Knowledge-Based Systems 251 (2022) 109266

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

slr-kit: A semi-supervised machine learning framework for systematic literature reviews

Tullio Facchinetti [*], Guido Benetti, Davide Giuffrida, Antonino Nocera

via A. Ferrata, 1, 27100 Pavia, Italy

ARTICLE INFO

Article history:
Received 11 April 2022
Received in revised form 10 June 2022
Accepted 11 June 2022
Available online 17 June 2022

Keywords:
Natural language processing
Topic modeling
Systematic literature review
Tagging
Performance evaluation

ABSTRACT

Systematic Literature Review (SLR) is nowadays a challenging task due to the large number of papers that typically compose the scientific material of the topic to review. Recently, a lot of research effort has been devoted to automate, even partially, the stages of an SLR. This paper proposes the design and implementation of a workflow and a set of tools – called slr-kit – to support key tasks in an SLR. The proposed approach leverages a semi-supervised strategy, in which time-consuming processes are carried out using automatic tools, whereas manual tasks have been optimized by carefully designed support tools to reduce the overall required effort. Important parts of the workflow include the extraction of key terms directly from the abstracts of the papers to survey, and the subsequent topic modeling that allows for a thematic clustering of the corpus of papers. In the proposed workflow, the former task is carried out by exploiting a novel tool, called FAst WOrd Classifier (FAWOC). The latter, instead, is designed to be automatically carried out by leveraging an ad-hoc solution based on the application of the Latent Dirichlet Allocation (LDA) algorithm. The result of the process consists in a set of statistics regarding the relationship among papers, topics, and their trend of publication on journals and conference proceedings. The validity of the method is demonstrated with an application to a dataset related to the scientific field of NLP, while its accuracy is assessed by the manual examination of the results by domain experts.

1. Introduction

A Systematic Literature Review (SLR, for short) is a scientific task devoted to identify, analyze and interpret all available research studies related to a specific research question. As suggested by Kitchenham and Charters [1], the activities to perform a systematic literature review includes planning, conducting and reporting the review. More in detail, given a research question of interest, the SLR involves systematically locating, appraising, and synthesizing evidence from scientific studies based on prespecified criteria. For this reason, systematic reviews are typically time consuming and labor intensive activities. Nonetheless, the availability of all-encompassing reports dedicated to provide a summarized answer to specific research questions is extremely useful for research communities [2]. For this reason, a lot of research effort has been devoted towards the development of software tools to support systematic review processes.

In the recent years, all the scientific fields have experienced a high increase in the production of proposals and papers, and suitable online databases and indexing systems have been promoted to favor their diffusion. Due to the world-scale volume of these repositories, the profitably study of specific subjects of given domains can be a challenging task. For this reason, SLRs has become a crucial activity to favor the diffusion of the scientific knowledge.

In general, an SLR can be seen as composed of two main tasks, namely: retrieving necessary/prominent sources, and extracting schematic information from them. The former concerns an Information Retrieval action whose objective is to identify the most influential and authoritative sources starting from a specific research query [3]. The recent scientific literature reports several advanced approaches to optimize this objective by leveraging querying extension strategies and semantic approaches [4,5]. As for the latter task, instead, it focuses on an Information Extraction action devoted to the identification of semantic data from the previous sources to ease their consultation for interested researchers [6]. However, managing and carrying out such reviews can still become a time consuming and challenging task with thousands of references to screen.

This paper provides a contribution in this setting and proposes a workflow and a set of tools for carrying out some of the tasks required in an SLR. The peculiarity of the workflow is that it is semi-supervised, i.e., most of the generally time-consuming

[*] Corresponding author.
E-mail addresses: tullio.facchinetti@unipv.it (T. Facchinetti), guido.benetti01@universitadipavia.it (G. Benetti), davide.giuffrida01@universitadipavia.it (D. Giuffrida), antonino.nocera@unipv.it (A. Nocera).

https://doi.org/10.1016/j.knosys.2022.109266
0950-7051/© 2022 Elsevier B.V. All rights reserved.

procedures are carried out using automatic strategies. Manual operations have been optimized to reduce the total time required to obtain the final results of the SLR. The proposed tool, called `slr-kit`,[1] leverages ad-hoc natural language processing techniques and algorithms to perform the automatic processing of the text. In particular, the key aspect of the proposed workflow is a guided classification of terms extracted from the abstracts of the documents, which are relevant for the specific research domain under analysis in the SLR and can hence be referred as keywords for the domain. The classification process is designed to minimize the effort of human experts in the identification of relevant versus irrelevant terms. Therefore, a novel tool – called FAWOC[2] – is introduced, which allows an efficient classification of terms thanks to the specific design of its user interface. The classification enables an effective application of a subsequent topic modeling activity to identify important concepts and organize the research papers involved in the SLR. Arguing that the statistical distribution of keywords inside the abstracts is the most important element to identify the topics of a considered domain, our solution leverages the LDA topic modeling algorithm and exploits relevant terms, identified during the first part of the workflow, to improve the model confidence. As a result, research papers are clustered based on the topics identified by the LDA.

In addition, `slr-kit` is able to generate a set of reports about the analyzed set of papers and their topics, such as the trend of popularity of the identified topics and the association of topics with their publication venues. The reports are intended to be directly incorporated in the SLR.

This paper describes the whole proposed workflow and validates its suitability as an SLR support tool. The validation consists in the application of proposed solution to a real case study in which the considered research domain is the Natural Language Processing, possibly limited to the papers published in the computer science area.

To check the accuracy of the proposal we adopted a supervised strategy involving human experts of the considered case study. Following a shared rating scheme, each expert manually evaluated the soundness of the classification and topic modeling produced by our system. The results show the quality of the proposal, demonstrating the effectiveness of `slr-kit` as a companion in carrying out the SLR task.

The remaining part of this paper is organized as follows. Section 2 discusses the relevant related works. The details of an SLR are presented in Section 3, with the indication of the steps that are addressed by `slr-kit`. The model of the proposed workflow is explained in Section 4, while the description of the workflow itself is reported in Section 5. Section 6 presents the characteristics of FAWOC, the tool devised to support the labeling, while Section 7 is dedicated to the evaluation of the workflow on a real case study. Finally, Section 8 concludes the paper.

2. Related works

As stated in the Introduction, in the recent years a lot of researchers focused their efforts to define approaches for automating, completely or partially, all the phases of SLR processes [7]. In particular, some proposals have been devoted to the application of Machine Learning (ML) to automate the citation screening phase [8,9]. In many cases, these approaches proposed the use of text mining and NLP techniques to achieve an adequate automation level in this task [10].

Over the years, different phases were identified by researchers that can benefit from automatic or semi-automatic tools to support the development of an SLR [11,12]. Such phases comprise the crucial steps of identifying, selecting, and assessing primary studies. However, when it comes to the definition of these support tools, there is a growing attention from the scientific community. However, the scientific literature is still missing a strong contribution and only some preliminary studies have been performed so far, as highlighted by the authors of [12]. Indeed, even if specific software have been designed to facilitate some of the phases above, complete solutions embracing the entire SLR process are still missing [13].

In particular, to the best of our knowledge, none of the existing approaches support the whole SLR process by providing both solutions for reviewers to manually identify important keywords and concepts and semi-supervised approaches to consequently organize documents by topic to ease the review process. The framework proposed in this paper tackles these issues by focusing on specific steps of the process, as detailed in Section 3, and adopting an NLP-based solution.

The application of NLP for citation screening in SLR is a relatively young and growing field. Specifically, when it comes to identifying scientific papers to be included in a review, researchers are interested in automated techniques for discovering whether a given study (described by a title and abstract) is relevant [14,15]. Some of the assumptions of the application of NLP in other contexts do not hold when transferred to the review of scientific papers. Indeed, although some authors suggest that it could be possible to reduce between 30% and 70% the workload associate with an SLR [10], this would cause a loss of about 5% in the associated recall. However, in the context of SLR, a high recall (i.e., the identification of all the related publications) is a crucial aspect to achieve even if this may cause the inclusion of a vast number of irrelevant proposals (lower selection precision). To solve the trade off between the need of high recall and the overall workload of SLR, NLP solutions have started to be adopted in a range of areas [14–17]. In this context, one of the most common strategy adopted is classification, in which the information about the suitability of a paper for a systematic review is learned on the basis of data derived from manual screening [18]. Under this assumption semi-automated screening processes can be obtained [19,20]. Other works have exploited such a classification strategy to reduce the set of relevant papers (i.e., exclude less useful ones) by preserving a recall of at least 95% for the overall SLR [21,22]. In particular, techniques based on active learning text classification have been developed. These approaches adopt iterative algorithms to incrementally identify eligible papers to review. In more details, generally these algorithms start by considering an initial set of manually labeled papers to train a baseline classifier. After that, the active learning approach works by executing several optimization phases to improve the classifier accuracy. During each of these phases, the obtained model is exploited to label a set of new papers candidate for the systematic review. Such a set is then validated by a human expert to obtained a new training sample that will be provided in input to the next optimization phase. This iterative optimization is driven by a convergence criterion whose objective is the achievement of a suitable recall threshold [23].

Another interesting research direction in this scenario is the evaluation of active learning strategies for topic modeling instead of classification. Indeed, the capability of organizing candidate papers for an SLR, according to the set of topics they deal with, can significantly reduce the workload associated with the screening phase. However, several challenges must be faced to develop solutions in this direction, such as the problem of imbalanced data typical of an SLR. In this setting, unsupervised topic modeling techniques such as LDA can help boosting the performance of active learning also in the case in which no manual annotations are available for training [20]. Indeed, approaches based on Latent Dirichlet Allocation (LDA) have been successfully adopted

[1] Source code available at https://github.com/robolab-pavia/slr-kit.

[2] Source code available at https://github.com/robolab-pavia/fawoc.

Table 1
The steps of a classical SLR and the indication of which steps are supported by the `slr-kit` framework presented in this paper.

