An open-source integrated framework for the automation of citation collection and screening in systematic reviews.

A Preprint

Angelo D'Ambrosio
Institute for Infection Prevention and Hospital Hygiene
Freiburg University Hospital
Freiburg, Germany
angelo.d.ambrosio@uniklinik-freiburg.de; a.dambrosioMD@gmail.com

Hajo Grundmann
Institute for Infection Prevention and Hospital Hygiene
Freiburg University Hospital
Freiburg, Germany
hajo.grundmann@uniklinik-freiburg.de

Tjibbe Donker
Institute for Infection Prevention and Hospital Hygiene
Freiburg University Hospital
Freiburg, Germany
tjibbe.donker@uniklinik-freiburg.de

February 19, 2022

Abstract

The exponential and prolific growth of published original scientific contributions makes secondary literature abridgements increasingly demanding. We introduce a new open-source framework for systematic reviews that significantly reduces time and human resource allocation for text identification, collection, and screening of scientific literature.

The framework provides three main tools: 1) an automatic citation search engine and manager that allows collecting records from multiple citation databases (Pubmed, WOS, SCOPUS, EMBASE, IEEE) with a unified query syntax, 2) a citation screening tool based on Bayesian active machine learning and natural language processing that requires users feedback only on uncertain classifications to iteratively increase predictive accuracy and, 3) a semi-automatic, data-driven query generator to create new search queries from existing reviewed citation data sets.

The framework was tested on an example topic to collect and screen citations of related scientific production. To evaluate the performance of the active machine learning classifier, we estimated the median posterior sensitivity and efficiency [90% Credible Intervals] using Bayesian simulation to predict the distribution of potentially relevant matches among the records not manually reviewed.

A total of 17,755 unique records were collected through the framework citation manager; 101 of 766 records were found to be relevant after manual evaluation, while the rest were excluded by automatic classification; the theoretical efficiency was 95.6% [95.3%, 95.7%] with a sensitivity of 100% [93.5%, 100%].

A new search query was generated from the labelled dataset, and 82,579 additional records were collected; only 567 records required human review after automatic screening, and six additional positive matches were found. Including the additional records, the overall expected sensitivity decreased to 97.3% [73.8%, 100%] while the efficiency increased to 98.6% [98.2%, 98.7%].

For large studies, the framework can significantly reduce the human resources required to conduct systematic reviews by simplifying citation collection and screening while demonstrating exceptional sensitivity. Such a tool can improve the standardization and repeatability of systematic reviews.

Keywords Systematic review automation · Citation management · Online data collection · Active machine learning · Natural language processing · Bayesian modeling

1 Introduction

Scientific production has experienced continuous exponential growth in the last decades (Larsen and Von Ins 2010; Bornmann and Mutz 2015). This is especially true for biomedical research, a trend further accelerated by the COVID-19 pandemic, thanks to faster article' processing time by publishers and the greater use of preprint databases (Aviv-Reuven and Rosenfeld 2021; Horbach 2020; Hoy 2020). Consequently, it has become harder for researchers and practitioners to stay up to date on the latest findings in their field. Secondary research is of paramount importance in this scenario in that it provides valuable summaries of the latest research results; however, it is becoming ever more challenging in terms of time and human resources required (Allen and Olkin 1999; Borah et al. 2017; A. M. Cohen et al. 2010; Bastian, Glasziou, and Chalmers 2010).

The article collection and screening phases of a systematic review are particularly demanding (Babar and Zhang 2009). First, relevant published research must be collected from scientific databases using appropriately built search queries (retrieval phase); secondly, the scientific citations collected must be screened, selecting only those that are relevant to the topic (appraisal phase) (Bannach-Brown et al. 2019; Tsafnat et al. 2014; Higgins et al. 2019).

Search queries construction is a complex task (Lefebvre et al. 2011; Hammerstrøm et al. 2010), requiring both expertise in the scientific field of interest and some knowledge of the database query languages. The goal is to obtain a set of results that contains all relevant articles (high sensitivity) while keeping the total number of records low (high specificity), possibly focusing on the first at the expense of the second (Hammerstrøm et al. 2010).

If an integrated search tool is not used, manual work is required to download, store and organise the publication data; this approach is complicated by limits to the number of records that can be downloaded at any one time and the need to harmonise different formats and resolve record duplication (Marshall and Wallace 2019).

The citation screening phase is usually the more resource-demanding task in a systematic review: even with appropriately built search queries, the results may easily range in the tens of thousands, of which just a small fraction are actually relevant (Lefebvre et al. 2011). It has been estimated that labelling 10,000 publications can take up to 40 weeks of work and that the average clinical systematic review takes 63 weeks to complete (Bannach-Brown et al. 2019; Borah et al. 2017; Allen and Olkin 1999). A consequence of this is systematic reviews are often already out-of-date by the time they are published (E. M. Beller et al. 2013).

The field of Data Science applied to evidence synthesis and acquisition has greatly maturated in the last years (Marshall and Wallace 2019; E. Beller et al. 2018; Tsafnat et al. 2014). By applying natural language processing (NLP), it is possible to transform free text into quantitative features, with various levels of abstraction and generalisation (Ananiadou and McNaught 2006; K. B. Cohen and Hunter 2008); using machine learning, such text-derived data can be used to map and reproduce human judgment, automating the screening of citations (Ikonomakis, Kotsiantis, and Tampakas 2005).

Automation of systematic reviews has made significant improvements in the last years (Ananiadou et al. 2009; O'Mara-Eves et al. 2015; Tsafnat et al. 2013; Jonnalagadda, Goyal, and Huffman 2015), and it is possible foreseeable that it will become the standard approach in the field (E. Beller et al. 2018), with many solutions already being implemented into commercial or free-to-use tools (see Marshall and Wallace 2019, table 1).

In this manuscript, we present an open source, production-ready framework that further contributes to the state-of-the-art in systematic review automation (SRA) and helpers (SRH) tools. We improve the "retrieval phase" by providing a unified framework for the automated collection and management of scientific literature from multiple online sources. For the citation screening (appraisal) phase, we built an active machine learning-based protocol (Settles 2009; Miwa et al. 2014), which utilises a Bayesian framework to efficiently identify potentially relevant documents that require human review, while automatically screening-out the vast majority of clearly non-relevant ones; the algorithm then requires human review to iteratively increase

classification accuracy. Finally, we included a tool to generate new search queries based on an already labelled citation data set, to identify relevant research that may possibly have been missed by manually-made queries. We tested the framework in the retrieval and appraisal phases of an example topic of interest to our group: the evaluation of the mathematical modelling of patient referral networks among hospitals and their impact on the diffusion of healthcare-associated pathogenic microorganisms; the protocol is published in (Sadaghiani et al. 2020).
In the Methods, we give an overview of the framework, in the Result, we show the outputs and performance of the framework applied to the example topic, and in the Discussion, we explain the methodological rationale for the different components and features of the framework.

2 Methods

2.1 General description

We built an R (R Core Team 2020) based framework to simplify two aspects of systematic literature review: record acquisition and classification. The code used to generate the results in the manuscript is available at `https://github.com/AD-Papers-Material/BART_SystReviewClassifier`, while an updated and ready to use version of the framework is distributed as an R package at `https://github.com/bakaburg1/BaySREn`.
The framework includes several modules that communicate through intermediate outputs stored in standard formats, which make it possible for users to extend the framework or easily integrate it with other tools in their pipeline. See Supplemental Material S1 for an in-depth description of the framework and how to use it.
The tasks carried out by the framework are grouped into "sessions", which comprise obtaining scientific citation data (records) using a search query and then labelling them as relevant ("positive" in the rest of the text) or not ("negative") for the topic of interest with the help of a machine learning engine (Fig. 1). The initial search query should be built using domain knowledge, trying to achieve a high relevant/non-relevant record ratio.
The framework can then generate a new data-driven query from this labelled set to perform a new session to find records possibly missed by the first query.

2.2 Record acquisition and initial labelling

We built a set of tools to allow users to automatically search and download citation data from three major scientific databases ("sources"): Pubmed (`https://pubmed.ncbi.nlm.nih.gov/`), Web Of Science (WOS, `https://apps.webofknowledge.com/`) and the Institute of Electrical and Electronics Engineers (IEEE, `https://ieeexplore.ieee.org/Xplore/home.jsp`). The framework takes care of authorization management for non-open databases like WOS and IEEE. It is also possible to download and import records in the framework manually; this is particularly useful to acquire records from the SCOPUS (`https://www.scopus.com/`) and EMBASE databases (`https://www.embase.com/`), for which a comprehensive API interface was not easy to build. An extra manual search was also necessary for Pubmed, since the API and the web interface have different rule expansion algorithms and return slightly different results ('NCBI Insights : Updated Pubmed e-Utilities Coming in April 2022!', n.d.). A short guide on how to set up the framework for each database supported is available in Supplemental Material S3.
The collected records are merged into a single database, resolving duplicates and different formatting between sources. The records are ordered according to the frequency of the positive query terms (e.g., not preceded by a *NOT* modifier) in the title and abstract ("simple query ordering").
The researcher is then asked to label a number of records to create the "initial training set" needed to start the automatic classification. We recommend manually labelling the first 250 records (see "hyperparameter optimization" later). Simple query ordering increases the positivity rate in the initial training set (Wallace, Small, et al. 2010), leading to higher sensitivity during automatic classification (Chawla, Japkowicz, and Kotcz 2004).

