write-up. Thus, all relevant information from the whole review can be downloaded as Latex or can be opened directly at Overleaf [MBR+19].

Feature Set 4: Process Management Thoth allows multiple users to work on a single review and allows multiple projects to be undertaken. The tool also assists with management of roles. The access rights of other users are managed by the administrator and only users with admin rights have the ability to assign other users for certain activities. The activities of each assigned user are recorded and tracked. However, the tool does not support the ability to share data between the tasks and processes among the users.



	1	2	3	General Score	Score	Status
	Partially	Yes	Yes	good	2.5	Accepted
	Yes	Yes	Yes	good	3	Accepted
	No	Partially	Partially	poor	1	Rejected
	Partially	Yes	Yes	good	2.5	Accepted
	Yes	Partially	Yes	good	2.5	Accepted
	Yes	No	Partially	fair	1.5	Accepted
	Yes	Yes	Yes	good	3	Accepted
	Yes	Partially	Partially	fair	2	Accepted
	Partially	Partially	Partially	fair	1.5	Accepted
	Yes	Yes	Yes	good	3	Accepted
	1	2	3	General Score	Score	

Figure 5.16: Quality assessment procedure of Thoth.

Feature Set 5: Other Non-functional features Thoth uses a MVC architecture and all the features are implemented in the model layer [MBR+19]. As the source code of the tool is open-source, there are possibilities to extend the tool for further missing features. In addition, the tool is secure and follows a registration process and a log-in system. However, it seems that the tool has not been updated since the first release in 2019. Therefore, maintainability is not supported.

Buhos

Table 5.22 shows the scores for Both.

Feature Set 1: Economic Buhos is a web-based tool, which is developed in Ruby and offers functionalities to manage the whole process of the SLR. The tool is freely available on the developers website[15] and can be installed on multiple operating systems. The source code of the tool is also available via GitHub[16] and can therefore used to install the tool directly from the source code.

[15] https://www.buhos.org/manual/en/

[16] https://github.com/clbustos/buhos/tree/master/public/ViewerJS

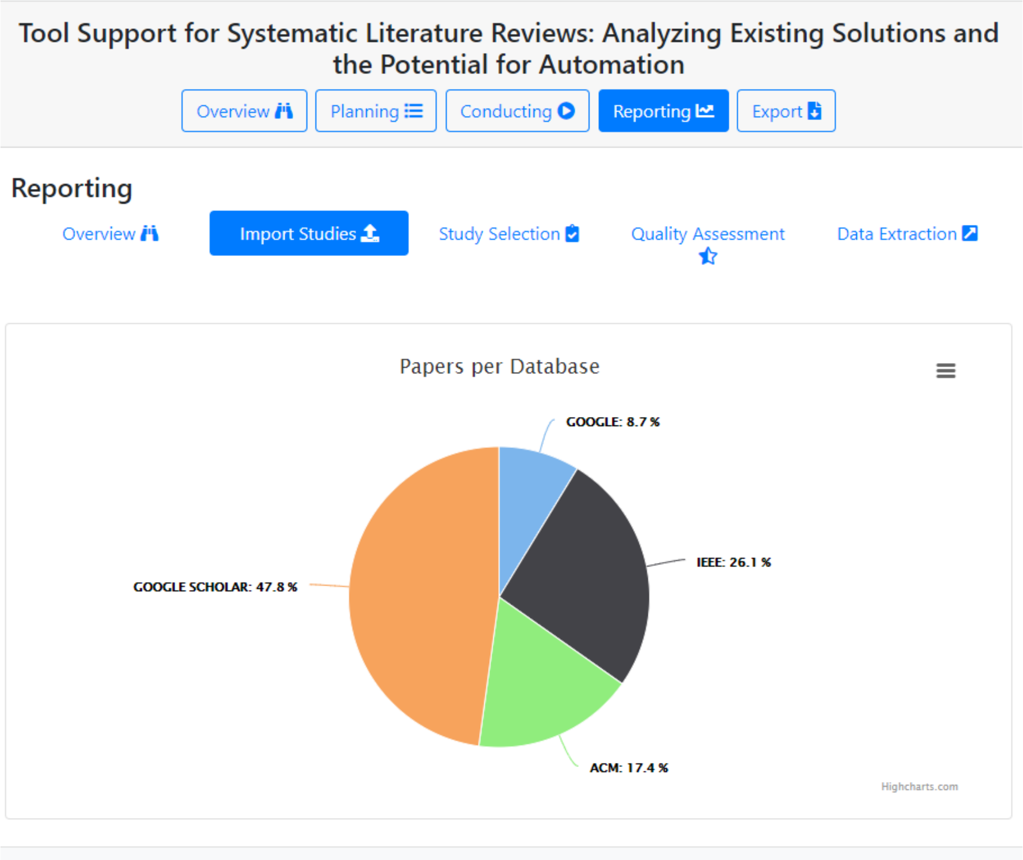


Figure 5.17: Automatic generation of statistical informations with Thoth.

Feature Set 2: Ease of Introduction and Setup Buhos does not require any advanced hardware. The tool runs on Linux, Windows and MacOS. However, it is necessary to install some compilation requirements, operating environments and dependencies to run the tool correctly. Hence, it is required to install Ruby 2.4 and development libraries for Ruby, ghost script for extraction of images from PDFs and MySQL or SQLite3 are needed. The configuration and setup of the tool was a bit challenging, as it is required beforehand, to create a SQLite database for local use, or a MySQL database for distributed use and the connection of the application to the database did not work properly in the first instance. The tool includes a configuration and installation guide and an introduction for the tool with its main features is also available.

Feature Set 3: SLR Activity Support Buhos supports a reasonable template for researchers to develop the start-up protocol for the SLR. Accordingly, the objectives, the inclusion and exclusion criteria, a time window in which the SLR is undertaken, the approaches to be adopted, can be defined in the planning phase. In addition, Buhos supports collaborative work. Therefore, reviewers can be assigned in the beginning to review, evaluate and analyze papers. Moreover, groups can be generated for the organization of the review process and each reviewer can participate in one or more reviews. After recording all the required information in the planning phase, the search stage of the conducting phase follows. The tool does not support automated search, but the searches can be directly imported in BibTex format. The obtained articles from different databases can be integrated into a common list and the tool eliminates the duplicates. Each paper that is imported, is

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	4.5/5	90%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	0.5	0.5/1		
FS3	F-FS3-01	S	0.5	1.5/2	13.5/24	56%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0.5	1/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	0.5	0.5/1		
	F-FS3-06	HS	1	3/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	1	1/1		
	F-FS3-10	S	0.5	1.5/2		
	F-FS3-11	S	0.5	1/2		
	F-FS3-12	N	1	1/1		
FS4	F-FS4-01	HS	1	3/3	8/11	73%
	F-FS4-02	S	1	2/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0.5	2/3		
FS5	NF-FS5-01	HS	0	0/3	2/6	33%
	NF-FS5-02	S	0.5	1/2		
	NF-FS5-03	N	1	1/1		
Total Score			**Overall %**			
30/48			62.5%			

Table 5.21: Scores for Thoth.

assigned as “canonical document” that corresponds to the unique reference in the system to that document. Accordingly, if diverse papers with the same text come up in different searches, Buhos assigns all such papers to one same “canonical document”. Moreover, the web-based tool supports title, abstract, reference and full-text screening of the included papers. The inclusion and exclusion process for reviewing the papers in different screening stages is also supported by the tool. The most imported feature that is supported by Buhos, is backward snowballing. Before the snowballing procedure can be performed, all relevant information must be obtained on the references cited in the included papers in the stage of reference screening [BMSD18]. Papers imported through reference manager tools do not include all this relevant information on references. Therefore, this information on references must be firstly downloaded from CrossRef to perform the backward snowballing properly. Besides, Buhos partially supports to assess quality of the included papers. Therefore, the tool enables the development of suitable quality assessment criteria. However, the evaluation of the quality cannot be performed, even though it provides a facility, to show the unresolved and resolved disagreements. After the full-text review process is completed, various information can be extracted from the data. Here, the tool supports the definition of the data extraction forms as well and to each included document can be added tags and comments. Nevertheless, Buhos does not have the feature to analyze quantitative information automatically [BMSD18].