#	Description	Addressed by `slr-kit`
1	Identification of the research question	NO
2	Definition of inclusion and exclusion criteria	NO
3	Search for studies	NO
4	Selection of studies for inclusion based on pre-defined criteria	PARTIAL — Some support by features not discussed in this paper
5	Extraction of data from included studies	YES — Topic modeling
6	Evaluation of the risk of bias of included studies	YES — In relation to the topic modeling
7	Presentation of results	YES
8	Assessment of the quality of evidence	YES

in scenarios in which little supervised information is available. LDA is a generative statistical model that represents documents as a mixture of hidden topic distributions so that each word in the document can be attributed to one of the topic identified for it. In the context of SLR, the complexity of the considered scenario introduces important limitations in the direct use of LDA-based solutions. For this reasons, many authors have started to collect training material by extracting data from existing manual screen processes. An interesting outcome of this strategy is that, as the number of existing SLRs increases, also the size of available training sets will grow. This ultimately would allow for a continuous improvement of existing solutions. Indeed, for instance, in [20] the authors proposed the exploitation of an existing manually-processed dataset to improve the labeling of the documents therein. To do so, they applied a clustering algorithm to the whole dataset to derive cluster-based features that could hence be leveraged to boost LDA performance in the extraction of topics for the involved documents. Interestingly, if, on one hand, the extraction of cluster-based features allows for an improvement of LDA performance, the authors proved that the subsequent labeling of existing documents with topics from LDA reduces the overall cost of the manual intervention for the screening, on the other hand. Experimental results showed that the approach achieves a very high recall and that it can be used to support SLRs. Similarly, the approach presented in [24] clusters research papers using information about the likelihood that they share similar subjects. This solution leverages the keywords associated with paper and, separately, adopts LDA for topic modeling. After this, it uses both the information pieces to apply K-means clustering algorithm to classify research papers according to TF-IDF values for both keywords and LDA topics.

It is worth noticing that, although our approach shares some similarities with the approaches based on LDA described above, the goals of our proposal are wider. Indeed, we propose semi-automated solutions for both supporting human-experts to derive important keywords from paper abstracts, and also an LDA-based strategy leveraging such keywords to cluster papers on the basis of shared topics. Differently, from the approaches presented above, LDA is driven by derived keywords for the identification of topics, which ultimately leads to an important increase of the performance. Finally, the proposed solution supports SLR by building advanced statistics on classified papers, automatically. Such analysis is intended to be directly included in the SLR task.

3. Systematic literature reviews and `slr-kit`

This section shortly recalls the standard procedure to carry out an SLR as presented in [25], and outlines the parts in which our proposed workflow brings relevant contributions in terms of original and efficient methods to manage the information to process.

Table 1 summarizes the steps required by a classical SLR, and indicates the steps that are supported by the proposed `slr-kit` framework. As can be seen in the table, our proposed method focuses on specific steps of an SLR. In particular, it does not focus on the identification of the research question, the definition of inclusion and exclusion criteria, and the search of studies (steps from 1 to 3). In other words, the set of papers to review represents the input to the proposed workflow. On the other hand, `slr-kit` provides some features to select the studies to be included based on the selection of keywords, although the description of such features is beyond the scope of this paper (step 4).

The main contribution of `slr-kit` is targeted to the extraction of data related to the language used in the papers in order to perform the topic modeling, and the generation of results that can be derived by the association among papers, topics, keywords and publication venues (steps 5 and 8).

4. Model of the workflow

The workflow proposed by `slr-kit` operates on a set $\mathcal{D}$ of documents, which are the abstracts of the papers related to the considered research question. Each document $D_i \in \mathcal{D}$ is modeled as a list of words $W(D_i)$. The set of all the possible words is denoted with $\mathcal{W}$, where $\mathcal{W} = \bigcup_i W(D_i)$. Without impacting on the generality of the model, it can be assumed that $W(D_i)$ is composed by the words obtained after a process of stemming and lemmatization of the original abstracts. We denote with $w_{i,j} \in W(D_i)$ the jth word of the ith document D_i.

The goal is to associate a list of topics to each document and to use the aggregation of documents by topics to derive meaningful insights regarding the research question. The set of all possible topics is denoted with $\mathcal{T}$. Each topic $T_i \in \mathcal{T}$ will be associated with a set of words $W(T_i)$ with $W(T_i) \subset \mathcal{W}$.

We denote with $\mathcal{S}$ the set of stopwords, which is a set of words that are not relevant for the modeling of the topics.

The processing of the documents follows the workflow depicted in Fig. 1. The remainder of this section introduces the notation that is used to denote the relevant elements used along the workflow.

Extraction of tokens. The first automatic step consists in the extraction of the tokens that will be provided to the expert for the manual classification. The tokens are word level n-grams with $n \in [1, N]$.

The token $t^n_{i,j} \in Tok(D_i)$ is defined as $t^n_{i,j} = [w_{i,j}, w_{i,j+1}, \dots, w_{i,j+n}]$ such as $\forall j, w_{i,j} \notin \mathcal{S}$. Informally, the token $t^n_{i,j}$ is the n-gram starting from the jth word of the document D_i that does not contain any stopword from the set $\mathcal{S}$.

We denote the set of tokens obtained from the document D_i as $Tok(D_i)$. The total set of tokens is $TOK = \bigcup_i Tok(D_i)$. When necessary, we will indicate with t_k the kth element of TOK.

Classification of tokens. This is a manual operation in which the expert knowledge is exploited to identify the words that are more relevant to the targeted research question. The purpose of the manual classification is to divide the set of tokens TOK into two subsets, denoted with $relev(TOK)$ and $noise(TOK)$. We refer to the tokens in the two sets as *relevant* and *noise* tokens, respectively. The two sets are disjoints, i.e.,

$$relev(TOK) \cap noise(TOK) = \emptyset$$

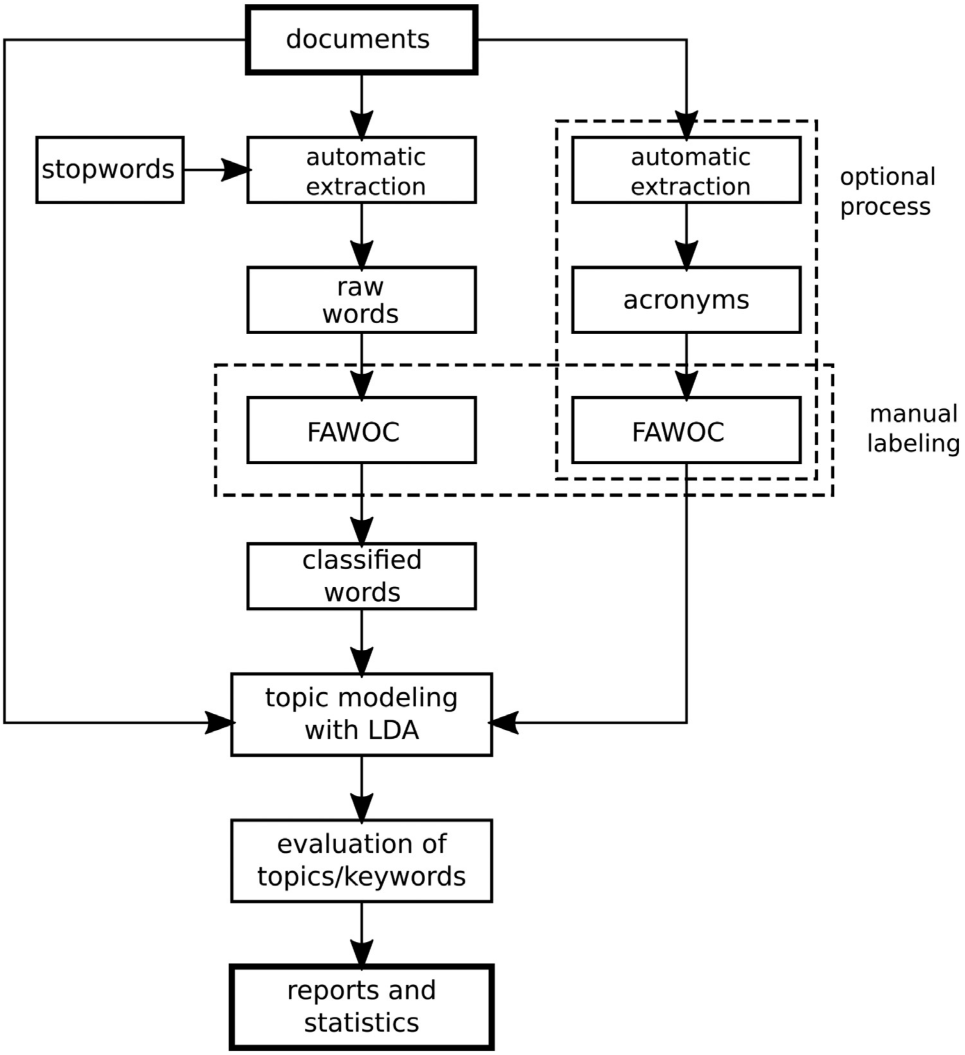


Fig. 1. Block diagram showing the proposed workflow.

meaning that a given token t_k can only belong to one of the two sets.

A token shall be labeled as noise if either (1) it is a common term that does not contribute in the characterization of an argument related to the intended research domain or (2) the token is syntactically or semantically wrong. The latter case typically applies to multi-grams, for which the extraction does not take into account the syntactical correctness of the terms. Examples of case (1) are terms such as *in our paper*, *the proposed technique* or *results*, which appear often in the abstracts of scientific papers but do not add any value in the characterization of the topic. Examples of case (2) are terms such as *in our* or *method we propose*, which are made by consecutive words in the text, but they do not have any semantic meaning. On the other hand, relevant tokens are those that may characterize a topic. Such tokens strongly depend on the research domain that is considered in the SLR. For example, if the research domain is the NLP, possible relevant tokens may be *deep learning* or *sentiment analysis*. While the former identifies a family of algorithms used to the field of NLP, the latter is related to one of the most common analysis performed on free text.