2.3 Text feature extraction

The collected citation data contain a number of fields characterising a scientific publication. The framework models the relevance of a record based on the following fields: title, abstract, authors, keywords, MESH terms (Lipscomb 2000). A range of Natural Language Processing (NLP) techniques (Baeza-Yates, Ribeiro-Neto, et al. 1999; Marshall and Wallace 2019; Ananiadou and McNaught 2006) are employed to convert the textual

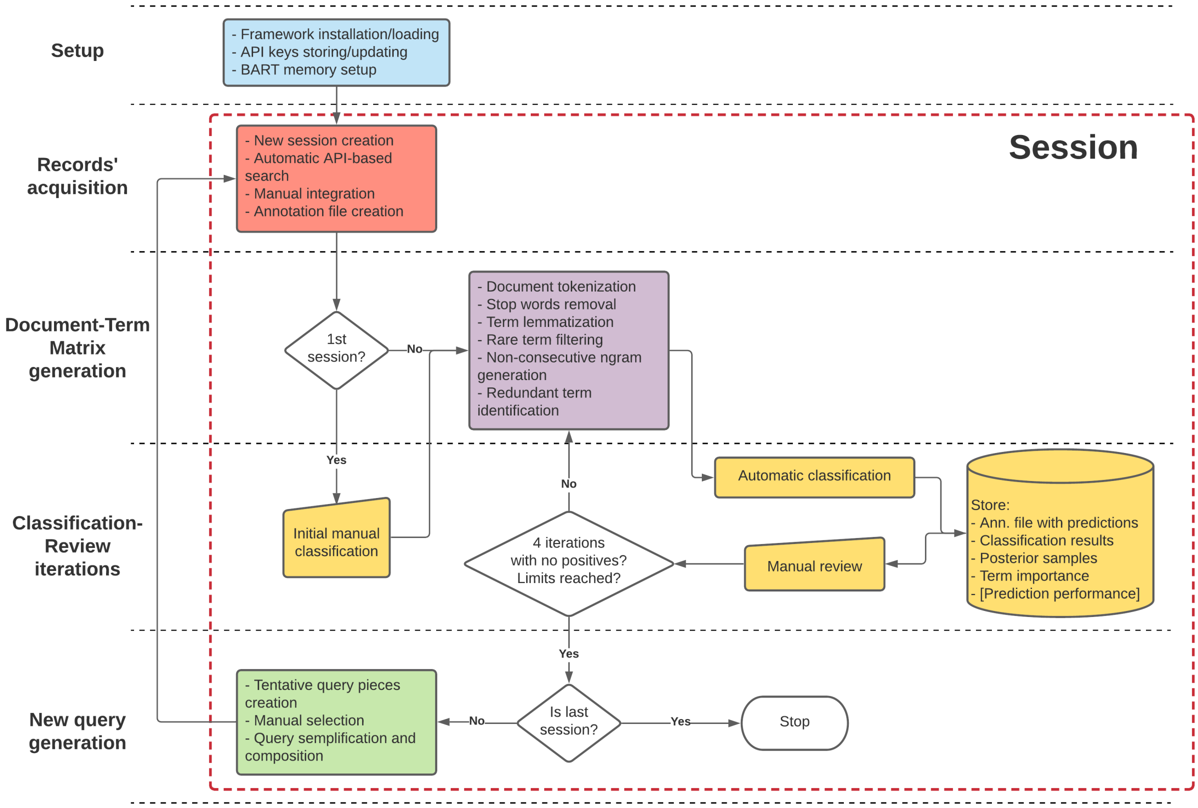
Setup
- Framework installation/loading
- API keys storing/updating
- BART memory setup
Session
Records' acquisition
- New session creation
- Automatic API-based search
- Manual integration
- Annotation file creation
Document-Term Matrix generation
1st session?
No
- Document tokenization
- Stop words removal
- Term lemmatization
- Rare term filtering
- Non-consecutive ngram generation
- Redundant term identification
Yes
Classification-Review iterations
Initial manual classification
No
Automatic classification
Store:
- Ann. file with predictions
- Classification results
- Posterior samples
- Term importance
- [Prediction performance]
4 iterations with no positives? Limits reached?
Manual review
Yes
New query generation
- Tentative query pieces creation
- Manual selection
- Query semplification and composition
No
Is last session?
Yes
Stop

Figure 1.Framework's visual depiction.

information in these fields into features for machine learning through a bag-of-words approach (Marshall and Wallace 2019). Processing of free text fields (title, abstract) includes: tokenization (i.e., extracting the terms), removal of common stopwords (i.e., sentence components having no semantic value), part-of-speech filtering (only nouns, adjectives, verbs and untagged terms are retained), and lemmatisation of terms (i.e., reduction to their base grammatical form). Text processing for authors, keywords and MESH terms identifies logical units (e.g., authors' full names, composite keywords) and extracts them.

Terms appearing in less than 5% of the labelled documents are removed from negative records. All terms in the positive set are kept to increase sensitivity at the cost of specificity.

Some terms tend to co-appear in records (non-consecutive ngrams, nc-ngrams), often carrying a particular meaning when they do co-occur. To detect nc-ngrams, we generated a word network representation (Francois Rousseau 2015) with edges occurring between terms with a cosine similarity in terms of docuement co-occurrence > 0.5. We extracted the maximal cliques in the network (Eppstein, Löffler, and Strash 2010) representing highly correlated groups of terms; these groups are added to the dataset as individual features. Only nc-ngrams comprising a maximum of ten terms are kept.

A second network is built using a co-occurrence threshold of 0.9. In this case, the cliques represent terms that always appear together and can therefore be considered redundant (i.e., they do not need to be considered separately). These terms are merged to increase computational efficiency and reduce overfitting.

The output is a Document-Term Matrix (DTM), with N_d rows representing the records (D_i), N_t terms column for the t_{field} terms (divided by record field) and $0, 1$ values whether $t_{field} \in D_i$. We also enriched the DTM with features referencing the number of terms in each field to help the model scale term importance based on the field length.

2.4 Label prediction

We used a Bayesian Additive Regression Trees (BART) machine learning "classification model" (Chipman et al. 2010) (in the implementation of Kapelner and Bleich 2013) to predict the probability of a record being

relevant, given the information coded into the enriched DTM and the initial training set. We set up the BART model to use 2,000 MCMC iterations (after 250 burn-in iterations) and 50 trees; we used a k value of 2 to regularize extreme prediction and let the model use missing fields in the DTM as features (Kapelner and Bleich 2015). Positive records are oversampled ten times to increase sensitivity (Batista, Prati, and Monard 2004).
The output is a posterior predictive distribution (PPD) for each record describing the probability of it being relevant (i.e., a positive match). An ensemble of ten models was fitted to improve prediction stability by averaging the PPD between models (Zhou 2021; Dietterich 2000).

To assign the labels, we employed an "active learning" approach (Settles 2009; Miwa et al. 2014), where a human reviews a specific subset of predictions made by the machine, which is then retrained on the manually reviewed dataset. This process is carried out iteratively to reduce prediction uncertainty.
Label assignment is done through identification of an "uncertainty zone" the construction of which is possible thanks to the Bayesian nature of BART, which provides full PPDs instead of point-wise predictions for each record.
To describe the process formally, we define

$$\pi_i = \frac{1}{M}\sum_{j=1}^{M} Pr(L_i = 1|DTM, m_j)$$

as the PPD of a record D_i being relevant (i.e, having a positive label, $L_i = 1$), averaging the PPDs of the ensemble of $M = 10$ models m, and

$$\pi_{i,l} = \{\pi_i : Pr(\pi_i) = 1\%\}$$
$$\pi_{i,u} = \{\pi_i : Pr(\pi_i) = 99\%\}$$

respectively as the lower and upper boundaries of the 98% quantile interval of π_i (98% Predictive Interval, 98% PrI).
Then we identify the "uncertainty zone" as

$$U_\pi = [\max \vec{\pi}_u^-, \min \vec{\pi}_l^+]$$

with $\vec{\pi}_u^-$ being the vector of $\pi_{i,u}$ with a negative label and $\vec{\pi}_l^+$ the vector of $\pi_{i,l}$ with a positive label. That is, U_π defines a range of values between the smallest $\pi_{i,l}$ in the set of already labelled positive records L_p and the largest $pi_{i,u}$ related to the negative ones L_n, noting that the two limits can appear in any order.
Consequently, a record D_i will be labelled as positive if

$$\pi_{i,l} > \max_{\pi \in U_\pi} \pi$$

that is, the record lower 98% PrI boundary should be higher than every value in the uncertainty zone. In other words, for a record to be labelled positive, its PPD should be within the range of the mixture of PPD of the records previously labelled positive, and should not cross the distributions of the negative records.
Conversely, a record is labelled as negative if

$$\pi_{i,u} < \min_{\pi \in U_\pi} \pi$$

All other records are labelled as "uncertain".

Manual review is then necessary for: 1) uncertain records, 2) positive records (to avoid false positives), and 3) records whose predicted label differs from the existing manual one. The last case helps identify human errors or inconsistent labelling criteria.

The automatic classification and manual review steps alternate in a loop (CR iterations) until no new positive matches are found in four consecutive iterations.

2.5 Relevant term extraction

With the BART model, it is possible to extract the proportion of times a term is used in a posterior tree over the sum of total inclusions of all variables (Kapelner and Bleich 2013), as a measure of importance of the features in the model. We extracted the terms, the portion of the citation data in which they were used, the average "inclusion rate" among the ensemble models (over 10,000 inclusions) and its ratio over the standard deviation of this inclusion (inclusion stability, IS). For each term we ran a Poisson regression to get the linear association with a positive label and reported it as Relative Risk (RR) with the number of standard errors as significance index (Statistic); the comparison between the inclusion rate in the BART models and the linear association allows to spot relevant non-linear effects (i.e, the feature is relevant only in association with others). In the Results, we only listed the first 15 terms with IS > 1.5, (in order of inclusion rate), while the first fifty terms regardless of inclusion stability are listed in Supplemental Material S2.

