OFORI-BOATENG, R., ACEVES-MARTINS, M., WIRATUNGA, N. and MORENO-GARCIA, C.F. [2024]. Towards automation of systematic reviews using natural language processing, machine learning, and deep learning: a comprehensive review. *Artificial intelligence review* [online], (accepted). To be made available from: https://doi.org/10.1007/s10462-024-10844-w

Towards automation of systematic reviews using natural language processing, machine learning, and deep learning: a comprehensive review.

OFORI-BOATENG, R., ACEVES-MARTINS, M., WIRATUNGA, N. and MORENO-GARCIA, C.F.

2024

Towards Automation of Systematic Reviews Using Natural Language Processing, Machine Learning, and Deep Learning: A Comprehensive Review

Regina Ofori-Boateng[1*], Magaly Aceves-Martins[2], Nirmalie Wiratunga[1], Carlos Francisco Moreno-Garcia[1*]

[1*]School of Computing, Robert Gordon University, Aberdeen, Scotland.
[2]The Rowett Institute, University of Aberdeen, Aberdeen, Scotland.

*Corresponding author(s). E-mail(s): r.ofori-boateng@rgu.ac.uk; c.moreno-garcia@rgu.ac.uk;

Abstract

Systematic reviews (SRs) constitute a critical foundation for evidence-based decision-making and policy formulation across various disciplines, particularly in healthcare and beyond. However, the inherently rigorous and structured nature of the SR process renders it laborious for human reviewers. Moreover, the exponential growth in daily published literature exacerbates the challenge, as SRs risk missing out on incorporating recent studies that could potentially influence research outcomes. This pressing need to streamline and enhance the efficiency of SRs has prompted significant interest in leveraging Artificial Intelligence (AI) techniques to automate various stages of the SR process. This review paper provides a comprehensive overview of the current AI methods employed for SR automation, a subject area that has not been exhaustively covered in previous literature. Through an extensive analysis of 52 related works and an original online survey, the primary AI techniques and their applications in automating key SR stages, such as search, screening, data extraction, and risk of bias assessment, are identified. The survey results offer practical insights into the current practices, experiences, opinions, and expectations of SR practitioners and researchers regarding future SR automation. Synthesis of the literature review and survey findings highlights gaps and challenges in the current landscape of SR automation using AI techniques. Based on these insights, potential future directions are discussed. This review aims to equip researchers and practitioners with a foundational understanding of the basic concepts, primary methodologies, and recent advancements in AI-driven SR automation while guiding computer scientists in exploring novel techniques to invigorate further and advance this field.

Keywords: Systematic review, Artificial intelligence, Natural language processing, Machine learning, Deep learning, Systematic review automation, Active learning

1 Introduction

Literature reviews constitutes an essential part of academic research, serving as a critical foundation across various fields. A literature review may be conducted for various reasons, such as providing a general overview of a particular research topic, identifying existing theories and methodologies gaps, equipping a researcher with adequate information for decision-making, or even substantiating why a research topic must be studied, among others (Snyder, 2019). Predominantly, there exist two main types of literature reviews: the *narrative or traditional review* and the *systematic review (SR)*, with the latter being considered the gold standard and more credible approach in numerous disciplines (Booth et al, 2016). SR, primarily used in healthcare research and other disciplines such as software engineering (SE) or humanities (Kitchenham et al, 2009; Davis et al, 2014), allows literature revision to be performed transparently, organised, and comprehensively. The systematic steps involved in an SR ensure an unbiased synthesis of relevant literature, thus providing robust evidence to support practitioners, policymakers, and academics (Egger and George Davey Smith, 2001). The general steps involved while conducting an SR include 1) Development of protocol, 2) identification of relevant databases and developing a search strategy, 3) screening of titles and abstracts obtained after searching, 4) full-text screening of relevant abstracts to scout those that meet the exclusion/inclusion criteria stated in the protocol, 5) Extracting relevant data of studies meeting the inclusion criteria, 6) critical appraisal/risk of bias (RoB) assessment to check the quality of the included studies, 7) synthesis and interpretation of results (Aromataris and Pearson, 2014).

SR, rather than a product, is a process. However, the SR process is inherently time-consuming and susceptible to human error due to its orderly and well-structured nature. Reviewers have the overwhelming task of planning, searching, screening titles and abstracts, reading the full texts, and synthesising data from many publications. Averagely, the typical timeframe reported for an SR to be completed and published is approximately 15 months (Borah et al, 2017). With the exponential growth in daily published literature (Bornmann and Mutz, 2015), most SRs fall behind, missing out on incorporating recent studies that could have influenced the research outcomes (Gates et al, 2018; van de Schoot et al, 2021). This highlights a pressing need for innovative solutions to streamline and enhance the efficiency of SRs. On the other hand, this rapid growth in the number of studies published daily, coupled with the demanding requirements of SR, has prompted significant interest in the deployment of Artificial Intelligence (AI). Specifically, three broad aspects of AI, Natural Language Processing (NLP), Machine Learning (ML), and Deep Learning (DL), have been explored for their potential to automate various stages of the SR process (Marshall and Wallace, 2019). However, it is unclear what specific methods are being implemented and what are the benefits of using AI methods during SR (Blaizot et al, 2022). To address these challenges, this review paper seeks to explores the application of AI in automating the SR process and to provide a comprehensive overview of the current AI techniques proposed. Thus, this paper aims to equip researchers with a foundational understanding of the basic concepts, primary methodologies, and advancements in SR AI automation.

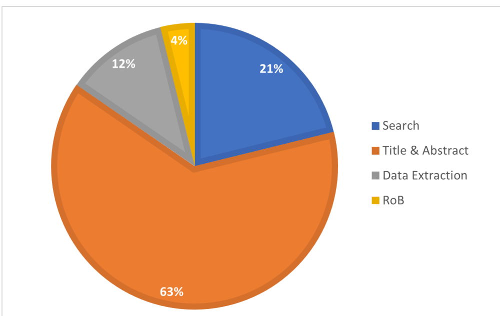
To the best of knowledge, there exists only one study by Jaspers et al (2018) that provides a detailed overview of the ML approach employed in SR. However, the study focuses on only one branch of AI and only partially covers the NLP and DL aspects of the AI used for SR automation. Additionally, the review focused on ML techniques used for only SRs within the domain of the Education and Skills Funding Agency (ESFA). Thus, this review seeks to bridge the gap by summarising the AI methods used to automate SR in fields such as the medical and software engineering (SE) domain.

1.1 Contributions of this study

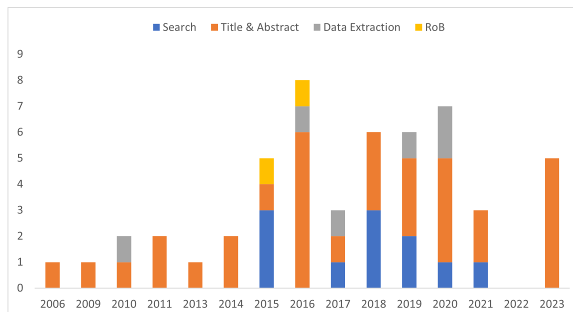
Overall, the main contributions and structure of this survey paper are summarised as follows: 1) to provide a comprehensive overview of the current AI methods used in SR automation, a subject area that has not been exhaustively covered in previous literature, 2) presenting empirical results from an original online survey which provides practical insights into the current practices, experiences, opinions and expectations of SR practitioners and researchers for future SR automation, 3) combining the results of the original survey as well as the comprehensive overview to provide recommendations for future AI SR automation. Overall, this paper is organised as follows: Section 2 discusses the fundamentals of AI actively used for SR automation. Section 3 presents an overview of how these methods described in Section 2 are deployed in the studies found for the four most reported stages (search, screening, data extraction, and RoB) of the SR process. Section 4 presents the online AI survey on SR automation. Section 5, summarises the public datasets and codes available for automating these four stages and provided an assessment summary for the most common evaluation metric in Section 3, used on similar public datasets. Section 6 discusses potential limitations, challenges, and future directions for SR automation .

1.2 Search criteria and eligibility criteria

To identify relevant studies, 31 papers were retrieved from current systematic reviews on SR automation by van Dinter et al (2021) and Blaizot et al (2022). These SRs focused on finding studies that targeted automating any of the SR's stages but did not describe the AI methods deployed in these studies. Additionally, databases such as PubMed, Scopus, Google Scholar, IEEE, Elsevier, Springer, ACM, and ScienceDirect were queried using relevant Boolean strings keywords (e.g., "systematic review" AND ("machine learning", "text mining/classification" OR "deep learning" OR "natural language processing" OR "automation" OR "active learning"). To gather other relevant papers, the concept of snowballing was used. Papers that did not principally focus on SR automation and explain the AI methodology used were excluded. The last update for the included articles was in 2024. From the search database, 21 new papers were added to the 31 previously recruited papers, resulting in 52 papers. Among these, 11 papers targeted the automation of the search phase, 33 addressed the screening phase, six focused on data extraction automation, and two on the automation of the RoB. These papers are generally summarised in Figure 1a and Figure 1b. Despite the recent prominence of large language models (LLMs) such as ChatGPT[1], papers utilising ChatGPT were excluded from this analysis due to the selection criteria emphasising papers with a detailed explanation of the AI methods used. However, it is noted in Figure 1b that other LLMs have been employed in some of the identified papers included in this review.



(a) Number of papers for each stage that met the eligibility criteria



(b) Number of papers with respect to the years for each stage

Fig. 1: Analysis of paper criteria and year distribution

2 Fundamentals of AI used in SR automation

The application of AI in the automation of SRs has increased significantly in recent years. As detailed in Section 1, NLP, ML, and DL constitute the core AI techniques employed to accelerate the SR process. The 52 papers found for the four stages of the SR (search, title/abstract screening, data extraction and RoB) highlight NLP as the predominant technique used in SR automation. Thus, this section elucidates the foundational NLP techniques commonly utilised in this context. To describe the interlinkage of ML and DL with the NLP concept, Sections 2.5 and 2.6 expatiate this basis. NLP involves statistical and graphical methods that facilitate systems' understanding of human language. Among the primary NLP tasks that underpin SR automation, *text classification* is the most predominant (Marshall and Wallace, 2019). This task involves categorising text segments based on their content, such as during the title/abstract screening phase of the SR process, where abstracts and titles are classified as relevant or irrelevant. Another example of where this task is deployed is categorising the methods design of included studies as having a high/low bias, thus facilitating the RoB assessment . Additionally, text classification supports the search phase by filtering and categorising documents pertinent to specific research questions, thereby alleviating the screening burden, for example, by identifying randomised control trials (RCT) from databases.

[1] https://chat.openai.com/

Information retrieval (IR) represents another essential NLP task, particularly vital in health research for literature searches (Nadkarni, 2002). During the search phase, a prominent IR technique discussed in related literature discussed in Section 3 query expansion (QE), which extends search strings to include related terms, further improving original queries and resulting in richer and more relevant results (Aklouche et al, 2019). *Information extraction* is another vital SR automation task, primarily used during the data extraction phase. This process involves extracting specific information. In the medical domain, these include elements of the PICO framework (Population, Intervention, Comparator, and Outcome), sample size, setting details, and research questions from included studies. One of the earliest techniques proposed for automating the data extraction stage is template filling, where data is extracted based on sample templates such as CONSORT (Moher, 2001). Furthermore, this task aids in extracting supporting statements for study design evaluations, thereby automating the RoB assessment. Additionally, some related works to be discussed employed these tasks to automate the search stage. That is, extracting information from seed studies to develop query strings. Lastly, another aspect of NLP used for SR automation is *Visual Text Mining (VTM)*. VTM combines text mining techniques such as IE and IR with visuals. In SR, VTM is mainly used to automate the search stage and, sometimes, for screening/selecting primary studies (Felizardo et al, 2012).

In summary, the integration of NLP techniques in SR automation follows a sequence of processes known as the NLP pipeline, as illustrated in Figure 2. The subsequent subsections will discuss the stages of the NLP pipeline (Figure 2) and their application in the automation of SR processes across the 52 identified studies.

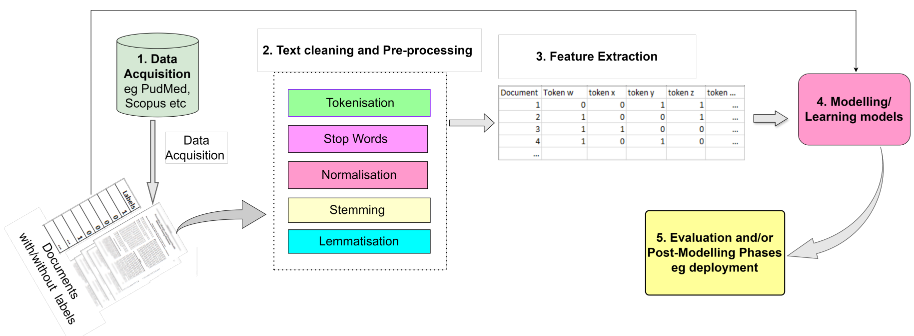


Fig. 2: The NLP Pipeline for Systematic Review Automation (Training Phase)

2.1 Data Acquisition

To train the learning models for SR automation, a crucial initial step, as depicted in Figure 2, involves acquiring data from pertinent sources and databases. Among the 52 related studies, PubMed[2] abstracts and Medline[3] full-text data are most frequent source utilised to train models across the four identified stages of SR reviewed in this study, especially for title and abstract screening. Additional data sources include the CLEF eHealth Technology Assisted Reviews (TAR)[4] and the TREC Precision Medicine dataset[5], which offer queries, abstracts, and relevance scores to enhance the automation of the search stage. For the RoB and data extraction, text summaries from the Cochrane Database of Systematic Reviews (CDSR)[6] is the source employed in related studies to train and validate the AI model.

[2]https://pubmed.ncbi.nlm.nih.gov/
[3]https://www.nlm.nih.gov/medline/medline_overview.html
[4]https://clefehealth.imag.fr/
[5]https://trec.nist.gov/data/clinical.html
[6]https://www.cochranelibrary.com/cdsr/about-cdsr

2.2 Text Cleaning and Pre-processing

The principal aim of this stage in the pipeline is to remove noise from the text data, ensuring that clean data is fed into subsequent stages. This section highlights some of the most frequent approaches identified in related studies for SR automation, including sentence and word tokenisation, stop word removal, stemming and lemmatisation, normalisation, and Part-of-speech (POS) Tagging. In RCT SRs, stemming and/or lemmatisation are not always applied to tokens, as they can lead to the loss of critical information in the text. For instance, during stemming, the term “trials” in an RCT SR report might be reduced to “trial,” potentially altering the meaning and implying it is part of a single RCT report rather than an SR of multiple RCTs (Bannach-Brown et al, 2019). To demonstrate how these pre-processing techniques work significantly, and to help our non-technical readers, a sample SR abstract on juvenile obesity by Aceves-Martins et al (2021) is used to describe these in Figure 3 visually.

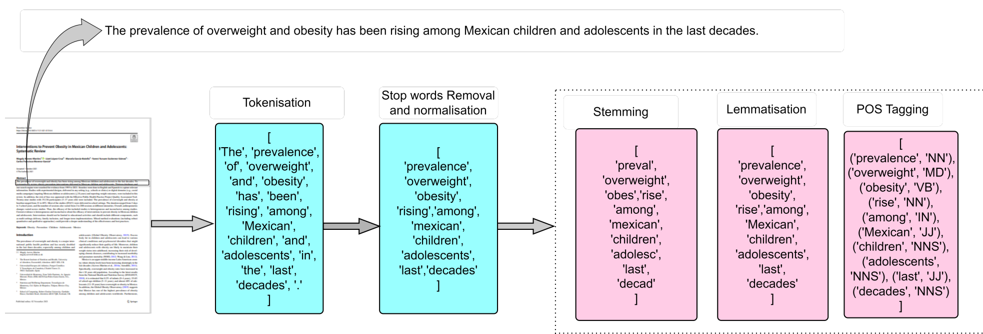


Fig. 3: Demonstration of how some pre-processing techniques are deployed for SR automation using a sample abstract by Aceves-Martins et al (2021)

2.3 Feature Extraction

Figure 4 summarises the various feature extraction methods used in the related studies for automating the four stages: search, screening, data extraction and RoB. This section aims to provide deeper insights into these methods' comparative strengths and limitations. Under traditional feature extraction techniques, examples of these methods used include BoW, Bag of N-gram as 2-gram (bi-gram), 3-gram (trigram) and TF-IDF are extensively utilised due to their simplicity and effectiveness in handling large datasets(Walkowiak et al, 2018). BoW, being used in the screening processes as shown in Figure 4, is advantageous for its ease of implementation but is limited by its inability to capture semantic meanings between words. In contrast, N-gram models, which also appear frequently in the screening phase, offer a balance by capturing some context within the data, though at a computational cost that scales with the size of the n-gram. TF-IDF, on the other hand, stands out in Figure 4, demonstrating its robustness in distinguishing relevant terms in large text corpora by emphasising unique terms in documents. This method is computationally efficient and often serves as a baseline for feature relevance assessment in text mining applications (Walkowiak et al, 2018). Advanced embedding techniques like Word2Vec and GloVe, noted less frequently in the screening stages, offer rich semantic representations of text but require more computational resources. Even though these models capture deeper linguistic contexts, making them suitable for applications needing nuanced text interpretation, they could be more practical for large datasets or limited-resource settings. Transformer-based methods, such as BERT and s-BERT, represent the cutting edge in feature extraction. Their lower frequency of use as feature extractors, as indicated in Figure 4, may be due to their computational demands or because the model is directly used for fine-tuning the SR tasks. However, their ability to understand context and nuance in text is unparalleled. Thus, the choice of feature extraction method significantly impacts the computational efficiency and effectiveness of SR automation. While traditional methods like BoW and TF-IDF are computationally less demanding and thus more prevalent in larger datasets, advanced methods like BERT provide superior contextual understanding, suggesting a trade-off between performance and computational overhead.

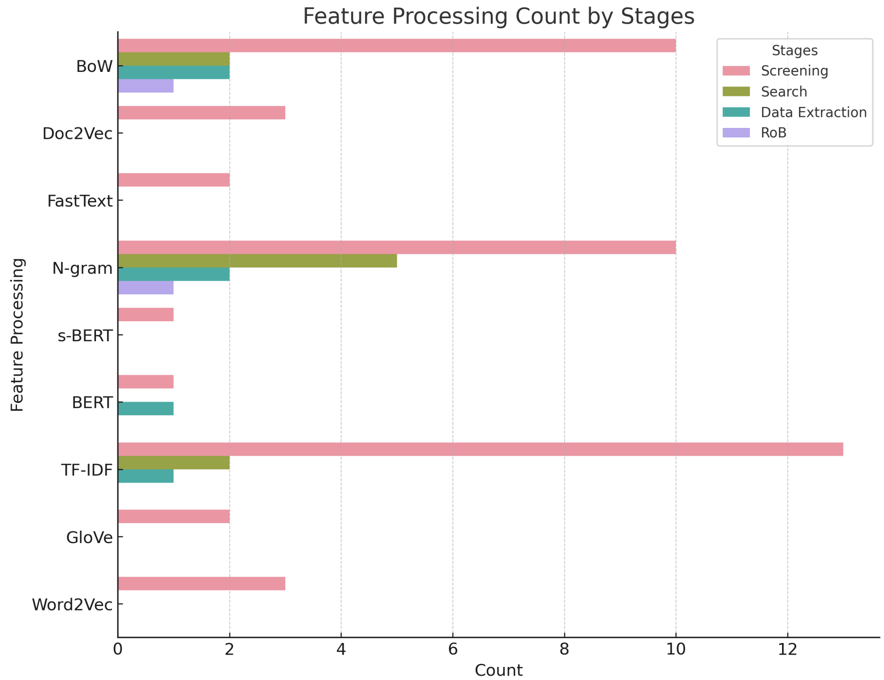


Fig. 4: Summary of proposed feature extraction techniques in identified papers obtained

2.4 Modelling/Learning models

Continuing with the NLP pipeline depicted in Figure 2, the subsequent stage following text vectorisation is typically modelling. The three main AI learning models identified in the related works for SR automation include the rule-based approach, ML and DL, a subclass of ML (Song et al, 2020). The rule-based approach involves explicit, well-defined guidelines comprising logical statements that dictate actions under specific conditions. Standard techniques observed in the related works include word lists, string matching, and regular expressions (AHO, 1990). Specifically in SRs, rule-based methods, particularly regular expressions, are primarily used in the data extraction phase to identify and extract data from included studies (Marshall et al, 2016, 2017). Although rule-based methods are effective and provide a straightforward foundation for developing NLP models, a significant drawback is their static nature; they do not adapt or learn over time, often necessitating the development of new rules as the system evolves. In contrast, ML and DL models overcome these limitations by utilising adaptive learning and pattern recognition capabilities (Song et al, 2020). Nonetheless, rule-based approaches can also complement ML and DL models, for example, by extracting information as input for these models or by removing special characters from text during the preprocessing stage. Given the prominence of ML and DL in the studies reviewed, these models will be discussed in detail as focal points in this subsection. Training of these learning models is primarily categorised into three approaches: 1) supervised, where all training documents are manually annotated, such as classifying text as either relevant or irrelevant, or assessing whether a study is an RCT or if the methodology of an included study has high or minimal bias. The advantage of supervised learning in SR automation is its accuracy and predictability in performance. However, it requires a substantial amount of labelled data to train the learning model, which can be costly; 2)unsupervised, where no labels are used to discover hidden patterns and 3) semi-supervised, where a small proportion of training documents are labelled compared to the unlabelled ones, helping to mitigate the label scarcity problem by leveraging unlabelled data. In SR automation, semi-supervised learning is encapsulated in the concept of *active learning*, described in Section 2.5.3. The discussed papers in Section 3 showcase numerous applications of these training methods across different stages of

SR automation. Figure 5 illustrates that supervised training is predominantly used in the search phase, while semi-supervised training is prevalent in the screening, data extraction, and RoB stages.