It is worth to note that, we do not require that all the tokens are classified either as relevant or noise. In formal terms, we do not require that

$$relev(TOK) \cup noise(TOK) = TOK$$

This is due to the fact that, during the classification, some tokens could be classified neither as relevant or noise. In fact, some tokens may be classified with other labels. For example, some of them may be classified as stopwords possibly not initially included in $\mathcal{S}$. However, this feature is allowed for the sake of convenience during the classification, but it does not affect the knowledge that is necessary to carry out the proposed workflow. Some tokens may not be classified at all. This typically may happen when the expert evaluates that the frequency of the remaining tokens is too low to be meaningful. More information about these aspects are provided in Section 5.

Filtering of relevant tokens. This step relates to the application of a filter that removes from the documents all the tokens that are not explicitly classified as *relevant*.

Formally, the set of the remaining tokens after filtering the document D_i can be defined as $Tok'(D_i) = \{t^n_{i,j} : t^n_{i,j} \in relev(Tok(D_i)), \forall j, n\}$.

The set of all the filtered tokens is denoted with $TOK' = \bigcup_i Tok'(D_i)$. When necessary, we will indicate with t'_k the kth element of TOK'.

The filtering process transforms each $D_i \in \mathcal{D}$ in a filtered document D'_i, which is composed by the tokens $Tok'(D_i)$.

Topic modeling. The topic modeling approach adopted in this work consists in the plain application of the LDA algorithm proposed in [26] to the set of documents composed by the documents obtained after the filtering. Therefore, the topic modeling is applied to the set of $\mathcal{D}'$ documents. While the description of the LDA algorithm is out of the scope of this paper, it is important to notice that the application of the modeling to a set of documents that are composed by relevant words only, has the ability to automatically exclude from the words associated with topics all those words that do not adequately characterize the research domain under investigation.

The application of LDA is able to generate the set of topics $\mathcal{T}$ and the association between topics and documents. The hth topic is defined as T_h, and $\mathcal{T} = \{T_h\}$.

Every token $t_k \in TOK'$ is associated with every topic $T_h \in \mathcal{T}$. The relationship strength of a token with a topic is expressed through a probability of the token to belong to the specific topic. This probability indicates the relevance of the token in the corresponding topic. We indicate the probability of the token t_k to be relevant for the topic T_h as $P_{TT}(k, h)$.

Similarly, the outcome of LDA is also an association between topics and documents. Formally speaking, each document $D_i \in \mathcal{D}$ is associated with a topic $T_h \in \mathcal{T}$. The probability that T_h is associated with the document D_i is denoted by $P_{DT}(i, h)$.

5. Discussion on the workflow

With the availability of the model presented in Section 4, this section discusses the workflow summarized in Fig. 1. As in traditional SLRs, the process begins by selecting the set $\mathcal{D}$ of papers to include in the review. Notice that the application of the proposed method only requires the abstracts of the papers. This is an important aspect for the applicability of the method, since the abstracts are usually freely available together with other bibliographic details, while this is not the case for the full text of the paper. Moreover, the selection of the documents is a step that is outside the scope of the proposed workflow, in the sense that the abstract of papers can be obtained using any available service or platform, such as Scopus, Web of Science, Google Scholar or other services.

One key aspect of our framework is that it requires the bibliographic information regarding the papers. Bibliographic information include title, authors, year and journal or proceedings of the publication, and – optionally – the number of citations at the time of the data acquisition.

Extraction of tokens. Once the documents are available, two operations are applied to the text of the abstracts: the extraction of the set TOK of tokens from the set $\mathcal{W}$ of raw words which composes the set $\mathcal{D}$ of documents and, optionally, the identification of the acronyms. The extraction of tokens consists in the automatic extraction of n-grams from $\mathcal{W}$ to build the TOK set, with a selectable value of n. In our experiments, we empirically found that an adequate value for n is 4, which is a suitable trade-off between the wish to capture the meaning of parts of the text, and to avoid the generation of too many tokens.

During the automatic extraction of tokens, we replace the stopwords from $\mathcal{S}$ with placeholders. This approach allows us to improve the extraction of multi-grams, since a multi-gram that contains a placeholder is automatically eliminated from $\mathcal{W}$. The implementation of this feature leads to a reduction of the number of multi-grams that are generated for the subsequent manual selection.

To clarify the role of $\mathcal{S}$, consider the example reported in Table 2, which shows an excerpt of the abstract of [27]. The table shows the original text, the text obtained with a processing that uses basic and standard text manipulations (elimination of punctuation, making the text lower-case, etc.), and the list of stopwords that have been replaced in the original text. In the text, the stopwords have been replaced with the placeholder "@".

Acronyms are items that appear relatively frequently in the abstracts of scientific papers. For this reason, we envisage a dedicated path of the workflow to deal with acronyms. Acronyms can be easily extracted using established techniques such as the one proposed in [28]. After the extraction, the list of acronyms can be selected or discarded with the same method used for multi-grams (tokens), as described in the next paragraph.

It is worth to note that the extraction and classification of acronyms is an optional step. In fact, if the acronyms are not processed as "special" terms after their extraction, they can be processed as regular multi-gram words together with all the tokens t_k extracted from the documents. In general, it is a good approach to deal with the acronym separately, in order to avoid that the list of regular terms is "polluted" by partially extracted – and thus meaningless – acronyms.

Manual classification. Once the list of tokens t_k has been extracted, they are manually classified as either being part of $relev(TOK)$ or to $noise(TOK)$. This classification is done using a specifically developed tool called FAWOC. FAWOC is a key component in the workflow and a novel contribution of this framework. Section 6 is dedicated to the overview of this tool.

As mentioned above, the same tool can be used to assess the correctness of the extracted acronyms. In practice, FAWOC can be used to validate the correctness of the acronyms that are automatically extracted by using the dedicated algorithm.

As a final remark, to further improve the quality of this task, synonyms can also be evaluated. In the current workflow, any existing strategy for synonym detection among keywords can be adopted, e.g. [29,30]. To smoothly handle the identified synonyms, it could be possible to associate a *virtual token* with them, and use it to feed the subsequent topic modeling step based on LDA.

Topic modeling. Once the classification of relevant tokens has been done, we apply the standard LDA topic modeling algorithm to extract the topics and to associate the topics to the documents. The documents processed by LDA consist in the text of the abstracts that have been filtered by removing all the tokens that are not classified as `relevant` with FAWOC. This is a key aspect of the workflow, which allows to consider only tokens that are strictly relevant for characterizing the topics of the considered research domain. The manual verification of the topics generated by LDA confirms that the ability of the algorithm in finding meaningful topics is greatly improved by working on documents that are composed by tokens that have been identified as relevant for the considered research domain.

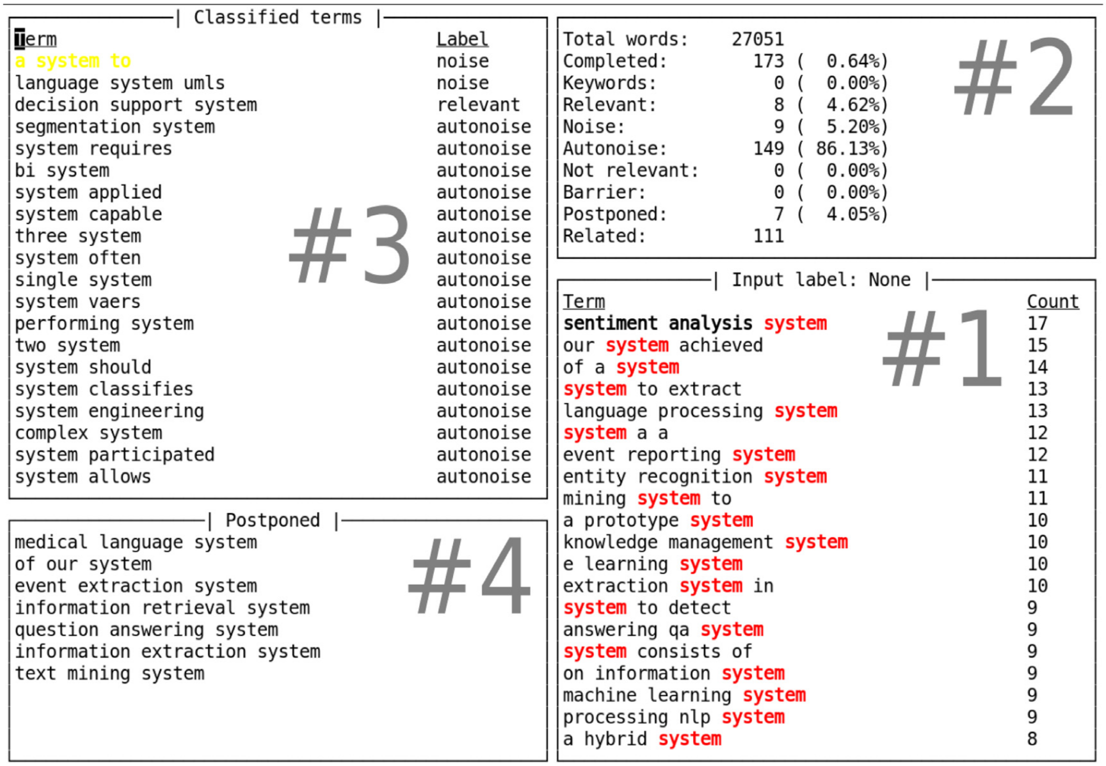
The tools devised to process the text produce an output that can be conveniently inspected by the user, in order to let the expert check the quality of the topics. To validate the modeling, the user can easily check the association between tokens and topics, and between topics and documents.

It is worth to note that the topic extraction is an interesting outcome of our framework per-se, since it gives an objective overview on the main subjects investigated in the considered research domain.

Generation of statistics. The generation of statistics leverages the clustering of papers made by the association between papers and topics. These statistics are automatically generated in the form of a report. There are several different "views" of the information that can be associated with the set of topics and the papers. Currently, the available statistics include:

Table 2
Excerpt of the preprocessing applied to the abstract of [27]. The symbol "@" identifies a stopword.