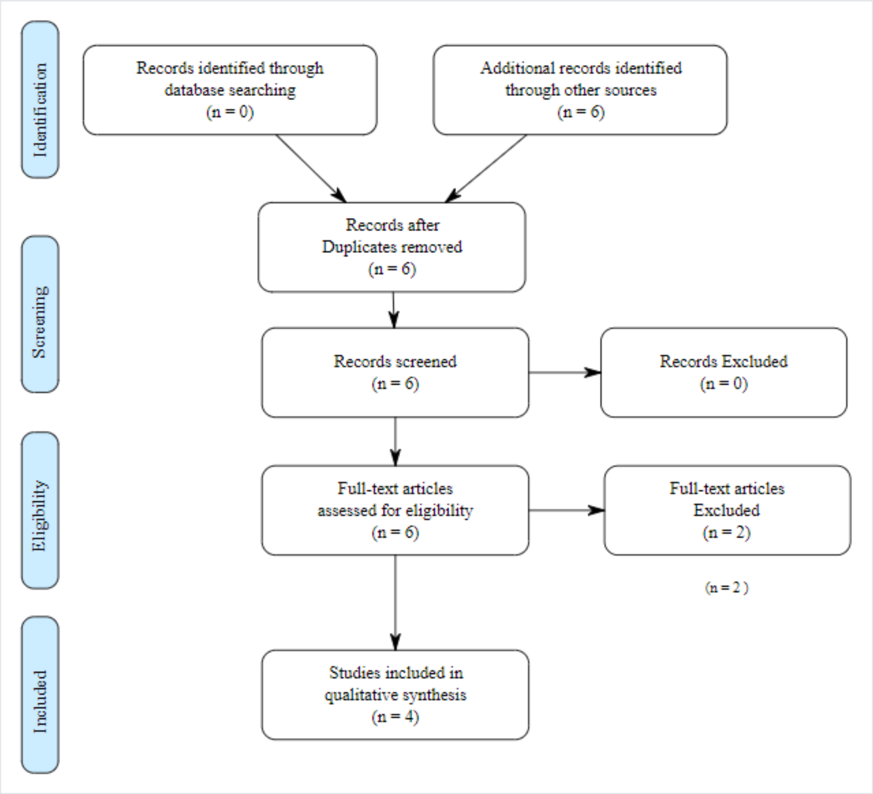

Identification
Screening
Eligibility
Included
Records identified through database searching (n = 0)
Additional records identified through other sources (n = 6)
Records after Duplicates removed (n = 6)
Records screened (n = 6)
Records Excluded (n = 0)
Full-text articles assessed for eligibility (n = 6)
Full-text articles Excluded (n = 2)
(n = 2)
Studies included in qualitative synthesis (n = 4)

Figure 5.18: PRISMA flow diagram supported by Buhos.

For the reporting phase, the tool allows to create a PRISMA flow diagram, where a summary of the papers that were included and excluded at each stage of the SLR process is indicated (see Figure 5.18) [BMSD18]. Therefore, this diagram can be used to assist the write-up. A process report can also be exported in the form of a web page or an Excel file, which includes all decisions of the reviewers of the selection process. The extracted information is also reported and shows the information for all screening stages completed by each reviewer [BMSD18].

Feature Set 4: Process Management As mentioned before, collaboration is supported by Buhos. Therefore, the tool allows multiple users work on a SLR, supports different reviewers to perform specific reviewing tasks and users can perform multiple SLRs at the same time. Moreover, the tool supports an internal messaging system for decision processing and all assignments of the reviewers are tracked and therefore the traceability is partially supported [BMSD18]. However, the tool does not support to share data among the SLR team.

Feature Set 5: Other Non-functional features The automated code review available in Code Climate[17] shows that Buhos is well maintained. In addition, the test coverage of Buhos is higher than 90% [BMSD18]. Moreover, the API of Buhos is well documented and can be used for further extensions. The web-based tool includes a log-in system and therefore the tool can be

[17] https://codeclimate.com/github/clbustos/buhos

seen as secure. However, Buhos has some usability problems and needs improvements in the user interface, especially the assignment of papers and the resolution of acceptance or rejection of several instruments [BMSD18].

Feature Set	Feature ID	Importance	Interpretation Score	Weighted Score	Feature Set Score	Feature Set Score in %
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	0.5	1/2	3/5	60%
	NF-FS2-02	S	0.5	1/2		
	NF-FS2-03	N	1	1/1		
FS3	F-FS3-01	S	0.5	1/2	15.5/24	65%
	F-FS3-02	HS	0	0/3		
	F-FS3-03	S	0.5	1/2		
	F-FS3-04	HS	1	3/3		
	F-FS3-05	N	1	1/1		
	F-FS3-06	HS	1	3/3		
	F-FS3-07	HS	1	3/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0.5	0.5/1		
	F-FS3-10	S	1	2/2		
	F-FS3-11	S	0	0/2		
	F-FS3-12	N	1	1/1		
FS4	F-FS4-01	HS	1	3/3	7/11	64%
	F-FS4-02	S	1	2/2		
	F-FS4-03	N	1	1/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0.5	1/3		
FS5	NF-FS5-01	HS	1	3/3	6/6	100%
	NF-FS5-02	S	1	2/2		
	NF-FS5-03	N	1	1/1		
Total Score			**Overall %**			
33.5/48			69.8%			

Table 5.22: Scores for Buhos.

REviewER

Table 5.23 shows the scores for REviewER.

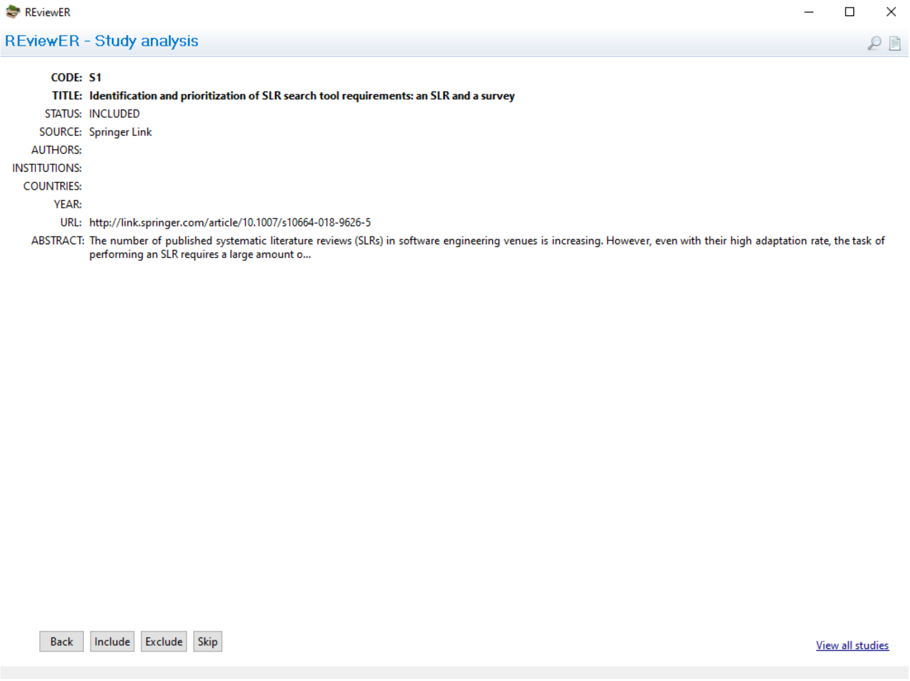
Feature Set 1: Economic REviewER helps researchers to assist the conducting phase of SLR. The binaries for Linux, Windows and MacOS can be downloaded freely on the delevoper's website[18]. The open-source code is also available via GitHub[19].

[18] https://sites.google.com/site/eseportal/tools/reviewer

[19] https://github.com/bfsc/reviewer

Feature Set 2: Ease of Introduction and Setup The setup of the tool was not complicated. All you have to do is to install the downloaded binary. After the installation, the application can be executed. Therefore, the tool does not require any advanced hardware or software for the developed functionalities. However, an introduction guide or a guideline for the usability is not offered.

Feature Set 3: SLR Activity Support REviewER does not support any functionalities for the protocol development in the planning stage of SLR. The tool was especially developed to provide an automated search function which enables the search in multiple databases for computer science. However, the automated search does not work as well as indicated in the developer's website. The automatic search feature of the tool does only work for Springer Link, although it is stated that the automated search works for ACM Digital Library, IEEE Xplore, Scopus, Engineering Village, Springer Link and Science Direct. Moreover, the entered query strings for the automated search do not perform orderly. Accordingly, boolean operators, like AND or OR to specify the search, are not supported and the entered query string must be similar to an existing paper reported in Springer Link. Moreover, only a small subset of papers is fetched from the database and therefore relevant literature can be missing for the research. In consequence of this, the automated search is partially supported.



CODE: **S1**
TITLE: **Identification and prioritization of SLR search tool requirements: an SLR and a survey**
STATUS: INCLUDED
SOURCE: Springer Link
AUTHORS:
INSTITUTIONS:
COUNTRIES:
YEAR:
URL: http://link.springer.com/article/10.1007/s10664-018-9626-5
ABSTRACT: The number of published systematic literature reviews (SLRs) in software engineering venues is increasing. However, even with their high adaptation rate, the task of performing an SLR requires a large amount o...

Figure 5.19: Screening process of REviewER.

After the results from the automated search are imported, the papers can be screened for the inclusion and exclusion criteria. The screening process of papers is also partially supported, because the tool does only support a small subset of meta information about the papers and the abstracts are not displayed correctly. As shown in Figure 5.19, in this process, the papers can be labeled as excluded or included. However, REviewER does not support pilot of inclusion and exclusion criteria and a semi-automated selection of studies is also not provided. After the selection procedure is completed, the results can be exported as an Excel file.

Feature Set 4: Process Management None of the process management features are supported by REviewER. Thence, the collaborative work is not supported. In addition, the tool does not allow to share data and traceability is also not provided.

Feature Set 5: Other Non-functional features The last release of the tool was in 2013. Since then, there have been no updates for the tool, even though the tool has some bugs and some features are not working orderly. Because of this, REviewER seems to be outdated and not maintained. Moreover, information to extend the tool could not be found.