2.6 New search query generation

We developed an algorithm that generates a new search query to find further relevant publications missed in the initial search, possibly at a reasonable cost to specificity (i.e., a higher number of negative results).
The algorithm encompasses a number of steps:

- A partition tree (Therneau and Atkinson 2019) is built between the DTM and 800 samples from the PPD; if a term is present multiple times in the DTM (e.g., both in the title and abstract), it is counted just once, and field term count features are removed. This step generates a list of rules composed by AND/NOT "conditions" made of terms/authors/keywords/MESH tokens, which together identify a group of records.
- For each rule, negative conditions (i.e., NOT statements) are added iteratively, starting from the most specific one, until no conditions are found that would not also remove positive records.
- The extended set of rules is sorted by positive-negative record difference in descending order. The cumulative number of unique positive records is computed and used to group the rules. Rules inside each group are ordered by specificity.
- The researcher is then asked to review the rule groups and select one or more rules from each group (useful if they convey different meaning), or edit them if overly too specific positive or negative conditions were included. It is possible to exclude a group of rules altogether, especially those with the poorest sensitivity/specificity ratio.
- The selected rules are joined together by OR statements, defining a subset of records with a sensibly higher proportion of positive records than the original set
- Redundant (i.e., rules whose positive records are already included in more specific ones) and non-relevant rules (i.e., conditions that when removed do not impact sensitivity and specificity) are removed.
- Finally, the rules are re-elaborated in a query that can be used in the major scientific databases.

Because the algorithm is data-driven, it creates queries that effectively select positive records from the input dataset, but may be not specific enough when applied to actual research databases. Therefore we added an extra subquery in AND, that specifies the general topics of our search and narrows the search domain.
The new query was used to initiate a second search session.

2.7 Performance evaluation

We trained a simple Bayesian logistic regression (surrogate model) on the reviewed records to evaluate the consistency of classification model (see Discussion for the theoretical justification). The surrogate model uses as predictor the lower boundary of the 98% PrI of the PPD of the records with weakly regularizing, robust priors for the intercept (Student T with $\nu = 3, \mu = 0, \sigma = 2.5$) and the linear coefficient (Student T with $\nu = 3, \mu = 0, \sigma = 1.5$).
The quality of the model was evaluated through the Bayesian R^2 (Gelman et al. 2019), of which we reported the posterior median and 90% Credible Interval [90% CrI]. The R^2 also provides an evaluation of the consistency of the original classification model. Given that this model is conditional only on the BART predictions and not on the DTM, it is characterized by more uncertainty, providing plausible worst-case scenarios.
The surrogate model is then used to generate the predictive cumulative distribution of the number of total positive records in the whole dataset. This distribution allows to estimate the expected total posterior "Sensitivity" and "Efficiency" of the classification model in the full (unreviewed) dataset. Efficiency is

summarised by the "Work saved over random" (WSoR) statistic: one minus the ratio between the number of record manually reviewed and those that would be required to find the same number of positives if classification were performed choosing records randomly; this last quantity is estimated through a negative hypergeometric distribution (Chae 1993) over the predicted number of positive records.
For the number of predicted positive records, sensitivity and efficiency, we reported the "truncated 90% PrI" [trunc. 90% PrI], i.e., the uncertainty interval bounded by the number of observed total positive records (i.e., there cannot be fewer predicted positive records than observed).

2.8 Hyperparameter evaluation

Our classification algorithm has a number of hyperparameters:

- Size of the initial training set: 50, 100, 250, 500 records;
- Number of models in the ensemble: 1, 5, 10, 20, 40, 60 repetitions;
- Oversampling rate of positive records: (1x (i.e., no oversampling), 10x, 20x;
- PrI quantiles for building the uncertainty zone: 80%, 90%, 98%;
- Source of randomness between models in the ensemble: MCMC sampling only (Robert and Casella 2004), MCMC plus data bootstrapping (Breiman 1996) of the training set.

To evaluate the hyperparameter effect of performance, we set up a "grid search" (Claesen and De Moor 2015; L. Yang and Shami 2020) on a prelabelled "validation set" derived from the first 1,200 records of the first session dataset. Each hyperparameter combination was tested until four CR iterations were completed with no positive records or the whole dataset was labelled.
For each combination, a performance score was computed as the product of "Efficiency" (1 minus the ratio of records that required reviewing over the total number of records) and "Sensitivity" (number of positive records found over the total number of positive records). We then used a partition tree (Therneau and Atkinson 2019) to identify homogeneous "performance clusters" of scores given hyperparameter values. For the rest of the study, we used the best hyperparameter set in terms of Sensitivity followed by Efficiency from the cluster with the highest average score.

3 Results

3.1 First session

The initial search query for the example topic was:

((model OR models OR modeling OR network OR networks) AND (dissemination OR transmission OR spread OR diffusion) AND (nosocomial OR hospital OR "long-term-care" OR "long term care" OR "longterm care" OR "long-term care" OR "healthcare associated") AND (infection OR resistance OR resistant))

selecting only results between 2010 and 2020 (included). Results were collected from Pubmed, WOS, IEEE, EMBASE and SCOPUS, using the framework tools as described in the Methods and Supplemental Material S1.

The first search session returned a total of 27,600 records, specifically 12,719 (71.6% of the total) records from the EMBASE database, followed by 9,546 (53.8%) from Pubmed, 3,175 (17.9%) from SCOPUS, 2,100 (11.8%) from WOS, and 60 (0.34%) from IEEE (Table 1). There were various degrees of overlapping between sources, with 38.4% of records being present in more than one database, and EMBASE and IEEE being the databases with the higher uniqueness ratios. The final data set was composed from 17,755 unique records.
The first 250 records (based on "simple query ordering") were labeled manually. Of these 43 (17.2%) were labeled as positive, and 207 (82.8%) as negative.

The labeled records were used to train the Bayesian classification model used to label the remaining records. After seven classification and review (CR) iterations (three resulting in new positive matches and four extra replications to account for stochastic variability), a total of 101 positives matches were found, requiring manual review of 766 records (13.2% positivity rate).
It is noticeable how the number of records that required manual review decreased rapidly between iterations (Table 2), indicating that the engine was converging while the uncertainties were resolved.
This phenomenon is better illustrated in Fig. 1 of the Supplemental Material S2. It shows the mixture

Table 1. Distribution of retrieved records by source and session. For each source it is reported the number of records, percentage over the session total (after removing duplicates), and number or records specific for a source as absolute value and as percentage over the source total. All session shows records after joining and deduplication of the Session 1 and Session 2 data set.

Session	Source	Records	% over total	Source specific records	% over source total
Session1	Total	17,755			
	Embase	12,719	71.6%	6,683	52.5%
	Pubmed	9,546	53.8%	3,457	36.2%
	Scopus	3,175	17.9%	298	9.39%
	WOS	2,100	11.8%	473	22.5%
	IEEE	60	0.34%	29	48.3%
Session2	Total	82,579			
	Embase	48,396	58.6%	40,826	84.4%
	Pubmed	28,811	34.9%	18,021	62.5%
	Scopus	17,070	20.7%	4,908	28.8%
	WOS	12,956	15.7%	2,817	21.7%
	IEEE	61	0.074%	22	36.1%
All Sessions	Total	98,371			
	Embase	59,604	60.6%	46,942	78.8%
	Pubmed	37,278	37.9%	21,371	57.3%
	Scopus	19,353	19.7%	5,181	26.8%
	WOS	14,367	14.6%	3,175	22.1%
	IEEE	108	0.11%	48	44.4%

distribution of the PPDs of the records, specifically for records that were manually reviewed, before and after the classification step: it can be seen how the distribution of uncertain records shrinks (i.e., it becomes concentrated in a shorter probability range) and shifts toward the negative zone as more positive matches are found and reviewed.

Table 2. Results of the automatic classification and manual review rounds. For each iteration, the cumulative number of positives and negative records and their sum (Total labelled) and percentage over total are shown. Also, the number of changes after review and their description is reported."Unlab." indicates unlabelled records marked for review. For each Iteration, the number of features used by the engine is also reported. The first row reports the results of the initial manual labelling of records, which served as input for the automatic classification in Iteration 1. In Session 2, the engine uses the labels at the end of Session 1 to classify the newly added records.

Session	Iteration	Positives	Negatives	Total labelled (%)	Unlab. -> y	Unlab. -> n	Unlab. -> *	n -> y	Changes	N. features
Session1 (n = 17755)	Initial labelling	43	207	250 (1.41%)	43	207	0	0	250	2,289
	1	93	529	622 (3.5%)	50	322	0	0	372	2,289
	2	100	614	714 (4.02%)	6	86	0	1	93	3,750
	3	101	625	726 (4.09%)	1	11	0	0	12	3,834
	4	101	648	749 (4.22%)	0	23	0	0	23	3,856
	5	101	651	752 (4.24%)	0	3	0	0	3	3,856
	6	101	660	761 (4.29%)	0	9	0	0	9	3,856
	7	101	665	766 (4.31%)	0	5	0	0	5	3,856
Session2 (n = 98371)	1	106	934	1040 (1.06%)	5	270	998	0	1,273	4,729
	2	107	1,123	1230 (1.25%)	1	189	0	0	190	4,729
	3	107	1,176	1283 (1.3%)	0	53	0	0	53	4,733
	4	107	1,200	1307 (1.33%)	0	24	0	0	24	4,729
	5	107	1,209	1316 (1.34%)	0	9	0	0	9	4,729
	6	107	1,226	1333 (1.36%)	0	17	0	0	17	4,729

We extracted the 15 term more relevant for the classification model, described as: Term (citation part): Inclusion Rate (Inclusion Stability) [linear Relative Risk, Statistic].