Original	Large number of studies that must be analyzed. Different approaches have been investigated to support SLR processes, such as: Visual Text Mining or Text Classification. But acquiring the initial dataset is time-consuming and labor intensive. Objective: In this work, we proposed and evaluated the use of Text Classification to support the studies selection activity of new evidences
Processed	Large number of study @ @ be analyzed different approach @ @ investigated to support slr process @ a visual text mining @ text classification @ acquiring @ initial dataset @ time consuming and labor intensive objective in @ work @ @ and evaluated @ use of text classification to support @ study selection activity of new evidence
Stopwords	that, must, have, been, such, or, but, the, this, we, proposed



```
| Classified terms |
Term                                        Label
a system to                                 noise
language system umls                        noise
decision support system                     relevant
segmentation system                         autonoise
system requires                             autonoise
bi system                                   autonoise
system applied                              autonoise
system capable                              autonoise
three system                                autonoise
system often                                autonoise
single system                               autonoise
system vaers                                autonoise
performing system                           autonoise
two system                                  autonoise
system should                               autonoise
system classifies                           autonoise
system engineering                          autonoise
complex system                              autonoise
system participated                         autonoise
system allows                               autonoise
```

```
| Postponed |
medical language system
of our system
event extraction system
information retrieval system
question answering system
information extraction system
text mining system
```

```
Total words:     27051
Completed:         173 (  0.64%)
Keywords:            0 (  0.00%)
Relevant:            8 (  4.62%)
Noise:               9 (  5.20%)
Autonoise:         149 ( 86.13%)
Not relevant:        0 (  0.00%)
Barrier:             0 (  0.00%)
Postponed:           7 (  4.05%)
Related:           111
```

```
| Input label: None |
Term                                        Count
sentiment analysis system                   17
our system achieved                         15
of a system                                 14
system to extract                           13
language processing system                  13
system a a                                  12
event reporting system                      12
entity recognition system                   11
mining system to                            11
a prototype system                          10
knowledge management system                 10
e learning system                           10
extraction system in                        10
system to detect                            9
answering qa system                         9
system consists of                          9
on information system                       9
machine learning system                     9
processing nlp system                       9
a hybrid system                             8
```

Fig. 2. The user interface of FAWOC.

- The trend over the years of a given topic; this is calculated from the number of papers associated with the topics and their year of publication.
- The journals and conferences that have dealt with a specific topic; this is calculated by counting the number of papers published in a given venue.

6. Efficient token classification with FAWOC

To support the workflow described in Section 5, a dedicated tool called FAWOC has been developed. This section provides details of the solutions implemented in FAWOC to make the classification of tokens as efficient as possible.

FAWOC is a terminal program whose user interface is shown in Fig. 2. The graphical front-end is organized in 4 panes, containing the following information:

1. The list of tokens *TOK* to be labeled.
2. Some statistics regarding the tokens under classification, including: the total number of tokens; the number of tokens that have been already classified; the tokens classified with a specific label.
3. The list of labeled tokens *TOK*′.
4. The list of postponed tokens.

In pane #1, the next token that will be classified is highlighted at the top of the list. The tokens are sorted inversely w.r.t. their frequency in the documents, i.e., from the most to the least frequent. This means that the next token to handle by the user is the topmost frequent token. Other sorting strategies are possible, for example based on a relevance calculated using metrics such as the Term Frequency–Inverse Document Frequency (TF-IDF) [31, 32].

One useful feature of the sorting is that, once a 1-gram token has been classified, the program presents to the user the list of all the "related" tokens, which are other n-grams that contain the token t_k. The common word between the tokens is highlighted using a red color, which makes the identification of the word easier within the n-gram. We noticed that this approach tends to keep the concentration of the user on a specific argument instead of requiring a change of focus at every new token to classify.

When these "related" multi-grams are under processing, the last line in pane #2 displays the total number of related tokens to let the user have an idea of the amount of work to be done to complete them.

FAWOC allows to assign to the tokens the `keyword` and `noise` labels, which corresponds – respectively – to tokens that will be included and discarded in the topic modeling step.

Additionally, the following special labels are available: `not-relevant`, `stopword` and `postponed`.

The `not-relevant` label is used to identify a token that is clearly not relevant for the domain of interest. The presence of this kind of tokens may be due to the approximation of a search query that is typically performed to retrieve the documents (false positive). These words can be labeled as `not-relevant` so that they can help to identify the papers that are not relevant for the considered subject.

The `stopword` label is used to tag a token that should have been a stopword since the beginning. In fact, it often happens that not all the stopwords may be available when the abstracts are elaborated to extract the words. Although many words can be stopwords in almost all the domains, this is not guaranteed a-priori. Therefore, if the initial processing is based on a limited set of stopwords, it is possible to mark tokens as stopwords during the labeling step, so that in future refinements of the classification, or in classifications made by different experts, the new stopwords can be suitably included in the set of tokens to filter automatically.

The `postponed` label allows to mark a token so that it can be handled later. This is useful to postpone the classification of tokens that may require longer time for a decision about the most suitable label to assign. In this way, the user can proceed without blocking his mental flow. Note that the `postponed` label is available purely for convenience during the manual classification; on the other hand, it does not impact or change the overall proposed workflow.

FAWOC can also be used to review the tokens already classified. Moreover, it allows the re-labeling of such tokens. This feature enables the collaboration of multiple experts in order to reduce chances of bias in the manual classification phase.

7. Case study

This section shows the use of `slr-kit` to obtain relevant statistics of the publications in a specific research domain by applying the proposed workflow.

The selected research domain is related to the techniques and methodologies that have been proposed for the NLP.[3]

7.1. Selection of papers

We obtained the set of papers to analyze through a query on the Scopus online bibliographic database.[4] We used the following query:

`natural AND language AND processing AND "text mining"`

The goal of the query is to select the papers that are related to the NLP by roughly discarding the publications that come from fields that are different from the engineering domain. In the case of NLP, there is a huge literature on its *applications*, in particular in the domain of medical science, which are deemed not interesting for our scope.

Although, in this work, we are not interested to discuss possible approaches used for the best possible refinement of the search results, we noticed that a poor set of documents, i.e., a set of documents that contains many items with poor relationship with the topic of interest, can greatly affect the quality of the results and thus the applicability of the method. Therefore, the set of papers obtained from the original query submitted to Scopus has been manually filtered to eliminate the ones with the poorest relationship with our intended research domain. The manual filtering has been performed with the help of some features of `slr-kit` that allow to eliminate entire sets of papers published on specific journals that do not belong to our domain of interest. These features are beyond the scope of this paper.

The original query returned a total of 4427 papers. As a result of the filtering, 2471 papers were left.

Table 3
Counting of the different types of elements in the considered dataset: total number of words and stopwords in all the documents, number of unique tokens, count of the number of n-grams and number of unique tokens classified as keywords.

Type of element	Count	
1-grams (all)	303,745	
Stopwords (total number)	81,437	
Classified tokens	**Count**	**Keywords**
1-grams	3 531	13
2-grams	5 000	234
3-grams	1 205	40
4-grams	277	0
Total	10,013	287

7.2. Extraction and classification of terms

The total number of words $W(D_i)$ extracted from the abstracts of the $\mathcal{D}_i$ documents is 10,013. We considered the n-grams with $n \in [1, 4]$. The number of different types of n-grams is shown in Table 3. The table also reports the total number of words and stopwords in all the documents. The column "keywords" shows how many of each n-grams are classified with the `keyword` label using FAWOC.

Before running LDA, the words in the abstracts are filtered to retain only the words that have been classified as keywords in the previous stage. Fig. 3 (left graph) shows the distribution of the number of *words* that are present in the abstracts before the filtering, while the right graph reports the distribution of the number of *keywords* that are present in the abstracts after the filtering.

As a final point, we also investigated the possibility of adopting an automatic filter-based strategy for the extraction of *keywords*. In particular, we used statistics from the manual classification to identify common patterns in the obtained *keywords* that could be used to define ad-hoc filtering strategies. From our exploratory analysis, we observed that the 98.6% of obtained *keywords* are nouns. Therefore, a baseline automatic strategy can be obtained by selecting nouns and n-grams containing at least one noun. As typically done in this context, to reduce the probability of selecting noise, we also applied a filtering condition based on the term frequency.

Hence, we evaluated the suitability of the above solution by comparing the obtained results with the expert-based manual classification, described at the beginning of this section, by computing the *precision* and *recall* metrics [33,34]. In particular, we computed the two metrics with different percentages of the most frequent keywords, i.e., from 10% to 50%. The results reported in Fig. 4 clearly show that the application of automatic strategies for the identification of *keywords* is not adequate. Indeed, the precision is extremely low and remains below the value 0.1. The comparison with an, even naive, baseline automatic approach for *keyword* extraction confirmed our intuition that, because of the uniqueness of each considered domain, expert-based classification of terms is the most adequate strategy in our application context.

7.3. Evaluation of topics

After the n-grams in the NLP dataset have been classified, they are used to filter the words contained in the abstracts. The filtered text is then provided as input to the LDA algorithm.

The execution of the LDA algorithm mainly depends on three parameters: α and β, which control the statistical relationship among words and documents, and the number of topics to generate (see the original paper [26] for details). Moreover, the GenSim

[3] The complete dataset, including the labeling, is available at https://github.com/robolab-pavia/slrkit_NLP_Nocera.

[4] URL: https://www.scopus.com/.