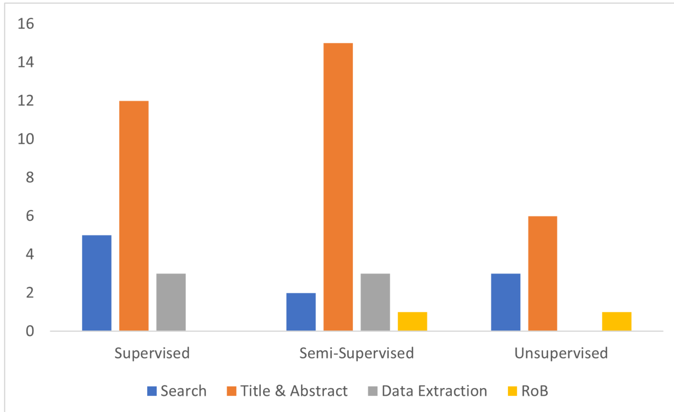


Fig. 5: Summary of techniques used in training NLP model to automate some stages in the SR process from 51 out of the identified papers that explicitly stated the training type used

2.5 Machine Learning (ML)

ML is a branch of AI that allows models to learn directly from given data and experiences, e.g. instructions and observations(Mitchell, 1997). This learning process is facilitated through four primary techniques: supervised, unsupervised, semi-supervised, and reinforcement learning (Jha et al, 2021), each defining a unique training approach. Interestingly, from the 52 related works found, only one study focused on reinforcement learning; this will be discussed in Section 3. In short, reinforcement learning comprises algorithm learning, which is achieved by being given an observation of a particular activity rather than a label itself. . The ultimate purpose is for the algorithm to use the information from the environment to raise awareness and minimise the danger or maximise the acquisition (Kaelbling et al, 1996; Gosavi, 2009). Figure 6 summarises the best-proposed ML algorithms in the 52 related works across the SR stages, elucidating which models excel in each stage. The following subsection provides a brief overview of these models deployed for SR automation, focusing on their suitability for the different stages.

2.5.1 Supervised Machine Learning Algorithms

This subsection discusses the underpinning of the popular supervised learning classification algorithms deployed in SR automation, as summarised from the identified papers in Figure 6. Supervised algorithms are extensively utilised across all stages of SR automation due to their ability to learn from labelled data. For a detailed explanation of these techniques, readers are referred to the study by (Sarker, 2021).

- **Support Vector Machine (SVM)**: is extensively utilised across various stages of the SR, as illustrated in Figure 6. This algorithm identifies an optimal hyperplane that segregates input data points by their class (e.g. relevant or irrelevant as in the case of automating the screening stage or classifying the input as having a high-risk or low-risk bias) within an N-dimensional space (Cortes and Vapnik, 1995) by employing a range of mathematical functions known as kernels. These kernels include linear, sigmoid, Gaussian, polynomial, nonlinear, and radial basis functions (Mahendra and Azizah, 2023). The linear SVM is predominantly used in LR automation (Joachims, 2006). Additional variations of SVM, such as the soft-margin polynomial and Evolutionary SVM (EvoSVM), have been proposed in other studies to enhance performance (Timsina et al, 2015).
- **Logistic Regression (LR)**: remarkably proposed for automating the title/abstract screening stage, as illustrated in Figure 6., is a probabilistic statistical model that uses a sigmoid function, the algorithm's core, to make predictions (Cessie and Houwelingen, 1992). Automatically, it performs binary

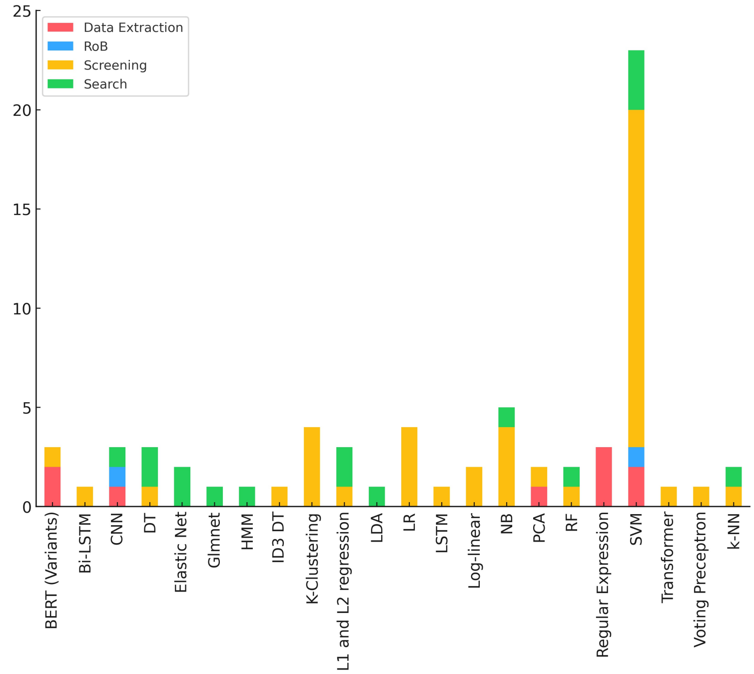


Fig. 6: Summary of the common algorithms used in SR automation from related works per each stage; SVM=Support Vector Machine, KNN=K Nearest Neighbours, LDA= Latent Dirichlet Allocation, RF = Random Forest, PCA= Principal Component Analysis, LR= Logistic Regression, DT= Decision Tree, CNN= Convolutional Neural Network, LSTM=Long Short Term Memory, NB= Naïve Bayes, HMM=Hidden Markov Model

classification and is thus appropriate for text classification tasks, hence explains why it is proposed for SR screening automation; relevant or irrelevant. However, recent advances have been made to support multi-class classification(Abramovich et al, 2021). Readers are referred to the work done Iparragirre et al (2023) for a detailed explanation of the LR model.

- **Naive Bayes (NB)**: notably proposed for automating both the screening stage and the search stage of the SR process is a probabilistic classifier uses the Bayes theorem seen in Equation 2.2. Various variants of NB classifiers exist, including Gaussian, Bernoulli, Multinomial, Complement, and Categorical (Baranwal et al, 2022). Specifically, the Complement NB (cNB) is the type of NB employed in SR automation to address class imbalance, a significant challenge in training datasets (O'Mara-Eves et al, 2015)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad \text{where } P(B) \neq 0 \tag{2.2}$$

- **K Nearest Neighbours (KNN)**: though less common in SR automation, has been proposed for automating both the screening and the search stage. It makes predictions based on the similarity between the input data and the desired outcome (Guo et al, 2003).
- **Decision Tree (DT) and Random Forest (RF)**: DT is an algorithm that learns from a training dataset by emulating the structure of a tree based on conditions and rules (Kotsiantis, 2011). A variant of DT deployed in SR is Iterative Dichotomiser 3 (ID3), shown as in Figure 6 used to automate the screening phase of the SR. Though DT is easy to understand, one main challenge is that it is prone to over-fitting and may be unstable to noisy datasets (Kotsiantis, 2011). RF is an advancement and ensemble method of the decision tree algorithm that solves the over-fitting issue (Popuri, 2022). In SR automation, RF is proposed for automating the search and screening stage. Readers are referred to the work by Popuri (2022) for a detailed explanation of how these models work.

- **Latent Dirichlet Allocation (LDA)**: is a dimensionality reduction supervised learning approach which is used to reduce the number of input features present in the training dataset proposed by (Blei et al, 2003). As illustrated in Figure 6, LDA has been proposed for automating the search stage in the SR process. This is because LDA supports thematic understanding that enables latent topic discovery Jelodar et al (2018). As a result, it aids in refining search queries and enhances the relevance of documents. An application of LDA used in expediting SRs is topic modelling described in Section 3 of this paper.

2.5.2 Unsupervised Machine Learning Algorithms

Here, the most commonly used unsupervised learning techniques in automating SRs are summarised as identified in related works. The primary categories of these algorithms include clustering and dimensionality reduction. A summary of the popular unsupervised algorithms follows:

- **K-Means Clustering**: is one of the most utilised unsupervised models for automating SR, particularly the screening stage (Figure 6). This method partitions observations into distinct clusters based on similar behaviours or patterns. As a result, K-means clustering supports organising large sets of SR datasets, e.g. abstracts, into clusters based on similarities in their text content. This grouping helps identify patterns or themes common to certain clusters, which can indicate relevance to the research questions or criteria of the SR. While K-Means is computationally efficient, determining the optimal number of clusters remains challenging Ahmed et al (2020).
- **Principal Component Analysis (PCA)**: is a dimensionality reduction technique that simplifies the complexity of high-dimensional data while retaining trends and patterns. It reduces the dataset dimensions by transforming the original variables into a new set of variables, which are linear combinations of the original variables, known as principal components. The technique is proper for exploratory data analysis and feature extraction as such, PCA is proposed for automating the search and the screening stage in the SR process(Paul et al, 2013; Jolliffe, 2014).

2.5.3 Semi-Supervised Machine Learning Algorithms and Active Learning (AL)

Supervised and unsupervised machine learning techniques typically require a significant amount of data randomly sampled from the underlying population distribution, representing a passive approach to learning (Thrun, 1995). The challenge lies with the cost (time, resource) involved in getting this large amount of data, especially labelled data, for supervised ML models, which is the core of SR automation. In automating SRs, researchers must manually label a substantial dataset for model training, further burdening the SR process. This challenge has spurred the adoption of Active Learning (AL), a semi-supervised technique that involves initially labelling only a small subset of data to make predictions on unseen data. This technique allows humans or oracles within the cycle, thus known as *humans in the loop*. Unlike passive learning, where the model learns from a random sample, AL allows it to select the most beneficial data points for faster learning. These selected data points are then presented to a human or oracle for labelling, constituting a more targeted and informative sampling approach than random sampling (August, 2001). This process of selection is referred to as a query. The primary goal of AL is to minimise the volume of labelled data required to train a model effectively. In contrast to passive learning, which solely relies on the input data provided, AL actively seeks new information or data to enhance the model's predictive capabilities.

Fig 7 illustrates the active learning cycle used in SR automation. There are three principal settings through which the model, referred to as the learner, queries the human or oracle for additional data or information: 1) membership query strategy, the earliest form of this approach (Angluin, 1988), 2) stream-based selective sampling (Cohn et al, 1994), and 3) pool-based sampling (Lewis, 1998), which has proven particularly effective in text classification (Hoi et al, 2006) and is the most frequently employed method in SR automation. Pool-based sampling operates under the assumption that a large reservoir of unlabelled data is available, from which queries are made using an informative measure known as a query strategy.

The query strategy enables the learner to select the most informative sample or instance from the unlabelled data or choose which instance to learn from. One example used in computerising SR is

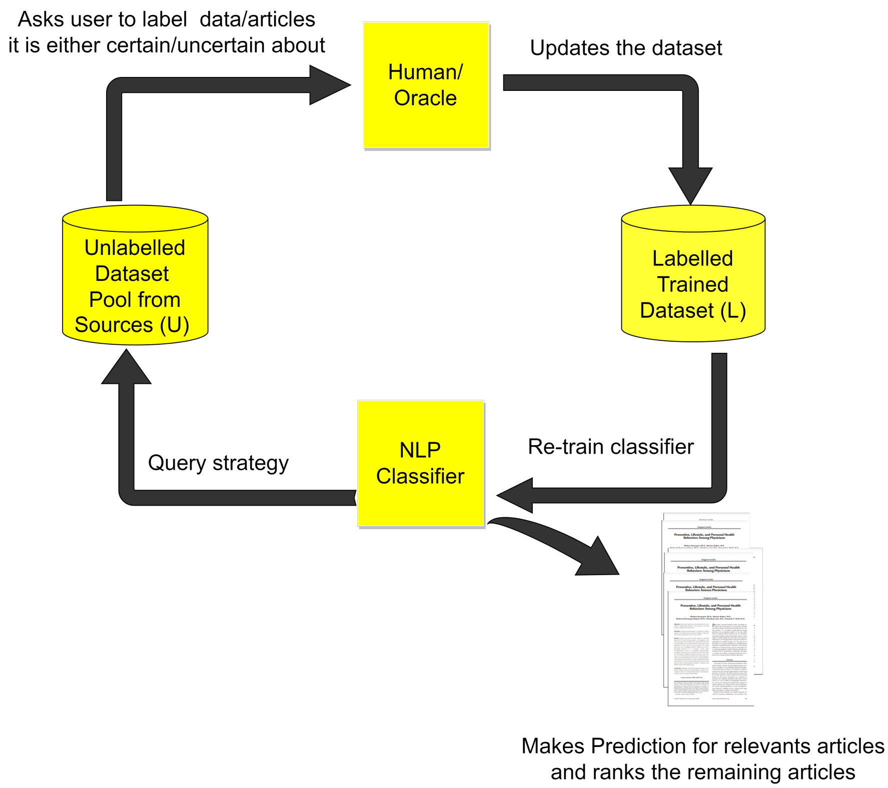


Fig. 7: Active learning cycle for SR automation

uncertainty sampling (Lewis, 1998). The rationale behind this strategy is to present or select instances where it has minimal confidence in its expected output or prediction. In so doing, three main probabilistic approaches were used. The first is the least confidence method, mathematically written as, where is the instance, is the expected label, and is the probability of y happening if x has transpired, and $H(x)$ is the uncertainty value. The learner queries are outputs with higher $H(x)$ values. One limitation of this approach is that it considers only one of the many possible expected probabilities of an instance to calculate the uncertainty value whilst ignoring the rest. To solve this, the margin of sampling query strategy is used (Scheffer et al, 2001). It calculates the uncertainty level using the expected label's highest and second-highest probability. The formula used for this method is $H(x) = P(y_1 \mid x) - P(y_2 \mid x)$. The third approach used is entropy sampling (Shannon, 1948). This uncertainty sampling method uses a summation of an instance's probability labels instead of finding the uncertainty value using some selected values. Certainty-based sampling (Miwa et al, 2014) is another query strategy, which is the inverse of uncertainty sampling. Here, the learner queries the user on data it is most confident about its expected output. In SR, this type of query is helpful because the goal would be to present relevant articles for querying, thus minimising the workload. Other types include the query-by-committee and expected model change, among others. A detailed explanation of how AL works is found in the survey by (McGreevy and Church, 2020). AL is the most used method in automating the screening phase from the related works, especially for methods deployed as tools.

2.6 Deep Learning (DL)

DL is a subfield of AI that employs neural networks with multiple layers to address complex problems that are challenging for traditional ML algorithms, especially beneficial for handling larger datasets. The simplest form of neural network used in DL is a perceptron, which consists of a single layer coming together to form multiple layers. The following summarises the basic DL model proposed for SR automation, illustrated in Figure 6. :

- **Convolutional Neural Network (CNN)**: Apart from SVM, CNN is the model proposed to automate three (data extraction, RoB and search) out of the four SR stages. The general architecture of a CNN (Lecun et al, 1998) model comprises a convolutional layer with activation functions, a pooling layer, and a fully connected layer to learn from the training data and make future predictions. In the search phase, CNNs are proposed to determine the relevance of textual content by recognising patterns that match the strings or queries. Resulting that CNNs are known for superior pattern recognition capabilities (Albawi et al, 2017), they are proposed as a learning model to extract specific information from both structured or semi-structured research studies Marshall et al (2017).
- **Recurrent Neural Network (RNN)**: These are models suitable for sequential data and tasks where the order of the data points is crucial, such as text processing and time series analysis. However, they struggle with long sequences due to the vanishing gradient problem, which is mitigated by advanced architectures like Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Units (GRU) (Cho et al, 2014). In SR automation, LSTM and Bi-LSTM are the two types of RNNs used to automate SRs, primarily the search stage as depicted in Figure 6.
- **Transformers**: Introduced by Vaswani et al (2023), transformers use self-attention mechanisms to weigh the importance of each word in a sequence relative to others, allowing more effective handling of long-range dependencies in text data. Transformers, primarily BERT (Devlin et al, 2019) and GPT (Radford et al, 2019), are increasingly used in SR automation for tasks such as text classification and data extraction (van de Schoot et al, 2021).