Patient Transport (Keyword): 61.2 (3.77) [99.1, 21.3], Transfer (Abstract): 57 (3.93) [22.5, 15.4], Network (Title): 56.5 (2.91) [18, 14.2], Network & Patient (Abstract): 54.2 (4.66) [26.3, 15.2], Donker T (Author): 53.5

(4.56) [159, 16.5], Worker (Abstract): 50 (3.33) [0.421, -1.21], Hospitals (Keyword): 49.8 (4.31) [27.8, 16.5], Movement (Abstract): 47.8 (2.7) [27.2, 15], Spread (Title): 46.6 (2.25) [16.2, 12.1], Facility (Abstract): 45 (2.22) [19.6, 14.8], Orange County (Keyword): 44.3 (3.19) [199, 17.2], Conduct (Abstract): 42.6 (3.7) [0.221, -2.57], Patient (Abstract): 42 (3.61) [27.6, 7.23], Perform (Abstract): 41.9 (2.38) [0.342, -2.55], Hospital (Title): 39 (1.95) [12.5, 12.5].

The "&" indicates nc-ngrams, i.e., terms strongly co-occurrent in the documents.
The engine was able to pick up the central concept of the research topic, i.e., "patient transport" or "transfer" through a "network" of "facility"ies that facilitates the "spread" of infections, and even one of the authors of this study (Donker T.) as well as the region of interest ("Orange County") of another research group active on the topic of pathogen spreading over hospital networks. Some terms were considered highly relevant by the BART models (e.g., "Worker" in 6th position out of more than 3800 terms considered) although in a simpler linear model their effect would hardly be significant (statistic: -1.21 s.e.); these are terms that are only relevant in conjunction with other terms but not by their own, highlighting the extra predictive power achievd through the use of advanced, non-linear machine learning.
A more extensive set of terms is presented in Table 1 of the Supplemental Material S2.

3.2 Second session

The results of the first classification session were used to create a second, data-driven query with the objective of performing a more extensive search to find records that may have been missed during the first search session. The resulting query was as follows:

(((Donker T) NOT (bacterium isolate)) OR ((network patient) AND (resistant staphylococcus aureus) NOT (monte carlo) NOT isolation) OR (facility AND (network patient) AND regional NOT hospitals NOT increase NOT (patient transport) NOT (control infection use)) OR ((patient transport) NOT (Donker T) NOT worker) OR (hospitals AND (network patient) NOT (patient transport) NOT regional NOT clinical) OR (facility AND (network patient) NOT hospitals NOT (patient transport) NOT regional NOT prevention NOT medical) OR ((healthcare facility) NOT (Donker T) NOT worker NOT positive) OR (hospitals NOT (network patient) NOT medical NOT environmental NOT outcome NOT global) OR ((network patient) NOT facility NOT hospitals NOT (patient transport) NOT therapy NOT global)) AND ((antimicrobial resistance) OR (healthcare infection))

The final piece *AND ((antimicrobial resistance) OR (healthcare infection)* was added manually to better define the search domain, since the algorithm was trained on documents that were all more or less related to these topics.
The generated query also provides a more nuanced understanding of the engine's internal classification logic, and this is helpful to spot possible biases in the model.

The search was done with the same year filter and procedures used in the first session.

The new search produced 107,294 records (Table 1), of which 48,396 (58.6%) from the EMBASE, followed by 28,811 (34.9%) from Pubmed, 17,070 (20.7%) from SCOPUS, 12,956 (15.7%) from WOS, and 61 (0.074%) from IEEE; compared with the first session, the relative weight of EMBASE and Pubmed decreased, while the level of content specificity greatly increased, as it was for SCOPUS. After removal of duplicates, 82,579 unique records were obtained. The newly collected records were joined with those from the first session and duplicates were removed. We obtained 98,371 unique records, with just 1,963 shared records between searches, which equates to 2% of the total. The percentage of records shared by two or more source dropped to 22%.

Six CR rounds were necessary to complete the second session classification, with just 6 new positive found after reviewing 568 extra records. The first CR iteration required the user to review a substantial number of records (1,273); however, just labelling 275 of them (the suggested 250 plus 25 alredy labelled for the framework hyperparameter tuning) was sufficient to reduce this number to just 190 in the subsequent round. An evaluation of the convergence (Supplemental Material S2, Fig. 1) showed that, in addition to the dynamics already observed in session 1 (shrinkage and negative shift), a second mode appeared in the mixture distribution of the records to be reviewed, centred in a highly positive zone. The interpretation is that as the number of negative training records increases, the engine becomes more and more sceptical and even asks to review some records labelled as positive in the initial training set generated during Session 1.

This behaviour can be useful to spot classification errors and inconsistencies. Considering both sessions, 1,333 records were manually reviewed and 107 (8.03%) confirmed positive matches were found.

Again, the evaluation of the inclusion rate of the terms showed that the engine is quite capable of internalizing the concepts behind the research topic. A subsample of the relevant terms used by the model in the second session is reported in Table 2 of Supplemental Material S2.

3.3 Hyperparameter selection

As described in the methods, hyperparameters were selected by evaluating sensibility and efficiency through a grid search on a validation set of 1,200 manually labelled records. The analysis suggested that the following parameter combination performed best: an initial input of 250 labelled records with 10x oversampling of positive matches, an averaged ensemble of 10 models, no bootstrapping and an uncertainty zone defined by the 98% predictive interval. This combination of parameters was associated with a sensitivity of 98.8% (81 / 82 positive matches found) and an efficiency of 61.5% (462 / 1200 records evaluated). The detailed results of the hyperparameter tuning analysis are reported in Table 3 of Supplemental Material S2. Fig. 2 in Supplemental Material S2 demonstrates that the positive record oversampling rate, the number of ensemble models and the size of the initial training set were the parameters that mostly impacted performance.

3.4 Performance evaluation

To evaluate the theoretical performance of the engine, a surrogate Bayesian logistic regression model was trained on the manually reviewed labels using only the lower boundary of the record PPDs as predictor (see the Methods for details). The surrogate model showed the high predictive power of the scores produced by the classification model (Bayesian R2: 98.1% [97.4%, 98.3%] for session 1 and 98.2% [97.6%, 98.3%] for session 2).

Fig. 2 presents the actual and predicted (from the surrogate model) cumulative number of positive matches, ordered by the initial simple ordering query: the median of surrogate models' cumulative predictive distributions matches quite well the actual number of positive records found. It is striking how many more records would have required manual evaluation to find the same number of positive matches without using a smart classification tool, with some positive matches found close to the end of the heuristically ordered list of records.

Table 3 shows various performance indexes for both sessions, both descriptive (Total records, Reviewed records, Observed positive matches) and estimated through the surrogate model (Expected efficiency, Predicted positive matches, Expected sensitivity, R^2).
In session 1 we observe an expected total number of positives of 101 [101, 108] for an estimated sensitivity of 100% [93.5%, 100%] and efficiency of 95.6% [95.3%, 95.7%]. In session 2 we observed a drop in expected sensitivity, especially in the lower credibility boundary (97.3% [72.8%, 100%]), due to the fact that as the number of records increases, even a small probability of being a positive match can, in the worst case scenario, lead to a relevant number of missed positive matches (147 in this case). To ensure that no obvious positive matches were missed, we evaluated 100 additional records among those not reviewed with the highest median predicted probability produced by the engine and found no additional positive matches.

4 Discussion

We propose a new integrated framework to help researchers collect and screen scientific publications that is characterised by high performance and versatility. This framework joins the joining the growing field of systematic review automation (SRA) and helpers (SRH) tools (A. M. Cohen et al. 2006, 2010; Ananiadou et al. 2009; O'Mara-Eves et al. 2015). This framework implements standard approaches and uses ad-hoc solutions to deal with common SRA issues. By freely sharing the tool as an open-source R package and by following a modular design, we sought to adopt some of the so-called Vienna Principles advocated by the International Collaboration for the Automation of Systematic Reviews (ICASR) (E. Beller et al. 2018).
The framework consists of four main components: 1) an integrated query-based citation search and management engine, 2) a Bayesian active machine learning-based citation classifier, and 3) a data-driven search query generation algorithm.

Table 3. Estimated performance summary. The table reports for each session, the number of reviewed records and the percentage over the total. Also, the posterior expected number of positive records, "Sensitivity" and "Efficiency" (as WSoR) are reported, with their 90% PrI truncated to the observed realization in the dataset [trunc. PrI] (see. methods). Finally the median Bayesian R^2 [90% CrI] of the logistic models is reported. PrI: Predictive Intervals; CrI: Credibility Intervals.

Indicator	Session 1	Session 2
Total records	17,755	98,371
Reviewed records (% over total records)	766 (4.31%)	1,333 (1.36%)
Expected efficiency (over random) [trunc. 90% PrI]	95.6% [95.3%, 95.7%]	98.6% [98.1%, 98.6%]
Observed positive matches (% over total records)	101 (0.57%)	107 (0.11%)
Predicted positive matches [trunc. 90% PrI]	101 [101, 108]	110 [107, 147]
Expected sensitivity [trunc. 90% PrI]	100% [93.5%, 100%]	97.3% [72.8%, 100%]
Simple Model R^2 [90% CrI]	98.1% [97.4%, 98.3%]	98.2% [97.6%, 98.3%]