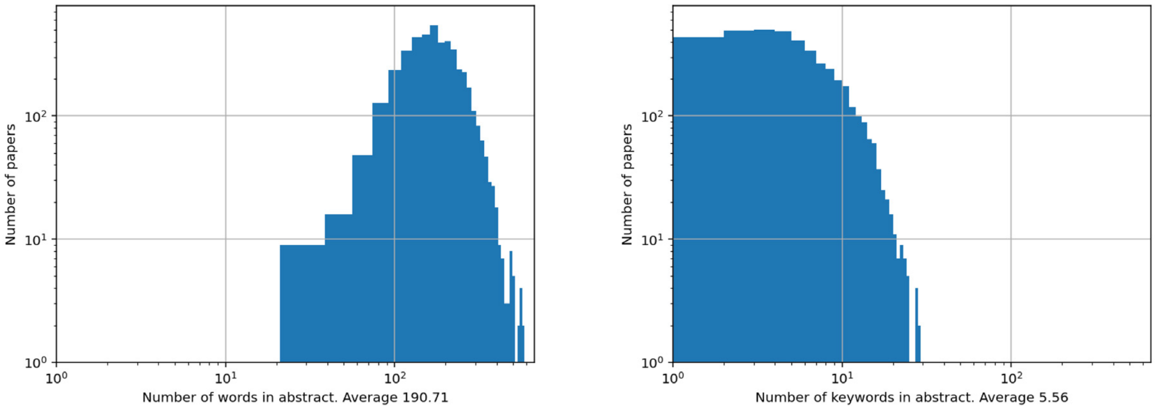


Fig. 3. Distribution of the number of words in the abstracts. The graph on the left refers to the distribution measured after the preprocessing stage, while the graph on the right considers the terms classified as `keyword`. The X scale is logarithmic to manage the different magnitude of the two results.

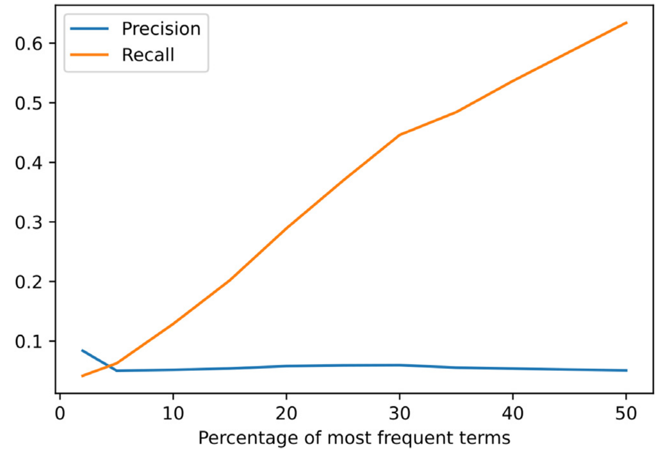


Fig. 4. The performance of a baseline filter-based automatic strategy for keyword extraction as a function of the percentage of the most frequent keywords.

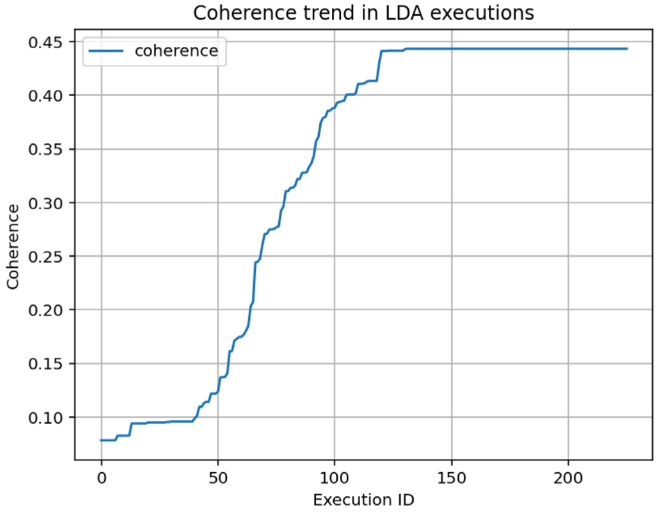


Fig. 5. Topic coherence trend in the optimization of the LDA execution.

library[5] used in this work introduces two additional parameters: `no_below` and `no_above` which are respectively the minimum number and the maximum fraction of documents that have to contain a token in order to include the token itself in the calculations. Basically, the two parameters are used to filter the tokens that appear too often or too rarely.

`slr-kit` implements an optimization method based on a Genetic Algorithm to find the combination of the aforementioned parameters that maximize the *coherence* of topics. Gensim uses the coherence as proposed in [35] to evaluate the quality of the model. The coherence takes values between 0 and 1; higher values correspond to better results.

Fig. 5 reports the trend of the coherence during the optimization. As can be seen in the figure, the highest reached coherence is slightly less than 45%, which is generally deemed very good in the case of the LDA algorithm.

This section shows a full report generated by `slr-kit` in its application to the NLP dataset, which represents the output of the whole workflow. The chosen execution is the one with the highest topic coherence from the optimization of LDA, with a value of 0.4433, achieved with the following parameters: $\alpha = 0.2175$, $\beta = 0.0344$, `no_above` $= 1$, `no_below` $= 1$. The number of topic generated by this execution is 20.

The list of generated topics is reported in Table 4. The topics are sorted in descending order of coherence. The table also lists the tokens that are associated with each topic, including the degree of relevance between the token and the corresponding topic. The degree of relevance is reported within parenthesis alongside the token. It is a number in the interval [0, 1], where an higher value corresponds to an higher relevance.

It is well known that the LDA algorithm cannot establish the actual meaning of the retrieved topics. For this reason, the topics in Table 4 are identified with a generic identifier T_h, where h is an incremental number generated by `slr-kit`. The interpretation of the meaningfulness of the topics is always left to the expert. The remainder of this section reports some considerations regarding the meaning of the topics identified by LDA that arise from their manual evaluation.

First of all, a general comment that holds for every topic is that the value of the metric is relatively high for the topmost 1 to 5 tokens associated with each topic, where the exact number depends on the topic. For example, the tokens associated with the first topic in Table 4, i.e., topic T_{18}, have a value of the metric that is 0.314 for the first token, the second and the third terms weight 0.199 and 0.1 respectively, which is approximately half of value for the first token, and then the remaining tokens have a metric's value that is significantly lower. For this reason, a rough approach to associate a meaning to a topic would be to consider

[5] GenSim homepage: https://radimrehurek.com/gensim/.

Table 4
List of topics generated by LDA and selected among the results with the highest coherence after the optimization of parameters. The degree of relevance of tokens to the corresponding topic is reported within parenthesis.

Topic	Coherence	Tokens
T_{18}	0.57252	information_retrieval (0.314), retrieval (0.199), free_text (0.1), unstructured_information (0.033), systematic_review (0.031)
T_{14}	0.56296	precision_and_recall (0.155), arabic_text (0.126), classification_model (0.105), support_vector_machine (0.062), manually_annotated (0.049)
T_2	0.52363	label (0.241), knowledge_discovery (0.103), knowledge_base (0.101), word_sense_disambiguation (0.053), text_clustering (0.049)
T_1	0.52202	short_text (0.184), training_data (0.151), unstructured_text (0.08), word_segmentation (0.061), test_data (0.058)
T_{13}	0.48192	document_clustering (0.115), clustering_algorithm (0.107), extraction_method (0.105), labeled_data (0.059), unlabeled_data (0.057)
T_{19}	0.47631	stemming (0.287), information_retrieval (0.084), vector_space (0.068), domain_knowledge (0.058), cosine_similarity (0.043)
T_{16}	0.47517	review (0.424), sentence (0.398), online_review (0.027), topic_detection (0.016), evaluation_metric (0.016)
T_{15}	0.47059	corpus (0.803), annotated_corpus (0.033), word_representation (0.016), manually_annotated (0.014), sentence (0.012)
T_3	0.46767	knowledge (0.745), feature_extraction (0.038), knowledge_extraction (0.023), text_representation (0.016), association_rule_mining (0.015)
T_{10}	0.44748	feature_selection (0.157), f_score (0.146), neural_network (0.109), sentence (0.074), support_vector_machine (0.051)
T_6	0.44623	semantic_similarity (0.165), similarity_measure (0.126), sentence (0.09), semantic (0.055), training_set (0.041)
T_8	0.44219	topic_modeling (0.16), question_answering (0.157), supervised_learning (0.084), part_of_speech (0.077), classification_method (0.075)
T_0	0.42253	sentiment (0.352), sentiment_analysis (0.351), review (0.046), sentiment_classification (0.028), opinion (0.022)
T_{17}	0.42082	topic_model (0.278), customer_review (0.097), linguistic_feature (0.073), knowledge_source (0.062), topic_modeling (0.044)
T_{11}	0.40871	semantic (0.718), machine_learning_technique (0.046), knowledge (0.034), knowledge_base (0.024), semantic_information (0.021)
T_7	0.40060	opinion (0.492), opinion_mining (0.123), sentiment_analysis (0.088), text_summarization (0.079), review (0.039)
T_5	0.35663	ontology (0.674), semantic_web (0.064), semantic (0.043), knowledge (0.042), network_analysis (0.03)
T_{12}	0.35100	machine_learning (0.251), deep_learning (0.096), word_embeddings (0.078), word_embedding (0.05), word_vec (0.05)
T_4	0.33988	information_extraction (0.288), unstructured_data (0.098), document_collection (0.053), structured_data (0.052), classification_accuracy (0.042)
T_9	0.27812	extraction (0.654), extraction_system (0.058), test_set (0.042), precision_recall (0.029), f_score (0.027)

the topmost tokens in each topic. Another way could imply the supervision of an expert that gives a name to the topics based on the tokens associated with the topic itself.

According to the latter approach, the following considerations can be done for some of the most interesting and peculiar topics:

- T_{18}: the topic seems related to the ability to retrieve information from free text; it is thus reasonable to associate this topic with "information retrieval" in general.
- T_{14}: the topic seems related to the use of classification algorithms used in the context of NLP. The bi-gram "arabic text" could be a little out of context, however there are ongoing efforts in the scientific literature to handle the texts in Arabic language. Probably there were not enough tokens marked as `keyword` and belonging to the Arabic language topic to build a topic on its own.
- T_{13}: almost every n-gram in this topic is associated with clusters, unsupervised methods and labels. Therefore, this topic is clearly related to clustering approaches applied to NLP.
- T_{10} and T_{12}: both the topics clearly point out to machine learning approaches, but neither of the two topics is narrow enough to focus on a more specific aspect of machine learning.
- T_6: the topmost terms are *semantic_similarity* and *similarity_measure*. This suggests that the corresponding papers are likely to mention or to use such techniques.
- T_0 and T_7: their 10 top most n-grams are quite similar. However T_0 has higher probability score associated with words regarding sentiment analysis, while T_7 is more focused on opinion mining.
- T_{17}: the token *ontology* has a metric value that is hugely higher than the other terms. This suggests that there is a very focused relationship of the corresponding papers with topics associated with ontologies.