2.7 Evaluation and/or Post-Modelling Phases

Table 1 defines the most common metrics for evaluating NLP models built for SR automation. These metrics are derived from the fundamental concepts of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP refers to the number of relevant articles correctly identified by the model, while TN represents the number of irrelevant articles correctly identified. Conversely, FP, a Type I error, refers to the number of irrelevant articles incorrectly predicted as relevant. FN, known as a Type II error, indicates the number of relevant articles incorrectly predicted as irrelevant. In some active learning approaches, these concepts are denoted as TP^L, TN^L, FP^L, FN^L, where L represents data labelled by the oracle, and U represents unlabelled data whose labels are inferred by the classifier for the remaining citations. In Section 3, where all 52 identified papers are summarised w.r.t the various AI techniques used in the NLP pipeline, metrics such as precision, recall, and f-beta score are frequently reported across the four SR stages. Another principal metric used in SR automation is *Work Saved Over Sampling (WSS)*, particularly in the screening stage and sometimes during the search stage. WSS, first introduced by Cohen et al (2006), measures the reduction in human labour at a given recall level compared to random sampling. This metric estimates the proportion of irrelevant articles researchers do not have to manually review because the model has correctly identified them as irrelevant. The calculation of WSS is mathematically defined in Equation 1, where the most commonly targeted recall (R) levels are 95% and 100%. A recall of 95% is widely considered satisfactory in SRs as proposed by Cohen et al (2006), acknowledging that approximately 5% of relevant studies might be missed. Furthermore, Yu et al (2018) argues that no algorithm can guarantee 100% recall unless all candidate studies are examined, which supports the rationale for not always targeting a 100% recall level. Nevertheless, some SR automation studies report achieving WSS at 100% (van de Schoot et al, 2021). Ultimately, the higher the WSS value, the more effectively the algorithm reduces the workload of human screening. In certain active learning studies, this metric is analogous to yield.

$$\text{WSS@R} = \left(\frac{TN + FN}{N}\right) - (1 - R) \quad \text{where} \quad N = TP + TN + FP + FN \tag{1}$$

2.8 Techniques to Alleviate Over-Fitting of ML and DL for SR automation

Both ML and DL SR models face two main challenges: over-fitting and under-fitting O'Mara-Eves et al (2015). By default, most NLP models suffer from overfitting Marshall and Wallace (2019). In this section, we present some approaches used to curb overfitting for SR automation from related works:

Table 1: Common evaluation metrics used for SR automation

Evaluation Metric	Definition	Calculation
True Positive (TP)	Number of relevant articles/citations correctly identified	TP
True Negative (TN)	Number of irrelevant articles correctly identified	TN
False Positive (FP)	Number of irrelevant articles predicted as relevant (Type I error)	FP
False Negative (FN)	Number of relevant articles incorrectly predicted as irrelevant (Type II error)	FN
Precision (P)	Exactness of AI model, focusing on Type I error	$\frac{TP}{TP+FP}$
Recall (R)	Measures number of relevant records identified correctly (Type II error)	$\frac{TP}{TP+FN}$
Specificity (S)	Estimates number of irrelevant records correctly identified	$\frac{TN}{TN+FP}$
False Positive Rate (FPR)	Inverse of specificity, measures irrelevant articles predicted as relevant	$\frac{FP}{FP+TN}$
Accuracy	General performance of the model	$\frac{TP+TN}{TP+TN+FP+FN}$
Work Saved Over Sampling (WSS)	Reduction of manual screening at a specific recall level	$WSS@R = \frac{TP}{TP+FN+FP} - (1.0 - R)$
Portion Missed (PM)	Relevant articles incorrectly classified as irrelevant	$\frac{FN}{TP+FN}$
Matthews Correlation Coefficient (MCC)	Measures performance on imbalanced datasets	$\frac{(TP \times TN - FP \times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$
F beta score	Harmonic mean of recall and precision	$\frac{(1+\beta^2) \times P \times R}{\beta^2 \times P + R}$
Yield	Percentage of relevant records recognised by the algorithm	$\frac{TP}{TP+FN}$
Burden	Percentage of citations that must be screened manually	$\frac{TP+TN+FP}{N}$
Utility	Assesses yield and burden, taking user preference into account	$\frac{\beta \times Yield + (1 - Burden)}{1+\beta}$
Precision@k (P@k)	Precision at the k-th prediction	$\frac{TP@k}{TP@k+FP@k}$
Recall@k (R@k)	Recall at the k-th prediction	$\frac{TP@k}{TP@k+FN@k}$
Average Precision (AP)	Assess precision over top-ranked forecasts	AP definition
Mean Average Precision (MAP)	Mean of AP across different rankings/queries	MAP definition
Normalized Discounted Cumulative Gain (NDCG)	Compares relevance of one result set to another	NDCG definition

- **Weight regularisation**: In SR automation, this approach constrains the model to minimise the loss function by tuning some hyper-parameters to add weight penalties to the loss function. Examples deployed in SR automation include Lasso regression (L1) and ridge regression (L2) to regularise LR (Simon et al, 2019). A combination of both methods proposed for SR automation is the elastic net regression model (Hans, 2011; Allot et al, 2021).
- **Cross Validation**: proposed for SR automation works by dividing the training data into folds, where some data is used for training and others for testing. This helps to compare how different ML and DL models will work, evaluate their performance on unseen data, and help select the best model for a task (Cohen et al, 2006; Bekhuis and Demner-Fushman, 2012; Timsina et al, 2015).
- **Dropout**: This is a regularisation approach by randomly omitting some units during training neural networks to prevent over-fitting during the training phase. The purpose is to enable the model to study a sparse representation.
- **Use of Ensemble Techniques**: This technique proposed for SR automation has proven to obtain better predictive performance in their models, e.g., the combination of DT and LR to form a Logistic model tree (LMT) for automating the search phase (Almeida et al, 2016; Marshall et al, 2018)
- **Data Balancing Techniques**: One major challenge in SR is class imbalance resulting from the training set having less number of "relevant" data. This involves re-sampling techniques such as over-sampling and undersampling or using cost-sensitive classifiers such as the use of algorithms like cNB (Timsina et al, 2015)

2.9 Overview of techniques used in SR for maintaining recall high whilst increasing precision

In SR, achieving a recall of $\geq 95\%$ is crucial to minimise the omission of relevant articles (i.e., reducing false negatives, FN) (O'Mara-Eves et al, 2015). However, a precision-recall trade-off exists where increasing recall decreases precision and vice versa. Consequently, some studies have employed techniques to enhance precision while maintaining high recall rates. These techniques include feature enrichment, resampling methods, and query expansion. Table 2 summarises the methods proposed in relevant studies to maintain recall rates and improve precision.

3 Summary of the NLP methods proposed for SR automation

This section provides a comprehensive summary of how NLP methods, as discussed in Section 2, have been utilised across the stages of systematic review (SR) in each identified study. The 52 related works reveal that the most automated phases in SR are the search, screening, and data extraction stages. Thus, discussion will be centred around the AI methods used in these four stages. To ensure a thorough discussion of the NLP approaches, the technical stages proposed in each included paper w.r.t the NLP pipeline, i.e. text pre-processing, feature extraction, and modelling techniques, are outlined. The methods discussed are summarised in detail in relation to the various stages of the NLP pipeline. While some related studies have implemented the NLP concepts as either web services or desktop applications, the focus remains on discussing the underlying AI techniques rather than the specific tools. For a deeper exploration of SR automation tools and software, readers are directed to the scoping review by Khalil et al (2022) or the survey conducted by Marshall and Wallace (2019), which comprehensively lists and describes these automation tools.

3.1 Summary of NLP methods proposed in related works for automating the search phase

This section highlights the NLP methods proposed in the related studies for automating the search phase. 11 out of the 52 associated works targeting the automation of the search phase reveal that most proposed NLP automation techniques fall under three major categories: *search prioritisation*, *text classification*, and *information retrieval (with and without visualisation)*. The subsequent subsections delve into these NLP categories and techniques proposed in related studies across various stages of the NLP pipeline. Although various algorithms and vectorisation techniques were explored by researchers, this work only

Table 2: Summary of methods used to increase precision and recall from the related works

Approach	Explanation	Stage	Reference
Query Expansion	Extension of search phrases to include related terms, which further improves original queries, resulting in more affluent and more relevant results	Search	(Bui et al, 2015; Aklouche et al, 2019)
Feature Enrichment	Addition of Medical Subject Headings (MeSH)	Search	(Bui et al, 2015; Cohen et al, 2015)
		Screening	(Cohen et al, 2009; Wallace et al, 2010; Almeida et al, 2016; Howard et al, 2016; Kontonatsios et al, 2020)
Feature Enrichment	Addition of publication type (PT)	Search	(Marshall et al, 2018)
		Screening	(Cohen et al, 2006)
Feature Enrichment	Addition of registry Number	Search	(Allot et al, 2021)
Feature Enrichment	Use of keywords	Search	(Ros et al, 2017; Allot et al, 2021)
		Screening	(Wallace et al, 2010; Miwa et al, 2014)
			(Ros et al, 2017; Weißer et al, 2020)
Feature Enrichment	Addition of references and bibliometric features	Screening	(Gulo et al, 2015; Rúbio and Gulo, 2016; Olorisade et al, 2019)
Feature Enrichment	Use of Unified Medical Language System (UMLS) terms	Search	(Scells et al, 2020)
Feature Enrichment		Screening	(Wallace et al, 2010; Frunza et al, 2011a)
			(Timsina et al, 2015)
Acronym disambiguation module	Expansion of abbreviation to prevent vagueness, especially for short acronyms.	Search	(Soto et al, 2018)
Combination of Sampling techniques	Use of SMOTE + undersampling	Screening	(Timsina et al, 2015)

presents the best-performing methods, except in cases involving ensemble techniques. Tokenisation, as a fundamental process in NLP, is prevalent across articles in this category, with most employing it on their training dataset. Table 3 and Table 4 provide a detailed summary of these proposed approaches for automating the search stage under each category.

3.1.1 Search prioritisation techniques for search automation

Search prioritisation is one of the primal techniques proposed for automating the search phase in the SR process. It is a semi-supervised text classification approach that re-orders articles in the remaining unlabelled dataset such that articles eligible for inclusion are ranked higher. Cohen et al (2015), one of the earliest studies found and solely under this of automation of the search phase, proposed the use of search prioritisation as a method of ranking citations as being RCT studies with a confidence score ranging from 0 to 1. Using the Medline RCT filter as a comparator, the researchers proposed using SVM to train a 5 million dataset retrieved from Medline, , with partially labelled data. Performance metrics obtained from the AUC, average precision, F1-score, and accuracy highlighted the potential of the approach over the traditional Medline RCT filter with a precision metric obtained from their pilot testing spanning from 0.85, AUC ROC was between 0.971 - 0.978 and accuracy of 0.98.

3.1.2 Text classification techniques for search automation

Automating the search phase of the SR process has transitioned from ranking-based search prioritisation to binary text classification methods. Compared to Cohen et al (2015), Marshall et al (2018) aimed at training an ensemble model to classify citations as RCT studies. However, instead of a ranking score as output, the methodology proposed by the latter was binary (whether a study was RCT (1) or not (0)). Using the Cochrane Highly Sensitive Search Strategy (HSSS), SVM and CNN as a benchmark, the proposed ensemble method trained with CNN+SVM with PT yielded the best results in terms of AUC ROC, recall, and precision. In contrast to training a model with RCT data, Simon et al (2019) and Allot et al (2021) proposed the use of PubMed IDs to classify abstracts as relevant or irrelevant to the research question aiming to reduce search output obtained from the database. Simon et al (2019), was the first study found in the automation of the search stage to propose using an ensemble of classifiers to accommodate the complex nature of the search SR reviews. These classifiers included SVM, maximum entropy, elastic net model, RF, scaled LDA, Boosting, DT, kNN, and NB classifiers trained with abstracts to classify PubMed IDs. Selecting the best-performing model was based on the concept of cross-validation. In the study by Allot et al (2021), which is a comparative study to Simon et al (2019), beyond training the learning models with PubMed IDs, the use of abstracts, registry numbers, and keywords were added as a feature enrichment methods. Similarly, variant classifiers such as elastic net and ridge classifiers were proposed, with the output fed into an LR classifier. Compared to Simon et al (2019), the results obtained on the public LitCovid dataset (Chen et al, 2020), resulted in an AUC of 0.067, recall of 0.144, precision of 0.007, and an F1-score of 0.089 higher.

3.1.3 Information extraction methods for SR search automation

In this category, Mergel et al (2015) proposed the use of an iterative VTM method to extract relevant terms from selected included studies. As such, refining the initial search string to be used in the search phase.. The proposed method was to be introduced during screening, where, as titles and abstracts are screened, essential words/terms are extracted using the TF-IDF approach. The TF-IDF terms extracted with scores are visually displayed using a Heat Map, with higher scores indicating words more likely to be included as refined search strings. Similarly, in the study conducted by Ros et al (2017), a five-step iterative method was proposed. For automating the search phase, in the first step, a set of accepted papers was used as the initial seed to train an ID3 algorithm for generating search strings from terms in the title, abstract, and keywords. A novelty of the proposed method was using the Scopus database to automatically download articles, which later became part of the initial training set based on queries from term extraction.

Likewise, Scells et al (2020) presented a novel approach to automatically explore how to formulate Boolean queries from an SR protocol. The proposed framework comprised 1) query logic composition,

Table 3: Summary of NLP methods

Proposed NLP Task	**Reference**	**Discipline**	**Pre-processing**	**Feature Extraction**	**Training part**	**Training Technique**	**Learning Model**	**Public code**	**Dataset**	**Evaluation Metrics**	**Deployed/ Name**
Screening prioritisation	(Cohen et al, 2015)	Medicine	Tokenisation	N-Gram Chi-squared	Title Abstracts MeSH	Semi-Supervised	SVM	No	Private	Precision Accuracy AUC ROC F1	Yes RCT Tagger
Text Classification	(Marshall et al, 2018)	Medicine	Tokenisation	N-gram Word-Embedding	Title Abstract RCT PT	Supervised	CNN+SVM	Yes	Private	AUC ROC Recall Precision Specificity F1-Score	Yes Robot Search
Text Classification	(Simon et al, 2019)	Medicine	Tokenisation Stop words Stemming	BoW TF-IDF	Abstract	Supervised	SVM RF Glmnet NB L1, L2 model Elastic Net	Yes	Private	AUC ROC F1-Score	Yes Bio-reader
Text Classification	(Allot et al, 2021)	Medicine	Tokenisation	Title N-Gram Registry Keywords	BoW N-Gram	Supervised	LR Elastic Net L1, L2 model	No	Public	Recall Precision AUC ROC F1-Score	Yes Lit-suggest
Information extraction String/-Query Formation	(Mergel et al, 2015)	SE	Tokenisation	TF-IDF Heat Map Visualisation	Title Abstract	Supervised	Not stated	Yes	Private	Not explicitly stated	Yes SLR.qub
Information extraction String/-Query Formation	(Ros et al, 2017)	SE	Stemming	N-grams TF-IDF	Title Abstract keyword	Semi-Supervised	DT (ID3)	No	Private	Accuracy Recall Precision F1-score	No
Information extraction String/-Query Formation	(Scells et al, 2020)	Medicine	Tokenisation	Not explicitly stated	Review statement (protocol+ seed citations)	Supervised	Not stated	No	Private	Precision F1 score Recall WSS	No

Table 4: Summary of NLP methods

Proposed NLP Task	Reference	Discipline	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Information retrieval and Query Expansion	(Bui et al, 2015)	Medicine	Stemming	Not explicitly stated	MesH	Unsupervised	Not stated	No	Private	MAP NDCG P@10	No
	(Aklouche et al, 2019)	Medicine	Tokenisation Stop words	Stemming Word2vec	Title	Unsupervised	Used Word2Vec	No	Private	MAP NDCG P@10	No
Information retrieval with visuals (VTM)	(Russell-Rose et al, 2019)	Medicine	Tokenisation	N-grams Word2vec-(PubMed trigram)	Not stated	Unsupervised	Not stated	No	Private	Recall Precision F1 score	Yes 2D Search
Information retrieval Query Formulation	(Soto et al, 2018)	Medicine	Named Entity Recognition (NER)	Not explicitly stated	Abstract	Not stated	HMM	No	Public	infNDCG P@10 R-prec	Yes Thalia

a logical hierarchy to extract statements describing the protocol using an English probabilistic context-free grammar (PFCG) (Klein and Manning, 2003), which was to convert the logics extracted to noun phrases, 2) extraction of entity and representation as ULMS terms, 3) optional expansion of the entities represented, 4) mapping of entities to keywords and, 5) and post-processing using techniques like stemming. It was realised that this study is the first to have reported WSS for the search phase. Overall, the results obtained from evaluation metrics precision, recall, F1 score and WSS indicate the method's potential to automate the SR search phase using the SR protocol.

3.1.4 Information retrieval techniques for search automation

Moving to the most used approach for automating the search phase, in this category, , it was noticed that the two main techniques deployed were: QE and ranking. Another observation noted is the variation in evaluation metrics across studies, including precision@k (P@k) and mean average precision (MAP), as depicted in Table 1. Bui et al (2015) presented an unsupervised QE method and ranking approach, with PubMed QE expansion as the comparator. The researchers proposed adding MeSH terms to PubMed queries for QE and suggested using an ensemble classifier of NB and SVM for ranking. The proposed approach achieved comparative results using MAP, NDCG, and P@10. Similarly to Bui et al (2015), Aklouche et al (2018) proposed using an unsupervised iterative QE and ranking method as an extension of PubMed's search engine. The study aimed to present a novel technique of QE by training a Word2Vec embedding model. Suggesting a 4-stage pipeline, the method included 1) data pre-processing, 2) training of the model, 3) QE, and 4) ranking of relevant articles from PubMed search. To rank the documents, Aklouche et al (2018) proposed using Okapi BM25 (Zhang et al, 2009), a probabilistic weighting to find the most significant articles analogous to TF-IDF. Russell-Rose et al (2019) likewise presented the use of a meta-search engine which maps the API of some databases, such as Google Scholar, PubMed, and Elastic Net, to expand queries. The studies aimed to propose a method to serve as an alternative to conventional "advanced searches." Here, the researchers suggested the addition of a 2-D canvas where queries can be manipulated. The study investigated word embedding, Glove, and Word2Vec on Wikipedia, Google News and PubMed (Chiu et al, 2016) to expand queries. The validation results concluded that word2vec trained on PubMed data produced the best QE and search string recommendation results. Finally, Soto et al (2018) also proposed using a semantic search engine that expands queries to identify articles from the PubMed database as part of its methodology. The NLP processing suggested was named entity recognition (NER) to extract medical entities. In the study by Soto et al (2018), the entities were limited to only eight main concepts in search words to be typed by the user (chemicals, species, drugs, metabolites, diseases, genes, proteins, and anatomical entities).

3.2 Summary of NLP methods proposed in the related works for automating the screening phase

The 33 related studies aiming to automate the screening phase can be categorised under four main approaches: *screening prioritisation*, *text classification*, *active learning (human-in-the-loop)* and *reinforcement learning*. Primarily, most of the proposed methods to be discussed that are deployed as software (desktop/web) use *active learning*. In contrast, those not deployed predominantly use *text classification*, including state-of-the-art LLMs-based approaches. Throughout the various papers, the most common evaluation metric that runs through the related works is the *WSS*. The subsequent subsections delve into how the various approaches were proposed in related studies across various stages of the NLP pipeline. A detailed summary and comparison of the related works for studies that proposed screening prioritisation and reinforcement learning is provided in Table 5. Similarly, Table 6 and Table 7 also provide a comprehensive summary of the various text classification methods proposed as well Table 8 for the active learning methods.

3.2.1 Screening prioritisation technique for screening automation

Screening prioritisation is a ranking-based method that assigns a confidence score to each citation instead of a binary label. Most studies in this section deployed topic modelling and clustering methods. Cohen et al (2009) proposed a novel topic modelling technique known as cross-topic learning, combining topics

Table 5: Summary of methods in related studies proposed for automating the screening stage

Proposed NLP Task	Reference	Discipline	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Screening-Prioritisation	(Cohen et al, 2009)	Medicine	Tokenisation Stop words	N-gram	Title Abstract MeSH	Semi-supervised	SVM Clustering	No	Public	AUC	No
Screening-Prioritisation	(Howard et al, 2016)	Medicine	Tokenisation	N-gram TF-IDF	Title Abstracts MesH	Unsupervised	LDA Log-linear	No	Public	WSS @95	Yes SWIFT-Reviewer
Screening-Prioritisation	Gonzalez-Toral et al (2019)	SE	Tokenisation Stop words Stemming Lemmatisation	N-gram TF-IDF	Abstract	Unsupervised	PCA	No	Public	Recall AUC ROC	No
Screening Prioritisation	(Kontonatsios et al, 2020)	Medicine SE	Stemming Stop words	Autoencoders+ feed-forward	Title Abstract MesH	Semi-supervised	SVM	Yes	Public	No	No
Screening Prioritisation	(Weißer et al, 2020)	Multi-disciplinary	Tokenisation Stop word Stemming	TF-IDF	Title Keywords Abstract	Unsupervised	K means	No	Private	Silhouette-score (SSC) Sum of squared errors (SSE)	No
Screening Prioritisation	(Cawley et al, 2020)	Medicine	Tokenisation Stop words	Not explicitly stated	Title Abstract	Semi supervised	K means	No	Public	Recall	No
Reinforcement Learning	(Ros et al, 2017)	SE	Stemming Tokenisation	TF-IDF N-gram	Abstract Title Keywords	Semi-Supervised	LR	No	Private	Recall F1-score Precision WSS Accuracy	No
Visual Text Mining	(Felizardo et al, 2012)	Not stated	Tokenisation	Not explicitly stated	Title Abstract Keyword References	Unsupervised	Clustering	No	Private	Performance Effectiveness	No

from specific topic training datasets with information from other SR topics to train an SVM.To reduce classifier bias, more specific topics with fewer non-specific topics were recommended. Results from the AUC metric demonstrated how cross-topic learning can aid in automating the screening phase. Howard et al (2016) also suggested using topic modelling to discover citation keywords for training a log-linear supervised model. Bag of n-grams with TF-IDF, was proposed as a feature extraction method alongside the use of LDA to facilitate topic modelling. Likewise, the study by Kontonatsios et al (2020) aimed to project the use of a novel supervised neural-based extraction method compared to the standard feature extraction methods. The architecture of the proposed deep learning feature extraction had a denoising autoencoder and a feed-forward network, which was used to train an SVM to rank the unlabelled part of the dataset using a confidence score. The scores were calculated based on the "soft-margin" distance of features for a particular citation to the hyperplane of the SVM. Their proposed model indicated a promising result compared to 5 other baseline models, BoW-LDA, BoW-SVD, BoW-MeSH, BoW-LDA, BoW-PV, and BoW-SVD-LDA-PV. On the other hand, Gonzalez-Toral et al (2019) also investigated how using unsupervised clustering of words in citations can reduce and prioritise the words in citations that may apply to the research question. Different experiments were done using LDA, embedding techniques such as (Word2Vec, Doc2Vec, FastRead) and PCA with BM25. Experimental results showed that using PCA for ranking words in citations outperformed all the other experimental models. Similarly, the work by Weißer et al (2020) introduced an unsupervised method, k-means clustering, for filtering abstracts. The clustering algorithm trained using a large metadata set comprised of titles, abstracts, keywords, and authors' names. The NLP pipeline included tokenisation of documents with stop words removal, stemming, and TF-IDF vectorisation, with Latent Semantic Analysis (LSA) employed for dimensionality reduction. Evaluation metrics such as average TF-IDF score per word per cluster, the sum of squared errors (SSE), and silhouette score (SSC) were computed. Results showed that clustering using titles yielded promising results compared to abstracts or keywords, suggesting that abstract and keyword text may be too complex for effective dimensionality reduction. Finally, Cawley et al (2020) suggested a semi-supervised clustering method to identify relevant studies. This technique utilised a set of "initial seeds" or relevant studies for training and clustering algorithms to rank clusters on new datasets. Using and ensemble approach of nonnegative matrix factorisation (NMF) and k-means with cluster sizes of 10, 20, and 30, the experimental results indicated the prospective of the proposed method for expediting citation screening. Although screening prioritisation has proven effective in automating abstract screening tasks, more recent studies is geared toward automating the screening tasks as a binary task, *text classification*, rather than a screening prioritisation task.