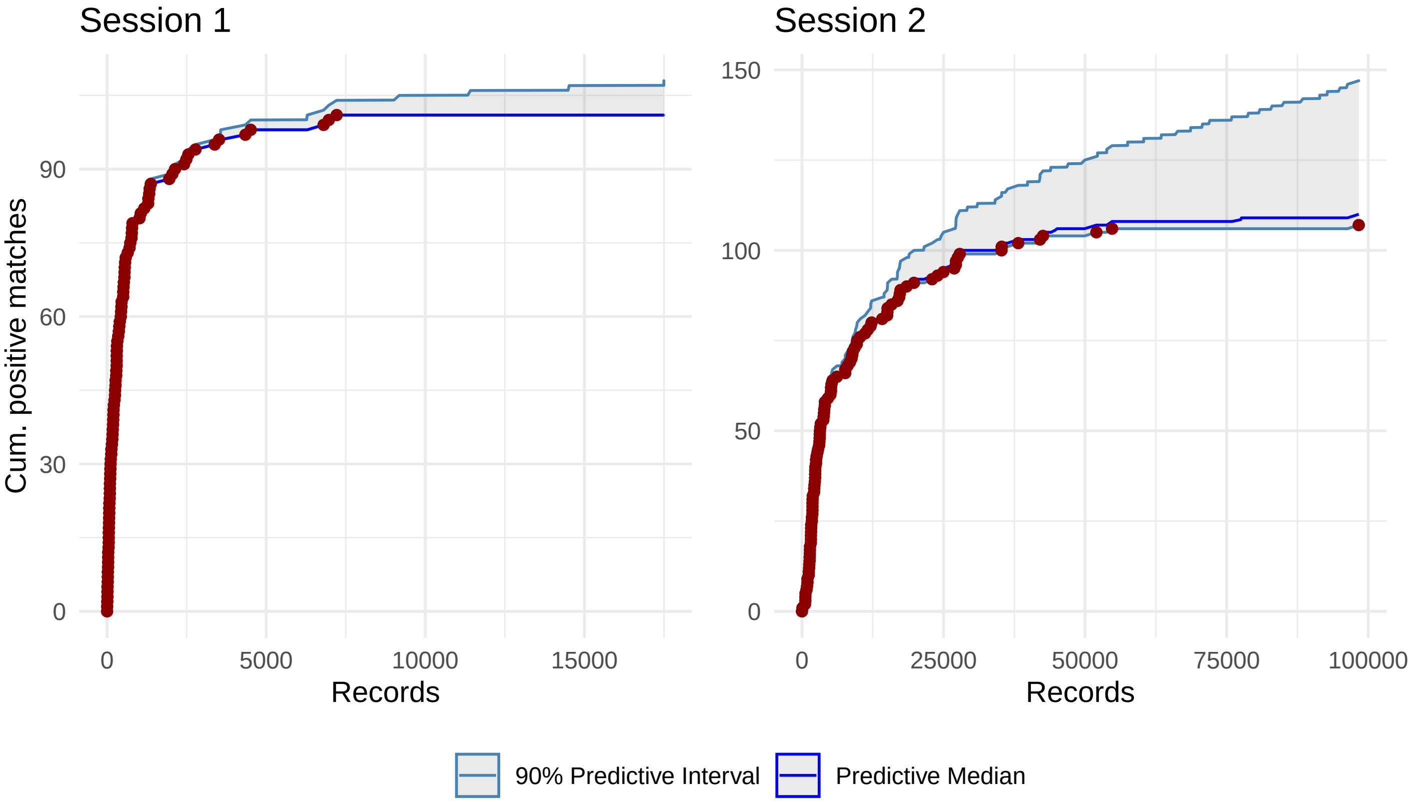

Figure 2. Observed cumulative number of positive matches (red dots) sorted by simple query ordering. The [trunc. 90% PrI] of the cumulative positive matches estimated by the logistic Bayesian model is shown as a shaded area delimited by the 95% quantile of the PrI and by the observed number of positive matches (light blue lines). The median of the PrI is represented by a darker blue line.

The search engine module used by the framework is able to automatically collect citation data from three well-known scientific databases (i.e., Pubmed, Web of Science, and the database of the Institute of Electrical and Electronics Engineers) asand to process manually downloaded results from both the two more databases (SCOPUS, EMBASE). In comparison, most commercial or free SRH tools, rely on internal databases (e.g., Mendeley `https://www.mendeley.com/`) sometimes focusing only on a particular topic (Visser 2010) or on a single external data source (Thomas and Brunton 2007; Poulter et al. 2008; Soto, Przybyła, and Ananiadou 2019).
Mixing different databases is essential to obtain a more comprehensive view of the literature (Bajpai et al. 2011; Wilkins, Gillies, and Davies 2005; Woods and Trewheellar 1998): in our results, 18.7% of the positive

matches were found in only one of the different data sources, and no positive record was present in all the sources (data not shown).
The framework online search algorithms are efficient enough to manage tens of thousands of search results, using various solutions to overcome the limitations of citation databases in terms of traffic and download quotas. The results are then automatically organized, deduplicated and arranged by "simple query ordering" in a uniform corpus. The preliminary ordering allows to increase the positivity rate in the initial training set (Wallace, Small, et al. 2010).

For the framework's record classification module, we developed an active machine learning approach (Settles 2009; Miwa et al. 2014) which is based on the best practices from other SRA studies but also brings further improvements at various levels.
The feature extractor module uses modern NLP techniques (Ananiadou and McNaught 2006; K. B. Cohen and Hunter 2008) to transform text into input data for machine learning. We did not include classical n-grams (Schonlau and Guenther 2017); rather we used network analysis to find non-consecutive, frequently associated terms, a generalisation of n-grams that relaxes the term adjacency assumption. This approach can also be used to find term connections across parts of the records, e.g., whether a term has a different relevance when associated with a particular author. The same technique, albeit with different parameters, was used to merge redundant terms to make the model estimation more efficient and reduce noise.
The use of concurrency network-driven modelling of text is not new (Francois Rousseau 2015; Violos et al. 2016; François Rousseau, Kiagias, and Vazirgiannis 2015; Ohsawa, Benson, and Yachida 1998) and is a valuable tool to extract semantic information that is not evident in one-word or consecutive n-gram models.

The automatic classification algorithm is based on Bayesian Additive Regression Trees (BART) (Chipman et al. 2010; Kapelner and Bleich 2013). Like other boosted trees algorithms (Hastie, Tibshirani, and Friedman 2009), the BART method can explore complex non-linearities, perform variable selection, manage missing data while maintaining high predictive power.
However, the Bayesian foundation of the method provides further benefits: lower sensitivity to the choice of hyperparameters, natural regularisation through priors, and, most imporantly, predictive distributions as output instead of point-wise predictions (Soria-Olivas et al. 2011; Joo, Chung, and Seo 2020; Jospin et al. 2020). By selecting relatively tight prior distributions, we discouraged overly deep trees, long tree sequences, and extreme predicted probabilities, this reducing the risk of overfitting.
The algorithm runs multiple replications of the model and averages their predictive distributions creating an "ensemble"; this technique has been shown to improve out-of-sample predictive performance (Zhou 2021; Dietterich 2000), as we were able to confirm during the hyperparameter evaluation (Supplemental Material S2). Ensembling reduces the uncertainty in the predictive distribution tails related to the randomness in the MCMC fit (Robert and Casella 2004), generating a shift in the probability mass towards the distribution centre and stabilising it (i.e., reducing variance without impacting bias). On the other hand, simply imposing more robust uninformative priors against extreme predictions would have reduced variance but also shifted the distribution towards a non-decision zone, increasing bias (Hansen et al. 2000).
Since the number of model replications has a significant impact on computation times, we decided to use ten replicas, the lower value after which performance stabilised, as resulted from the evaluation of the hyperparameters (Supplemental Material S2, Fig. 2).
We also investigated whether bootstrapping between replications (Breiman 1996) would improve performance; however, contrary to theory (Díez-Pastor et al. 2015), it appeared to be slightly detrimental in our case (Supplemental Material S2, Fig. 2) compared to simple ensembling.

A low rate of relevant matches (class imbalance) is typical for literature reviews (Sampson, Tetzlaff, and Urquhart 2011; Wallace, Trikalinos, et al. 2010; O'Mara-Eves et al. 2015), and a strong imbalance between positive and negative records can affect sensitivity (Khoshgoftaar, Van Hulse, and Napolitano 2010; Chawla, Japkowicz, and Kotcz 2004).
To overcome this problem, we oversampled (Batista, Prati, and Monard 2004) the positive records ten times before model fitting. The hyperparameter analysis showed that the oversampling rate, together with model ensembling, was the parameter with the greatest impact on performance.
A known risk with positive oversampling is the misclassification of negative records (Ramezankhani et al. 2016). However, since all predicted positives in our approach are reviewed manually, we are always guaranteed to achieve 100% specificity/positive predictive value: the only price for the increased sensitivity due to oversampling is a larger number of records to be reviewed.

An alternative to oversampling would be to apply different weights and/or costs to the classes (Abd Elrahman and Abraham 2013; Díez-Pastor et al. 2015), but the BART implementation we used did not have this feature; furthermore, using simple oversampling allows for broader compatibility with different modelling engines (Galar et al. 2011; Roshan and Asadi 2020).
Finally, sorting the records by query term frequency (simple query ordering) produces a much higher rate of relevant records in the initial training set (17.2%) compared to the overall data (0.11%), which boosts the sensitivity of the model.

One of the key innovations we have introduced is the concept of "uncertainty zone", thie implementation of which is possible thanks to the Bayesian foundation of the classification model.
This construct guides the selection of records to be manually review and gets dynamically updated and reduced after each CR iteration, as more uncertain predictions are evaluated (Supplemental Material S2 Fig. 1).
This approach overcomes the usual requirement of dataset-specific hard thresholds in active machine learning, and also allows to review multiple items at once between iterations (Laws and Schütze 2008; Miwa et al. 2014; Zhu et al. 2010). The parameters required by our algorithm are general and non task-specific, like the PPD intervals based on which the uncertainty zone is built, and the maximum number of iterations without positive matches after which a session is concluded; evaluation of the hyperparameters shows that the algorithm is robust against variations in these parameters and we expect the default values to perform well on most datasets.
Since researchers are asked to review both records with a positive predicted label and those inside the uncertainty zone, this method can be considered as a unifying synthesis of the "certainty" and "uncertainty" paradigms of active learning (Miwa et al. 2014).