The above discussion represents the type of considerations that shall be done by a human expert to characterize a topic on the basis of the terms associated with the topic by LDA.

It is worth to recall that the results, in terms of graphs and tables, represent the direct output of the `slr-kit` tool, which are thus available for a direct inclusion in technical and scientific papers or websites.

7.4. Clustering and reports

This section shows the results that can be obtained by `slr-kit` at the end of the classification and clustering process.

The topics obtained in the previous step are used to group the papers into clusters according to their degree of relationship with the corresponding topic.

Fig. 6 and Table 5 report an evaluation of the popularity of the topics during the years. The popularity in each year is estimated by weighting the papers published in the year, considering their relevance to the topic.

In order to gather the results, at first the papers are clustered according to their publications year. Afterwards, for each topic, the coherence of every paper to that topic is summed, obtaining a total number of papers per year for the topic.

More formally, considering a topic T_h and an year y, the popularity measure $pop(y, T_h)$ associated with each cell in Table 5 is calculated as follows:

$$pop(y, T_h) = \sum_i C^i_{y,T_h}$$

being C^i_{y,T_h} the coherence value of the paper D_i published in the year y to the topic T_h.

The obtained results show a steady increase in the number of papers published per year about every topic. This is justified by the fact that the NLP has seen an increasing interest, in particular starting from 2007–2008, according to the trend depicted in Fig. 6. The drop in the number of papers towards the end of the considered period is due to the fact that the query was submitted at the beginning of March 2021, thus the numbers related to the 2021 are not meaningful.

Tables 7 and 8 report the evaluation of the amount of literature published in different journals about each topic. In the tables, the journals are identified by a label to make the presentation more compact. The association between the label and the corresponding journal is reported in Table 6.

To collect the data shown in the tables, the top 10 most relevant journals (by number of publications) of the dataset were gathered. Then a weighted sum by topic coherence has been computed for every paper linked to those top journals, considering the topics which they were assigned to.

More formally, considering a journal J_k and the topic T_h, we denote with $v(J_k, T_h)$ the metric that assesses the amount of

Table 5
Evolution of topics over the years.

Topic	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
0	10.38	13.36	20.7	26.18	35.71	37.59	48.46	37.16	27.94	3.6
1	6.8	8.53	9.87	12.09	14.61	17.61	19	22.01	19.32	2.45
2	8.62	7.89	11.47	15.62	16.23	17.3	18.44	20.58	21.97	2.21
3	10.7	15.32	15.99	16.91	22.5	21.65	24.83	30.62	25.12	2.37
4	8.08	11.15	13.51	17.28	17.2	18.08	18.58	22.52	22.52	1.91
5	9.62	10.39	13.05	14.5	15.49	15.38	18.06	20.48	17.04	1.53
6	6.37	8.54	12.88	13.11	15.87	16.35	18.88	21.17	20.1	2.65
7	7.57	11.3	14.28	14.3	22.27	24.05	25.76	26.21	22.53	4.78
8	7.21	8.8	9.9	11.33	15.96	14.3	18.38	21.02	23.02	1.94
9	8.21	10.68	14.23	20.73	20.17	18.46	20.61	26.37	23.98	2.17
10	6.91	7.81	10.19	12.93	16.74	17.02	20.48	26.74	25.76	2.86
11	11.26	14.48	16.63	18.92	17.61	20.4	24.39	25.48	21.37	2.65
12	5.24	9.18	11.87	14.55	18.06	26.76	32.39	40.56	45.56	5.71
13	6.94	8.56	9.82	14.64	13.73	15.55	15.37	17.63	17	1.4
14	7.07	6.65	9.39	11.11	15.78	14.41	18.21	20.11	17.31	2.69
15	13.44	13.21	16.99	23.57	23.53	21.94	27.66	29.77	27.25	2.93
16	11.85	16.14	17.72	17.94	25.89	27.58	31.95	35.25	32.58	3.16
17	5.04	7.01	9.13	13.06	13.24	14.13	19.54	20	16.47	2.3
18	8.6	10.15	13.85	15.69	17.23	19.14	22.77	23.02	21.92	2.06
19	8.09	8.87	11.52	14.56	16.18	16.32	18.23	21.3	18.26	1.63

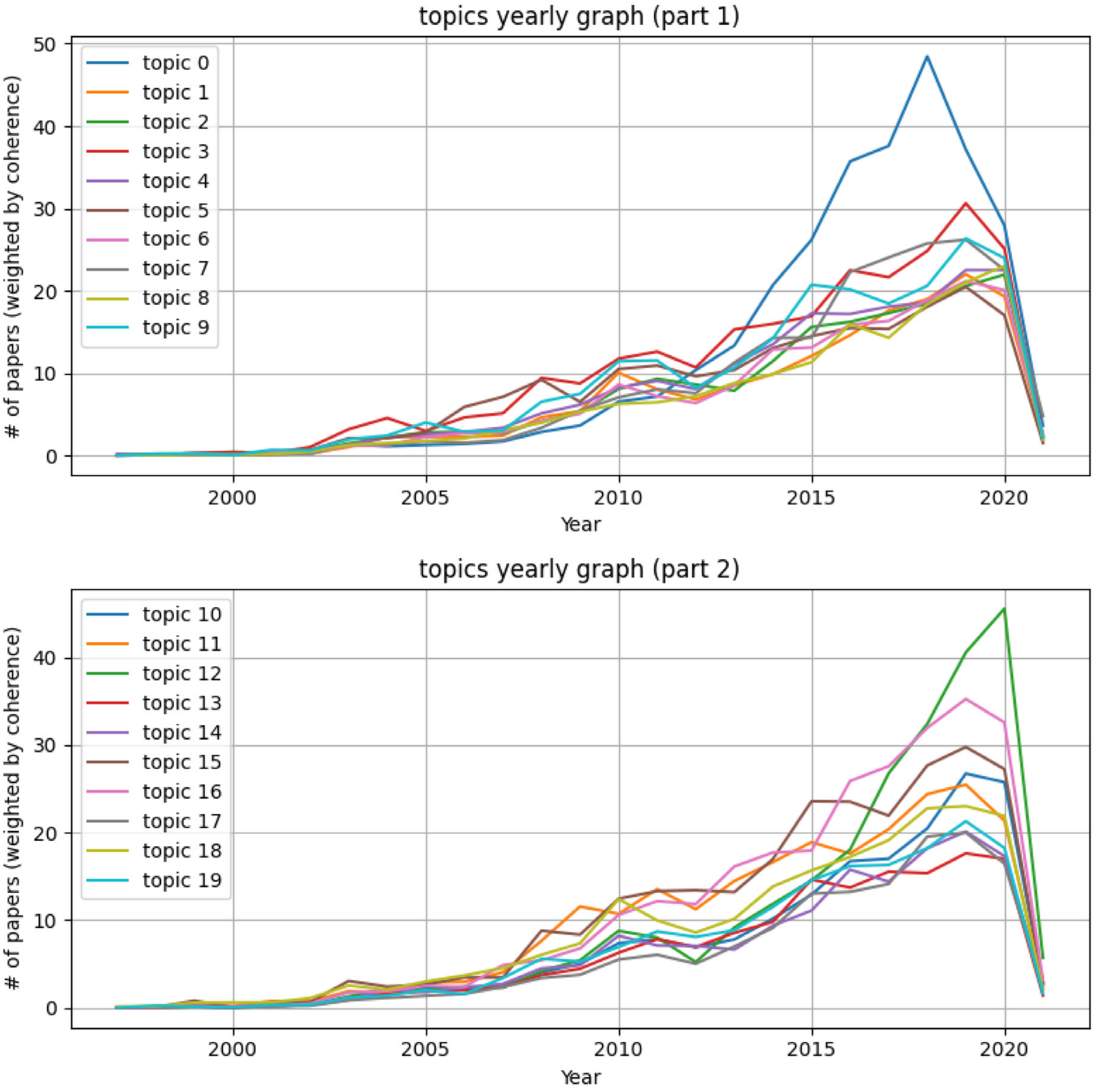


Fig. 6. Evolution of topics over the years.

Table 6
Legend for Journals and their labels.

Journal	Title
J_0	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)
J_1	CEUR Workshop Proceedings
J_2	BMC Bioinformatics
J_3	Advances in Intelligent Systems and Computing
J_4	Journal of Biomedical Informatics
J_5	ACM International Conference Proceeding Series
J_6	Communications in Computer and Information Science
J_7	Studies in Health Technology and Informatics
J_8	IEEE Access
J_9	Expert Systems with Applications

literature published on the journal J_k about the topic T_h. The value of $v(J_k, T_h)$ is calculated as:

$$v(J_k, T_h) = \sum_i C^i_{J_k,T_h}$$

being $C^i_{J_k,T_h}$ the coherence value of the paper D_i published by journal J_k to the topic T_h.

As it can be seen in the tables, there are clearly some journals that tend to prefer publication linked to certain topics, as well as some that have a more homogeneous distribution of topics.

Finally, Table 9 reports the number of papers published in each year. It can be noted that there is an overall growth for most of the top journals, while the bottom part of the table looks more homogeneous.

The application of the LDA algorithm allows to obtain an association between documents and topics. In particular, for each document D'_i, each topic T_h has an associated probability $P_{DT}(i, h)$ of being relevant for the document. A qualitative evaluation of the results shows that the ideal situation is when an article has few "characterizing" topics (typically 1 to 3 topics), i.e., topics having a high probability $P_{DT}(i, h)$. On the opposite, a situation that can be considered unsatisfactory is when the topics associated with the document have a probability distribution close to the uniform distribution, i.e., all the topics have almost the same probability. In fact, in this latter case, there is no possibility to clearly associate the document with specific topics.