3.2.2 Text classification techniques for screening automation

In this category, Cohen et al (2006) is one of the earliest studies found. This study introduced having a recall $\geq$ 95% in screening classification and calculating WSS@95%. The pre-processing technique involved the use of stemming and stop words on the most occurring 300 tokens from titles, abstracts, MESH, and Medline PT in the training dataset. The training utilised a voting perceptron-based approach with a linear kernel. Results indicated that recall ≥ 0.95 was achievable for the screening task however, reported a trade-off where an increase in recall resulted in a reduction in WSS@95. Tomassetti et al (2011) proposed using the Linked Data approach, a method of using an existing technology within the area of the semantic web to enrich the domain of studies obtained in the search phase with the information to select relevant studies. This method was later used to train an NB classifier to classify unseen studies as relevant or irrelevant to the research question. The researchers proposed using BoW after applying pre-processing techniques like stop words and stemming for feature extraction. They presented the use of the title, introduction, abstract and conclusion for training based on the studies by Cohen et al (2006), which suggests that the essential terms in documents appear at the beginning and the end. Similarly,Frunza et al (2011b) presented the addition of the research question to classify medical citations. Comparing the addition of the research question to the proposed classifier, NB, with the same classifier built without the research question, they found that the addition improved the evaluation metrics, precision, and recall. Likewise, they also projected from their comparative study that combining ULMS terms and BoW for feature extraction improves results. The investigation by Bekhuis and Demner-Fushman (2012) focused on examining the impact of different citation portions (title + abstract, full citations i.e., title + abstract

Table 6: Summary of text-classification methods in related studies for automating the screening stage

Proposed NLP Task	Reference	Discipline	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Text-Classification	(Cohen et al, 2006)	Medicine	Stemming Stop-words	BoW	Title Abstract MeSH Medline PT	Supervised	Voting-perceptron with linear-kernel	No	Public Creators	F1 Precision Recall WSS@95	No
Text-Classification	(Frunza et al, 2011a)	Medicine	Stop-words Normalisation	BoW	Abstracts Research-question UMLS	Supervised	NB	No	Private	Precision Recall	No
Text-Classification	(Tomassetti et al, 2011)	Medicine	Stemming Stop-words	BoW	Title Abstract Introduction Conclusion	Supervised	NB	No	Private	Recall	No
Text-Classification	(Bekhuis and Demner-Fushman, 2012)	Medicine	Tokenisation Normalisation Stop-words Stemming	BoW N-gram	Title Abstracts Metadata	Supervised	EvoSVM cNB	No	Private	Recall Precision F3 score	No
Text-Classification	(Gulo et al, 2015)	Medicine	Stop-words Normalisation	TF-IDF	Bibliometric-features	Supervised	ID3 NB	No	Private	Recall Accuracy Precision	No
Text-Classification	(Almeida et al, 2016)	Medicine	Tokenisation	BoW IDF Odds Ratio	MesH Keywords Title Abstract	Supervised	LMT (DT+LR)	Yes	Private	Recall Precision F1 and F2	No
Text-Classification	(Timsina et al, 2015)	Medicine	Tokenisation	BoW	Title Abstract UMLS	Supervised	SoftMax-SVM	No	Public	F1 Precision Recall WSS@95	No
Text-Classification	(Bannach-Brown et al, 2019)	Medicine	Tokenisation	TF-IDF N-gram	Title Abstracts	Supervised	SVM with SDG	no	Public Creators	Precision Recall Accuracy WSS@95	no
Text-Classification	(Olorisade et al, 2019)	Medicine SE	Stop-words	BoW TF-IDF Word2Vec	References	Supervised	SVM	No	Public	Precision Recall Accuracy WSS@95 MCC	No

Table 7: Summary of text-classification methods in related studies for automating the screening stage

Proposed NLP Task	Reference	Discipline	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Text-Classification	(Natukunda and Muchene, 2023)	Medicine	Tokenistion Stop-words	Topic-modelling	Title Abstract	Unsupervised	LDA	No	Private	True positive-rate against the no of topics	No
Text-Classification	(Hasny et al, 2023)	Medicine	Not stated	BERT tokenizer	Title Abstract	Supervised	BERT SciBERT MedBERT PubMedBERT	Yes	Private	AUC ROC Recall %Reduction	No
Text-Classification	(Ofori-Boateng et al, 2023)	Medicine	Tokenisation Stop-words	GloVe	Title Abstract	Supervised	LSTM Bi-LSTM	No	Private/ Public	Precision Recall F1 WSS@95	No
Text-Classification	(Moreno-Garcia et al, 2023)	Medicine	Tokenisation	GloVe FastText Doc2Vec	Title Abstract	Supervised	SVM Zero-Shot	No	Private/ Public	Precision Recall F1	No
Text-Classification	(Orel et al, 2023)	Medicine	Stop-words	Topic-modelling Clustering	Abstract	Unsupervised	K-NN	No	Private	WSS@95	Yes LiiteRev

+ metadata, and title + abstract) on automation processes. Additionally, the study explored the influence of Bag of Words (BoW), bi-grams, and tri-grams on training. It evaluated the effectiveness of kNN, NB, cNB, and EvoSVM algorithms in screening automation under these variations. Furthermore, the study delved into the effects of optimisation techniques and cross-validation on model performance. The results suggested that optimising and cross-validating BoW with full citations (title + abstract + metadata) or with title + abstract, using either cNB or EvoSVM, yielded the most favourable outcomes in terms of automation performance. Rúbio and Gulo (2016) also presented bibliometric features as a method of finding relevant studies instead of training the model with studies obtained during the search. These include publications metadata linked with an article's relevance, e.g., the citation number, reference number, media type, year and type of publication. Like all other tasks, the dataset was passed through a series of classifiers, such as DT, NB, ID3 and KNN, where ID3 was the best-performing algorithm. Using their previous study as a benchmark (Gulo et al, 2015), where the researchersproposed using references for text classification with an NB classifier but not with SR data, their latter experiment concluded that the combination of references and bibliometric features has the potential to expedite the screening phase. On the other hand, a comparative study by Timsina et al (2015) was conducted, building upon the work of Cohen et al (2006). The researchers advocated for ULMS as a feature extraction method from the titles and abstracts within the training dataset. Five algorithms were compared in the constructed models: SoftMax SVM, SVM, Perceptron, EvoSVM, and Naïve Bayes. The researchers reported that SoftMax SVM outperformed the other algorithms across four public datasets. In addressing the research question concerning enhancing precision while maintaining high recall rates, they explored various re-sampling techniques such as SMOTE, under-sampling, and a combination of SMOTE + under-sampling. Results derived from using SMOTE + under-sampling demonstrated the highest scores for F1, precision, recall, and WSS@95 when employing a 5X2 cross-validation technique.

Similarly, investigations by Almeida et al (2016) delved into the potential of various re-sampling techniques, feature extraction methods, and feature selection techniques to aid in automating the screening stage. The undersampling technique was proposed to address class imbalance. Regarding feature extraction, the researchers explored the effectiveness of using BoW alongside either MeSH terms or keywords in conjunction with the title and abstract to enhance evaluation metrics. Moreover, different methods were evaluated for dimensionality reduction and feature selection, including Information Gain (IG), Inverse Document Frequency (IDF), and odds ratio techniques. Among the classifiers considered (Logistic Model Tree (LMT), SVM, NB), the results highlighted that employing BoW + MeSH with the LMT classifier using IDF demonstrated potential in automating the screening stage based on precision, F1, F2, and recall metrics. Additionally, Bannach-Brown et al (2019) proposed the utilisation of tri-grams with TF-IDF for their approach. The dataset utilised was curated by the authors. The proposed method employed SVM with Stochastic Gradient Descent (SGD) to automate the screening phase. Similarly, Olorisade et al (2019) aimed to demonstrate the potential of feature enrichment in improving citation screening. The researchers investigated the impact of adding references/bibliography to each citation on evaluation metrics. The study used 19 public datasets, comprising 15 clinical reviews and four software engineering datasets, to create two data sets: one with reference data and one without. Regarding the learning model, different configurations of SVM (BoW with non-linear kernel, word2vec with linear kernel, and word2vec with non-linear kernel) were explored. This study is the first to report the Matthews Correlation Coefficient (MCC) metric. Experimental results depicted that adding reference data has potential in the automation of citation screening.

More recently, text classification for abstract screening has shifted towards the use of RNNs and LLMs. Hasny et al (2023), is one of the newer papers to investigate the use of BERT and its biomedical variants for title and abstract screening for complex SR datasets. To fine-tune the BERT models for this classification challenge, the study employs two intricate datasets, encompassing human, animal, and in-vitro studies. Backtranslation, a data augmentation technique, is used to address issues of class imbalance. The study compares the performance of BERT models and their variants on both original and augmented data sets. The findings indicate that BERT models and their variants offer an accessible and efficient solution for the screening phase of SR. Natukunda and Muchene (2023) also presented the use of an LDA-based topic model to identify relevant topics from titles and abstracts, and the establishment of a scoring threshold for determining the relevance of documents for full-text review. The methodology was retrospectively applied to two systematic review datasets: one on Helminth and the other on Wilson disease. The results showed varying degrees of sensitivity and specificity. In the helminth dataset, the method achieved a sensitivity of 69.83% against a false positive rate of 22.63%. In the Wilson disease dataset, the sensitivity was 54.02%, with a specificity of 67.03%. Moreno-Garcia et al (2023) presented the use of traditional machine learning SVM combined with a zero-shot classification approach. GloVe, FastText and Doc2vec were explored as the feature extraction method combined with a zero-shot classification threshold output. In summary, the results showed that the combination of the output of the zero-shot method as input to the SVM model showed promising results. Orel et al (2023) also introduced LiteRev, a tool that collects relevant metadata, including abstracts or full texts. It then processes this text data and transforms it into a TF-IDF matrix. Employing dimensionality reduction and clustering techniques, LiteRev uses a k-NN algorithm to suggest potentially relevant papers. Out of 613 papers suggested for screening (31.5% of the total corpus), LiteRev correctly identified 64 relevant papers (73.6% recall rate) compared to the manual abstract screening. For full-text screening, LiteRev had a recall rate of 87.5%, accurately identifying 42 relevant papers out of 48 found manually. This resulted in a total work-saving oversampling of 56%. The study demonstrates LiteRev's effectiveness as an automation tool. Finally,Ofori-Boateng et al (2023), presented the use of LSTM and Bi-LSTM, coupled with GloVe for vectorisation, in streamlining the abstract screening stage. Additionally, to address the precision-recall trade-off—a common challenge in classification tasks—the study incorporates attention mechanisms into these classifiers. This enhancement is aimed at boosting precision while maintaining a recall rate of at least 95%. The experimental results demonstrate that the Bi-LSTM model with the added attention mechanism shows promising potential in accelerating the citation screening process.

In summary, although these text classification methods have shown great potential in automating abstract screening, they are fully automated and, as such, do not allow humans-in-the-loop or user input. The next subsection discusses how the concept of active learning(humans-in-the-loop), is deployed in most existing AI screening automation software (deployed as a web/desktop) from the related works.

3.2.3 Active learning (AL) techniques for screening automation

As stated in Section 2.5.3, AL allows humans in the loop. However, a significant challenge faced by many AL models identified in this review and reiterated in the study conducted by (Marshall and Wallace, 2019) is the absence of a precise threshold for human intervention in screening processes. The calculation of WSS often assumes that users possess prior knowledge of when optimal recall levels are achieved, a situation rarely encountered in real-world scenarios (Przybyła et al, 2018). Notably, only two studies in this review attempted to tackle this challenge. An SR AL screening review conducted by Yu et al (2018) identified three state-of-the-art methods (Wallace et al, 2010; Miwa et al, 2014; Cormack and Grossman, 2014), serving as foundational frameworks for other AL screening methods. These methods primarily address four key areas crucial for AL implementation: 1) when the classifier starts training, 2) which studies to query next, 3) whether to stop training or continue and 4) how to balance the training data. For 1), i.e., when to start training, two main suggestions that are proposed are "patient" (P) and "hasty" (H). In P, the algorithm keeps random sampling until a specified number or an adequate number of the "relevant" studies are obtained or retrieved from the dataset. In H, the reverse of P, the classifier begins training as soon as one "relevant" study is found. Compared to P, H is of tremendous advantage since it causes the algorithm to learn faster, thus saving time to make predictions on the remaining articles (Cormack and Grossman, 2014; van de Schoot et al, 2021). Similarly, 2) has two leading suggestions already described in Section 2.5.3. These are U for "uncertainty sampling", and C for "certainty sampling". In 3), the two main suggestions proposed for SR automation are whether the algorithm should continue training (T) or stop training (S). In T, the algorithm never stops training, but when the query strategy used is U, the algorithm only switches to C after the classifier attains stability. On the other hand, in S, the algorithm stops training immediately after the classifier achieves stability. This stability is reached based on a specified number of "relevant studies" that the classifier can find from the training data. Finally, in 4), these papers propose four primary suggestions for data balancing; no balancing (N), aggressive under-sampling (A), weighting (W) before and after the algorithm reaches stability, and M for "mixing of W and A". Where the balancing is M, W is first applied before the classifier attains stability, and A is used after. The AL techniques summarised in related studies are detailed based on these state-of-the-art methods in Table 9.

Table 8: Summary of active learning methods in related studies proposed for automating the screening stage

Proposed NLP Task	Reference	Discipline	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Active Learning	(Wallace et al, 2010)	Medicine	Tokenisation	N-gram TF-IDF	Title Abstract MeSH Keywords UMLS	Semi-Supervised	SVM	Yes	Private	Yield Burden	Yes Abstrackr
Active Learning	(Cormack and Grossman, 2014)	Humanities	Tokenisation	Not explicitly stated	Abstract	Semi-Supervised	SVM	No	Private	Recall	No
Active Learning	(Miwa et al, 2014)	Medicine Social-sciences	Stop words Tokenisation	Topic modelling	Title Abstract Keywords	Semi-Supervised	LDA SVM+L2 LR	No	Private	Yield Burden Utility AUC ROC	No
Active Learning	(Hashimoto et al, 2016)	Medicine	Tokenisation	Doc2Vec Topic-modelling	Abstract	Semi-Supervised	SVM	No	Private	Yield Burden WSS@95	No
Active Learning	(Ouzzani et al, 2016)	Medicine SE Social-science	Stop words Stemming	N-grams	Title Abstract MeSH	Semi-Supervised	SVM	No	Not stated	AUC ROC WSS@95	Yes Rayyan
Active Learning	(Cheng et al, 2018)	Medicine	Tokenisation	Word2Vec	Title Abstract	Semi-Supervised	SVM with SGD	No	Private	Not stated	Yes Colandr
Active Learning	(Przybyła et al, 2018)	Medicine	Stop words Lemmatisation Clustering	TF–IDF BoW	Title Abstracts	Semi-Supervised	SVM LDA	No	Private	WSS@95	Yes Robot Analyst
Active Learning	(Yu et al, 2018)	SE	Tokenisation Stop words	BoW TF-IDF	Title Abstract	Semi-Supervised	SVM	Yes	Public	WSS@95	Yes FASTREAD
Active Learning	(Howard et al, 2020)	Medicine	Tokenisation	N-gram TF-IDF	Title Abstracts	Semi-Supervised	Log-linear	No	Public Private	WSS@95 Recall	Yes SWIFT Active-Screener
Active Learning	(van de Schoot et al, 2021)	Medicine SE	Tokenisation Normalisation	Doc2Vec TF–IDF N- gram sBERT	Title Abstracts	Semi-Supervised	SVM NB DNN LR LSTM RF	Yes	Public	WSS@100 WSS@95	Yes AsReview
Active Learning	(Chai et al, 2021)	Medicine	Tokenisation	Doc2vec N-gram TF-IDF	Title Abstracts	Semi-Supervised	Transformer	No	Private	WSS@95	Yes Robot Screener

Table 9: Summary of AL techniques in related works used in SR automation where P = Patient, H = Hasty, S = Stop training, T = Continue training, A = Aggressive sampling, N= No balancing, W = Weighting, M = Mixed

Active Learning Studies	**When to Start Training**	**Which Document to Query Next**	**Whether to Stop Training (or not)**	**How to Balance the Training Data**
(Wallace et al, 2010)	P	U	S	A
(Cormack and Grossman, 2014)	H	C	T	N
(Miwa et al, 2014)	P	C	T	W
(Hashimoto et al, 2016)	N/A	C	N/A	W
(Ouzzani et al, 2016)	Not explicitly stated	Not stated explicitly but uses five- star score rating	S	N/A
(Cheng et al, 2018)	P	C	T	M
(Przybyła et al, 2018)	P	U	T	Not stated
(Yu et al, 2018)	H	U	T	M
(Howard et al, 2020)	P	C	S	N/A
(van de Schoot et al, 2021)	H	U	T	M
(Chai et al, 2021)	P	C	N/A	N/A

The study by Wallace et al (2010) is noted as an early advocate of AL for screening automation, where the PUSA was introduced alongside an SVM classifier. The SVM model utilised manual annotations for classification (relevant, borderline, or irrelevant) to rank remaining citations asynchronously. Feature extraction involved N-Gram with TF-IDF for titles, abstracts, and MeSH terms enriched by UMLS terminology. Results indicated AL's potential in screening automation, especially with UMLS enrichment, reducing human effort while maintaining screening efficacy(Gates et al, 2018). Similarly, Cormack and Grossman (2014) advocated for the HCTN approach, favouring quicker initiation of training over patient strategies. It is one of the initial studies to show the potential of using "Hasty" generalisation instead of "Patient" when the algorithm should start training. Miwa et al (2014) contributed an AL method employing PCTW, combining L2-regularised SVM and logistic regression. The work emphasised certainty sampling's advantages over uncertainty sampling and introduced evaluation metrics like yield, burden, coverage, and utility for AL models. Hashimoto et al (2016) proposed paragraph vectors for topic detection in AL, contrasting with traditional LDA. This method's context awareness enhanced the grouping of similar words, improving WSS@95 and reducing the workload. Also, Ouzzani et al (2016) focused on N-gram features and MeSH terms with an SVM classifier, employing a five-star rating system for query strategy.