Feature Set	**Feature ID**	**Importance**	**Interpretation Score**	**Weighted Score**	**Feature Set Score**	**Feature Set Score in %**
FS1	NF-FS1-01	S	1	2/2	2/2	100%
FS2	NF-FS2-01	S	1	2/2	4/5	80%
	NF-FS2-02	S	1	2/2		
	NF-FS2-03	N	0	1/1		
FS3	F-FS3-01	S	0	0/2	5.5/24	23%
	F-FS3-02	HS	0.5	1/3		
	F-FS3-03	S	0	0/2		
	F-FS3-04	HS	0.5	1.5/3		
	F-FS3-05	N	0	1/1		
	F-FS3-06	HS	0.5	2/3		
	F-FS3-07	HS	0	0/3		
	F-FS3-08	N	0	0/1		
	F-FS3-09	N	0	0/1		
	F-FS3-10	S	0	0/2		
	F-FS3-11	S	0	0/2		
	F-FS3-12	N	0	0/1		
FS4	F-FS4-01	HS	0	0/3	0/11	0%
	F-FS4-02	S	0	0/2		
	F-FS4-03	N	0	0/1		
	F-FS4-04	S	0	0/2		
	F-FS4-05	HS	0	0/3		
FS5	NF-FS5-01	HS	0	0/3	0/6	0%
	NF-FS5-02	S	0	0/2		
	NF-FS5-03	N	0	0/1		
Total Score			**Overall %**			
11.5/48			24%			

Table 5.23: Score for REviewER.

5.3.4 Discussion of the Results

This section shows a discussion of the feature analysis spotlighting the main strengths and weaknesses of the evaluated tools. Therefore, an overview of the overall scores is presented, to highlight the tools which cover the SLR process best or worst. Moreover, features that are considered as highly significant from the perspective of the researchers, obtained from the semi-structured interviews, are summarized to perceive how well these features are covered by the tools.

Table 5.24 shows an overview of the feature set scores and overall scores of the evaluated tools. Buhos achieves the highest overall score with 69.8% and therefore can be considered as the most suitable tool to support SLR in SE. The main strengths of Buhos are:

- provides collaborative work
- eliminates automatically duplications
- most of the activities in the conducting stage are supported
- supports backward snowballing
- the tool is well maintained and the API is well documented

Buhos main weaknesses is especially the complex installation, as it is required to install some compilation requirements and dependencies to run the tool correctly. Moreover, the user interface of Buhos is structured a bit complicated and has some usability problems. Therefore, the tool needs some improvements in the user interface.

		F1		**F2**		**F3**		**F4**		**F5**		**Overall score**	
		(scores out of 2)		**(scores out of 5)**		**(scores out of 24)**		**(scores out of 11)**		**(scores out of 6)**		**(scores out of 48)**	
1	**Buhos**	2	100%	3	60%	15.5	65%	7	64%	6	100%	33.5	69.8%
2	**Thoth**	2	100%	4.5	90%	13.5	56%	8	73%	2	33%	32	62.5%
3	**Sysrev**	1	50%	5	100%	13	54%	8	73%	2.5	42%	29.5	61.5%
4	**Parsifal**	2	100%	4.5	10%	12	50%	6	55%	4	67%	28.5	59.4%
5	**ReLis**	2	100%	3.5	70%	9	38%	9	81%	1.5	25%	25	52.1%
6	**ASReview**	2	100%	5	100%	7	29%	1	9%	4	66%	19	39.6%
7	**SurVis**	2	100%	4	80%	7	29%	2	18%	1	17%	16	33.3%
8	**SLR-Tool**	2	100%	4.5	90%	5	21%	1	9%	1	17%	13.5	28.1%
9	**FASTREAD**	2	100%	5	100%	6	25%	0	0%	0.5	8%	13.5	28.1%
10	**REviewER**	2	100%	4	80%	5.5	23%	0	0%	0	0%	11.5	24%

Table 5.24: Feature set scores and overall scores.

The second most suitable tool to support SLR with an overall score of 62.5% is Thoth. Its main strengths are:

- provides team-based support for the whole SLR process
- UI is well structured and organized
- wide range of options for inclusion/exclusion criteria and quality assessment
- good support of a template to assist the write-up

One of Thoth's weaknesses is that the automated analysis of extracted data does not work, even though it provides data extractions forms. The tool can illustrate several processes with graphs to get an overview of the selected studies, duplications and studies that are assessed for the quality, but an automated analysis of the extracted data is unfortunately not supported. Additionally, Thoth does not seem to be well maintained. as there cannot be found any updates since the last release.

Sysrev occupies the third place of appropriate tools to support the SLR process with an overall score of 61.5%. Sysrev is especially conceptualized for the healthcare discipline but can also used for the SE context with the disadvantage that the supported automated search is only provided for the medicine context and not for other disciplines. However, the tool supports good features for the SLR process. The main strengths of Sysrev are:

- good support for the conducting stage with automatic duplicate detection
- provides several automatically generated analytics
- supports collaboration and tracks member activities

The main weakness of Sysrev is that it requires a payment to use the advanced features, like analytics features or the ability to create unlimited private projects. As long as you use the free basic version of the web-based tool, the created projects are public. Moreover, the tool supports only the conducting phase of SLR's. Templates to assist protocol development or report development are not supported.

A similar overall score to Sysrev has been achieved by Parsifal with an overall score of 59.5%. The online tool was designed for the SE context and helps through all SLR steps. The main strengths of Parsifal are:

- provides a step-by-step guide and helps to remind important elements during the SLR process
- semi-automated search, where the search queries automatically access to the databases
- provides collaborative work

However, the results from the semi-automated search in Parsifal cannot be integrated and used for the study selection. Studies to review for the study selection stage must be imported. Therefore, this can be seen as a main weakness, even though a semi-automatic search is provided.

ReLis and ASReview are two tools that offer similar support. While ReLis has an overall score of 52.1% , ASReview has an overall score of 39.6%. Both tools provide good support for the conducting stage, where the imported papers can be screened for the inclusion and exclusion criteria. Moreover, both tools partially support features for data extraction and automated analysis of quantitative data. The difference of ASReview and ReLis is that ReLis supports additionally collaborative work, while this is not provided by ASReview. Yet, the setup of ReLis is a quite complex, as it is required to set up a configuration model by defining the semantic methods to use the supported features. On the other hand, the setup of ASReview is relatively simple.

SurVis is developed for the purpose of analyzing and structuring literature collections and is not conceptualized as a tool to support SLR. Consequently, the tool offers great support for visual analysis of imported set of papers but except of the analysis options and the screening of the imported papers, the support for SLR activity features or process management features are quite limited. Therefore, SurVis has an overall score of 33.3%.

The SLR activity features of SLR-Tool by A. Hinderks and FASTREAD are quite limited. Both tools provide a facility to screen the imported papers for the inclusion and exclusion procedure. Because of this limited support, both tools have an overall score of 28.1%.

REviewER has the lowest overall score of 24%. Even if the tool has promising features which were described on the developer's website [Mem], the tool fails to implement them effectively. An automated search function which enables the search in multiple databases within the SE context, was a primary objective of REviewER. However, the automated search does not work well as indicated on the developer's website and the features for the study selection process are partially supported. Due to this, most of its support for SLR process is quite limited.

	Automated Search (F-FS3-02)	**Abstract/Text screening (F-FS3-04)**	**Inclusion/Exclusion (F-FS3-06)**	**Snowballing (F-FS3-07)**	**Support for multiple users (F-FS4-01)**	**Traceability (F-FS4-05)**	**Maintainability (NF-FS5-01)**
Parsifal	Partially	Yes	Yes	No	Yes	Partially	Yes
Buhos	No	Yes	Yes	Partially	Yes	Partially	Yes
ReLis	No	Yes	Yes	No	Yes	Yes	No
Sysrev	No	Yes	Yes	No	Yes	Partially	Partially
Thoth	No	Yes	Yes	No	Yes	Partially	No
ASReview	No	Yes	Partially	No	No	No	Yes
FASTREAD	No	Yes	Partially	No	No	No	Partially
SLR-Tool	No	Yes	Partially	No	No	No	No
SurVis	No	Yes	No	No	No	Partially	No
REviewER	Partially	Partially	Partially	No	No	No	No

Table 5.25: Highly significant features supported by the tools.

Table 5.25 shows the highly significant features supported by the tools. As you can see, support for an automated search is very limited and can be considered as poorly supported. Only Parsifal and REviewER provide partially an automated search. Furthermore, the snowballing procedure is only supported by Buhos. However, this feature is not fully supported by Buhos, as it only allows to perform backward snowballing. The screening stage is in generally supported by all evaluated tools. There are some tools which have problems to screen the abstracts or meta-information of the imported papers correctly but these bugs were not considered as critical. However, these small errors could be improved. The study selection process with the inclusion and exclusion criteria is supported by most of the tools. SurVis is the only tool that does not support this feature. Support for collaboration is provided by half of the analyzed tools. Accordingly, Parsifal, Buhos, ReLis, Sysrev and Thoth provide effective facilities for collaborative work amongst multiple researchers. Moreover, support for traceability is also limited. Most of the tools have a different definition of traceability and therefore support this feature only partially. ReLis is the only tool which has effective support for traceability. Furthermore, the maintainability of the tools is also restricted, as there could not be found information on maintainability for most tools. Parsifal, Buhos and ASReview are the only tools which are well maintained and regularly updated. Worthy of note is that Parsifal supports all features that were assigned as highly significant, except the snowballing feature, although it occupies the third place in the overall score (see Table 5.24). Within the SE context, Parsifal supports the most important requirements, but the support for an integrated search is still missing.