We assessed performance as the ability of the screening procedure (automatic classification plus manual review) to find the largest number of relevant records while requiring manual reviewing for as few of them as possible (i.e., sensitivity × efficiency).
We avoided the classical out-of-sample approaches such as train-test sampling, out-of-bag bootstrapping or cross-validation (Kohavi et al. 1995; James et al. 2013). Such methods primarily assume that the rate of positivity is the same on average in every possible random subset of the data (Tashman 2000); this uniformity is broken by how the initial training set and the subsequent reviewed records are selected by the query-based ordering and active learning algorithm, resulting in a lower positivity rate in the unlabelled records (Fig. 2). Moreover, a literature corpus is unique per search query/database combination, and therefore any out-of-sample performance estimate is not replicable since no new data can be acquired related to the current corpus.
To estimate overall sensitivity, we instead applied simple Bayesian regression (surrogate model) to the manually reviewed data to abstract the classification model predictions and achieve a maximum entropy (Harremoës and Topsøe 2001) estimate of the number of missed positive matches among the unreviewed records in the whole dataset. This simple surrogate model fitted the data very well (R^2 consistently above 97%) using only the lower 98% PrI boundary of the PPDs as predictor, indicating predictive consistency in the classification model. The posterior predictive distribution of the surrogate model could be used to explore worse case scenarios in terms of sensitivity.

By screening only a very small fraction of all records, our framework achieves very high sensitivity, bringing a meaningful reduction in workload.
Based on the surrogate model, we predicted a predicted median sensitivity of 100% [93.5%, 100%] in the first session (screening 4.29% of records) and of 97.3% [73.8%, 100%] in the second (screening 1.34% of records): efficiency increased significantly in the second session as only a few new positive matches were found; however, given the large number of records, uncertainty about sensitivity also increased, as expected. Both results are above the usual performance in this field (O'Mara-Eves et al. 2015) and are in line with the average sensitivity of 92% estimated after human-only screening (Edwards et al. 2002). In one interesting case, the model detected a human-caused misclassification error, demonstrating its robustness and value as a second screener, a role already suggested for SRA tools in previous studies (Frunza, Inkpen, and Matwin 2010; Bekhuis and Demner-Fushman 2012, 2010). Although "simple query ordering" concentrated most relevant matches in the first 20-25 thousand records, without the tool support the remaining relevant records would have been missed withouth manually screeening almost the entire dataset.

The model required ~5-20 minutes per iteration to perform the predictions in session 1 (17,755 documents) and 20-40 minutes in session 2 (98,371 documents) on an eight-core, 2.5 GHz, 16 GB RAM, 2014 laptop; including manual record review, one session required 1-3 days of work, for a total of 1-2 weeks for the whole process (including record collection). This is a considerable time saving compared to the several months typically required for the screening phase of systematic reviews (Bannach-Brown et al. 2019; Borah et al. 2017; Allen and Olkin 1999). To our knowledge, the amount of data processed (~100,000 records) was larger than what is typical of most SRA studies (O'Mara-Eves et al. 2015; Olorisade et al. 2016), highlighting the scalability of the tool in real-world scenarios.

The last module of our framework is an algorithm for data-driven search query generation. Generating an efficient and effective search query is a complex task (Lefebvre et al. 2011; Hammerstrøm et al. 2010); it requires building a combination of positive and negative terms to maximise the number of relevant search results while minimising the total number of records to be reviewed. Our solution combines a sensitivity-driven subquery proposal engine based on concurrent decision trees (Blanco-Justicia and Domingo-Ferrer 2019; Moore et al. 2018) built on the BART ensemble PPD, with a human review step and an efficiency-driven query builder. The aim is to generate a new search query to help find records missed in the first search session. The generated query did indeed to retrieve a few more relevant records that were not found in session 1, but at the cost of a significant increase in the number of documents.

An interesting aspect of this feature is that it provides a human-readable overview of the classification rules learned by the classification model, showing which combination of terms was particularly relevant and even spotting authors and geographical locations associated with the study topic. The generated query, therefore, served also as a means for machine learning explainability (Bhatt et al. 2020; Burkart and Huber 2021), useful for understanding and detecting biases in black-box classification algorithms (Malhi, Knapic, and Främling 2020); explainability is often required or even legally mandatory for high-stake machine learning applications (Bibal et al. 2021, 2020).

It is important to note that this process is entirely data-driven. The algorithm is only aware of the "world" defined by the dataset used as input, which is generated by a specific search query focused on a particular topic. Therefore, the new query may not be specific enough when applied to an unbounded search domain, and may return an unmanageable amount of unrelated results. The solution we found was to add another component to the query, specifying the general topic (antimicrobial resistance and healthcare-associated infections) of our research.

As mentioned early, our framework builds on modularity. We have designed it in such a way that full independence of the main modules can be easily implemented in future iterations; it will be possible for users to add custom features such as citation search and parsing for other scientific databases, alternative text processing algorithms or machine learning modules. We consider such interoperability to be extremely relevant: the main strength of our tool lies in the composition of many solutions such as the idea of Bayesian active machine learning and the exploit of the derived uncertainty in defining the records needing human review.

However, each components could benefit considerably from the recent improvements in text mining and machine learning.

For example, our text processing approach, based on the "boolean bag-of-words" paradigm, is quite simple, and indeed could be improved by more nuanced text representations. It might be considered whether feature transformations such as TF-IDF (Baeza-Yates, Ribeiro-Neto, et al. 1999; Ananiadou and McNaught 2006) could be advantageous, although we hypothesise that tree-based classification algorithms like BART are robust enough not to require such operations. Instead, it might be worth exploring the application of word embedding: this technique transforms terms into semantic vectors derived from the surrounding text (Turian, Ratinov, and Bengio 2010; Bollegala, Maehara, and Kawarabayashi 2015; Minaee et al. 2021) and could be used to reduce noise by merging different terms that are semantically similar and enhance signal by distinguishing identical terms with different meaning given the context. Another option would be to employ unsupervised learning models like Latent Dirichlet Analysis, Latent Semantic Analysis (Pavlinek and Podgorelec 2017; Q. Chen, Yao, and Yang 2016; Landauer, Foltz, and Laham 1998) or graph-of-word techniques (Ohsawa, Benson, and Yachida 1998; Francois Rousseau 2015) to extract topics to expand the feature space.

Our classification algorithm is applicable with any Bayesian supervised machine learning method that provides full PPDs; therefore, alternative classification models, such as Gaussian Processes which are known for their flexibility (Jayashree and Srijith 2020; S.-H. Chen et al. 2015), could be evaluated. It would be even more interesting to test advanced learning algorithms that go beyond the bag-of-words approach and take

into consideration higher-level features in the text such as term context and sequences, long-distance term relationships, semantic structures, etc., (Cheng et al. 2019; Minaee et al. 2021; Li et al. 2020; J. Yang, Bai, and Guo 2020; Lai et al. 2015; Farkas 1995), provided that a Bayesian implementation of such algorithms is available (for example C. Chen, Lin, and Terejanu (2018)).
Finally, a natural improvement would be to provide a graphical user interface to make the framework easy to use also for less technical users.

The field of literature review automation is evolving rapidly, and we anticipate an increasing use of such technologies to address the accelerating pace of scientific production. We believe it is encouraging that a wide variety of tools are being made available to allow researchers and policymakers find the approach that best fits their needs.
We are contributing to this field with an innovative framework that provides excellent performance and easy integration with existing systematic review pipelines. The value of this work lies not only in the framework itself, which we make available as open-source software, but also in the set of methodologies we developed to solve various SRA issues and which can also be used to improve already existing solutions.

References

Abd Elrahman, Shaza M, and Ajith Abraham. 2013. 'A Review of Class Imbalance Problem'. *Journal of Network and Innovative Computing* 1 (2013): 332–40.

Allen, I Elaine, and Ingram Olkin. 1999. 'Estimating Time to Conduct a Meta-Analysis from Number of Citations Retrieved'. *Jama* 282 (7): 634–35.

Ananiadou, Sophia, and John McNaught. 2006. *Text Mining for Biology and Biomedicine*. Citeseer.

Ananiadou, Sophia, Brian Rea, Naoaki Okazaki, Rob Procter, and James Thomas. 2009. 'Supporting Systematic Reviews Using Text Mining'. *Social Science Computer Review* 27 (4): 509–23.

Aviv-Reuven, Shir, and Ariel Rosenfeld. 2021. 'Publication Patterns' Changes Due to the COVID-19 Pandemic: A Longitudinal and Short-Term Scientometric Analysis'. *Scientometrics*, 1–24.

Babar, Muhammad Ali, and He Zhang. 2009. 'Systematic Literature Reviews in Software Engineering: Preliminary Results from Interviews with Researchers'. In *2009 3rd International Symposium on Empirical Software Engineering and Measurement*, 346–55. IEEE.

Baeza-Yates, Ricardo, Berthier Ribeiro-Neto, et al. 1999. *Modern Information Retrieval*. Vol. 463. ACM press New York.

Bajpai, Akhilesh, Sravanthi Davuluri, Haritha Haridas, Greta Kasliwal, H Deepti, KS Sreelakshmi, Darshan Chandrashekar, et al. 2011. 'In Search of the Right Literature Search Engine (s)'. *Nature Precedings*, 1–1.

Bannach-Brown, Alexandra, Piotr Przybyła, James Thomas, Andrew SC Rice, Sophia Ananiadou, Jing Liao, and Malcolm Robert Macleod. 2019. 'Machine Learning Algorithms for Systematic Review: Reducing Workload in a Preclinical Review of Animal Studies and Reducing Human Screening Error'. *Systematic Reviews* 8 (1): 1–12.

Bastian, Hilda, Paul Glasziou, and Iain Chalmers. 2010. 'Seventy-Five Trials and Eleven Systematic Reviews a Day: How Will We Ever Keep Up?'. *PLoS Medicine* 7 (9): e1000326.

Batista, Gustavo EAPA, Ronaldo C Prati, and Maria Carolina Monard. 2004. 'A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data'. *ACM SIGKDD Explorations Newsletter* 6 (1): 20–29.