Cheng et al (2018) introduced the PCTM method for training an SVM with SDG, suggesting the commencement of training after identifying 100 "relevant" studies, which may be limiting for studies with fewer inclusions. Also, Przybyła et al (2018) recommended the PUT method for screening, focusing on automated keyword extraction from titles and abstracts to train SVM models. Feature enrichment included utilising the GENIA tagger for lemma and POS tracking and adopting the C-value to improve keyword identification. The study's novelty was real-time evaluation during an ongoing review, showcasing potential workload reduction from 7% to 71% based on WSS@95 metrics across 22 citation collections. Likewise, Yu et al (2018) also suggested the usage of HUTM for screening citations from the title and abstract. Like all other studies, basic pre-processing techniques were deployed. The main aim of the studies was to compare the three state-of-the-art screening AL methods and how different combinations from these suggestions could outperform the original techniques. Thus, their result found that the HUTM method outperforms the three state-of-the-art methods. Howard et al (2020) contributed to the PCS approach, introducing a recall-based stopping criterion using the negative binomial distribution to determine the safe threshold for halting screening, ensuring a recall rate of 95%. This study is the first to propose a method to handle the "safe" threshold faced by AL SR methods. Their method showed promising results with an average WSS@95 of 35% across 26 heterogeneous datasets.

van de Schoot et al (2021) also proposed using HUTM like Yu et al (2018) for screening. The study's novelty is that it allows a wide range of classifiers to be implemented, allowing it to accommodate the varying complexity of SR projects, thus having higher flexibility. The classifiers proposed by the researchers are SVM, NB, the default algorithm, LSTM, LR, and RF. Interestingly, this study is the only one we found in this review that uses transformer models for feature extraction, Sentence BERT, from the titles and abstracts. Their study also showed the use of multi-feature extraction techniques that the oracle could select TF–IDF Embedding-IDF, Doc2Vec with the default TF-IDF and BoW. van de Schoot et al (2021) is the first study we found to have reported WSS@100 compared to the most used WSS@95. In evaluating their approach on four SR datasets created by the authors, the WSS@100 obtained was within 38.2% - 92.6% and WSS@95 was also within 67-92%. Chai et al (2021) introduced the use of PC, although the specifics of data balancing and stopping criteria for training were not explicitly detailed. Similar to Howard et al (2020), one of the study's objectives was to establish a "safe stopping" threshold for the oracle. For feature extraction, Doc2Vec was proposed by the researchers for titles and abstracts. The proposed algorithm engages users by presenting articles in batches of fifty, then used as input for AL algorithms to re-rank subsequent batches of fifty articles. The rationale for this batch size stemmed from preliminary experiments indicating that immediate algorithm retraining after user labelling led to accelerated re-ranking, potentially causing relevant articles to be pushed down in the ranking order and overlooked. Sensitivity analyses were conducted across nine SR datasets to determine the optimal screening threshold. A five-step interval approach was used to assess the capture rate of final relevant articles at different intervals (5%, 10%, 15%, 20%, and so forth). For example, in a sensitivity analysis of the "Low back pain - lifting" dataset with 2249 references, where only 13 were deemed relevant, the algorithm identified nine relevant studies after screening 5% of the papers, with similar trends observed at subsequent intervals. This analysis indicated that the percentage of relevant articles screened ranged from 5% to 35%, with an average of 12.8%, suggesting a viable screening threshold of 50%. These findings were supported by WSS@100 results, implying that researchers could confidently halt screening after approximately 40 rounds of citations, assuming a researcher is dealing with an SR study involving 4000 citations. Across nine SR projects, WSS@95 results ranged from 6% to 46%, while WSS@100 showed a 28% to 44% improvement over other AL methods like van de Schoot et al (2021). These studies collectively demonstrate evolving strategies in AL for screening automation, emphasising nuanced approaches in training initiation, query strategies, evaluation metrics, and feature enrichment to optimise screening efficacy while minimising human effort. With the rise in alignment methods such as reinforcement learning, the next subsection discusses a related work found that proposes this approach.

3.2.4 Reinforcement learning technique for screening automation

In this review, the study by Ros et al (2017) is the first and only paper found that proposes the use of reinforcement learning for screening automation. The study contrasted the outcomes achieved using RL paired with LR classifiers against the more commonly employed active learning (AL) approach with SVM classifiers. The results obtained from their investigation indicated that employing RL alongside LR classifiers led to a notable reduction in human effort during screening processes, demonstrating promising outcomes. Moving further, Felizardo et al (2012) contributed to the field by proposing the utilisation of a Visual Topic Model (VTM) for citation screening. They advocated for the adoption of innovative visualisation techniques, including the document map, citation network, and edge bundles, to streamline screening processes. The document map, functioning as a 2-D visual representation, aids reviewers in comprehending the content and identifying similarities among primary studies under consideration. Through clustering methodologies, documents sharing commonalities in titles, abstracts, and keywords are grouped together, enhancing efficiency in analysis. The edge bundle technique, depicted as a hierarchical tree, visually portrays nodes (representing primary studies) and node links (depicting citations), providing insights into the relationships within the literature. Furthermore, the citation network introduced by Felizardo et al (2012) serves to elucidate the intricate relationships between primary studies and their cited references. Their evaluation framework proposed assessing performance metrics, such as time spent identifying relevant studies, and effectiveness metrics, gauging the alignment of included or excluded studies with expert opinions in SRs These methodological innovations underscore ongoing efforts to enhance the efficacy, accuracy, and interpretability of screening processes in research reviews.

Table 10: Summary of Data Extraction and RoB in related studies proposed for automating the screening stage

Proposed NLP Task	Reference	Pre-processing	Feature Extraction	Training part	Training Technique	Learning Model	Public code	Dataset	Evaluation Metrics	Deployed/ Name
Information-Extraction	(Kiritchenko et al, 2010)	Sentence-splitting Stop-words	N-Gram	Abstracts Methodology Results section-from HTML tags	Semi-supervised	Regular-Expression SVM	No	Private	Precision Recall	Yes ExaCT
Information-Extraction	(Marshall et al, 2016)	Tokenisation Stop-words	BoW	Full-texts	Semi-supervised	SVM	No	Private	Precision Recall F1	No
Information-Extraction	(Bui et al, 2016)	Tokenisation Stop-words	BoW	Full text of pdfs	Not stated	SVM with BoW+ Context+ Semantic) Regular-matching	No	Private	Recall Precision F1 score	No
Information-Extraction	(Marshall et al, 2016)(RoB) (Marshall et al, 2017)(Data Extraction)	Stop words Tokenisation	N-grams	Full text of pdfs	Semi-supervised	CNN+SVM PCA Regular-expression	Yes	Private	Not explicitly stated	Yes Robot-Reviewer
Information-Extraction	(Norman et al, 2019)	Tokenisation Stop-words	N-grams BERT-tokenizer	Abstracts of RCT	Semi-supervised	BioBERT Logistic-Regression	No	Private	Precision	No
Information-Extraction	(Marshall et al, 2020)	Tokenisation	N-grams	RCT abstracts from PubMed WHO ICTRP	Semi-supervised	Rule-based Logistic-Regression	No	Private	Recall Precision C-statistics	Yes Trailstreamer
Information Extraction	(Schmidt et al, 2020)	Not explicitly stated	BERT-tokenizer	Abstracts	Supervised	SCiBERT mBERT	No	Private	Recall F1 Precision	No

3.3 Summary of NLP methods proposed in the related studies for automating the data extraction and RoB phase

Eight related works were found for this category. These associated works are summarised in detail in Table 10. One of the earliest studies found to automate the data extraction stage is by Kiritchenko et al (2010). The study's primary purpose was to extract PICO elements and other pertinent information, such as DOI, publication date, funding number, and early stopping of trials, from full texts of RCTs. SVM was proposed to highlight necessary sentences from HTML files with a high probability of containing targeted information. These sentences were highlighted based on the algorithm's identification of their intended information, extracting the best five sentences ranked from high to low, excluding publication details (DOI, DOP, author name). Additionally, a template based on CONSORT statements (Moher, 2001) was proposed, with regular expressions used to extract wordings from highlighted sentences to fill the template.

In comparison, Bui et al (2016) proposed a method for extracting data from PDFs instead of HTML using a nine-stage pipeline. The architecture of their proposed method included 1) text extraction from PDF documents using the open-source tool PDFBox to break down texts into snippets, and 2) classification and filtering of snippets using a multi-pass sieve method to automatically classify the snippets into five categories: title, body text, abstract, metadata, and semi-structure. Normalisation of snippets, identification of IMRAD sections, segmenting sentences, and filtering irrelevant sentences were performed. They proposed using BoW combined with contextual or semantic information to train an SVM for ranking and prioritisation of sentences. Key phrase extraction using regular expressions, noun phrase chunking, and post-processing to filter out lengthy extracted phrases as part of the methodology. Results indicated combining BoW and contextual information for ranking achieved higher recall and precision. Marshall et al (2016) proposed the use of ML based on the standard Cochrane Risk of Bias (RoB) Tool, which assesses seven common types of bias in clinical trials. The system was built using distant supervision, utilising data from the Cochrane Database of Systematic Reviews (CDSR), a vast repository of systematic reviews. This data was used to pseudo-annotate a corpus of approximately 2,200 clinical trial reports in PDF format. Marshall et al (2016, 2017) stand as the only study found in this review to automate both RoB assessment and the data extraction phase. The study aimed to classify RCT articles as having a high/unknown or minimal risk of bias and provide supporting text for that prediction. Additionally, the study aimed to extract PICO elements and general information such as author names and article titles. The Cochrane RoB tool's six domains by Higgins et al. (Higgins et al, 2011) were used for RoB assessment, and distant supervision was employed to obtain labels and rationale for RoB assessment without manual annotation. Distant supervision automates label acquisition through heuristics like regular expressions, which link and extract author judgments and PICO elements. The CNN and Softmax SVM ensemble method was proposed for multi-variant task classification. Additionally, PCA was presented to aid in visualising PICO embeddings. Similarly, Norman et al (2019) also explored automating data extraction for diagnostic test accuracy (DTA) using distant supervision, comparing its effectiveness with direct supervision. They created a dataset of about 90,000 sentences, with experts manually annotating 1,000 sentences. BioBERT and logistic regression models were tested for ranking sentences, showing distant supervision's effectiveness comparable to or exceeding direct supervision. Marshall et al (2020) proposed Trailstreamer, combining ML and rule-based methods to find and categorise new RCT reports automatically. The system extracts trial PICO elements, maps them to Medical Subject Headings (MeSH) terms, predicts the risk of bias, and extracts critical findings. Finally, Schmidt et al (2020) explored BERT variants for PICO extraction in English and multilingual contexts. They treated data extraction as question-answering and sentence classification tasks, achieving high F1 scores across models and domains and addressing ambiguity in PICO sentence prediction tasks through diverse training datasets.

Overall, these studies showcase the evolving landscape of automated data extraction techniques, leveraging machine learning, distant supervision, and advanced LLMs to enhance the speed, accuracy, and scalability of data extraction and RoB assessment in SR.

4 Systematic literature review survey

4.1 Overview

As discussed in Section 3, the automation of stages in the SR process has been targeted by numerous studies. However, it is still unclear which stage in the review process is considered the most burdensome from the perspective of SR reviewers, as existing studies are based on estimations derived from related works. For example, the RoB stage was proposed to be burdensome for reviewers in the SR process by Marshall et al (2016), as it was estimated that an average of 20 minutes is required for a sole study that successfully passes the screening stage to be critically evaluated (RoB). Similarly, an average of 30 - 90 seconds was estimated by Howard et al (2020) for a skilled systematic reviewer to screen a single abstract. Additionally, Przybyła et al (2018) estimated that an average of 80 - 125 hours is required for screening 5,000 publications retrieved from searching, among other estimations. Thus, in the next section, results from an online survey are presented that aim to bridge this gap identified by presenting which stage in the review process SR researchers and practitioners think future AI automation will help, rather than from a point of estimation. Similar methods were followed, and some questions were recruited from the SR survey by Scott et al (2021), which focused on understanding automation tools. However, the aim of our survey is not to understand these tools but to gather the opinions of systematic reviewers. This enables us to identify which stages they find challenging and gather their suggestions on which SR stage AI methods can benefit the most. Additionally, the survey aimed to understand how abreast these reviewers were with AI, targeting their knowledge of automation tools and which stages reviewers apply these SR automation tools. The survey also intended to capture the challenges faced while using the tools and gather general feedback on whether automation tools have been of great benefit to them in the review process. The following subsections discuss the methods and procedures that were followed.

4.2 Study design

The survey was implemented on the JISC platform and comprised 10 main questions provided in Appendix A. The questions asked could be grouped into five main sections. Knowing the location and affiliation of participants was the first aspect. The second aspect was knowing the type of review performed by participants and how long they have been doing it. The third was to assess the level of ease or difficulty associated with the different stages involved in the SR. The fourth was to capture the participant's knowledge of AI through automation tools. Finally, the fifth aspect captured the participants' recommendations for any future AI automation for SR. The estimated time to complete the survey was 5-10 min.

4.3 Participants and distribution

Participation in the survey was entirely voluntary. Researchers who have performed or were performing SRs and were at least 18 years old were targeted by the survey. The team of SR reviewers in the School of Nursing, Midwifery and Paramedic Practice and the School of Health Sciences at Robert Gordon University and The Rowett Institute, University of Aberdeen, were involved in distributing the survey to their networks, such as the Joanna Briggs Institute (JBI), Cochrane Collaboration, etc. The survey was opened on 23rd April 2022, and responses inputted before 1st June 2022 were analysed. Nonetheless, the survey [7] is still open to systematic reviewers who want to share their opinions.

4.4 Result and discussion

The survey results are presented in two formats: a bar chart and statistics. The results for all five aspects of the survey are in Additional File 1 as a bar chart, and statistical values are in Additional File 2.

4.4.1 First and second aspect: Geographic location and type(s) of SRs conducted

In all, 60 responses were obtained from institutions across the globe. The geographical distribution of the participants is indicated in Figure 8. From the responses, it was noticed that 10 (16.7%) of the

[7] https://robertgordonuniversity.onlinesurveys.ac.uk/automating-systematic-literature-review-with-artificial-in

respondents had performed over 10 systematic reviews (SRs) over the past five years, 4 (6.7%) had conducted 7-10 reviews, while 22 (36.7%) had participated in 4-6 SRs and 24 (40%) had been involved in 1-3 SRs over the past years. Likewise, it was also noticed that the type of SR review most commonly performed by the respondents was systematic reviews, with 50 (83.3%) conducting SRs, scoping reviews being the second highest at 28 (46.7%), and meta-analyses the third highest at 26 (43.3%).

Summarising the first and second aspects of this survey, the result gave a general impression that most of the participants were indeed involved in SRs. Thus, on average, had performed at least 3- 6 SRs over the past 5 years, which was beneficial to the overall results to be obtained from the survey.

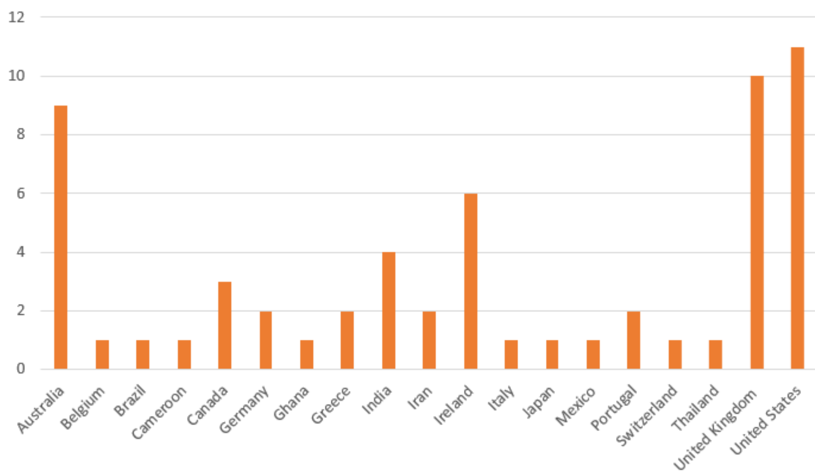


Fig. 8: Results of demographical visualisation of survey respondents

4.4.2 Third aspect: Rating of stages as respondents perform SR

The results obtained for this section focused on knowing the level/difficulty associated with each stage in the SR process using the Likert scale [8] from 1-5 (1 for “very easy”, 2 – “very easy”, 3 – “neutral”, 4 – “difficult”, 5 – “very difficult”. The results are summarised in Appendix B and the statistical summary in Additional File 2. For the development of the protocol, it was observed that, on average, most respondents find this stage neutral. For the search phase, 22 (36.7%) of the respondents rated this stage as difficult, while 6 (10%) rated this stage as extremely difficult. Both 15 (25%) rated this stage as neutral and easy; thus, the level of ease is likewise neutral but more complex, with a mean value of 3.25. For the title and abstract screening, 31 (51.7%) of the respondents rated this stage as easy, while 13 (21.7%) rated this stage as complex. The mean rank was 2.57, indicating that most respondents consider this stage easy. For data extraction and synthesis, 35 (59.3%) rated this stage as complex, and 3 (5.1%) also rated this stage as extremely difficult. Thus, the mean ranking was 3.56. Likewise, the mean rank for the RoB was 3.67. In conclusion, most respondents rated the RoB stage as the most challenging stage they encountered during the SR process, followed by the data extraction stage, with the screening stage. as the easiest. The next subsection sheds more light on why respondents may have given these ratings.

4.4.3 Fourth aspect: Respondent’s knowledge of AI through automation tools

The results from this section are fully recapitulated in Figures B3, B4 and B5 . Concerning the results from this aspect, 33 (55%) of the 60 respondents were familiar with automation tools and utilised them to expedite one or more stages in the SR process. Of those who had not used any automation tool, 27 (45%) of the respondents were aware of automation tools. However, factors such as cost prevented 7 (58.3%) out of the 13 respondents from using such tools. Others, 4 (33.3%), also stated that the lack of availability in their institution prevented them from using such tools. Additionally, one respondent was comfortable

[8] https://en.wikipedia.org/wiki/Likert_scale

with the traditional SR method, and others claimed they were pleased to work with spreadsheets. On the other hand, 14 (51.9%) out of the 27 respondents were unfamiliar with AI automation tools. However, rating their willingness on a scale of 1-10 to accept and use AI, 13 (95.8%) rated above 5, indicating their willingness to use AI tools. Of the 33 respondents who used any AI automation software, 21 (63.6%) mostly used the Covidence tool. The results from the initial question on where in the SR stage the respondents deployed these tools showed that the most used stage was the title and abstract screening, 22 (66.7%), followed by the data extraction, 14 (48.5%); with the search and interpretation of literature as the most miniature stage where the respondents applied these tools, 5 (15.2%). It can be inferred that most respondents probably stated that the title and abstract screening is the easiest stage in (b) because most automation has been developed in that area. It was also realised that most of the 33 respondents learned how to use these tools personally, 14 (42.4%), while others also learned it from conferences, workshops, etc. Overall, 16 (48.5%) of the respondents reported that using automation in SR saves a lot of time, while 15 (45.5%) also stated it saves some time. Additionally, 22 of the 33 respondents encountered no challenges while using the tool. However, 7 out of the 11 suggested that using AI automation for SR was a challenge because some tools required technical knowledge. The conclusion drawn from these results is that automation is indeed a significant benefit in SR automation.

To summarise these results, it can be inferred that most systematic reviewers do have a fair idea of existing available AI automation software. A trend in the tools being used, as seen in Figure B4, is human-in-the-loop. This implies that most reviewers prefer tools that allow them to be a part of the process rather than to be fully automated.

4.4.4 Fifth aspect: Participant's recommendations for future AI automation techniques for SR

Results in this section captured participants' thoughts on which stage is suggested would chiefly benefit from AI automation (Q: Based on your experience as a systematic reviewer, which particular stage in the SR process do you think would be of the most benefit using an automation method or tool?). As seen in Figure 9, 18 (30%) of the 60 respondents indicated that the title and abstract screening would benefit most from using AI. Although most respondents rated this stage as easy, they still recommend it as the most beneficial stage. This confirms that the screening phase is the most time-consuming stage in the process (Booth et al, 2016; Przybyła et al, 2018). Although there are existing methods, exploring this stage is still necessary for reviewers. Additionally, 15% of the respondents suggested that the search phase would be the second most beneficial stage if automated. Both results from the survey in this aspect and the rate of ease/difficulty suggest that the search is another difficulty in SR that needs much exploration. The third proposed stage to benefit from AI automation is the data extraction stage, 13 (21.7%). In Table B1, further comments on future suggestions for AI automation from respondents are indicated.

Based on the results for this aspect, it can be concluded that the title and abstract screening phase is the stage in the SR process reviewers find laborious, followed by the search/information retrieval and the data extraction phase. Hence, these results can inform and direct future AI automation methods rather than from estimations.