6 Discussion

With the conducted semi-structured interviews, several observations made by the participants associated to the SLR process were recognized. The observations showed a series of challenging and problem related activities that appear during the SLR process. In relation to these activities, different tool requirements were identified by the participants which are important for the SLR tool support. Subsequently, these requirements were adapted to the feature analysis to evaluate a series of existing tools. The feature analysis had the intention to analyze and compare these tools with respect to the supported features and steps of the SLR process. The results of the feature analysis show that there are existing tools to support SLR, but the support is still limited. For most of the tools, the limitations are common. The support for protocol development is only assisted effectively by Parsifal, Thoth and Buhos. Furthermore, full support for automated search is still lacking in all evaluated tools, which is a frequently stressed issue in the area of SLR research [AC19; HCHA16]. In addition, the support for including grey literature, which receives more and more attention for recent topics within the SE context [GFM19], is not supported by any tool. According to the conducting phase, most of the tools support the screening and study selection procedure. In general, each tool has its own strengths. However, a holistic tool that automates various tasks for the entire SLR process is still missing, especially to reduce the manual effort of conducting a SLR and simultaneously to realize these tasks reliable. Therefore, in this chapter the most important areas for the automation of SLR will be discussed to suggest a holistic tool design by combining existing tools that can make certain steps more efficient and reliable. Finally, important factors that could be a threat to the validity of the conducted studies and the performed feature analysis will be explained.

6.1 Automatable SLR Activities (RQ 3)

As confirmed in the interviews and presented in related works, the SLR process is time and effort consuming due to the large amount of manual tasks [Al-17]. Therefore, automation approaches would reduce the effort and SLR researchers would benefit by these provided features in a SLR tool. However, it is important to note that the SLR process includes creative tasks for decision making processes as well as tasks that can be performed automatically. In consequence to this, it is important to separate these tasks which could and could not be automated. This section discusses the SLR activities that have the most potential for automation.

6.1.1 Development of the Protocol

The protocol development is one of the challenging activities [BKB+07] and must be created in a qualitative and creative way to ensure a high-quality SLR. Especially the research questions drive the entire review process and are therefore very important. An automation of this activity is very difficult, as it requires creative and decision making steps for the protocol development which drives the entire SLR. However, an automated support through text analysis can help to identify existing SLR's which largely cover the same defined research questions [FT20]. As it is necessary to identify the need for a SLR, this automation support can reduce the effort to establish any gaps or existing SLR's which are of insufficient quality. Moreover, to avoid research bias and to conduct the SLR in a systematic way, it is important to follow the guidelines of the SLR methodology. In this situation, this activity can also be developed as a "recipe" to check automatically for consistency and integrity and to ensure that all relevant artefacts of the review protocol are formally defined in a consistent and unbiased way [TGC+14].

Parsifal, Buhos and Thoth are three SLR tools that provide a good protocol template. Therefore, these tools include standard fields that remind the researcher to cover all important components to reach the desired level of empirical rigor.

6.1.2 Conducting the Search

According to the undertaken interviews, most of the problems occur during the search stage. One of the problems is the definition of the search string. For researchers, it is often difficult to combine keywords and to find appropriate synonyms that fit the search purpose. One reason for this is the inconsistent terminology within SE, which often overlaps with other domains [AC19]. Therefore, an automated search strategy by using the PICOC (Population, Intervention, Comparison, Outcome, Context) framework [Kee07] could be beneficial to derive appropriate search strings from the defined research questions. This proposal can also ensure that the search strings are consistent with the research questions [TGC+14].

Another issue is the combined use of academic and non-scientific search engines. Often, grey literature is also included additionally to find informative material for the research that is available outside the literature databases. Due to this, the variation of the query languages among the search engines is too large. An automation support which converts the search strings to an appropriate format for each search engine and the import/export of studies in a standardized format would bring great benefit in terms of the overall time and effort reduction [AC19; FT20].

However, an automated search within a SLR tool is still lacking. As mentioned above, this poor support is related to digital libraries which mostly block the integrated search of the third-party tools [AC19] and therefore, it is difficult for the developed SLR tools to support this feature.

Parsifal is the only tool that provides a semi-automated search, where the defined search strings automatically access the databases. However, Parsifal is only integrated with Scopus and ScienceDirect and the results from the integrated search cannot be used for the study selection. In addition, Thoth is the only tool that converts a generic search string to an appropriate format for each search engine. The conversion works for ACM Digital Library, IEEE Xplore, Google, Google Scholar and more search engines.

6.1.3 Study Selection

The most challenging part in the study selection procedure is to assess the studies for the inclusion/exclusion criteria. Reading through all abstracts and texts and deciding to include or exclude the papers requires high effort and is one of the time-consuming tasks. For this activity, an automated decision support can help to highlight important information for researchers to simplify the task [TGC+14]. Furthermore, the speed of the study selection process can be increased with machine learning approaches.

FASTREAD is for example a tool which uses active learning algorithms and classification models to automatically classify the studies, after a number of studies have been trained manually. Such an approach can assist to minimize reading efforts. However, an automated classification of studies can be error-prone and still requires a human reviewer to make the final decision of the included or excluded studies. Due to this, the machine learning approach does not dramatically reduce the time on this task [OTT+19].

An automatic de-duplication of studies has also the potential to save time and to reduce the effort. This approach exists already in most of the reference manager tools and can be well adapted in a SLR tool. Mostly, information retrieval techniques and string matching algorithms are used to detect duplicates and to remove them automatically [AMV+15]. An automatic de-duplication is provided by Sysrev and Buhos.

As already found out in the interviews, the snowballing procedure is also a laborious and time-consuming task. Therefore, this task has potential for automation, especially with respect to automatically extracting the references from the selected studies [FT20]. According to the publication by Tsafnat et al. [TGC+14], this approach can increase the recall of more followed citations than a human being would follow. However, as it is required to find the correct set of papers for the manual snowballing procedure, this issue must also be applied to the automatic snowballing. In addition, with an automatic snowballing system, there is the danger that it can get out of control because too many studies are retrieved that are not actually important for the research. Due to this, it is important to develop additional stopping criteria where the iterations can be also defined [TGC+14].

Buhos is the only SLR tool that enables backward snowballing. After choosing a set of papers for the snowballing procedure, the tool uses CrossRef to obtain all relevant information on the references cited in the selected set of papers and searches automatically for common references [BMSD18].

6.1.4 Data Extraction and Synthesis

The extraction of data is one of the error-prone SLR activities, as it is a subjective process and different researcher do it in different ways. Minimizing the risk of incorrect data extraction requires that this step is performed in an accurate and qualitative way. Due to this, an automated extraction of high-quality data is not feasible. Nevertheless, an automated extraction and clustering methods for simple meta-information of the references and the creation of automated visualizations of these data seems to be possible. Felizardo et al. mention that the use of Visual Text Mining (VTM), especially in the SMS process, can reduce the effort of creating such visualizations [FNF+10].

For example, Sysrev provides automated analytic forms of basic data and gives a statistical overview of the overall research. Therefore, the tool screens the included and excluded papers, tracks member activities, shows the recent progress of the research, counts the answers for the data extraction forms and displays the most important terms, which appear in the articles. SurVis is also a visual analysis tool which automatically analyzes and structures literature collections based on the meta-information of the included references.

6.1.5 Reporting the Findings

Writing-up the review protocol is also a tedious task that requires more creative interventions. However, this activity involves sub-tasks that can be automated, such as a template creation to assist the write-up, typesetting figures or laying out the text [TGC+14].

A template to assist the write-up is for example supported by Thoth. All relevant information, formulated in the review protocol, like the objectives, research questions, search strategy, selection criteria, quality assessment and extraction forms can be downloaded as a Latex file. The automated creation of a report from the protocol can help to organize the write-up and to ensure that all important elements of the research are complete and consistent [TGC+14].

6.1.6 Collaboration

To reduce single researcher bias, one of the key elements of a SLR is collaborative work. Therefore, collaboration support in a SLR tool is needed, especially to reduce overlaps in the results and for conflict resolution among the SLR team. With respect to this, a traceability feature would be also beneficial where certain tasks of involved researchers are tracked.

The collaboration support is provided by half of the evaluated tools. For example, Buhos allows multiple users to work on a SLR, supports different reviewers to perform specific reviewing tasks and users can perform multiple SLRs at the same time. In addition, the tool provides an internal messaging system for decision processing and all assignments of the reviewers can be tracked. However, by most of the tools, the collaborative support does not quite fulfil its purpose. Working productively and asynchronously in a SLR team is still not ideally supported and requires specialized tool support.