Bekhuis, Tanja, and Dina Demner-Fushman. 2010. 'Towards Automating the Initial Screening Phase of a Systematic Review'. *MEDINFO 2010*, 146–50.

———. 2012. 'Screening Nonrandomized Studies for Medical Systematic Reviews: A Comparative Study of Classifiers'. *Artificial Intelligence in Medicine* 55 (3): 197–207.

Beller, Elaine M, Joyce Kee-Hsin Chen, Una Li-Hsiang Wang, and Paul P Glasziou. 2013. 'Are Systematic Reviews up-to-Date at the Time of Publication?'. *Systematic Reviews* 2 (1): 1–6.

Beller, Elaine, Justin Clark, Guy Tsafnat, Clive Adams, Heinz Diehl, Hans Lund, Mourad Ouzzani, et al. 2018. 'Making Progress with the Automation of Systematic Reviews: Principles of the International Collaboration for the Automation of Systematic Reviews (ICASR)'. *Systematic Reviews* 7 (1): 1–7.

Bhatt, Umang, McKane Andrus, Adrian Weller, and Alice Xiang. 2020. 'Machine Learning Explainability for External Stakeholders'. *arXiv Preprint arXiv:2007.05408*.

Bibal, Adrien, Michael Lognoul, Alexandre De Streel, and Benoît Frénay. 2021. 'Legal Requirements on Explainability in Machine Learning'. *Artificial Intelligence and Law* 29 (2): 149–69.

Bibal, Adrien, Michael Lognoul, Alexandre de Streel, and Benoît Frénay. 2020. 'Impact of Legal Requirements on Explainability in Machine Learning'. *arXiv Preprint arXiv:2007.05479*.

Blanco-Justicia, Alberto, and Josep Domingo-Ferrer. 2019. 'Machine Learning Explainability Through Comprehensible Decision Trees'. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, 15–26. Springer.

Bollegala, Danushka, Takanori Maehara, and Ken-ichi Kawarabayashi. 2015. 'Embedding Semantic Relations into Word Representations'. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Borah, Rohit, Andrew W Brown, Patrice L Capers, and Kathryn A Kaiser. 2017. 'Analysis of the Time and Workers Needed to Conduct Systematic Reviews of Medical Interventions Using Data from the PROSPERO Registry'. *BMJ Open* 7 (2): e012545.

Bornmann, Lutz, and Rüdiger Mutz. 2015. 'Growth Rates of Modern Science: A Bibliometric Analysis Based on the Number of Publications and Cited References'. *Journal of the Association for Information Science and Technology* 66 (11): 2215–22.

Breiman, Leo. 1996. 'Bagging Predictors'. *Machine Learning* 24 (2): 123–40.

Burkart, Nadia, and Marco F Huber. 2021. 'A Survey on the Explainability of Supervised Machine Learning'. *Journal of Artificial Intelligence Research* 70: 245–317.

Chae, Kyung-Chul. 1993. 'Presenting the Negative Hypergeometric Distribution to the Introductory Statistics Courses'. *International Journal of Mathematical Education in Science and Technology* 24 (4): 523–26.

Chawla, Nitesh V, Nathalie Japkowicz, and Aleksander Kotcz. 2004. 'Special Issue on Learning from Imbalanced Data Sets'. *ACM SIGKDD Explorations Newsletter* 6 (1): 1–6.

Chen, Chao, Xiao Lin, and Gabriel Terejanu. 2018. 'An Approximate Bayesian Long Short-Term Memory Algorithm for Outlier Detection'. In *2018 24th International Conference on Pattern Recognition (ICPR)*, 201–6. IEEE.

Chen, Qiuxing, Lixiu Yao, and Jie Yang. 2016. 'Short Text Classification Based on LDA Topic Model'. In *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, 749–53. IEEE.

Chen, Sih-Huei, Yuan-Shan Lee, Tzu-Chiang Tai, and Jia-Ching Wang. 2015. 'Gaussian Process Based Text Categorization for Healthy Information'. In *2015 International Conference on Orange Technologies (ICOT)*, 30–33. https://doi.org/10.1109/ICOT.2015.7498487.

Cheng, Y, Z Ye, M Wang, and Q Zhang. 2019. 'Document Classification Based on Convolutional Neural Network and Hierarchical Attention Network'. *Neural Network World* 29 (2): 83–98.

Chipman, Hugh A, Edward I George, Robert E McCulloch, et al. 2010. 'BART: Bayesian Additive Regression Trees'. *The Annals of Applied Statistics* 4 (1): 266–98.

Claesen, Marc, and Bart De Moor. 2015. 'Hyperparameter Search in Machine Learning'. *arXiv Preprint arXiv:1502.02127*.

Cohen, Aaron M, Clive E Adams, John M Davis, Clement Yu, Philip S Yu, Weiyi Meng, Lorna Duggan, Marian McDonagh, and Neil R Smalheiser. 2010. 'Evidence-Based Medicine, the Essential Role of Systematic Reviews, and the Need for Automated Text Mining Tools'. In *Proceedings of the 1st ACM International Health Informatics Symposium*, 376–80.

Cohen, Aaron M, William R Hersh, Kim Peterson, and Po-Yin Yen. 2006. 'Reducing Workload in Systematic Review Preparation Using Automated Citation Classification'. *Journal of the American Medical Informatics Association* 13 (2): 206–19.

Cohen, K Bretonnel, and Lawrence Hunter. 2008. 'Getting Started in Text Mining'. *PLoS Computational Biology* 4 (1): e20.

Dietterich, Thomas G. 2000. 'Ensemble Methods in Machine Learning'. In *International Workshop on Multiple Classifier Systems*, 1–15. Springer.

Díez-Pastor, José F, Juan J Rodríguez, César I García-Osorio, and Ludmila I Kuncheva. 2015. 'Diversity Techniques Improve the Performance of the Best Imbalance Learning Ensembles'. *Information Sciences* 325: 98–117.

Edwards, Phil, Mike Clarke, Carolyn DiGuiseppi, Sarah Pratap, Ian Roberts, and Reinhard Wentz. 2002. 'Identification of Randomized Controlled Trials in Systematic Reviews: Accuracy and Reliability of Screening Records'. *Statistics in Medicine* 21 (11): 1635–40.

Eppstein, David, Maarten Löffler, and Darren Strash. 2010. 'Listing All Maximal Cliques in Sparse Graphs in Near-Optimal Time'. In *International Symposium on Algorithms and Computation*, 403–14. Springer.

Farkas, Jennifer. 1995. 'Document Classification and Recurrent Neural Networks'. In *Proceedings of the 1995 Conference of the Centre for Advanced Studies on Collaborative Research*, 21.

Frunza, Oana, Diana Inkpen, and Stan Matwin. 2010. 'Building Systematic Reviews Using Automatic Text Classification Techniques'. In *Coling 2010: Posters*, 303–11.

Galar, Mikel, Alberto Fernandez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. 2011. 'A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches'. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42 (4): 463–84.

Gelman, Andrew, Ben Goodrich, Jonah Gabry, and Aki Vehtari. 2019. 'R-Squared for Bayesian Regression Models'. *The American Statistician*.

Hammerstrøm, Karianne, Anne Wade, Anne-Marie Klint Jørgensen, and Karianne Hammerstrøm. 2010. 'Searching for Studies'. *Education* 54 (11.3).

Hansen, Lars Kai et al. 2000. 'Bayesian Averaging Is Well-Temperated'. In *Proceedings of NIPS*, 99:265–71.

Harremoës, Peter, and Flemming Topsøe. 2001. 'Maximum Entropy Fundamentals'. *Entropy* 3 (3): 191–226.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. 'Boosting and Additive Trees'. In *The Elements of Statistical Learning*, 337–87. Springer.

Higgins, Julian PT, James Thomas, Jacqueline Chandler, Miranda Cumpston, Tianjing Li, Matthew J Page, and Vivian A Welch. 2019. *Cochrane Handbook for Systematic Reviews of Interventions*. John Wiley & Sons.

Horbach, Serge PJM. 2020. 'Pandemic Publishing: Medical Journals Strongly Speed up Their Publication Process for COVID-19'. *Quantitative Science Studies* 1 (3): 1056–67.

Hoy, Matthew B. 2020. 'Rise of the Rxivs: How Preprint Servers Are Changing the Publishing Process'. *Medical Reference Services Quarterly* 39 (1): 84–89.

Ikonomakis, M, Sotiris Kotsiantis, and V Tampakas. 2005. 'Text Classification Using Machine Learning Techniques'. *WSEAS Transactions on Computers* 4 (8): 966–74.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning*. Vol. 112. Springer.

Jayashree, P, and PK Srijith. 2020. 'Evaluation of Deep Gaussian Processes for Text Classification'. In *Proceedings of the 12th Language Resources and Evaluation Conference*, 1485–91.

Jonnalagadda, Siddhartha R, Pawan Goyal, and Mark D Huffman. 2015. 'Automating Data Extraction in Systematic Reviews: A Systematic Review'. *Systematic Reviews* 4 (1): 1–16.

Joo, Taejong, Uijung Chung, and Min-Gwan Seo. 2020. 'Being Bayesian about Categorical Probability'. In *International Conference on Machine Learning*, 4950–61. PMLR.

Jospin, Laurent Valentin, Wray Buntine, Farid Boussaid, Hamid Laga, and Mohammed Bennamoun. 2020. 'Hands-on Bayesian Neural Networks–a Tutorial for Deep Learning Users'. *arXiv Preprint arXiv:2007.06823*.

Kapelner, Adam, and Justin Bleich. 2013. 'bartMachine: Machine Learning with Bayesian Additive Regression Trees'. *arXiv Preprint arXiv:1312.2171*.