5 Systematic Review Dataset Repositories and Code

This section highlights some readily available datasets and repositories used for building and testing these SR automation methods in SE and medicine, which will be a starting point for future research. Almost all the dataset falls within the abstract and title screening domain, whilst few are in the other stages. Below is a list of these datasets:

1. **ASReview Repository** is a compilation of some title and abstract datasets within the medicine and SE discipline readily available on Github[9]. Table 11 shows a summary of these datasets within this repository. Four of the 26 available datasets are related to the SE domain, while the rest are related to healthcare for humans and animals. The size of datasets in the repository varies greatly, from as few as 310 papers (Antihistamines) to over 10,000 (Anxiety-Related Disorders). Larger datasets may

[9] https://github.com/asreview/systematic-review-datasets

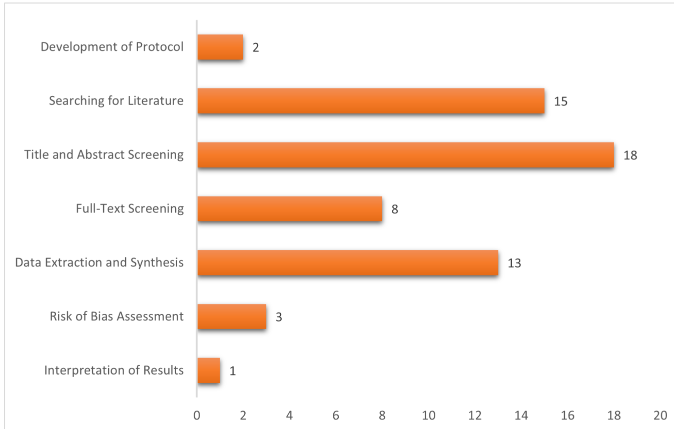


Fig. 9: Stage in the SR process proposed by participants where future AI automation would greatly benefit.

provide more robust training opportunities for machine learning models, while smaller datasets might not be as effective.

Analysis and comparison of the datasets AsReview Repository: The analysis and comparison of the datasets in the AsReview Repository reveal a class imbalance issue, as seen in Table 11. Various methods have been used to solve this issue before the algorithms are trained with data; however, further exploration of other class imbalance techniques is needed. In Table 12, where a comparison table is presented, the results of WSS@95 reported for experiments run on Table 11 are compiled with respect to three categories of methods proposed for the screening stage (text classification, screening prioritisation, and active learning). All proposed methods, text classification, screening prioritisation, and active learning, substantially gave positive results for WSS. It was noticed that the best-performing method across most of the datasets in Table 12 was the text classification approach, followed by screening prioritisation. An inference that can be drawn is that most text classification approaches, such as the study done by Timsina et al (2015), aimed at improving precision while maintaining a high recall, indeed helped increase the WSS@95 value. Nonetheless, no comparative analysis has been done on these similar datasets with LLMs, which is a future direction for future AI automation methods. Although no other comparative studies were found aside from Yu et al (2018) on the four SE data, the values of the WSS@95 were high. An exciting deduction that can be made from the study's aim stated in Section 3.2.3 was to find a faster AL technique compared to all the state-of-the-art approaches. The results showed that might indeed be valid. A future study could look at their proposed AL method on these health datasets instead of the SE dataset to explore its potential to reduce human burden.

2. **The TREC Track Repository**[10] comprises of benchmark datasets used for information retrieval tasks. In SR, the TREC Precision Medicine (PM) dataset is the used data for training learning models for automating the search stage. The PM TREC used for automating the SR search is the 2018. Soto et al (2018) partitioned into 2017 and 2018 datasets[11] containing 50 queries each. The TREC (PM) dataset is a collection of data and queries used in the TREC Precision Medicine track. It typically consists of queries that are clinically motivated questions, resembling the information needs of physicians. It also consists of a large set of documents that the search algorithms use to find relevant information. These documents can include scientific articles, clinical trial reports, and other related medical texts. Additionally, it consists of relevance judgments that are used to evaluate the performance of search systems which assess how well the documents retrieved by a search query meet the information need expressed in that query.

[10] https://trec.nist.gov/data.html
[11] https://trec.nist.gov/pubs/trec27/trec2018.html

Table 11: Summary of existing public title and abstracts screening dataset

Dataset ID	Topic	Total number of papers	Number included	Imbalance Ratio (IR)
Appenzeller-Herzog_2020	Wilson disease	3453	29	1: 118.07
Bannach-Brown_2019	Animal Model of Depression	1993	280	1: 6.12
Bos_2018	Dementia	5746	11	1: 521.36
Cohen_2006_ACEInhibitors	ACEInhibitors	2544	41	1: 61.05
Cohen_2006_ADHD	ADHD	851	20	1: 41.55
Cohen_2006_Antihistamines	Antihistamines	310	16	1: 18.38
Cohen_2006_AtypicalAntipsychotics	Atypical Antipsychotics	1120	146	1: 6.67
Cohen_2006_BetaBlockers	Beta Blockers	2072	42	1: 48.33
Cohen_2006_CalciumChannelBlockers	Calcium Channel Blockers	1218	100	1: 11.18
Cohen_2006_Estrogens	Estrogens	368	80	1: 3.60
Cohen_2006_NSAIDS	NSAIDS	393	41	1: 8.59
Cohen_2006_Opiods	Opiods	1915	15	1: 126.67
Cohen_2006_OralHypoglycemics	Oral Hypoglycemics	503	136	1: 2.70
Cohen_2006_ProtonPumpInhibitors	Proton Pump Inhibitors	1333	51	1: 25.14
Cohen_2006_SkeletalMuscleRelaxants	Skeletal Muscle Relaxants	1643	9	1: 181.56
Cohen_2006_Statins	Statins	3465	85	1: 39.76
Cohen_2006_Triptans	Triptans	671	24	1: 26.96
Cohen_2006_UrinaryIncontinence	Urinary Incontinence	327	40	1: 7.18
Hall_2012	Software Fault Prediction	8911	104	1: 84.68
Kitchenham_2010	Software Engineering	1704	45	1: 36.87
Kwok_2020	Virus Metagenomics	2481	120	1: 19.68
Nagtegaal_2019	Nudging	2019	101	1: 19.99
Radjenovic_2013	Software Fault Prediction	6000	48	1: 124.00
Wahono_2015	Software Defect Detection	7002	62	1: 111.94
Wolters_2018	Dementia	5019	19	1: 263.16
van_Dis_2020	Anxiety-Related Disorders	10953	73	1: 149.04

3. **LitCovid Hub**[12] is a readily available dataset of up-to-date scientific facts about the COVID-19 pandemic. This dataset is found in LitCovid, a curated literature hub. The dataset is updated daily as new articles related to COVID-19 are indexed in PubMed. This dataset was used by Simon et al (2019) to evaluate their proposed algorithms for automating the search stage.
4. **EBM-NLP dataset**[13] developed by Nye et al (2018) is the only readily available dataset with explicitly recognised PICO elements. This dataset contains approximately 4,993 annotated abstracts of PICO elements of medical journals outlining clinical trials. Since the annotation of the PICO is done on the abstract and not in full text, challenges may arise for journals with the PICO elements in the full text.

All the public codes found in the related studies are summarised in Table 13.

6 Gaps and recommendations

6.1 From Literature review

Putting it all together, from the 52 identified papers targeting the automation of the search, title and abstract screening, and data extraction, this section highlights the gap found and provides recommendations for the future. To begin, a wide gap was noticed in using large language models (LLMs) for SR automation. In Table 3, 4, 5, 6, 7, 8, 10 where all the related works are summarised with respect to the natural language processing (NLP) pipeline, it is clear that only a few studies have explored the use of LLMs for SR automation primarily for the title and abstract screening and data extraction phase (Hasny et al, 2023; Norman et al, 2019; Schmidt et al, 2020). Despite the growing prevalence of LLMs, their application in SR automation remains relatively nascent. These models can potentially redefine key SR stages such as title and abstract screening, search, data extraction, risk of bias (RoB) assessment, and

[12] https://www.ncbi.nlm.nih.gov/research/coronavirus/
[13] https://github.com/bepnye/EBM-NLP

Table 12: Comparison of proposed methods across the existing public datasets

Dataset ID	Task Type	Method	WSS@95
Cohen_2006_ACEInhibitors	Text classification	(Cohen et al, 2006)	0.56
	Text classification	(Timsina et al, 2015)	0.78
	Screening Prioritisation	(Howard et al, 2016)	0.80
	Text classification	(Olorisade et al, 2019)	0.81
	Active Learning	(Howard et al, 2020)	0.75
Cohen_2006_ADHD	Text classification	(Cohen et al, 2006)	0.68
	Screening Prioritisation	(Howard et al, 2016)	0.79
	Text classification	(Olorisade et al, 2019)	0.70
	Active Learning	(Howard et al, 2020)	0.74
Cohen_2006_Antihistamines	Text classification	(Cohen et al, 2006)	0.00
	Screening Prioritisation	(Howard et al, 2016)	0.13
	Text classification	(Timsina et al, 2015)	0.22
	Text classification	(Olorisade et al, 2019)	0.01
	Active Learning	(Howard et al, 2020)	0.07
Cohen_2006_AtypicalAntipsychotics	Text classification	(Cohen et al, 2006)	0.14
	Screening Prioritisation	(Howard et al, 2016)	0.49
	Text classification	(Olorisade et al, 2019)	0.18
	Active Learning	(Howard et al, 2020)	0.17
Cohen_2006_BetaBlockers	Text classification	(Cohen et al, 2006)	0.28
	Screening Prioritisation	(Howard et al, 2016)	0.43
	Text classification	(Olorisade et al, 2019)	0.47
	Active Learning	(Howard et al, 2020)	0.59
Cohen_2006_CalciumChannelBlockers	Text classification	(Cohen et al, 2006)	0.12
	Screening Prioritisation	(Howard et al, 2016)	0.45
	Text classification	(Howard et al, 2016)	0.24
	Active Learning	(Olorisade et al, 2019)	0.56
Cohen_2006_Estrogens	Text classification	(Cohen et al, 2006)	0.18
	Screening Prioritisation	(Howard et al, 2016)	0.47
	Text classification	(Olorisade et al, 2019)	0.25
	Active Learning	(Howard et al, 2020)	0.45
Cohen_2006_NSAIDS	Text classification	(Cohen et al, 2006))	0.50
	Screening Prioritisation	(Howard et al, 2016)	0.73
	Text classification	(Olorisade et al, 2019)	0.37
	Active Learning	(Howard et al, 2020)	0.62
Cohen_2006_Opiods	Text classification	(Cohen et al, 2006)	0.13
	Screening Prioritisation	(Howard et al, 2016)	0.83
	Text classification	(Olorisade et al, 2019)	0.61
	Active Learning	(Howard et al, 2020)	0.26
Cohen_2006_OralHypoglycemics	Text classification	(Cohen et al, 2006)	0.89
	Screening Prioritisation	(Howard et al, 2016)	0.11
	Text classification	(Olorisade et al, 2019)	0.04
	Active Learning	(Howard et al, 2020)	0.09
Cohen_2006_ProtonPumpInhibitors	Text classification	(Cohen et al, 2006)	0.28
	Screening Prioritisation	(Howard et al, 2016)	0.38
	Text classification	(Olorisade et al, 2019)	0.27
	Active Learning	(Howard et al, 2020)	0.40
Cohen_2006_SkeletalMuscleRelaxants	Text classification	(Cohen et al, 2006)	0.00
	Text classification	(Timsina et al, 2015)	0.72
	Screening Prioritisation	(Howard et al, 2016)	0.56
	Text classification	(Olorisade et al, 2019)	0.01
	Active Learning	(Howard et al, 2020)	0.29
Cohen_2006_Statins	Text classification	(Cohen et al, 2006)	0.25
	Screening Prioritisation	(Howard et al, 2016)	0.45
	Text classification	(Olorisade et al, 2019)	0.18
	Active Learning	(Howard et al, 2020)	0.40
Cohen_2006_Triptans	Text classification	(Cohen et al, 2006)	0.34
	Screening Prioritisation	(Howard et al, 2016)	0.41
	Text classification	(Olorisade et al, 2019)	0.03
	Active Learning	(Howard et al, 2016)	0.46
Cohen_2006_UrinaryIncontinence	Text classification	(Cohen et al, 2006)	0.26
	Screening Prioritisation	(Howard et al, 2016)	0.53
	Text classification	(Olorisade et al, 2019)	0.28
	Active Learning	(Howard et al, 2020)	0.41
Hall_2012	Active learning	(Yu et al, 2018)	0.91
Kitchenham_2010	Active learning	(Yu et al, 2018)	0.58
Radjenovic_2013	Active learning	(Yu et al, 2018)	0.85
Wahono_2015	Active learning	(Yu et al, 2018)	0.85

Table 13: Publicly available codes from related studies

Reference	Code availability (If https is not at the beginning, it implies that it is under github.com)
(Wallace et al, 2010)	bwallace/abstrackr-web
(Mergel et al, 2015)	gmergel/SLR.qub
(Almeida et al, 2016)	TsangLab
(Marshall et al, 2016)	ijmarshall/robotreviewer
(Marshall et al, 2018)	ijmarshall/robotsearch
(Yu et al, 2018)	fastread/src
(Kontonatsios et al, 2020)	gkontonatsios/DAE-FF
(van de Schoot et al, 2021)	1. https://zenodo.org/record/6258041#.YkRv-XrMLIW 2. asreview/asreview
(Hasny et al, 2023)	3. /ESA-RadLab/BERTCSRS

even the synthesis of findings by leveraging their deep contextual understanding. Thus, future research could explore how transformer models can be fine-tuned for these tasks.

Additionally, one general challenge identified across all the stages from the related works is the varying effectiveness of NLP techniques based on the specificity of the SR topic at hand. In Table 2, an approach used for handling this is domain knowledge integration, which includes feature enrichment methods such as the addition of MeSH headings, publication tags, and concatenation of UMLS embeddings with abstract embeddings, among others. In the other related studies that deployed state-of-the-art LLMs, variants of BERT pre-trained on medical domain corpora like SciBERT, PubMedBERT, and BioBERT were used as domain adaptability and knowledge integration. However, reported studies have shown that these LLMs are unable to capture medical concepts and terms required for biomedical data and treat these key terms as ordinary tokens (Xie et al, 2022). Additionally, since these LLMs were trained on the free biomedical corpus, they lack specific structured domain knowledge essential for biomedical domain tasks (Xie et al, 2022). This opens up an area of exploration on domain integration into LLMs for SR automation as a stand-alone together with human feedback in active learning methods (human-in-the-loop).

Discussing the automation of the search phase of SR, a prevalence of proposed methods such as text classification, information retrieval with and without visualisation (VTM), and information extraction was observed. For example, Cohen et al (2015) utilised search prioritisation, employing SVM to rank citations in a large dataset. Although effective in prioritising relevant studies, this technique showed limitations in processing complex queries. Similarly, Marshall et al (2018) and Allot et al (2021) applied text classification techniques, integrating CNN and SVM to classify citations. Despite their effectiveness in narrowing search results, these approaches still grapple with the challenge of accurately handling diverse and nuanced SR research topics. Future works can explore the use of LLMs for these tasks in terms of query generation and expansion for SR automation, as they are pre-trained in a broader range of datasets and thus can handle complex queries and provide more nuanced search results, overcoming limitations of traditional methods (Alaofi et al, 2023). Furthermore, summarising the main challenges associated with the text classification technique for the search stage, some identified studies were limited to automating publication from only PubMed, excluding articles or abstracts not indexed in PubMed and non-peer-reviewed publications. Other studies also focused on automating searches for only randomised controlled trials (RCTs). Thus, future works may be to find appropriate methodologies that may be examined to automate the search phase beyond PubMed or RCTs. Moving on to the abstract and screening stage, most studies deployed as tools use active learning. Recapitulating the main associated challenges aside from the use of LLMs and domain knowledge integration, is finding the apt threshold for a reviewer to stop screening. Only two studies under active learning-related studies have sought to address this. This, therefore, opens an exploration of further advanced statistical approaches to solve this issue, providing a user with the threshold at which screening can be stopped.

For data extraction and the RoB phase, the NLP methods are still in a nascent stage. Kiritchenko et al (2010) and Bui et al. Bui et al (2016) explored SVM for extracting data from texts, highlighting the potential of NLP in identifying key study elements like PICO. In automating the RoB assessment, Marshall et al (2016, 2017) utilised an ensemble of CNN and SVM and rule-based methods, indicating the feasibility of NLP in this domain. However, this area remains relatively unexplored and ripe for further development. Thus, the potential of LLMs in this area is immense. By training these models on datasets and incorporating domain-specific heuristics, LLMs can automate the extraction of complex data elements like PICO, and assess RoB with greater accuracy. Additionally, it was observed that studies that focused on automating the data extraction phase treated it as a sentence classification task. A future recommendation will be to explore this task as a question and answering task as the latter is built for contextual understanding and response to specific queries and to reduce ambiguity Rogers et al (2023). Furthermore, as seen in Section 3 and Table 10, few studies have targeted the data extraction stage. Yet, in Figure B4 and Table B1, it is seen that this is one necessity for SR reviewers in the review process. As such, future automation studies may need to target this stage. Finally, in automating the RoB, the two related works focused on RCTs; thus, such automation needs to be extended to non-RCTs. Another novel area of exploration could be exploring how the human-in-the-loop strategy, active learning, might help in RoB classification.

Also, one significant observation to be realised across all the related studies is that all focused on only English datasets except for Schmidt et al (2020); thus, current SR automation studies are skewed towards English datasets. This opens a novel field of exploring which concepts will best automate either partially or fully non-English SRs. The result that most of the existing NLP methods in Section **??** proposed for SR automation are predominantly focused on English language datasets overlooks the rich and diverse body of non-English scientific literature, which is crucial for comprehensive global SRs. Thus, developing and refining NLP algorithms that cater to multilingual datasets is an imperative frontier. This includes training models on diverse linguistic datasets and developing language-agnostic models capable of processing and analysing research in multiple languages effectively. Such advancements would significantly broaden the scope and inclusivity of SRs, ensuring a more global representation in research synthesis. Similarly, regarding available datasets for SR automation, there is still the need to develop more public datasets beyond the screening stage, specifically for the other automation stages such as data extraction, RoB, and the search phase. To the best of my knowledge, there exists only one publicly available dataset readily available for PICO data extraction synthesis (EBM-PICO) in English. As such, there is a need for the development of diverse, publicly available datasets that encompass the full scope of SR automation. These datasets should include varied SR research topics, multiple languages, and different types of studies to enhance the robustness and generalisation of future AI SR automation models.

Finally, in the data extraction stage, it was noticed that there is currently no evidence of data extraction in images that may be present in the articles; hence, this provides a future gap for further development in future AI automation tools. A significant proportion of valuable data in scientific articles is often encapsulated in images, graphs, and tables. Current NLP techniques predominantly focus on text analysis, leaving a gap in extracting and interpreting data presented visually. The development of NLP methods integrated with image processing algorithms could unlock this untapped data source. This integration would enable the extraction of quantitative data from graphical representations, the conversion of table data into analysable formats, and even the interpretation of complex images like medical imaging reports. Such a holistic approach to data extraction would enhance the comprehensiveness and depth of SRs, especially in fields where visual data plays a pivotal role.

6.2 Conclusion and practical insights from the survey

Overall, the survey sought to provide insights into the current state of AI tool automation usage in SR, the challenges faced by reviewers, and potential areas for future development and improvement. Integrating the insights from your survey with the literature review to provides a comprehensive understanding of the current state and possible areas for improvement in AI methods for systematic review (SR) automation for the search phase, in Table B1, part of the challenges raised by the SR reviewers, is handling diverse search queries, which aligns with the literature's identified limitations. Thus, there is a need for more advanced AI methods that can handle the complexity and variability of research topics. Though the abstract screening phase is the most automated phase, the survey results show that this is a major need for most SR practitioners. Similarly, though techniques for data extraction and risk of bias assessment, such as those proposed by Kiritchenko et al (2010) and Bui et al (2016), participants find data extraction still particularly burdensome, indicating an area where current literature falls short. It suggests a need for more sophisticated NLP techniques capable of accurately extracting and synthesising data from diverse sources. This highlights a significant opportunity for developing NLP methods specifically tailored for RoB assessment. Finally, the survey reveals potential areas for AI Automation development from the point of view of SR reviewers; the title and abstract screening, followed by the search phase and data extraction, as potential areas where AI automation will be most beneficial. This feedback can direct future research and development ensuring that the development of AI tools for SR is aligned with the actual needs of researchers and practitioners in the field rather than from estimation.