6.1.7 Summary

Figure 6.1 summarizes automatable tasks for the whole SLR process as a combination of existing tools that were evaluated in the feature analysis. Such a tool chain plus additional automatable tasks that were discussed in this section would provide improved support for the entire SLR process, especially to reduce the effort of time-consuming activities while obtaining evidence.

It is noteworthy that none of the single tools given in the chain support the entire SLR process and all automatable tasks depicted in Figure 6.1. Nevertheless, for the use of a single tool Buhos and Parsifal are two tools that are more on the holistic side as they support the largest number of important features (see Section 5.3.4. If a researcher would like to undertake a SLR with a single tool, Buhos or Parsifal are recommended.

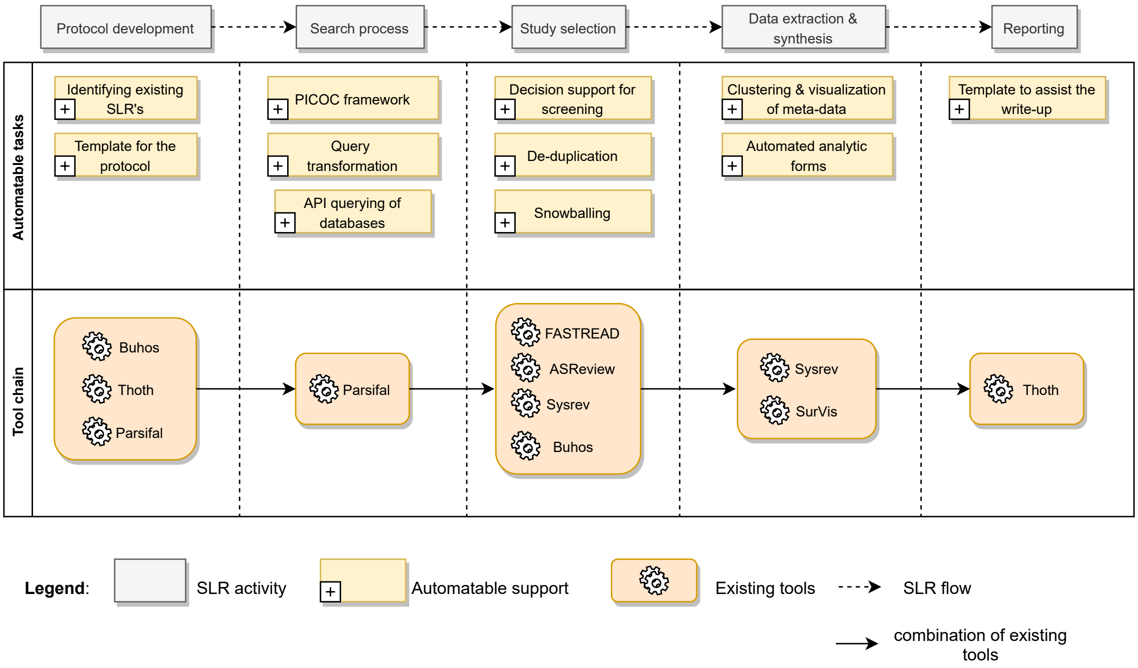


Figure 6.1: Automatable tasks and chain of existing tools.

6.2 Threats to Validity

In the following section, important factors that could be a threat to the validity of the conducted studies and the performed feature analysis will be explained.

Rapid Review

Due to its lightweight methodology, the undertaken RR leads to more threats to validity than traditional SLR's. One of the threats can be an insufficient search strategy that return undesirable research results and therefore not return the intended papers. To avoid this risk, a research protocol was developed and five search engines were used (see Chapter 4). However, as the number of the results returned was too large for Google Scholar, Google and ACM Digital Library, limits were set in order to provide a manageable amount of work. Therefore, there is the risk that not all relevant papers were detected. To mitigate this risk, snowballing was used and integrated additionally into the search strategy. Moreover, the entire selection procedure was conducted by a single researcher, which might introduce selection bias and a quality assessment on the selected studies was also not conducted. This might limit the reliability on the evidence.

Semi-Structured Interviews

Semi-structured interviews require strong communication skills to guide the interview process and to get all important information from the interviewees to answer the research questions. Due to this, there is the possibility that the quality of the data collected may be limited. To counter this threat, an interview guide was developed to organize and direct the interviews. Moreover, the general aim of the performed interviews were to explore the experiences and challenges that researchers are facing while performing SLR. Therefore, SE researchers with knowledge and experiences with any kind of literature reviews and researchers who had at least performed one SLR were involved. However, the participants might not have reported whatever they could remember which means that the results of interviews are based on the recollection of the participants. This might be a weakness but nevertheless, the results of the interviews were confident because most of the respondents were currently research associates who had experiences with the SLR methodology to a diverse set of topics in SE. In addition, the number of involved participants was seven. A higher number of participants could provide more reliability. Nonetheless, the involved participants were researchers with knowledge about the SLR process and who had at least performed one SLR for various SE topics. According to this, important data was obtained from the participants, which was rich enough. Another threat to validity can be the inaccuracy and incompleteness in hearing the respondents. To minimize this risk, the conducted interviews were recorded and afterwards transcribed.

Feature Analysis

The main threat to validity of the feature analysis is that the evaluation of the selected tools was a subjective process according to the author's judgment. However, strict attention was paid for the development of the analysis method. Therefore, feature sets with importance weightings and interpretation procedures for the scoring process based on the literature and the conducted interviews were developed. The scoring process was also a subjective task. Specially the interpretation of evaluation for partially supported features, wherethe features can score from 0.5 to 2.5. This kind of interpretation can lead to subjective bias. Nevertheless, care has been taken to ensure that the evaluation of each tool was proceeded in a consistent and fair way with respect to the developed criteria.

7 Conclusion

In this chapter, a summary of this thesis, recommendations and an outlook on possible future work is presented.

7.1 Summary

The main objectives of this thesis were to identify, collect and analyze existing tools that support certain steps of the SLR process and to investigate the challenges and experiences of SE researchers that arise during the SLR process. Using these results, the aim was to identify and analyze the most important areas for automation of the SLR process with tool support. To achieve these goals, research questions were defined in the research protocol (see Chapter 4).

To address the research questions **RQ 1 - RQ 1.2**, a RR was conducted to get insight into the current state of research regarding tools to support SLR in SE. Regarding **RQ 1**, 24 tools were identified that support the systematic review process in SE research. Concerning to **RQ 1.1** and **RQ 1.2** the supported SLR stages and the most important feature of these tools were discovered. The results showed that the most supported stage was the conducting phase. Therefore, abstract/text screening and the inclusion/exclusion procedure were the most supported features by the identified tools. Least supported features were automated search, snowballing and de-duplication. A brief overview of the identified tools with tool characteristics, supported stages and underlying approaches are presented in Table 5.2.

In **RQ 2** (including **RQ 2.1 - RQ 2.4**), the practices and experiences of SE researchers with SLR were investigated through semi-structured interviews. Therefore, 7 interviews were conducted with SE researchers who have knowledge of the SLR process. The results showed that one of the biggest problems of the SLR was the time-consuming nature of the SLR process. From the respondent's point of view, most of the complaints were related to the conducting stage. Furthermore, the most manual and effort-related acitivities occur during the study selection and the most error-prone activities occur while extracting data. Besides the experiences with conducting SLR's, participants' experiences with SLR tools were also investigated. However, the mentioned tools by the participants were mostly reference manager or productivity tools. Specific SLR tools could therefore not be detected. Based on the challenges of the participants, SLR requirements and features which have from the respondent's perspective the most potential for automation were inspected. The results showed that the most important requirements related to the SLR activities were automated search, inclusion/exclusion, screening procedure, snowballing and data extraction. Related to the process management and non-functional requirements, collaboration, traceability and maintainability were rated as the most important requirements. Regarding the automation potential of these activities, the opinions were different. However, the participants agreed that better tool support is needed to reduce the time-consuming effort of the SLR process.

Regarding **RQ 1.3**, a set of selected tools were analyzed and evaluated to find out the current degree of SLR practices in SE supported by existing tools. For this purpose, the requirements identified from the interviews and the literature were adapted to a feature analysis to evaluate these tools. The results from the feature analysis showed that there are existing tools that support certain features but an overall support is still limited. Especially an integrated search could not be identified in any tool.

Based on the results from **RQ 1** and **RQ 2**, in **RQ 3**, the most important areas for automation in order to suggest a holistic tool design from combination of existing tools were discussed. The results of the discussion part showed that a tool chain in combination of Buhos, Thoth, Parsifal, FASTREAD, ASREview, Sysrev and SurVis would provide a better support for the entire SLR process.

7.2 Recommendations and Future Work

Even if there exist tasks which have a great potential for automation, there are a few points that need to be considered. One point is the support of an integrated search in a SLR tool which is lacking in all of the evaluated tools. This feature was considered skeptically by the participants of the interviews and is a frequently stressed issue in the area of SLR research [AC19; HCHA16]. The poor support for this feature may be a consequence of the inherent difficulties associated with the digital libraries [BZB+07]. Often, the digital libraries block the integrated search and therefore it is difficult for the developed tools to support SLR to integrate this feature [AC19]. This problem has to be first addressed before an effective tool to support SLR can be developed.