7.5. Validation of results

The validation of results aims at the assessment of two aspect: (1) the relevance of each topic associated by LDA to the papers (precision), and (2) how well the set of topics associated with each paper is able to represent the actual topic of the paper (recall). The evaluation has been carried out manually by 3 domain experts for the $N_T = 50$ topmost cited papers in $\mathcal{D}'$.

Relevance of topics (precision)

To evaluate the relevance of topics, we want to assess whether the experts agree with LDA in considering the topics as relevant for the corresponding paper or not.

For this purpose, the jth expert – with $j \in \{1, 2, 3\}$ – is required to assign a score $c^j_{i,h}$ to each topic T_h associated with each paper D_i. The value of $c^j_{i,h}$ is integer and included in the range $[1, 5]$. The meaning of the score is reported in Table 10.

However, the score generated by LDA for each topic is a probability, which indicates how well the topic represents the content of the paper. Such a score is in the range $[0, 1]$, which is not directly comparable with the ratings assigned by the reviewers. Therefore, we determine an equivalent score $c^{LDA}_{i,h}$ that is suitable for the comparison. The score is obtained by converting the probability $P_{DT}(i, h)$ associated with each topic T_h and document D_i into an integer value in the range $[1, 5]$. This is done by dividing the range $[\min P_{DT}(i, h), \max P_{DT}(i, h)]$ into 5 equally sized bins $B_1, B_2, \ldots, B_5$. If the probability $P_{DT}(i, h)$ falls in the bin B_K, then $c^{LDA}_{i,h} = K$.

For example, given a paper D_i, let us consider $\min P_{DT}(i, h) = 0.2$ and $\max P_{DT}(i, h) = 0.7$. In this case, the five above defined bins are $B_1 = [0.2, 0.3)$, $B_2 = [0.3, 0.4)$, $\ldots$, $B_5 = [0.6, 0.7]$. Therefore, a topic T_h that has a probability equal 0.33 for the paper D_i falls in the bin B_2, and has thus a score $c^{LDA}_{i,h} = 2$.

The evaluation essentially consists in the comparison of the evaluation done by the experts with the score $c^{LDA}_{i,h}$ derived for LDA. First of all, we filtered out the topics having $c^{LDA}_{i,h} \leq 3$. The rational behind this choice is that a low probability, corresponding to lower scores, indicates a poor confidence of LDA on the suitability of the topic to the document. Therefore, we focus on the cases in which LDA has a high confidence.

Furthermore, we considered only the topics on which the reviewers are in agreement. To evaluate the agreement, we partitioned the score range in 3 intervals: *low* (scores 1 and 2), *neutral* (3) and *high* (4 and 5). We considered the reviewers in agreement if we had the majority of votes, i.e., 2 out of 3, in either the *low* or *high* bins. For example, if the three scores are 1, 3 and 4, there is no agreement.

For the topics for which the reviewers are in agreement, we considered the average score:

$$\overline{c_{i,h}} = \frac{1}{3} \sum_{j=1}^{3} c^j_{i,h}$$

This average determines the ground truth against which the LDA scores will be tested. The automatic topic assignment is considered correct for a document if the LDA score matches the ground truth for at least one topic in the document. We denote the *true positive set*, D_{TP}, as the set of documents for which the above condition holds. Analogously, we define the *false positive set*, D_{FP}, as the set of documents for which at least one topic assigned by LDA does not match the ground truth. Given these definitions, we compute the precision metric as

$$prec = \frac{|D_{TP}|}{|D_{TP} \cup D_{FP}|}$$

The rationale underlying this formulation is to capture the three cases in which *(i)* there is a perfect matching, i.e., the reviewer fully agrees with the topic assignment made by LDA, *(ii)* there is no matching at all, and *(iii)* there is a partial matching.

The precision obtained by the proposed method is $prec = 86.95\%$.

Coverage of papers (recall)

To assess how well the topics generated by LDA and associated with each paper are representative of the topics of the paper itself, the reviewers assigned a score to each paper D_i, indicated as *coverage*. The coverage of document D_i is denoted with $cvg_i \in \mathcal{N}$. Possible values for cvg_i are in the range $[0, 5]$, where 0 means that the topics do not represent at all the argument of the paper, while 5 means that the topics fully capture the content of the paper.

For example, `slr-kit` determined that T_{15} is the only topic sufficiently representative for the paper "Web-a-where: Geotagging Web content" [36], with a coherence score of 0.35. The three reviewers separately evaluated, using the ratings from 0 to 5, how much the document was covered by the topic T_{15}. In this specific case there was no agreement: one reviewer rated the paper with a coverage score of 5 out 5, which means that T_{15} completely encloses the topic of that paper. Another reviewer fully disagreed, giving a score of 1 out 5. The third reviewer layed in between, expressing with a score of 3 out 5 that the topic of the paper was only partially covered by T_{15}.

Table 7
Number of papers published by each journal according to topics (Part 1/2).

Journal	T_0	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}
J_0	16.94	11.86	13.76	16.87	12.96	14.88	12.89	14.21	12.88	12.78	12.88
J_1	8.49	7.71	6.05	7.79	6.5	10.88	6.01	7.53	6.93	8.21	7.84
J_2	2.35	4.06	4.22	7.7	4.3	9.27	3.69	2.27	3.05	11.95	4.84
J_3	10.21	4.07	3.8	5.52	5.46	3.61	4.01	7.41	4.67	4.93	4.15
J_4	3.18	3.74	4.43	5.78	3.91	4.15	4.17	2.66	3.15	9.76	6.19
J_5	6.49	4.45	3.82	5.24	4.97	3.85	5.25	5.54	3.84	4.04	4.58
J_6	5.87	2.1	2.78	4.81	3.15	2.76	2.63	4.75	2.39	3.2	2.67
J_7	1.34	1.76	1.79	2.35	2.78	1.74	1.81	1.49	1.57	2.35	1.78
J_8	2.53	1.51	1.12	1.47	1.58	1.37	1.37	2.45	2.14	2.66	1.95
J_9	3.16	2.14	1.44	1.61	2.97	1.4	0.99	2.23	0.75	2.44	1.05

Table 8
Number of papers published by each journal according to topics (Part 2/2).

Journal	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}	T_{16}	T_{17}	T_{18}	T_{19}
J_0	17.66	17.21	11.43	10.99	20.01	17.62	10.66	16.92	13.59
J_1	10.9	8.5	4.52	5.61	10.89	8.25	5.39	8.28	4.7
J_2	6.35	6.07	3.64	3.24	12.47	5.57	2.68	4.5	2.77
J_3	4.24	6.24	4.01	4.75	5.59	6.73	3.29	5.57	5.75
J_4	6.73	6.64	3.56	3.68	10.05	6.81	4.1	5.04	3.26
J_5	5.33	5.66	3.55	4.94	6.06	5.53	3.74	4.53	3.58
J_6	4.54	4.31	1.88	3.19	3.97	4.41	1.96	3.19	2.45
J_7	3.51	2.71	1.22	1.62	1.85	2.39	1.22	2.78	1.95
J_8	1.63	3.8	1.28	1.22	2.33	2.76	1.97	1.62	2.25
J_9	1.5	1.99	1.65	1.35	1.43	2.5	1.14	1.27	2.00

Table 9
Number of papers published by each journal in each year on topic of NLP.

Journal	2012	2013	2014	2015	2016	2017	2018	2019	2020
J_0	17	17	16	30	21	30	34	21	19
J_1	5	4	9	9	19	8	15	18	42
J_2	2	4	3	11	5	5	9	10	4
J_3	0	2	11	5	8	8	16	24	18
J_4	2	6	6	22	5	6	9	10	6
J_5	4	2	6	4	9	18	17	14	12
J_6	3	4	2	3	7	6	11	13	7
J_7	1	5	3	6	2	2	2	5	4
J_8	0	0	0	0	1	2	6	16	14
J_9	1	3	5	3	2	5	3	2	1

Table 10
Ratings and their meaning assigned by reviewers to topics.

Value of $c_{i,h}^j$	Meaning
1	The topic is not relevant at all to the document
2	The topic has some few relevance with the document
3	The reviewer is unsure about the relevance of the topic to the document
4	The topic is barely relevant to the document
5	The topic is definitely relevant to the document

The *recall* is computed as the mean of the coverage scores given by the reviewers. Formally:

$$recall = \frac{1}{5}\frac{\sum_{i=1}^{N_T}\overline{cvg_i}}{N_T}$$

where $\overline{cvg_i}$ is the average coverage for paper D_i among the three reviewers. The scaling factor equal to $1/5$ is used to obtain a result in the range $[0, 1]$ starting from scores in the range $[0, 5]$.

The recall obtained by the proposed method is $recall = 78.9\%$.

Comments

At the time of writing this paper, it was not possible to find any other study in the literature with similar assessment and metrics involved. This means that it is not possible to compare our results with others. Nonetheless, we believe that the achieved results are good enough to be used in practical applications to carry out systematic literature reviews, but with room for further improvements.

7.6. Evaluation of the effort of manual phases

The proposed approach includes two manual phases in a SLR: the classification of keywords, tokens and n-grams, and the characterization of topics. Other phases, such as the preprocessing, the extraction of tokens, the generation of topics and the generation of results are completely automatic. As the proposed workflow aims at optimizing the manual phases of the workflow, this section provides an indication of the effort required for the most demanding manual operations.

The most time-consuming phase is the classification of tokens described in Section 7.2. This operation is done using FAWOC. FAWOC logs all the operations, allowing an accurate profiling of the time required for the classification. The classification can be done in distinct sessions, according to the time availability of the expert. For the considered dataset, this phase required the classification of 10,013 terms extracted from 4427 documents. The classification was carried out in 6 different sessions. Two sessions were concluded in less than 2 minutes, three others lasted between 10 and 30 minutes, while there was a long one of 153 minutes. The total time for classifying all the tokens has been about 205 minutes. In average, around 1.23 s per token were required.

The second manual phase is the evaluation of the generated topics, which requires the evaluation of the topics generated by LDA. Since the number of topics is typically low, in the range of 10—20 topics, this phase required the analysis of a limited number of topics. Although for this phase we do not provide a dedicated tool which is able to track the exact duration of the process, we empirically estimated such a duration in 35—40 minutes.