Overall, the role of AI in automating SR indeed possesses numerous advantages.

7 Limitation of this study

While the study presents a comprehensive review of existing AI methods for SR automation, the literature included primarily provided information on SR health sciences, software engineering domains up until 2023. The findings and recommendations might not be fully applicable to SR in other fields with different types of data or research methodologies. Additionally, the study does not provide an overview of papers that deployed ChatGPT as an automation technique as our selection criteria was based on papers with detailed explanation on its AI methodology. Furthermore, with the rapidly evolving field of AI, the methods and tools discussed in this study might quickly become outdated as new advancements emerge. This limitation may affect the long-term applicability of the study's findings. Finally, the AI methods and tools discussed primarily focus on English language datasets. This limits applicability to systematic reviews involving non-English sources or multilingual datasets.

8 Conclusion

In conclusion, this review paper provided a comprehensive overview of the current AI methods, including NLP, ML, and DL, that are employed to automate various stages of the SR process. Through an extensive analysis of 52 related works identified from our search, we found that most studies focused on automating the screening stage, followed by the search, data extraction, and risk of bias (RoB) assessment stages. To complement the literature review, we conducted an original online survey to gather practical insights from SR practitioners and researchers regarding their experiences, opinions, and expectations for future AI-driven SR automation. By synthesising the findings from both the literature review and the survey results, we identified key gaps and challenges in the current landscape of SR automation using AI techniques. Based on these findings, we discussed potential future directions to bridge the identified gaps, such as exploring the application of LLMs for various SR stages, integrating domain knowledge into AI models, developing multilingual datasets and language-agnostic models, and incorporating image processing techniques for data extraction from visual representations in scientific literature. This review aimed to provide researchers and practitioners with a foundational understanding of the basic concepts, primary methodologies, and recent advancements in AI-driven SR automation. By highlighting the current state, limitations, and prospects, we anticipate that this work will not only aid non-technical researchers in comprehending the application of AI in SR automation but also guide computer scientists in exploring novel techniques to invigorate further and advance this field.

9 Acknowledgement

The authors would like to thank members of the COMO project [14] for supporting this research.

10 Conflict of Interest

The authors declare that they have no conflict of interest.

11 Supplementary Files

Additional File 1 and 2 contains the full details of the survey (questions and results).

[14] https://www.comoprojectmx

References

Abramovich F, Grinshtein V, Levy T (2021) Multiclass classification by sparse multinomial logistic regression. IEEE Transactions on Information Theory 67(7):4637–4646. https://doi.org/10.1109/tit.2021.3075137, URL http://dx.doi.org/10.1109/tit.2021.3075137

Aceves-Martins M, López-Cruz L, García-Botello M, et al (2021) Interventions to prevent obesity in mexican children and adolescents: Systematic review. Prevention Science 23(4):563–586. https://doi.org/10.1007/s11121-021-01316-6, URL http://dx.doi.org/10.1007/s11121-021-01316-6

Ahmed M, Seraj R, Islam SMS (2020) The k-means algorithm: A comprehensive survey and performance evaluation. Electronics 9(8):1295. https://doi.org/10.3390/electronics9081295, URL http://dx.doi.org/10.3390/electronics9081295

AHO AV (1990) Algorithms for Finding Patterns in Strings, Elsevier, p 255–300. https://doi.org/10.1016/b978-0-444-88071-0.50010-2, URL http://dx.doi.org/10.1016/b978-0-444-88071-0.50010-2

Aklouche B, Bounhas I, Slimani Y (2018) Query expansion based on nlp and word embeddings. In: Text Retrieval Conference, URL https://api.semanticscholar.org/CorpusID:155085448

Aklouche B, Bounhas I, Slimani Y (2019) Automatic query reweighting using co-occurrence graphs. In: Proceedings of the 16th International Conference on Applied Computing 2019. IADIS Press, AC 2019, https://doi.org/10.33965/ac2019_201912l005, URL http://dx.doi.org/10.33965/ac2019_201912l005

Alaofi M, Gallagher L, Sanderson M, et al (2023) Can generative llms create query variants for test collections? an exploratory study. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, SIGIR '23, https://doi.org/10.1145/3539618.3591960, URL http://dx.doi.org/10.1145/3539618.3591960

Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology (ICET), pp 1–6, https://doi.org/10.1109/ICEngTechnol.2017.8308186

Allot A, Lee K, Chen Q, et al (2021) Litsuggest: A web-based system for literature recommendation and curation using machine learning. Nucleic Acids Research 49:W352–W358. https://doi.org/10.1093/nar/gkab326

Almeida H, Meurs MJ, Kosseim L, et al (2016) Data sampling and supervised learning for hiv literature screening. IEEE transactions on nanobioscience 15(4):354–361. URL https://doi.org/10.1109/bibm.2015.7359733

Angluin D (1988) Queries and concept learning. Machine Learning 2:319–342. URL https://api.semanticscholar.org/CorpusID:11357867

Aromataris E, Pearson A (2014) The systematic review: An overview. American Journal of Nursing 114(3):53–58. https://doi.org/10.1097/01.NAJ.0000444496.24228.2c

August ST (2001) Active Learning : Theory and Applications. Stanford University 13(4):182

Bannach-Brown A, Przybyła P, Thomas J, et al (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. URL https://doi.org/10.1186/s13643-019-0942-7

Baranwal A, Bagwe BR, M V (2022) Machine Learning in Python: Diabetes Prediction Using Machine Learning, IGI Global, p 882–908. https://doi.org/10.4018/978-1-6684-6291-1.ch046, URL http://dx.doi.org/10.4018/978-1-6684-6291-1.ch046

Bekhuis T, Demner-Fushman D (2012) Screening nonrandomized studies for medical systematic reviews: a comparative study of classifiers. Artificial intelligence in medicine 55(3):197–207. URL https://doi.org/10.1016/j.artmed.2012.05.002

Blaizot A, Veettil SK, Saidoung P, et al (2022) Using artificial intelligence methods for systematic review in health sciences: A systematic review. Research Synthesis Methods 13(3):353–362. https://doi.org/10.1002/jrsm.1553, URL http://dx.doi.org/10.1002/jrsm.1553

Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. Journal of Machine Learning Research 3(null):993–1022

Booth A, Sutton A, Papaioannou D (2016) Systematic Approaches to a Successful Literature Review (2nd ed.). Sage

Borah R, Brown AW, Capers PL, et al (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):1–7. https://doi.org/10.1136/bmjopen-2016-012545, URL https://doi.org/10.1136/bmjopen-2016-012545

Bornmann L, Mutz R (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):2215–2222. https://doi.org/10.1002/asi.23329, URL https://doi.org/10.48550/arXiv.1402.4578, arXiv:1402.4578

Bui DDA, Jonnalagadda S, Del Fiol G (2015) Automatically finding relevant citations for clinical guideline development. Journal of Biomedical Informatics 57:436–445. https://doi.org/10.1016/j.jbi.2015.09.003, URL http://dx.doi.org/10.1016/j.jbi.2015.09.003

Bui DDA, Fiol GD, Hurdle JF, et al (2016) Extractive text summarization system to aid data extraction from full text in systematic review development. Journal of Biomedical Informatics 64:265–272. https://doi.org/10.1016/j.jbi.2016.10.014, URL https://doi.org/10.1016%2Fj.jbi.2016.10.014

Cawley M, Beardslee R, Beverly B, et al (2020) Novel text analytics approach to identify relevant literature for human health risk assessments: A pilot study with health effects of in utero exposures. Environment International 134:105228. https://doi.org/10.1016/j.envint.2019.105228, URL http://dx.doi.org/10.1016/j.envint.2019.105228

Cessie SL, Houwelingen JCV (1992) Ridge estimators in logistic regression. Applied Statistics 41(1):191. https://doi.org/10.2307/2347628, URL http://dx.doi.org/10.2307/2347628

Chai KE, Lines RL, Gucciardi DF, et al (2021) Research Screener: a machine learning tool to semi-automate abstract screening for systematic reviews. Systematic Reviews 10(1):1–13. https://doi.org/10.1186/s13643-021-01635-3

Chen Q, Allot A, Lu Z (2020) Litcovid: an open database of covid-19 literature. Nucleic Acids Research 49(D1):D1534–D1540. https://doi.org/10.1093/nar/gkaa952, URL http://dx.doi.org/10.1093/nar/gkaa952

Cheng SH, Augustin C, Bethel A, et al (2018) Using machine learning to advance synthesis and use of conservation and environmental evidence. https://doi.org/10.1111/cobi.13117

Chiu B, Crichton G, Korhonen A, et al (2016) How to train good word embeddings for biomedical nlp. In: Proceedings of the 15th Workshop on Biomedical Natural Language Processing. Association for Computational Linguistics, https://doi.org/10.18653/v1/w16-2922, URL http://dx.doi.org/10.18653/v1/w16-2922

Cho K, van Merrienboer B, Gulcehre C, et al (2014) Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, https://doi.org/10.3115/v1/d14-1179, URL http://dx.doi.org/10.3115/v1/d14-1179

Cohen AM, Hersh WR, Peterson K, et al (2006) Reducing workload in systematic review preparation using automated citation classification. Journal of the American Medical Informatics Association 13(2):206–219. https://doi.org/10.1197/jamia.m1929, URL https://doi.org/10.1197%2Fjamia.m1929

Cohen AM, Ambert K, McDonagh M (2009) Cross-topic learning for work prioritization in systematic review creation and update. Journal of the American Medical Informatics Association 16(5):690–704. https://doi.org/10.1197/jamia.m3162, URL http://dx.doi.org/10.1197/jamia.m3162

Cohen AM, Smalheiser NR, McDonagh MS, et al (2015) Automated confidence ranked classification of randomized controlled trial articles: an aid to evidence-based medicine. Journal of the American Medical Informatics Association 22(3):707–717. https://doi.org/10.1093/jamia/ocu025, URL http://dx.doi.org/10.1093/jamia/ocu025

Cohn D, Atlas L, Ladner R (1994) Improving generalization with active learning. Machine Learning 15(2):201–221. https://doi.org/10.1007/bf00993277, URL http://dx.doi.org/10.1007/bf00993277

Cormack GV, Grossman MR (2014) Evaluation of machine-learning protocols for technology-assisted review in electronic discovery. In: Proceedings of the 37th international ACM SIGIR conference on Research and development in information retrieval. ACM, SIGIR '14, https://doi.org/10.1145/2600428.2609601, URL http://dx.doi.org/10.1145/2600428.2609601

Cortes C, Vapnik V (1995) Support-vector networks. Machine learning 20(3):273–297

Davis J, Mengersen K, Bennett S, et al (2014) Viewing systematic reviews and meta-analysis in social research through different lenses. SpringerPlus 3(1). https://doi.org/10.1186/2193-1801-3-511, URL http://dx.doi.org/10.1186/2193-1801-3-511

Devlin J, Chang MW, Lee K, et al (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. 1810.04805

van Dinter R, Tekinerdogan B, Catal C (2021) Automation of systematic literature reviews: A systematic literature review. Information and Software Technology 136:106589. URL https://doi.org/10.1016/j.infsof.2021.106589

Egger M, George Davey Smith KO (2001) Systematic reviews in health care: meta-analysis in context(2nd ed.) p.9-12. Dover

Felizardo KR, Andery GF, Paulovich FV, et al (2012) A visual analysis approach to validate the selection review of primary studies in systematic reviews. Information and Software Technology 54(10):1079–1091. https://doi.org/10.1016/j.infsof.2012.04.003, URL http://dx.doi.org/10.1016/j.infsof.2012.04.003

Frunza O, Inkpen D, Matwin S, et al (2011a) Exploiting the systematic review protocol for classification of medical abstracts. Artificial intelligence in medicine 51(1):17–25. URL https://doi.org/10.1016/j.artmed.2010.10.005

Frunza O, Inkpen D, Matwin S, et al (2011b) Exploiting the systematic review protocol for classification of medical abstracts. Artificial Intelligence in Medicine 51(1):17–25. https://doi.org/10.1016/j.artmed.2010.10.005, URL http://dx.doi.org/10.1016/j.artmed.2010.10.005

Gates A, Johnson C, Hartling L (2018) Technology-assisted title and abstract screening for systematic reviews: A retrospective evaluation of the Abstrackr machine learning tool. Systematic Reviews 7(1):1–9. https://doi.org/10.1186/s13643-018-0707-8, URL https://doi.org/10.1186/s13643-018-0707-8

Gonzalez-Toral S, Freire R, Gualan R, et al (2019) A ranking-based approach for supporting the initial selection of primary studies in a systematic literature review. In: 2019 XLV Latin American Computing Conference (CLEI). IEEE, https://doi.org/10.1109/clei47609.2019.235079, URL http://dx.doi.org/10.1109/clei47609.2019.235079

Gosavi A (2009) Reinforcement learning: A tutorial survey and recent advances. INFORMS Journal on Computing 21(2):178–192. https://doi.org/10.1287/ijoc.1080.0305, URL http://dx.doi.org/10.1287/ijoc.1080.0305

Gulo CA, Rúbio TR, Tabassum S, et al (2015) Mining scientific articles powered by machine learning techniques. In: 2015 Imperial College computing student workshop (ICCSW 2015), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, URL https://doi.org/10.4230/OASIcs.ICCSW.2015.21

Guo G, Wang H, Bell D, et al (2003) KNN Model-Based Approach in Classification, Springer Berlin Heidelberg, p 986–996. https://doi.org/10.1007/978-3-540-39964-3_62, URL http://dx.doi.org/10.1007/978-3-540-39964-3_62

Hans C (2011) Elastic net regression modeling with the orthant normal prior. Journal of the American Statistical Association 106(496):1383–1393. https://doi.org/10.1198/jasa.2011.tm09241, URL http://dx.doi.org/10.1198/jasa.2011.tm09241

Hashimoto K, Kontonatsios G, Miwa M, et al (2016) Topic detection using paragraph vectors to support active learning in systematic reviews. Journal of Biomedical Informatics 62:59–65. https://doi.org/10.1016/j.jbi.2016.06.001, URL http://dx.doi.org/10.1016/j.jbi.2016.06.001

Hasny M, Vasile AP, Gianni M, et al (2023) BERT for Complex Systematic Review Screening to Support the Future of Medical Research, Springer Nature Switzerland, p 173–182. https://doi.org/10.1007/978-3-031-34344-5_21, URL http://dx.doi.org/10.1007/978-3-031-34344-5_21

Higgins JPT, Altman DG, Gotzsche PC, et al (2011) The cochrane collaboration's tool for assessing risk of bias in randomised trials. BMJ 343(oct18 2):d5928–d5928. https://doi.org/10.1136/bmj.d5928, URL http://dx.doi.org/10.1136/bmj.d5928

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Computation 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735, URL http://dx.doi.org/10.1162/neco.1997.9.8.1735

Hoi SCH, Jin R, Lyu MR (2006) Large-scale text categorization by batch mode active learning. In: Proceedings of the 15th international conference on World Wide Web. ACM, WWW06, https://doi.org/10.1145/1135777.1135870, URL http://dx.doi.org/10.1145/1135777.1135870

Howard BE, Phillips J, Miller K, et al (2016) Swift-review: a text-mining workbench for systematic review. Systematic Reviews 5(1). https://doi.org/10.1186/s13643-016-0263-z, URL http://dx.doi.org/10.1186/s13643-016-0263-z

Howard BE, Phillips J, Tandon A, et al (2020) SWIFT-Active Screener: Accelerated document screening through active learning and integrated recall estimation. Environment International 138(April 2019):105623. https://doi.org/10.1016/j.envint.2020.105623, URL https://doi.org/10.1016/j.envint.2020.105623

Iparragirre A, Barrio I, Aramendi J, et al (2023) Estimation of logistic regression parameters for complex survey data: a real data based simulation study. 2303.01754

Jaspers S, De Troyer E, Aerts M (2018) Machine learning techniques for the automation of literature reviews and systematic reviews in efsa. EFSA Supporting Publications 15(6). https://doi.org/10.2903/sp.efsa.2018.en-1427, URL http://dx.doi.org/10.2903/sp.efsa.2018.en-1427

Jelodar H, Wang Y, Yuan C, et al (2018) Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey. 1711.04305

Jha KK, Jha R, Jha AK, et al (2021) A brief comparison on machine learning algorithms based on various applications: A comprehensive survey. In: 2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS). IEEE, https://doi.org/10.1109/csitss54238.2021.9683524, URL http://dx.doi.org/10.1109/csitss54238.2021.9683524

Joachims T (2006) Training linear svms in linear time. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, KDD06, https://doi.org/10.1145/1150402.1150429, URL http://dx.doi.org/10.1145/1150402.1150429

Jolliffe I (2014) Principal component analysis. https://doi.org/10.1002/9781118445112.stat06472, URL http://dx.doi.org/10.1002/9781118445112.stat06472

Kaelbling LP, Littman ML, Moore AW (1996) Reinforcement learning: A survey. cs/9605103

Khalil H, Ameen D, Zarnegar A (2022) Tools to support the automation of systematic reviews: a scoping review. Journal of Clinical Epidemiology 144:22–42. https://doi.org/10.1016/j.jclinepi.2021.12.005, URL http://dx.doi.org/10.1016/j.jclinepi.2021.12.005

Kiritchenko S, de Bruijn B, Carini S, et al (2010) ExaCT: automatic extraction of clinical trial characteristics from journal publications. BMC Medical Informatics and Decision Making 10(1). https://doi.org/10.1186/1472-6947-10-56, URL https://doi.org/10.1186%2F1472-6947-10-56

Kitchenham B, Brereton OP, Budgen D, et al (2009) Systematic literature reviews in software engineering – a systematic literature review. Information and Software Technology 51(1):7–15. https://doi.org/10.1016/j.infsof.2008.09.009, URL https://doi.org/10.1016%2Fj.infsof.2008.09.009

Klein D, Manning CD (2003) Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - ACL '03. Association for Computational Linguistics, ACL '03, https://doi.org/10.3115/1075096.1075150, URL http://dx.doi.org/10.3115/1075096.1075150

Kontonatsios G, Spencer S, Matthew P, et al (2020) Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews. Expert Systems with Applications: X 6:100030. https://doi.org/10.1016/j.eswax.2020.100030, URL https://doi.org/10.1016%2Fj.eswax.2020.100030

Kotsiantis SB (2011) Decision trees: a recent overview. Artificial Intelligence Review 39(4):261–283. https://doi.org/10.1007/s10462-011-9272-4, URL http://dx.doi.org/10.1007/s10462-011-9272-4

Lecun Y, Bottou L, Bengio Y, et al (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324. https://doi.org/10.1109/5.726791, URL http://dx.doi.org/10.1109/5.726791

Lewis DD (1998) Naive (Bayes) at forty: The independence assumption in information retrieval, Springer Berlin Heidelberg, p 4–15. https://doi.org/10.1007/bfb0026666, URL http://dx.doi.org/10.1007/bfb0026666

Mahendra MFR, Azizah NL (2023) Implementation of machine learning to predict the weather using a support vector machine: Implementasi machine learning untuk memprediksi cuaca menggunakan support vector machine. Preprint https://doi.org/10.21070/ups.2889, URL http://dx.doi.org/10.21070/

ups.2889

Marshall I, Kuiper J, Banner E, et al (2017) Automating biomedical evidence synthesis: Robotreviewer. In: Proceedings of ACL 2017, System Demonstrations. Association for Computational Linguistics, https://doi.org/10.18653/v1/p17-4002, URL https://doi.org/10.18653%2Fv1%2Fp17-4002

Marshall IJ, Wallace BC (2019) Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. Systematic Reviews 8(1). https://doi.org/10.1186/s13643-019-1074-9, URL https://doi.org/10.1186%2Fs13643-019-1074-9

Marshall IJ, Kuiper J, Wallace BC (2016) RobotReviewer: Evaluation of a system for automatically assessing bias in clinical trials. Journal of the American Medical Informatics Association 23(1):193–201. https://doi.org/10.1093/jamia/ocv044, URL https://doi.org/10.1093/jamia/ocv044

Marshall IJ, Noel-Storr A, Kuiper J, et al (2018) Machine learning for identifying randomized controlled trials: An evaluation and practitioner's guide. Research Synthesis Methods 9(4):602–614. https://doi.org/10.1002/jrsm.1287, URL http://dx.doi.org/10.1002/jrsm.1287

Marshall IJ, Nye B, Kuiper J, et al (2020) Trialstreamer: A living, automatically updated database of clinical trial reports. Journal of the American Medical Informatics Association 27(12):1903–1912. https://doi.org/10.1093/jamia/ocaa163, URL http://dx.doi.org/10.1093/jamia/ocaa163

McGreevy KM, Church FC (2020) Active learning survey. https://doi.org/10.1037/t81767-000, URL http://dx.doi.org/10.1037/t81767-000

Mergel GD, Silveira MS, da Silva TS (2015) A method to support search string building in systematic literature reviews through visual text mining. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing. ACM, SAC 2015, https://doi.org/10.1145/2695664.2695902, URL http://dx.doi.org/10.1145/2695664.2695902

Mitchell TM (1997) Machine Learning. McGraw-Hill, New York

Miwa M, Thomas J, O'Mara-Eves A, et al (2014) Reducing systematic review workload through certainty-based screening. Journal of Biomedical Informatics 51:242–253. https://doi.org/10.1016/j.jbi.2014.06.005, URL http://dx.doi.org/10.1016/j.jbi.2014.06.005

Moher D (2001) The consort statement: Revised recommendations for improving the quality of reports of parallel-group randomized trials. JAMA 285(15):1987. https://doi.org/10.1001/jama.285.15.1987, URL http://dx.doi.org/10.1001/jama.285.15.1987

Moreno-Garcia CF, Jayne C, Elyan E, et al (2023) A novel application of machine learning and zero-shot classification methods for automated abstract screening in systematic reviews. Decision Analytics Journal 6:100162. https://doi.org/10.1016/j.dajour.2023.100162, URL http://dx.doi.org/10.1016/j.dajour.2023.100162

Nadkarni PM (2002) An introduction to information retrieval: applications in genomics. The Pharmacogenomics Journal 2(2):96–102. https://doi.org/10.1038/sj.tpj.6500084, URL http://dx.doi.org/10.1038/sj.tpj.6500084

Natukunda A, Muchene LK (2023) Unsupervised title and abstract screening for systematic review: a retrospective case-study using topic modelling methodology. Systematic Reviews 12(1). https://doi.org/10.1186/s13643-022-02163-4, URL http://dx.doi.org/10.1186/s13643-022-02163-4

Norman C, Leeflang M, Spijker R, et al (2019) A distantly supervised dataset for automated data extraction from diagnostic studies. In: Proceedings of the 18th BioNLP Workshop and Shared Task. Association for Computational Linguistics, https://doi.org/10.18653/v1/w19-5012, URL http://dx.

doi.org/10.18653/v1/w19-5012

Nye B, Li JJ, Patel R, et al (2018) A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, https://doi.org/10.18653/v1/p18-1019, URL https://doi.org/10.18653%2Fv1%2Fp18-1019

Ofori-Boateng R, Aceves-Martins M, Jayne C, et al (2023) Evaluation of attention-based lstm and bi-lstm networks for abstract text classification in systematic literature review automation. Procedia Computer Science 222:114–126. https://doi.org/10.1016/j.procs.2023.08.149, URL http://dx.doi.org/10.1016/j.procs.2023.08.149

Olorisade BK, Brereton P, Andras P (2019) The use of bibliography enriched features for automatic citation screening. Journal of biomedical informatics 94:103202. URL https://doi.org/10.1016/j.jbi.2019.103202

Orel E, Ciglenecki I, Thiabaud A, et al (2023) An automated literature review tool (literev) for streamlining and accelerating research using natural language processing and machine learning: Descriptive performance evaluation study. J Med Internet Res 25:e39736. https://doi.org/10.2196/39736, URL https://www.jmir.org/2023/1/e39736

Ouzzani M, Hammady H, Fedorowicz Z, et al (2016) Rayyan-a web and mobile app for systematic reviews. Systematic Reviews 5(1):1–10. https://doi.org/10.1186/s13643-016-0384-4, URL http://dx.doi.org/10.1186/s13643-016-0384-4

O'Mara-Eves A, Thomas J, McNaught J, et al (2015) Using text mining for study identification in systematic reviews: a systematic review of current approaches. Systematic reviews 4(1):1–22. URL https://doi.org/10.1186/2046-4053-4-5

Paul L, Suman A, Sultan N (2013) Methodological analysis of principal component analysis (pca) method. International Journal of Computational Engineering and Management 16:32–38

Popuri SK (2022) An approximation method for fitted random forests. ArXiv abs/2207.02184. URL https://api.semanticscholar.org/CorpusID:250279991

Przybyła P, Brockmeier AJ, Kontonatsios G, et al (2018) Prioritising references for systematic reviews with RobotAnalyst: A user study. https://doi.org/10.1002/jrsm.1311, URL https://doi.org/10.1002/jrsm.1311

Radford A, Wu J, Child R, et al (2019) Language models are unsupervised multitask learners. OpenAI URL https://api.semanticscholar.org/CorpusID:160025533

Rogers A, Gardner M, Augenstein I (2023) Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. ACM Computing Surveys 55(10):1–45. https://doi.org/10.1145/3560260, URL http://dx.doi.org/10.1145/3560260

Ros R, Bjarnason E, Runeson P (2017) A machine learning approach for semi-automated search and selection in literature studies. In: Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering. ACM, EASE'17, https://doi.org/10.1145/3084226.3084243, URL http://dx.doi.org/10.1145/3084226.3084243

Rúbio TR, Gulo CA (2016) Enhancing academic literature review through relevance recommendation: using bibliometric and text-based features for classification. In: 2016 11th Iberian Conference on Information Systems and Technologies (CISTI), IEEE, pp 1–6, URL https://doi.org/10.1109/cisti.2016.7521620

Russell-Rose T, Chamberlain J, Shokraneh F (2019) A visual approach to query formulation for systematic search. In: Proceedings of the 2019 Conference on Human Information Interaction and Retrieval. ACM, CHIIR '19, https://doi.org/10.1145/3295750.3298919, URL http://dx.doi.org/10.1145/3295750.3298919

Sarker IH (2021) Machine learning: Algorithms, real-world applications and research directions. SN Computer Science 2(3). https://doi.org/10.1007/s42979-021-00592-x, URL http://dx.doi.org/10.1007/s42979-021-00592-x

Scells H, Zuccon G, Koopman B, et al (2020) Automatic boolean query formulation for systematic review literature search. In: Proceedings of The Web Conference 2020. ACM, WWW '20, https://doi.org/10.1145/3366423.3380185, URL http://dx.doi.org/10.1145/3366423.3380185

Scheffer T, Decomain C, Wrobel S (2001) Active hidden markov models for information extraction. In: International Symposium on Intelligent Data Analysis, Springer, pp 309–318

Schmidt L, Weeds J, Higgins J (2020) Data mining in clinical trial text: Transformers for classification and question answering tasks. In: Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies. SCITEPRESS - Science and Technology Publications, https://doi.org/10.5220/0008945700830094, URL http://dx.doi.org/10.5220/0008945700830094

van de Schoot R, de Bruin J, Schram R, et al (2021) An open source machine learning framework for efficient and transparent systematic reviews. Nature Machine Intelligence 3(February):125–133. https://doi.org/10.1038/s42256-020-00287-7, URL http://dx.doi.org/10.1038/s42256-020-00287-7

Scott AM, Forbes C, Clark J, et al (2021) Systematic review automation tools improve efficiency but lack of knowledge impedes their adoption: a survey. Journal of Clinical Epidemiology 138:80–94. https://doi.org/10.1016/j.jclinepi.2021.06.030, URL https://doi.org/10.1016%2Fj.jclinepi.2021.06.030

Shannon CE (1948) A Mathematical Theory of Communication. Bell System Technical Journal 27(3):379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Simon C, Davidsen K, Hansen C, et al (2019) Bioreader: a text mining tool for performing classification of biomedical literature. BMC Bioinformatics 19(S13). https://doi.org/10.1186/s12859-019-2607-x, URL http://dx.doi.org/10.1186/s12859-019-2607-x

Snyder H (2019) Literature review as a research methodology: An overview and guidelines. Journal of Business Research 104(July):333–339. https://doi.org/10.1016/j.jbusres.2019.07.039, URL https://doi.org/10.1016/j.jbusres.2019.07.039

Song J, Lee JK, Choi J, et al (2020) Deep learning-based extraction of predicate-argument structure (pas) in building design rule sentences⋆. Journal of Computational Design and Engineering 7(5):563–576. https://doi.org/10.1093/jcde/qwaa046, URL http://dx.doi.org/10.1093/jcde/qwaa046

Soto AJ, Przybyła P, Ananiadou S (2018) Thalia: semantic search engine for biomedical abstracts. Bioinformatics 35(10):1799–1801. https://doi.org/10.1093/bioinformatics/bty871, URL http://dx.doi.org/10.1093/bioinformatics/bty871

Thrun SB (1995) Exploration in active learning. Handbook of Brain and Cognitive Science pp 381–384. URL http://robots.stanford.edu/papers/thrun.arbib-handbook.ps.gz

Timsina P, Liu J, El-Gayar O (2015) Advanced analytics for the automation of medical systematic reviews. Information Systems Frontiers 18(2):237–252. https://doi.org/10.1007/s10796-015-9589-7, URL https://doi.org/10.1007%2Fs10796-015-9589-7

Tomassetti F, Rizzo G, Vetro A, et al (2011) Linked data approach for selection process automation in systematic reviews. In: 15th Annual Conference on Evaluation and Assessment in Software Engineering (EASE 2011). IET, https://doi.org/10.1049/ic.2011.0004, URL http://dx.doi.org/10.1049/ic.2011.0004

Vaswani A, Shazeer N, Parmar N, et al (2023) Attention is all you need. 1706.03762

Walkowiak T, Datko S, Maciejewski H (2018) Bag-of-Words, Bag-of-Topics and Word-to-Vec Based Subject Classification of Text Documents in Polish - A Comparative Study, Springer International Publishing, p 526–535. https://doi.org/10.1007/978-3-319-91446-6_49, URL http://dx.doi.org/10.1007/978-3-319-91446-6_49

Wallace BC, Trikalinos TA, Lau J, et al (2010) Semi-automated screening of biomedical citations for systematic reviews. BMC Bioinformatics 11(1). https://doi.org/10.1186/1471-2105-11-55, URL https://doi.org/10.1186%2F1471-2105-11-55

Weißer T, Saßmannshausen T, Ohrndorf D, et al (2020) A clustering approach for topic filtering within systematic literature reviews. MethodsX 7:100831. https://doi.org/10.1016/j.mex.2020.100831, URL http://dx.doi.org/10.1016/j.mex.2020.100831

Xie Q, Bishop JA, Tiwari P, et al (2022) Pre-trained language models with domain knowledge for biomedical extractive summarization. Knowledge-Based Systems 252:109460. https://doi.org/10.1016/j.knosys.2022.109460, URL http://dx.doi.org/10.1016/j.knosys.2022.109460

Yu Z, Kraft NA, Menzies T (2018) Finding better active learners for faster literature reviews. Empirical Software Engineering 23(6):3161–3186. https://doi.org/10.1007/s10664-017-9587-0, URL https://doi.org/10.1007%2Fs10664-017-9587-0

Zhang D, Baclawski KP, J. Tsotras V (2009) B+-Tree, Springer US, p 197–200. https://doi.org/10.1007/978-0-387-39940-9_739, URL http://dx.doi.org/10.1007/978-0-387-39940-9_739

Appendix A Questions used for the survey

1. Please indicate your affiliation/institution
2. Select the country where your affiliation/institution is located
3. For how long have you been performing systematic reviews (SR)?
4. How many systematic reviews have you been involved in over the past 5 years?
5. Which type (s) of systematic reviews do you perform? Tick all that apply
6. Based on your experience, rate the level of ease/difficulty associated with each stage as you perform a systematic review (or other types of review) of the literature
7. Have you ever used automation software (any tool that is proposed to expedite any 7 stages of SR process e.g Rayyan, Abstrackr etc NOT a referencing managing tool e.g Zotero, Mendeley etc) while performing an SR?

If NO:

a. Are you aware of existing automation tools available for SRs
   IF YES:
   Kindly state your reason (s) for not using those tools. Tick all that apply IF NO:
   i. Considering that such tools are created to optimise the SR process, how willing would you be to accept and use one on a scale of 1 - 10?

IF YES:

a. In which stage (s) in the SR did you apply the tool?
b. On a scale of 1-10, how useful was the tool in the SR stage (s) you selected previously?
c. How did you learn to use the automation tool
d. Was there any Human checking while using the tool?
e. Based on your experience, how much time did the tools speed up the review process?
f. Did you encounter any challenges while using the tool?

IF YES:

a. What were some of these challenges (s)? Tick all that apply

8. Based on your experience as a systematic reviewer, which particular stage in the SR process do you think would be of the most benefit using an automation method or tool?

9. Any comments or suggestions you would like to see in future systematic review (or other review types) automation tool?

10. In your opinion, what makes a good SR, or what will you consider making the output of an SR a very good one.

Fig. A1: Summary of questions asked during the survey

Appendix B Some selected results from the survey

Q: Based on your experience, rate the level of ease/difficulty associated with each stage as you perform a systematic review (or other types of review) of the literature

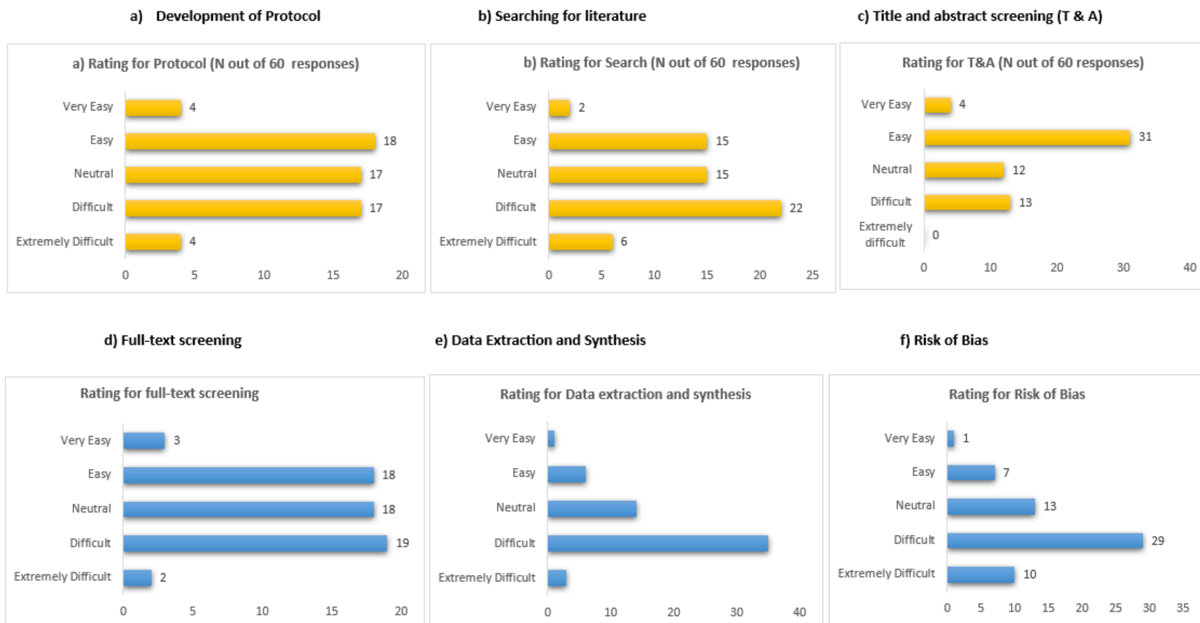


Fig. B2: Summary of results from respondents on ranking the degree of ease/difficulty associated with each stage as they perform SRs using the Likert scale.

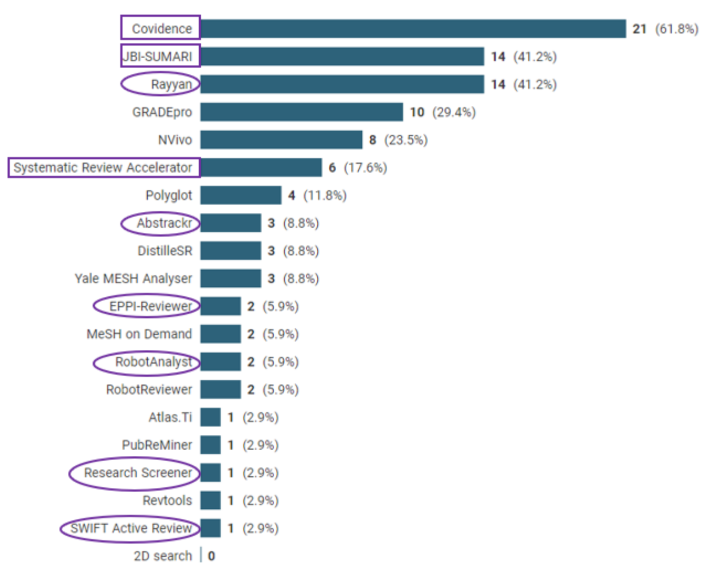


Fig. B3: Summary of the most used AI automation tools from the SR respondents.[15]

[15] The *squared* tools are those applied to multiple stages in the SR process, while the *circled* tools are those applied only to the title and abstract/citation screening stage and use the concept of active learning(human-in-the-loop)

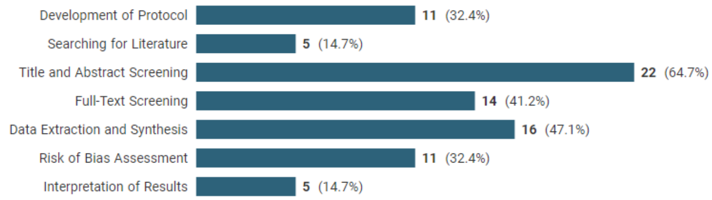


Fig. B4: Stage in the review process where participants deployed automation tools

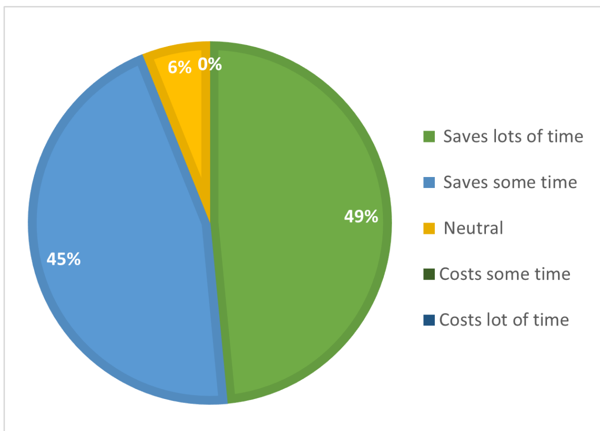


Fig. B5: Q: Based on your experience, how much time did the tools speed up the review process?

Table B1: Further suggestions from reviewers for future AI automation techniques

No	Suggestions from SR reviewers	Stage
1	*I think tools need to become more flexible and not just be built around what are effectively Cochrane standards and inocesses. For example, it would be helpful for text mining tools to reflect the fact that not all reviews require a comprehensive/exhaustive search (e.g. by helping prioritise terms?) and for tools designed to support screening to work with processes other than two independent reviewers screening 100interpretive/configurative reviews most often and this is reflected in my answer here. It would be really helpful in this particular field to have more flexible tools that can support processes to free up more time for interpretive work.*	Search and Screening
2	*Automation of data extraction and risk of bias would help speed up the conduct of SRs further.*	Data Extraction and RoB
3	*Retrieval of paper from all published data*	Search
4	*Need to communicate with health librarians to develop a suitable tool for searching across varying databases to find relevant literature.*	Search
5	*The manual extraction of outcomes will always need human input but might benefit from an initial AI attempt to save extraction time.*	Data Extraction
6	*Would be great to see a full-text screening and/or data extraction tool.*	Screening and Data Extraction
7	*Screening of title, abstract or full text could be an area to work on.*	Screening
8	*Automated data extraction would be great, but very difficult to implement well.*	Data Extraction
9	*An automation tool to develop search strategy specific to databases when keywords are provided. A tool for searching multiple databases*	Search