As a future work, the automatable tasks discussed in this thesis and the identified requirements can be investigated in more detail to develop a SLR tool within the SE discipline which supports the whole process. Considering the usually time-consuming effort of the SLR process, the successful development of such a tool would be beneficial.

However, another point which has to be considered is the mistrust of such automation tools, as the SLR requires creative tasks where the human still needs to make decisions. Moreover, trust usually fails if the tool is not well maintained. Therefore, an automation of a SLR tool which supports the whole process needs an open community which is further developed and kept up-to-date by the active community.

Bibliography

[10] "SLR-TOOL - A Tool for Performing Systematic Literature Reviews:" in: *Proceedings of the 5th International Conference on Software and Data Technologies*. 5th International Conference on Software and Data Technologies. University of Piraeus, Greece: SciTePress - Science, 2010, pp. 157–166. ISBN: 978-989-8425-22-5 978-989-8425-23-2. DOI: 10.5220/0003003601570166. URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0003003601570166 (visited on 10/21/2020) (cit. on pp. 29, 60).

[12] "Managing Literature Reviews Information through Visualization:" in: *Proceedings of the 14th International Conference on Enterprise Information Systems*. 14th International Conference on Enterprise Information Systems. Wroclaw, Poland: SciTePress - Science, 2012, pp. 36–45. ISBN: 978-989-8565-10-5 978-989-8565-11-2 978-989-8565-12-9. DOI: 10.5220/0004004000360045. URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0004004000360045 (visited on 10/28/2020) (cit. on pp. 29, 60).

[14] "Slrtool: A Tool to Support Collaborative Systematic Literature Reviews:" in: *Proceedings of the 16th International Conference on Enterprise Information Systems*. 16th International Conference on Enterprise Information Systems. Lisbon, Portugal: SCITEPRESS - Science, 2014, pp. 440–447. ISBN: 978-989-758-027-7 978-989-758-028-4 978-989-758-029-1. DOI: 10.5220/0004972204400447. URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0004972204400447 (visited on 10/21/2020) (cit. on p. 60).

[AC14] A. Al-Zubidy, J. C. Carver. "Review of Systematic Literature Review Tools". In: (2014), p. 4 (cit. on pp. 23, 29, 59).

[AC19] A. Al-Zubidy, J. C. Carver. *Identification and prioritization of SLR search tool requirements: an SLR and a survey*. Vol. 24. 1. Empirical Software Engineering, 2019, pp. 139–169. ISBN: 1066401896. DOI: 10.1007/s10664-018-9626-5 (cit. on pp. 18, 23, 25–29, 39, 97, 98, 104).

[Al-17] A. Al-Zubidy. "Vision for SLR tooling infrastructure: Prioritizing value-added requirements". In: *Information and Software Technology* (2017), p. 10 (cit. on pp. 23, 28, 97).

[AMV+15] R. Abilio, F. Morais, G. Vale, C. Oliveira, D. Pereira, H. Costa. "Applying Information Retrieval Techniques to Detect Duplicates and to Rank References in the Preliminary Phases of Systematic Literature Reviews". In: *CLEI Electronic Journal* (Aug. 1, 2015). ISSN: 0717-5000. DOI: 10.19153/cleiej.18.2.2. URL: http://www.clei.org/cleiej/index.php/cleiej/article/view/394 (visited on 10/27/2020) (cit. on p. 99).

[BB06] D. Budgen, P. Brereton. “Performing systematic literature reviews in software engineering”. In: *Proceeding of the 28th international conference on Software engineering - ICSE ’06*. Proceeding of the 28th international conference. Shanghai, China: ACM Press, 2006, p. 1051. ISBN: 978-1-59593-375-1. DOI: 10.1145/1134285.1134500. URL: http://portal.acm.org/citation.cfm?doid=1134285.1134500 (visited on 12/13/2020) (cit. on p. 15).

[BHB12] D. Bowes, T. Hall, S. Beecham. “SLuRp: a tool to help large complex systematic literature reviews deliver valid and rigorous results”. In: *Proceedings of the 2nd international workshop on Evidential assessment of software technologies - EAST ’12*. the 2nd international workshop. Lund, Sweden: ACM Press, 2012, p. 33. ISBN: 978-1-4503-1509-8. DOI: 10.1145/2372233.2372243. URL: http://dl.acm.org/citation.cfm?doid=2372233.2372243 (visited on 10/21/2020) (cit. on pp. 29, 61).

[BKB+07] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, M. Khalil. “Lessons from applying the systematic literature review process within the software engineering domain”. In: *Journal of Systems and Software* 80.4 (Apr. 2007), pp. 571–583. ISSN: 01641212. DOI: 10.1016/j.jss.2006.07.009. URL: https://linkinghub.elsevier.com/retrieve/pii/S016412120600197X (visited on 12/14/2020) (cit. on pp. 16–19, 23, 24, 26, 97).

[BKW16] F. Beck, S. Koch, D. Weiskopf. “Visual Analysis and Dissemination of Scientific Literature Collections with SurVis”. In: *IEEE Transactions on Visualization and Computer Graphics* 22.1 (Jan. 31, 2016), pp. 180–189. ISSN: 1077-2626. DOI: 10.1109/TVCG.2015.2467757. URL: http://ieeexplore.ieee.org/document/7192633/ (visited on 10/21/2020) (cit. on pp. 80, 81).

[BLK+09] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, S. Hellmann. “DBpedia - A crystallization point for the Web of Data”. In: (2009), p. 12 (cit. on pp. 29, 59).

[BMSD18] C. Bustos Navarrete, M. G. Morales Malverde, P. Salcedo Lagos, A. Díaz Mujica. “Buhos: A web-based systematic literature review management software”. In: *SoftwareX* 7 (Jan. 2018), pp. 360–372. ISSN: 23527110. DOI: 10.1016/j.softx.2018.10.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352711018300293 (visited on 10/21/2020) (cit. on pp. 89–91, 99).

[BS18] B. Bigendako, E. Syriani. “Modeling a Tool for Conducting Systematic Reviews Iteratively:” in: *Proceedings of the 6th International Conference on Model-Driven Engineering and Software Development*. 6th International Conference on Model-Driven Engineering and Software Development. Funchal, Madeira, Portugal: SCITEPRESS - Science and Technology Publications, 2018, pp. 552–559. ISBN: 978-989-758-283-7. DOI: 10.5220/0006664405520559. URL: http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006664405520559 (visited on 10/21/2020) (cit. on pp. 74–76).

[BZ09] M. A. Babar, H. Zhang. “Systematic literature reviews in software engineering: Preliminary results from interviews with researchers”. In: *2009 3rd International Symposium on Empirical Software Engineering and Measurement*. 2009 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM). Lake Buena

Vista, FL, USA: IEEE, Oct. 2009, pp. 346–355. ISBN: 978-1-4244-4842-5. DOI: 10.1109/ESEM.2009.5314235. URL: http://ieeexplore.ieee.org/document/5314235/ (visited on 01/29/2021) (cit. on p. 54).

[BZB+07] J. Bailey, C. Zhang, D. Budgen, M. Turner, S. Charters. "Search Engine Overlaps : Do they agree or disagree?" In: *Second International Workshop on Realising Evidence-Based Software Engineering (REBSE '07)*. Second International Workshop on Realising Evidence-Based Software Engineering (REBSE '07). Minneapolis, MN, USA: IEEE, May 2007, pp. 2–2. ISBN: 978-0-7695-2962-2. DOI: 10.1109/REBSE.2007.4. URL: http://ieeexplore.ieee.org/document/4273274/ (visited on 02/14/2021) (cit. on p. 104).

[CBSJ07] D. Cruzes, V. Basili, F. Shull, M. Jino. "Automated Information Extraction from Empirical Software Engineering Literature: Is that possible?" In: *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)*. First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). Madrid, Spain: IEEE, Sept. 2007, pp. 491–493. ISBN: 978-0-7695-2886-1. DOI: 10.1109/ESEM.2007.62. URL: http://ieeexplore.ieee.org/document/4343789/ (visited on 10/21/2020) (cit. on p. 60).

[CHHK13] J. C. Carver, E. Hassler, E. Hernandes, N. A. Kraft. "Identifying Barriers to the Systematic Literature Review Process". In: *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*. 2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Baltimore, Maryland: IEEE, Oct. 2013, pp. 203–212. ISBN: 978-0-7695-5056-5. DOI: 10.1109/ESEM.2013.28. URL: http://ieeexplore.ieee.org/document/6681353/ (visited on 01/21/2021) (cit. on pp. 23–25).

[CPS18] B. Cartaxo, G. Pinto, S. Soares. "The Role of Rapid Reviews in Supporting Decision-Making in Software Engineering Practice". In: *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*. EASE'18: 22nd International Conference on Evaluation and Assessment in Software Engineering 2018. Christchurch New Zealand: ACM, June 28, 2018, pp. 24–34. ISBN: 978-1-4503-6403-4. DOI: 10.1145/3210459.3210462. URL: https://dl.acm.org/doi/10.1145/3210459.3210462 (visited on 12/30/2020) (cit. on p. 33).

[CPS20] B. Cartaxo, G. Pinto, S. Soares. "Rapid Reviews in Software Engineering". In: *arXiv:2003.10006 [cs]* (Mar. 22, 2020). arXiv: 2003.10006. URL: http://arxiv.org/abs/2003.10006 (visited on 12/30/2020) (cit. on p. 33).