Overall, we estimate that the reported durations can generalize well to other datasets, and are largely independent from the number of papers to classify. In fact, the number of words that are extracted from the abstracts has an inherent limit due to the limited number of words that are commonly used in the writing of scientific texts. For example, in the classification of tokens for a different dataset related to real-time scheduling, we classified 19,973 tokens extracted from 9025 papers in slightly less than 4 hours.

7.7. Threats to validity

Due to the semi-supervised nature of the proposed framework, the involved manual phases may be subject to bias related to the expert who carries out the operations. In particular, the classification of tokens is the most sensitive phase to possible bias. For example, the results may change when different experts have heterogeneous levels of knowledge of the domain. This issue can be mitigated by carrying out the classification by multiple experts leveraging the features of FAWOC that allow the review and the re-labeling of tokens that have been already classified.

The application of the proposed methodology to other domains may also represent a possible threat to the generality of the method. For example, in case an SLR should be carried out on a scientific method that have applications in many domains (e.g., machine learning, deep learning, etc.), the terms appearing in the documents may be too broad to allow the topic modeling algorithm to identify specific topics. These aspects may be mitigated by the proper selection of the documents, which is – however – a phase that is independent from the proposed method and need to be refined with dedicated approaches.

8. Conclusions

This paper proposed a workflow based on semi-supervised machine learning to assist the derivation of statistics regarding the papers involved in a Systematic Literature Review.

The workflow is based on the efficient manual selection of important terms extracted from the abstracts of the documents. The selected terms can be made by multiple consecutive words. Once selected, the terms are used to filter the words in the abstracts. The filtered abstracts are processed by the LDA algorithm to automatically determine a list of topics. The papers are thus clustered according to the topics, and various statistics are derived from the clustered papers. Examples of statistics are the trend of publications per topic during the years, and the amount of papers published on different journals by topic.

While the proposed framework has been developed to help in carrying out an SLR, it can be easily adapted to perform the same kind of processing on different types of documents, such as Web content or blog posts. In fact, an intermediate product of the process is the list of terms that characterize a specific field of knowledge, such as the research papers on NLP as used in the experiments of this paper. Based on the list of terms, the topic modeling step can be applied to any set of documents, being not restricted to the abstracts of research papers. Moreover, the list of terms is a condensed representation itself of the corresponding field of interest, which could be used for other NLP tasks.

There are some elements that could be refined in the proposed tools. A important aspect is the evaluation of ad-hoc strategies for synonyms detection, which could lead to the finding of more refined relationships among papers.

CRediT authorship contribution statement

Tullio Facchinetti: Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Investigation, Formal analysis, Validation of obtained results. **Guido Benetti:** Design and implementation of the Software, Data curation, Visualization, Investigation, Formal analysis. **Davide Giuffrida:** Design and implementation of the Software, Data curation, Visualization, Writing – original draft. **Antonino Nocera:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Investigation, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Barbara Ann Kitchenham, Stuart Charters, Guidelines for Performing Systematic Literature Reviews in Software Engineering, Technical Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007.
[2] Cynthia D. Mulrow, Systematic reviews: rationale for systematic reviews, Bmj 309 (6954) (1994) 597–599.
[3] Mei Kobayashi, Koichi Takeda, Information retrieval on the web, ACM Comput. Surv. 32 (2) (2000) 144–173.
[4] Hiteshwar Kumar Azad, Akshay Deepak, Query expansion techniques for information retrieval: a survey, Inf. Process. Manage. 56 (5) (2019) 1698–1735.
[5] Yuncheng Jiang, Semantically-enhanced information retrieval using multiple knowledge sources, Cluster Comput. 23 (4) (2020) 2925–2944.
[6] Zara Nasar, Syed Waqar Jaffry, Muhammad Kamran Malik, Information extraction from scientific articles: a survey, Scientometrics 117 (3) (2018) 1931–1990.
[7] Nonthapat Pulsiri, Ronald Vatananan-Thesenvitz, Improving systematic literature review with automation and bibliometrics, in: 2018 Portland International Conference on Management of Engineering and Technology (PICMET), IEEE, 2018, pp. 1–8.
[8] Babatunde K. Olorisade, Ed de Quincey, Pearl Brereton, Peter Andras, A critical analysis of studies that address the use of text mining for citation screening in systematic reviews, in: Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering, 2016, pp. 1–11.
[9] Babatunde Kazeem Olorisade, Pearl Brereton, Peter Andras, Reproducibility of studies on text mining for citation screening in systematic reviews: evaluation and checklist, J. Biomed. Inform. 73 (2017) 1–13.
[10] Alison O'Mara-Eves, James Thomas, John McNaught, Makoto Miwa, Sophia Ananiadou, Using text mining for study identification in systematic reviews: a systematic review of current approaches, Syst. Rev. 4 (1) (2015) 1–22.
[11] Jefferson Seide Molléri, Fabiane Barreto Vavassori Benitti, Automated approaches to support secondary study processes: a systematic review, in: SEKE, 2012, pp. 143–147.
[12] Edgar Hassler, Jeffrey C. Carver, David Hale, Ahmed Al-Zubidy, Identification of slr tool needs–results of a community workshop, Inf. Softw. Technol. 70 (2016) 122–129.
[13] Yusra Shakeel, Jacob Krüger, Ivonne von Nostitz-Wallwitz, Christian Lausberger, Gabriel Campero Durand, Gunter Saake, Thomas Leich, (Automated) literature analysis-threats and experiences, in: 2018 IEEE/ACM 13th International Workshop on Software Engineering for Science (SE4Science), IEEE, 2018, pp. 20–27.
[14] James Thomas, John McNaught, Sophia Ananiadou, Applications of text mining within systematic reviews, Res. Synth. Methods 2 (1) (2011) 1–14.
[15] Sophia Ananiadou, Brian Rea, Naoaki Okazaki, Rob Procter, James Thomas, Supporting systematic reviews using text mining, Soc. Sci. Comput. Rev. 27 (4) (2009) 509–523.
[16] Guy Tsafnat, Adam Dunn, Paul Glasziou, Enrico Coiera, The automation of systematic reviews, 2013.
[17] Guy Tsafnat, Paul Glasziou, Miew Keen Choong, Adam Dunn, Filippo Galgani, Enrico Coiera, Systematic review automation technologies, Syst. Rev. 3 (1) (2014) 1–15.
[18] Yutaka Sasaki, Brian Rea, Sophia Ananiadou, Clinical text classification under the open and closed topic assumptions, Int. J. Data Min. Bioinform. 3 (3) (2009) 299–313.
[19] Siddhartha Jonnalagadda, Diana Petitti, A new iterative method to reduce workload in systematic review process, Int. J. Comput. Biol. Drug Des. 6 (1–2) (2013) 5–17.
[20] Makoto Miwa, James Thomas, Alison O'Mara-Eves, Sophia Ananiadou, Reducing systematic review workload through certainty-based screening, J. Biomed. Inform. 51 (2014) 242–253.
[21] Byron C. Wallace, Thomas A. Trikalinos, Joseph Lau, Carla Brodley, Christopher H. Schmid, Semi-automated screening of biomedical citations for systematic reviews, BMC Bioinformatics 11 (1) (2010) 1–11.
[22] Aaron M. Cohen, William R. Hersh, Kim Peterson, Po-Yin Yen, Reducing workload in systematic review preparation using automated citation classification, J. Am. Med. Inform. Assoc. 13 (2) (2006) 206–219.
[23] Kazuma Hashimoto, Georgios Kontonatsios, Makoto Miwa, Sophia Ananiadou, Topic detection using paragraph vectors to support active learning in systematic reviews, J. Biomed. Inform. 62 (2016) 59–65.
[24] Sang-Woon Kim, Joon-Min Gil, Research paper classification systems based on TF-IDF and LDA schemes, Human-Centric Comput. Inf. Sci. 30 (9) (2019).
[25] Rick W. Wright, Richard A. Brand, Warren Dunn, Kurt P. Spindler, How to write a systematic review. Clinical orthopaedics and related research, 2007, pp. 23–29.
[26] David M. Blei, Andrew Y. Ng, Michael I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (null) (2003) 993–1022.

[27] Willian Massami Watanabe, Katia Romero Felizardo, Arnaldo Candido, Érica Ferreira de Souza, José Ede de Campos Neto, Nandamudi Lankalapalli Vijaykumar, Reducing efforts of software engineering systematic literature reviews updates using text classification, Inf. Softw. Technol. 128 (2020) 106395.

[28] Ariel S. Schwartz, Marti A. Hearst, A simple algorithm for identifying abbreviation definitions in biomedical text, in: Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing, 2003, pp. 451–462.

[29] Alexander Yates, Oren Etzioni, Unsupervised methods for determining object and relation synonyms on the web, J. Artificial Intelligence Res. 34 (2009) 255–296.

[30] Kris Heylen, Yves Peirsman, Dirk Geeraerts, Dirk Speelman, Modelling word similarity: an evaluation of automatic synonymy extraction algorithms, in: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), 2008.

[31] Juan Ramos, Using tf-idf to determine word relevance in document queries, 2003.

[32] Lukáš Havrlant, Vladik Kreinovich, A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation), Int. J. Gen. Syst. 46 (1) (2017) 27–36.

[33] David M.W. Powers, Powers evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation, 2020, arXiv preprint arXiv:2010.16061.

[34] Claudia Diamantini, Antonino Nocera, Domenico Potena, Emanuele Storti, Domenico Ursino, Querying the iot using multiresolution contexts, IEEE Internet Things J. 8 (7) (2020) 6127–6139.

[35] Michael Röder, Andreas Both, Alexander Hinneburg, Exploring the space of topic coherence measures, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, in: WSDM '15, Association for Computing Machinery, New York, NY, USA, 2015, pp. 399–408.

[36] Einat Amitay, Nadav Har'El, Ron Sivan, Aya Soffer, Web-a-where: geotagging web content, in: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2004, pp. 273–280.