———. 2015. 'Prediction with Missing Data via Bayesian Additive Regression Trees'. *Canadian Journal of Statistics* 43 (2): 224–39.

Khoshgoftaar, Taghi M, Jason Van Hulse, and Amri Napolitano. 2010. 'Comparing Boosting and Bagging Techniques with Noisy and Imbalanced Data'. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 41 (3): 552–68.

Kohavi, Ron et al. 1995. 'A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection'. In *Ijcai*, 14:1137–45. 2. Montreal, Canada.

Lai, Siwei, Liheng Xu, Kang Liu, and Jun Zhao. 2015. 'Recurrent Convolutional Neural Networks for Text Classification'. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Landauer, Thomas K, Peter W Foltz, and Darrell Laham. 1998. 'An Introduction to Latent Semantic Analysis'. *Discourse Processes* 25 (2-3): 259–84.

Larsen, Peder, and Markus Von Ins. 2010. 'The Rate of Growth in Scientific Publication and the Decline in Coverage Provided by Science Citation Index'. *Scientometrics* 84 (3): 575–603.

Laws, Florian, and Hinrich Schütze. 2008. 'Stopping Criteria for Active Learning of Named Entity Recognition'. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, 465–72.

Lefebvre, C, E Manheimer, J Glanville, J Higgins, and S Green. 2011. 'Searching for Studies (Chapter 6)'. *Cochrane Handbook for Systematic Reviews of Interventions Version* 510.

Li, Qian, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S Yu, and Lifang He. 2020. 'A Survey on Text Classification: From Shallow to Deep Learning'. *arXiv Preprint arXiv:2008.00364*.

Lipscomb, Carolyn E. 2000. 'Medical Subject Headings (MeSH)'. *Bulletin of the Medical Library Association* 88 (3): 265.

Malhi, Avleen, Samanta Knapic, and Kary Främling. 2020. 'Explainable Agents for Less Bias in Human-Agent Decision Making'. In *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, 129–46. Springer.

Marshall, Iain J, and Byron C Wallace. 2019. 'Toward Systematic Review Automation: A Practical Guide to Using Machine Learning Tools in Research Synthesis'. In *Systematic Reviews*, 8:1–10. 1. Springer.

Minaee, Shervin, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. 'Deep Learning–Based Text Classification: A Comprehensive Review'. *ACM Computing Surveys (CSUR)* 54 (3): 1–40.

Miwa, Makoto, James Thomas, Alison O'Mara-Eves, and Sophia Ananiadou. 2014. 'Reducing Systematic Review Workload Through Certainty-Based Screening'. *Journal of Biomedical Informatics* 51: 242–53.

Moore, Alexander, Vanessa Murdock, Yaxiong Cai, and Kristine Jones. 2018. 'Transparent Tree Ensembles'. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 1241–44.

'NCBI Insights : Updated Pubmed e-Utilities Coming in April 2022!'. n.d. U.S. National Library of Medicine. `https://ncbiinsights.ncbi.nlm.nih.gov/2021/10/05/updated-pubmed-api/`.

O'Mara-Eves, Alison, James Thomas, John McNaught, Makoto Miwa, and Sophia Ananiadou. 2015. 'Using Text Mining for Study Identification in Systematic Reviews: A Systematic Review of Current Approaches'. *Systematic Reviews* 4 (1): 1–22.

Ohsawa, Yukio, Nels E Benson, and Masahiko Yachida. 1998. 'KeyGraph: Automatic Indexing by Co-Occurrence Graph Based on Building Construction Metaphor'. In *Proceedings IEEE International Forum on Research and Technology Advances in Digital Libraries-ADL'98-*, 12–18. IEEE.

Olorisade, Babatunde K, Ed de Quincey, Pearl Brereton, and Peter Andras. 2016. '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, 1–11.

Pavlinek, Miha, and Vili Podgorelec. 2017. 'Text Classification Method Based on Self-Training and LDA Topic Models'. *Expert Systems with Applications* 80: 83–93.

Poulter, Graham L, Daniel L Rubin, Russ B Altman, and Cathal Seoighe. 2008. 'MScanner: A Classifier for Retrieving Medline Citations'. *BMC Bioinformatics* 9 (1): 1–12.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Ramezankhani, Azra, Omid Pournik, Jamal Shahrabi, Fereidoun Azizi, Farzad Hadaegh, and Davood Khalili. 2016. 'The Impact of Oversampling with SMOTE on the Performance of 3 Classifiers in Prediction of Type 2 Diabetes'. *Medical Decision Making* 36 (1): 137–44.

Robert, Christian P, and George Casella. 2004. *Monte Carlo Statistical Methods.* Vol. 2. Springer.

Roshan, Seyed Ehsan, and Shahrokh Asadi. 2020. 'Improvement of Bagging Performance for Classification of Imbalanced Datasets Using Evolutionary Multi-Objective Optimization'. *Engineering Applications of Artificial Intelligence* 87: 103319.

Rousseau, Francois. 2015. 'Graph-of-Words: Mining and Retrieving Text with Networks of Features'. PhD thesis, Ph. D. dissertation.

Rousseau, François, Emmanouil Kiagias, and Michalis Vazirgiannis. 2015. 'Text Categorization as a Graph Classification Problem'. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1702–12.

Sadaghiani, Catharina, Tjibbe Donker, Xanthi Andrianou, Balázs Babarczy, Gerolf De Boer, Francesco Di Ruscio, Shona Cairns, et al. 2020. *National Health Care Infrastructures, Health Care Utilization and Patient Movements Between Hospitals - Networks Working to Improve Surveillance: A Systematic Literature Review.* PROSPERO. http://www.crd.york.ac.uk/PROSPERO/display_record.asp?ID=CRD42020157987.

Sampson, Margaret, Jennifer Tetzlaff, and Christine Urquhart. 2011. 'Precision of Healthcare Systematic Review Searches in a Cross-Sectional Sample'. *Research Synthesis Methods* 2 (2): 119–25.

Schonlau, Matthias, and Nick Guenther. 2017. 'Text Mining Using n-Grams'. *Schonlau, M., Guenther, N. Sucholutsky, I. Text Mining Using n-Gram Variables. The Stata Journal* 17 (4): 866–81.

Settles, Burr. 2009. 'Active Learning Literature Survey'.

Soria-Olivas, Emilio, Juan Gomez-Sanchis, José D Martin, Joan Vila-Frances, Marcelino Martinez, José R Magdalena, and Antonio J Serrano. 2011. 'BELM: Bayesian Extreme Learning Machine'. *IEEE Transactions on Neural Networks* 22 (3): 505–9.

Soto, Axel J, Piotr Przybyła, and Sophia Ananiadou. 2019. 'Thalia: Semantic Search Engine for Biomedical Abstracts'. *Bioinformatics* 35 (10): 1799–1801.

Tashman, Leonard J. 2000. 'Out-of-Sample Tests of Forecasting Accuracy: An Analysis and Review'. *International Journal of Forecasting* 16 (4): 437–50.

Therneau, Terry, and Beth Atkinson. 2019. *Rpart: Recursive Partitioning and Regression Trees.* https://CRAN.R-project.org/package=rpart.

Thomas, James, and Jeff Brunton. 2007. 'EPPI-Reviewer: Software for Research Synthesis'.

Tsafnat, Guy, Adam Dunn, Paul Glasziou, and Enrico Coiera. 2013. 'The Automation of Systematic Reviews'. British Medical Journal Publishing Group.

Tsafnat, Guy, Paul Glasziou, Miew Keen Choong, Adam Dunn, Filippo Galgani, and Enrico Coiera. 2014. 'Systematic Review Automation Technologies'. *Systematic Reviews* 3 (1): 1–15.

Turian, Joseph, Lev Ratinov, and Yoshua Bengio. 2010. 'Word Representations: A Simple and General Method for Semi-Supervised Learning'. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 384–94.

Violos, John, Konstantinos Tserpes, Evangelos Psomakelis, Konstantinos Psychas, and Theodora Varvarigou. 2016. 'Sentiment Analysis Using Word-Graphs'. In *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics*, 1–9.

Visser, Eelco. 2010. 'Performing Systematic Literature Reviews with Researchr: Tool Demonstration'. *Technical Report Series TUD-SERG-2010-010.*

Wallace, Byron C, Kevin Small, Carla E Brodley, and Thomas A Trikalinos. 2010. 'Active Learning for Biomedical Citation Screening'. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 173–82.

Wallace, Byron C, Thomas A Trikalinos, Joseph Lau, Carla Brodley, and Christopher H Schmid. 2010. 'Semi-Automated Screening of Biomedical Citations for Systematic Reviews'. *BMC Bioinformatics* 11 (1): 1–11.

Wilkins, Thad, Ralph A Gillies, and Kathy Davies. 2005. 'EMBASE Versus MEDLINE for Family Medicine Searches: Can MEDLINE Searches Find the Forest or a Tree?'. *Canadian Family Physician* 51 (6): 848–49.

Woods, David, and Kate Trewheellar. 1998. 'Medline and Embase Complement Each Other in Literature Searches'. *BMJ: British Medical Journal* 316 (7138): 1166.

Yang, JinXiong, Liang Bai, and Yanming Guo. 2020. 'A Survey of Text Classification Models'. In *Proceedings of the 2020 2nd International Conference on Robotics, Intelligent Control and Artificial Intelligence*, 327–34.

Yang, Li, and Abdallah Shami. 2020. 'On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice'. *Neurocomputing* 415: 295–316.

Zhou, Zhi-Hua. 2021. 'Ensemble Learning'. In *Machine Learning*, 181–210. Springer.

Zhu, Jingbo, Huizhen Wang, Eduard Hovy, and Matthew Ma. 2010. 'Confidence-Based Stopping Criteria for Active Learning for Data Annotation'. *ACM Transactions on Speech and Language Processing (TSLP)* 6 (3): 1–24.