[Duj] J. Dujmovi. "A Technique and Tool for Software Evaluation". In: (), p. 6 (cit. on p. 41).

[FNF+10] K. R. Felizardo, E. Y. Nakagawa, D. R. Feitosa, R. Minghim, J. C. Maldonado. "An Approach Based on Visual Text Mining to Support Categorization and Classification in the Systematic Mapping". In: 14th International Conference on Evaluation and Assessment in Software Engineering (EASE). Apr. 1, 2010. DOI: 10.14236/ewic/EASE2010.5. URL: https://scienceopen.com/document?vid=c36bb8b4-c1e0-4b74-a6c4-9e4e5cb0f144 (visited on 10/28/2020) (cit. on pp. 29, 60, 99).

[FNMM14] K. R. Felizardo, E. Y. Nakagawa, S. G. MacDonell, J. C. Maldonado. “A visual analysis approach to update systematic reviews”. In: *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering - EASE ’14*. the 18th International Conference. London, England, United Kingdom: ACM Press, 2014, pp. 1–10. ISBN: 978-1-4503-2476-2. DOI: 10.1145/2601248.2601252. URL: http://dl.acm.org/citation.cfm?doid=2601248.2601252 (visited on 10/25/2020) (cit. on pp. 29, 60).

[FT20] M. Felderer, G. H. Travassos, eds. *Contemporary Empirical Methods in Software Engineering*. Cham: Springer International Publishing, 2020. ISBN: 978-3-030-32488-9 978-3-030-32489-6. DOI: 10.1007/978-3-030-32489-6. URL: http://link.springer.com/10.1007/978-3-030-32489-6 (visited on 10/21/2020) (cit. on pp. 98, 99).

[GFM16] V. Garousi, M. Felderer, M. V. Mäntylä. “The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature”. In: *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*. EASE ’16: 20th International Conference on Evaluation and Assessment in Software Engineering. Limerick Ireland: ACM, June 2016, pp. 1–6. ISBN: 978-1-4503-3691-8. DOI: 10.1145/2915970.2916008. URL: https://dl.acm.org/doi/10.1145/2915970.2916008 (visited on 12/27/2020) (cit. on p. 22).

[GFM19] V. Garousi, M. Felderer, M. V. Mäntylä. “Guidelines for including grey literature and conducting multivocal literature reviews in software engineering”. In: *Information and Software Technology* 106 (Feb. 2019), pp. 101–121. ISSN: 09505849. DOI: 10.1016/j.infsof.2018.09.006. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584918301939 (visited on 12/25/2020) (cit. on pp. 21, 22, 97).

[HCHA16] E. Hassler, J. C. Carver, D. Hale, A. Al-Zubidy. “Identification of SLR tool needs – results of a community workshop”. In: *Information and Software Technology* 70 (Feb. 2016), pp. 122–129. ISSN: 09505849. DOI: 10.1016/j.infsof.2015.10.011. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584915001779 (visited on 12/28/2020) (cit. on pp. 23, 28, 39, 97, 104).

[HMTE20] A. Hinderks, F. J. D. Mayo, J. Thomaschewski, M. J. Escalona. “An SLR-tool: search process in practice: a tool to conduct and manage systematic literature review (SLR)”. In: *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings*. ICSE ’20: 42nd International Conference on Software Engineering. Seoul South Korea: ACM, June 27, 2020, pp. 81–84. ISBN: 978-1-4503-7122-3. DOI: 10.1145/3377812.3382137. URL: https://dl.acm.org/doi/10.1145/3377812.3382137 (visited on 10/25/2020) (cit. on pp. 77–79).

[HP14] N. R. Haddaway, A. S. Pullin. “The Policy Role of Systematic Reviews: Past, Present and Future”. In: *Springer Science Reviews* 2.1 (Dec. 2014), pp. 179–183. ISSN: 2213-7793, 2213-7793. DOI: 10.1007/s40362-014-0023-1. URL: http://link.springer.com/10.1007/s40362-014-0023-1 (visited on 12/13/2020) (cit. on p. 15).

[Int05] F. H. International. *Qualitative Research Methods - A Data Collector’s Field Guide*. Research Triangle Park, North Carolina 27709: Family Health International, 2005 (cit. on p. 37).

[JS09] A. S. Jadhav, R. M. Sonar. “Evaluating and selecting software packages: A review”. In: *Information and Software Technology* 51.3 (Mar. 2009), pp. 555–563. ISSN: 09505849. DOI: 10.1016/j.infsof.2008.09.003. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584908001262 (visited on 11/21/2020) (cit. on pp. 41, 42).

[KB13] B. Kitchenham, P. Brereton. “A systematic review of systematic review process research in software engineering”. In: *Information and Software Technology* 55.12 (Dec. 2013), pp. 2049–2075. ISSN: 09505849. DOI: 10.1016/j.infsof.2013.07.010. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584913001560 (visited on 01/21/2021) (cit. on pp. 23, 24, 26).

[KBT+09] B. Kitchenham, P. Brereton, M. Turner, M. Niazi, S. Linkman, R. Pretorius, D. Budgen. “The impact of limited search procedures for systematic literature reviews &#x2014; A participant-observer case study”. In: *2009 3rd International Symposium on Empirical Software Engineering and Measurement*. 2009 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM). Lake Buena Vista, FL, USA: IEEE, Oct. 2009, pp. 336–345. ISBN: 978-1-4244-4842-5. DOI: 10.1109/ESEM.2009.5314238. URL: http://ieeexplore.ieee.org/document/5314238/ (visited on 12/27/2020) (cit. on p. 22).

[KDJ04] B. A. Kitchenham, T. Dybå, M. Jørgensen. “Evidence-based Software Engineering”. In: *th International Conference on Software Engineering* (2004), p. 9 (cit. on p. 15).

[Kee07] S. Keele. “Guidelines for performing systematic literature reviews in software engineering”. In: *Technical report, Ver. 2.3 EBSE Technical Report. EBSE* (2007) (cit. on pp. 15–19, 98).

[Kita] B. Kitchenham. “Procedures for Performing Systematic Reviews”. In: (), p. 33 (cit. on p. 19).

[Kitb] B. A. Kitehenham. “Evaluating Software Engineering Methods and Tool Part 7: Planning Feature Analysis Evaluation 2”. In: *ACM SIGSOFT* 22.4 (), p. 4 (cit. on pp. 42, 43).

[KLL] B. Kitchenham, S. Linkman, D. Law. “DESMET: A method for evaluating Software Engineering methods and tools”. In: (), p. 7 (cit. on pp. 41, 42).

[KPB+09] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, S. Linkman. “Systematic literature reviews in software engineering – A systematic literature review”. In: *Information and Software Technology* 51.1 (Jan. 2009), pp. 7–15. ISSN: 09505849. DOI: 10.1016/j.infsof.2008.09.009. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584908001390 (visited on 12/13/2020) (cit. on pp. 15, 16).

[KPB+10] B. Kitchenham, R. Pretorius, D. Budgen, O. Pearl Brereton, M. Turner, M. Niazi, S. Linkman. “Systematic literature reviews in software engineering – A tertiary study”. In: *Information and Software Technology* 52.8 (Aug. 2010), pp. 792–805. ISSN: 09505849. DOI: 10.1016/j.infsof.2010.03.006. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584910000467 (visited on 12/25/2020) (cit. on p. 21).

[LSS05] T. C. Lethbridge, S. E. Sim, J. Singer. “Studying Software Engineers: Data Collection Techniques for Software Field Studies”. In: *Empirical Software Engineering* 10.3 (July 2005), pp. 311–341. ISSN: 1382-3256, 1573-7616. DOI: 10.1007/s10664-005-1290-x. URL: http://link.springer.com/10.1007/s10664-005-1290-x (visited on 01/29/2021) (cit. on p. 39).

[Lue19] T. Luechtefeld. *What is Sysrev?* 2019. URL: https://blog.sysrev.com/what-is-sysrev/ (visited on 02/13/2021) (cit. on p. 83).

[MB13] C. Marshall, P. Brereton. “Tools to Support Systematic Literature Reviews in Software Engineering: A Mapping Study”. In: *2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement*. 2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Baltimore, Maryland: IEEE, Oct. 2013, pp. 296–299. ISBN: 978-0-7695-5056-5. DOI: 10.1109/ESEM.2013.32. URL: http://ieeexplore.ieee.org/document/6681371/ (visited on 10/21/2020) (cit. on p. 28).

[MB15] J. S. Molléri, F. B. V. Benitti. “SESRA: a web-based automated tool to support the systematic literature review process”. In: *Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering - EASE ’15*. the 19th International Conference. Nanjing, China: ACM Press, 2015, pp. 1–6. ISBN: 978-1-4503-3350-4. DOI: 10.1145/2745802.2745825. URL: http://dl.acm.org/citation.cfm?doid=2745802.2745825 (visited on 10/21/2020) (cit. on pp. 29, 61).

[MBK14] C. Marshall, P. Brereton, B. Kitchenham. “Tools to support systematic reviews in software engineering: a feature analysis”. In: *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering - EASE ’14*. the 18th International Conference. London, England, United Kingdom: ACM Press, 2014, pp. 1–10. ISBN: 978-1-4503-2476-2. DOI: 10.1145/2601248.2601270. URL: http://dl.acm.org/citation.cfm?doid=2601248.2601270 (visited on 10/21/2020) (cit. on pp. 23, 29, 40).

[MBR+19] L. Marchezan, G. Bolfe, E. Rodrigues, M. Bernardino, F. P. Basso. “Thoth: A Web-based Tool to Support Systematic Reviews”. In: *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Porto de Galinhas, Recife, Brazil: IEEE, Sept. 2019, pp. 1–6. ISBN: 978-1-72812-968-6. DOI: 10.1109/ESEM.2019.8870160. URL: https://ieeexplore.ieee.org/document/8870160/ (visited on 10/21/2020) (cit. on pp. 85–87).

[Mem] E. Members. *REviewER: SLR*. URL: https://sites.google.com/site/eseportal/tools/reviewer (visited on 02/13/2021) (cit. on p. 96).

[MSS15] G. D. Mergel, M. S. Silveira, T. S. da Silva. “A method to support search string building in systematic literature reviews through visual text mining”. In: *Proceedings of the 30th Annual ACM Symposium on Applied Computing - SAC ’15*. the 30th Annual ACM Symposium. Salamanca, Spain: ACM Press, 2015, pp. 1594–1601. ISBN: 978-1-4503-3196-8. DOI: 10.1145/2695664.2695902. URL: http://dl.acm.org/citation.cfm?doid=2695664.2695902 (visited on 10/28/2020) (cit. on p. 60).

[NSEF19] G. T. G. Neto, W. B. Santos, P. T. Endo, R. A. Fagundes. “Multivocal literature reviews in software engineering: Preliminary findings from a tertiary study”. In: *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). Porto de Galinhas, Recife, Brazil: IEEE, Sept. 2019, pp. 1–6. ISBN: 978-1-72812-968-6. DOI: 10.1109/ESEM.2019.8870142. URL: https://ieeexplore.ieee.org/document/8870142/ (visited on 12/27/2020) (cit. on p. 22).

[OTG+18] A. M. O’Connor, G. Tsafnat, S. B. Gilbert, K. A. Thayer, M. S. Wolfe. “Moving toward the automation of the systematic review process: a summary of discussions at the second meeting of International Collaboration for the Automation of Systematic Reviews (ICASR)”. In: *Systematic Reviews* 7.1 (Dec. 2018), p. 3. ISSN: 2046-4053. DOI: 10.1186/s13643-017-0667-4. URL: https://systematicreviewsjournal.biomedcentral.com/articles/10.1186/s13643-017-0667-4 (visited on 12/28/2020) (cit. on p. 39).

[OTT+19] A. M. O’Connor, G. Tsafnat, J. Thomas, P. Glasziou, S. B. Gilbert, B. Hutton. “A question of trust: can we build an evidence base to gain trust in systematic review automation technologies?” In: *Systematic Reviews* 8.1 (Dec. 2019), p. 143. ISSN: 2046-4053. DOI: 10.1186/s13643-019-1062-0. URL: https://systematicreviewsjournal.biomedcentral.com/articles/10.1186/s13643-019-1062-0 (visited on 02/07/2021) (cit. on p. 99).

[PFMM08] K. Petersen, R. Feldt, S. Mujtaba, M. Mattsson. “Systematic Mapping Studies in Software Engineering”. In: 12th International Conference on Evaluation and Assessment in Software Engineering (EASE). June 1, 2008. DOI: 10.14236/ewic/EASE2008.8. URL: https://scienceopen.com/document?vid=6d552894-2cc3-4e2b-a483-41fa48a37ef8 (visited on 12/23/2020) (cit. on pp. 20, 21).

[PLLH15] F. Piroi, A. Lipani, M. Lupu, A. Hanbury. “DASyR(IR) - document analysis system for systematic reviews (in Information Retrieval)”. In: *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. 2015 13th International Conference on Document Analysis and Recognition (ICDAR). Tunis, Tunisia: IEEE, Aug. 2015, pp. 591–595. ISBN: 978-1-4799-1805-8. DOI: 10.1109/ICDAR.2015.7333830. URL: http://ieeexplore.ieee.org/document/7333830/ (visited on 10/21/2020) (cit. on p. 59).

[Sea99] C. Seaman. “Qualitative methods in empirical studies of software engineering”. In: *IEEE Transactions on Software Engineering* 25.4 (Aug. 1999), pp. 557–572. ISSN: 00985589. DOI: 10.1109/32.799955. URL: http://ieeexplore.ieee.org/document/799955/ (visited on 12/30/2020) (cit. on pp. 37, 39).

[SSS08] F. Shull, J. Singer, D. I. K. Sjøberg, eds. *Guide to advanced empirical software engineering*. OCLC: ocn173720742. London: Springer, 2008. 388 pp. ISBN: 978-1-84800-043-8 (cit. on p. 37).

[TD12] P. Thokala, A. Duenas. “Multiple Criteria Decision Analysis for Health Technology Assessment”. In: *Value in Health* 15.8 (Dec. 2012), pp. 1172–1181. ISSN: 10983015. DOI: 10.1016/j.jval.2012.06.015. URL: https://linkinghub.elsevier.com/retrieve/pii/S1098301512016555 (visited on 12/28/2020) (cit. on pp. 40, 41).

[Tea] T. Team. *Thoth- Help*. URL: http://200.132.136.13/Thoth/help (visited on 02/13/2021) (cit. on p. 86).

[Tea20a] A. C. D. Team. *ASReview: Active learning for Systematic Reviews*. 2020. URL: https://asreview.readthedocs.io/en/latest/plugins/plugin_visualization.html (visited on 01/15/2021) (cit. on p. 69).

[Tea20b] A. C. D. Team. *Pre-Screening - ASReview*. 2020. URL: https://asreview.readthedocs.io/en/stable/features/pre_screening.html (visited on 02/13/2021) (cit. on pp. 67, 68).

[TGC+14] G. Tsafnat, P. Glasziou, M. K. Choong, A. Dunn, F. Galgani, E. Coiera. "Systematic review automation technologies". In: *Systematic Reviews* 3.1 (Dec. 2014), p. 74. ISSN: 2046-4053. DOI: 10.1186/2046-4053-3-74. URL: https://systematicreviewsjournal.biomedcentral.com/articles/10.1186/2046-4053-3-74 (visited on 02/07/2021) (cit. on pp. 98–100).

[Wes19] M. J. Westgate. "revtools: An R package to support article screening for evidence synthesis". In: *Research Synthesis Methods* 10.4 (Dec. 2019), pp. 606–614. ISSN: 1759-2879, 1759-2887. DOI: 10.1002/jrsm.1374. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jrsm.1374 (visited on 11/04/2020) (cit. on p. 61).

[Woh14] C. Wohlin. "Guidelines for snowballing in systematic literature studies and a replication in software engineering". In: *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering - EASE '14*. the 18th International Conference. London, England, United Kingdom: ACM Press, 2014, pp. 1–10. ISBN: 978-1-4503-2476-2. DOI: 10.1145/2601248.2601268. URL: http://dl.acm.org/citation.cfm?doid=2601248.2601268 (visited on 12/25/2020) (cit. on p. 21).

[XW19] Y. Xiao, M. Watson. "Guidance on Conducting a Systematic Literature Review". In: *Journal of Planning Education and Research* 39.1 (Mar. 2019), pp. 93–112. ISSN: 0739-456X, 1552-6577. DOI: 10.1177/0739456X17723971. URL: http://journals.sagepub.com/doi/10.1177/0739456X17723971 (visited on 12/16/2020) (cit. on pp. 17–19).

[YFW+20] A. Yasin, R. Fatima, L. Wen, W. Afzal, M. Azhar, R. Torkar. "On Using Grey Literature and Google Scholar in Systematic Literature Reviews in Software Engineering". In: *IEEE Access* 8 (2020), pp. 36226–36243. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2971712. URL: https://ieeexplore.ieee.org/document/8984351/ (visited on 12/25/2020) (cit. on pp. 21, 22).

[YKM18] Z. Yu, N. A. Kraft, T. Menzies. "Finding better active learners for faster literature reviews". In: *Empirical Software Engineering* 23.6 (Dec. 2018), pp. 3161–3186. ISSN: 1382-3256, 1573-7616. DOI: 10.1007/s10664-017-9587-0. URL: http://link.springer.com/10.1007/s10664-017-9587-0 (visited on 10/21/2020) (cit. on pp. 69, 71).

[YYH+12] Yueming Sun, Ye Yang, He Zhang, Wen Zhang, Qing Wang. "Towards evidence-based ontology for supporting systematic literature review". In: *16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012)*. 16th International Conference on Evaluation & Assessment in Software Engineering

(EASE 2012). Ciudad Real, Spain: IET, 2012, pp. 171–175. ISBN: 978-1-84919-541-6. DOI: 10.1049/ic.2012.0022. URL: https://digital-library.theiet.org/content/conferences/10.1049/ic.2012.0022 (visited on 10/21/2020) (cit. on p. 60).

All links were last followed on March 5, 2021.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature