ContReviews: a Content-based Recommendation System for Updating *Living Evidences* in health care

Paolo Tenti[a], James Thomas[b], Rafael Peñaloza[a], Gabriella Pasi[a]

[a]*IKR3 Lab, University of Milano-Bicocca, Italy*
[b]*EPPI Centre, UCL Social Research Institute, University Colledge London, UK*

Abstract

Systematic reviews (SR) summarise the knowledge available in the literature related to a specific research topic. Keeping SRs up-to-date with new publications as soon as they become available is fundamental to avoid their early obsolescence. Recently, automated methods have been proposed to update one or more SRs. However, particularly in the health care domain, it is necessary to scale these methods to maintain *Living Evidences*, which comprise thousands of SRs. In this context, the main issue of using the current methods is that they are SR-specific, that is, they require manually designing and optimising search queries and eligibility assessment models for each SR. To address this challenge, the *ContReviews* system is proposed. *ContReviews* first leverages an academic knowledge graph to gather new publications, and then uses a content-based recommendation model to match these new publications to *all* the SRs in a *Living Evidence*. To faithfully represent new publications and SRs, multiple publication properties are used (i.e., title, abstract, citation network, and authors) and, for each of them, likelihoods of relevance are calculated and used to learn a relevance assessment function for an entire *Living Evidence*. *ContReviews* has been evaluated on a dataset of 6000+ Cochrane Reviews in the health care domain, reporting high efficiency and high effectiveness in recommending new publications to the Cochrane Reviews. Specifically, *ContReviews* has achieved an average precision of 98.1% with a recall of 100% on the considered Cochrane Reviews.

Keywords: systematic reviews update, living evidence, living systematic reviews, academic knowledge graph, content-based recommendation systems

1. Introduction

Literature reviews and evidence syntheses are important research practices that aim to advance science by accounting for the available knowledge. In this context, **Systematic Reviews** (SRs) have emerged as a distinctive approach in health sciences to provide a rigorous and comprehensive way of assessing the relevant literature [1, 2]. SRs are characterised by being methodical, comprehensive, transparent, and replicable; this systematic approach aims to minimise subjectivity and bias [3]. Unlike SRs, traditional literature reviews are often driven by the experience of their authors, missing a truly systematic approach.

Email address: p.tenti1@campus.unimib.it (Paolo Tenti)

The development of an SR involves identifying all the scientific publications that are relevant to the specific SR's scientific question. This task is particularly challenging and time consuming, due to the specificity of the scientific question, the complexity of searching through diverse data sources, the large volume of published studies, and the rigour necessary to follow SR protocols. Due to this type of complexity and to the rapid rate of publication, SRs can be of poor quality, duplicative, and out of date as soon as they are published [4, 5, 6]. To mitigate these issues, *living SRs* have been proposed [7, 8]. *Living SRs* involve the continuous surveillance of data sources and the integration of relevant publications as soon as they become available.

Current approaches to creating and updating an SR blend human and machine efforts [9, 10, 11]. To this aim, as illustrated in Figure 1, the following activities are usually performed to keep a SR current: (i) identify the bibliographic databases that are relevant to the SR; (ii) design highly specialized SR-specific Boolean queries to search for the most promising citations; (iii) leverage the publications already included in the SR to train models that can reduce the vast number of citations identified in the previous step (citation screening) and assess the most promising ones in greater details (abstract screening); (iv) manually assess the resulting publications for inclusion in the SR, considering their full text.

Some organisations rely on a significant number of SRs, which need to be of high quality to achieve their goals. For example, NICE uses thousands of SRs to maintain 350 guidelines, with the aim of providing recommendations on a wide range of topics related to public health and social care in England.[1] However, maintaining thousands of SRs is a complex task that requires significant human, financial, and computational resources, often beyond the reach of many organisations. As a result, **Living Evidences**, which consists of comprehensive collections of living SRs that span entire research areas, are proposed [6, 12]. As previously outlined, *Living Evidences* are especially crucial in the health care domain, where the Cochrane Database of Systematic Reviews [13] (CDSR) is the main resource. The CDSR is kept current with new and updated SRs being continuously published when ready. A 2010 report estimated that about 11 SRs were published every day [14], and a more recent study found an average of 10,000 SRs published annually in the last 22 years [10], although some of them may be old or not actively maintained.

In this context, state of the art approaches are usually designed to update individual SRs or a small group of them, but are not well suited to update large collections of SRs like *Living Evidences* [6]. Specifically, their critical issue is twofold. First, complex Boolean queries are explicitly developed for the target SR, and undergo continuous testing and redevelopment upon losing effectiveness. Second, supervised machine learning models for citation and abstract screening are developed, trained, and optimised for the target SR. However, a *Living Evidence* system, which considers thousands of SRs, would benefit from a more general approach that can be applied simultaneously to all the included SRs [15].

To address this challenge, *ContReviews*, an SR-independent system for updating *Living Evidences*, is introduced in this paper (Figure 2). First, as proposed in [16], *ContReviews* leverages an **academic knowledge graph** (AKG), which is a domain independent source of new publications, that replaces the searching of multiple and heterogeneous bibliographic databases. Specifically, OpenAlex [17] is used: it is the largest open-source AKG globally, and it is continuously updated by agents searching the Web and bibliographic databases. OpenAlex provides new publications that are structured in a relational format, which facilitates the extraction of various publication properties (e.g., title, abstract, authors, citations, journal, and venue). It is worth

[1] `https://www.nice.org.uk/process/pmg20/chapter/glossary#recommendations`

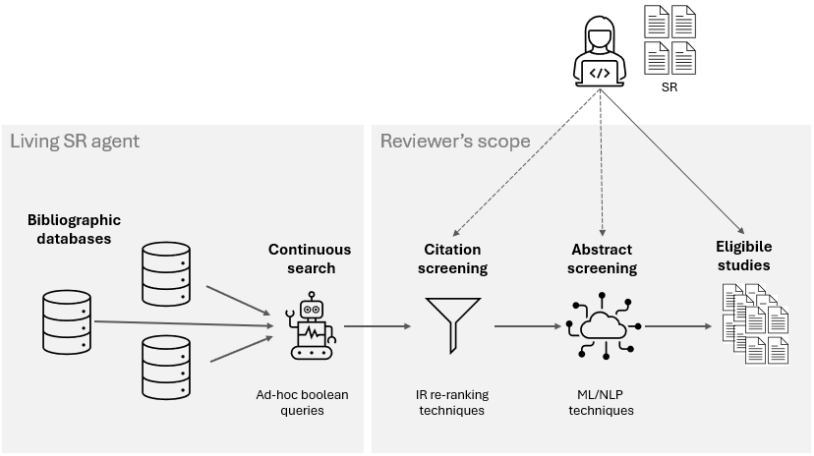

Figure 1: The common approach to maintain a *living SR* current.

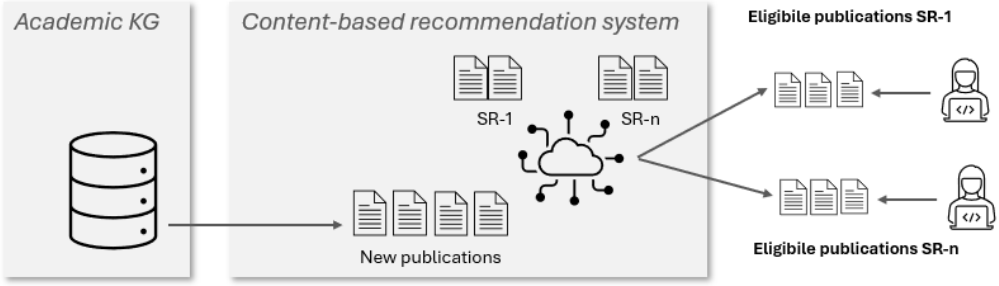

Figure 2: Workflow for *Living Evidence*.

outlining that entities, such as authors and citations, are disambiguated and provided by means of a unique identifier.

Second, a **content-based recommendation model** is proposed. While content-based recommendation models have been successfully applied in many different application domains, such as news articles [18], their adoption in the context of updating SRs is new. These systems are designed to recommend certain items to users, based on items' characteristics and user preferences [19, 20]. They differ in the methods used to (i) formally represent item profiles; (ii) formally represent user profiles; and (iii) match item profiles to user profiles to make recommendations. In the case of *Living Evidence*, the new publications are the items to recommend, and the SRs play the role of users. Publication profiles are described through their structured properties, which are provided by the AKG. In the context of this work, the title, abstract, authors, and citations of the publications are considered; however, other or different publication properties are also possible. SR profiles can also be described through the same publication properties, by considering their included publications. Therefore, a new publication is recommended for inclusion in an SR if it matches the SR profile.

As explained in Subsection 3.1, both publications and SRs are formally represented by means of multiple vectors, which are based on the considered publication properties. These vector repre-

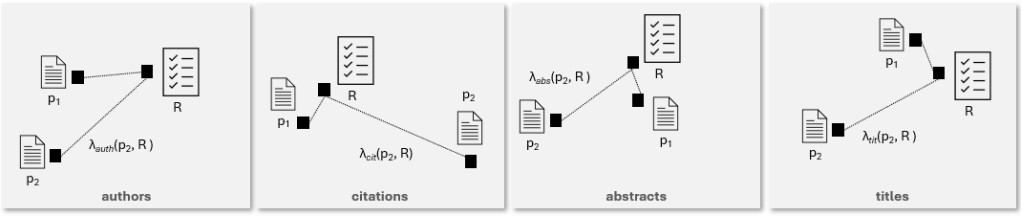

Figure 3: New publications (p_1 and p_2 in this picture) and SRs (R in this picture) are represented with one vector per publication property, which project them in multiple feature spaces (authors, citations, abstract, and title in this picture). For each publication and an SR their property-specific likelihoods of relevance $\lambda_\pi(p, r)$ are computed for each publication property π. The likelihood that a new publication is relevant to an SR is obtained through a relevance assessment function $\mathcal{M}$ which uses the property-specific likelihoods of relevance as parameters, i.e. $\lambda(p, r) = \mathcal{M}(\lambda_{auth}(p, r), \lambda_{cit}(p, r), \lambda_{abs}(p, r), \lambda_{tit}(p, r))$.

sentations map publications and SRs to multiple feature spaces, each corresponding to a different publication property, as shown in Figure 3. In each of these feature spaces, as discussed in Subsection 3.2, publications and SRs are matched and assigned with a **property-specific likelihood of relevance**. Finally, as described in Subsection 3.3, a relevance assessment function is computed by combining the property-specific likelihoods of relevance, to determine the likelihood that a new publication is relevant to an SR.

To evaluate the *ContReviews* system, experiments using a dataset of SRs published in the CDSR (that is, Cochrane Reviews) have been conducted. *ContReviews* provides the following research contributions in the *Living Evidence* context.

- To our knowledge this is the first approach to design a model for healthcare-related *Living Evidences*, as well as the first proposal to apply a content-based recommendation model to the extensive field of SR updating.
- As it is based on a content-based recommendation system, *ContReviews* can be applied to all SRs in a *Living Evidence*. In contrast to existing SR updating approaches, this method eliminates the need to manually design and test SR-specific Boolean search queries, as well as training SR-specific screening models.
- *ContReviews* leverages the entire *Living Evidence*'s data and, compared to most existing approaches to update individual SRs, it achieves higher efficiency with the same level of effectiveness. In fact, most health care related SRs include only a few dozen publications (see statistics in Table 1), therefore, training efficient supervised machine learning models for each of them, as most of the current methods do, is a significant challenge.
- The proposed content-based recommendation model leverages property-specific likelihoods of relevance to assess the relevance of publications to SRs. Property-specific likelihoods of relevance represent a novel approach to engineering model features in the space of methods for updating SRs.

2. State of the Art

In recent years, there has been a growing interest in automating the process of SR updating, especially in health care, through the application of information retrieval, ML and NLP

techniques. Most works in the state of the art focus on methods for identifying the new publications and automating citation screening. However, systematic comparison of different methods is challenged by the lack of standardised datasets and common evaluation criteria [10].

To produce or update high-quality SRs, complex Boolean queries are often developed to find useful publications. The quality of queries in this setting is crucial, as it affects how many documents must be verified for inclusion in the final SR by human reviewers [21]. The problem of creating effective Boolean queries to make SRs in health care has been tackled with MeSH term suggestion methods [22], data-driven query generation from existing citations [23], pre-trained generative models [24] and seeding methods [25]. Screening prioritisation, the process of ranking the set of documents found through Boolean queries, is also studied, as prioritising the most relevant documents ensures that the next review steps can be done more quickly and accurately. Screening prioritisation methods at the current state of the art use the final SR's title as a query to rank the publications, using neural methods based on language models [26, 27].

With the rise of *living SR*, the notion of *continuous surveillance* of data sources has been proposed [16, 28, 29] in contrast to periodically rewriting Boolean queries to update SRs. Recently, freely available academic knowledge graphs (AKG) have also emerged [30, 17]. They maintain continuous web-scale search over bibliographic databases and publicly available repositories in multiple domains. The Microsoft Academic Graph [30] has been shown to be sufficiently comprehensive to maintain a living map of COVID-19 research, as eligible records could be identified with an acceptably high level of specificity [16]. Trialstreamer [29] implements a continuous aggregated database search with push notifications, covering the majority of the health care literature and focusing on randomised controlled trials.

Continuous surveillance, similarly to approaches based on Boolean queries, still produces many publications that need to be evaluated by SR owners; thus, prioritisation is still needed. Recent works tackle this challenge by focusing on the eligibility assessment of the identified publications, treating it as a classification problem. To this aim, several models have recently been proposed to perform abstract screening. These models leverage the representations of title and abstract and classification models, based on ML [28, 31, 32, 33, 34] and, more recently, on deep learning [35, 36, 37, 38]. Methods for representing the title and abstract of the publication are generally based on embedding models, such as BERT [39], BioMed-RoBERTa [40], SciBERT [41], and PubMedBERT [42]. Variations of these embedding models are also used to represent in a more effective manner paragraphs (SBERT [43], text paragraph embeddings [44]) or to exploit graph information (Node2Vec [45]).

Most of these works focus on individual SRs, considering each of them as an independent dataset. For this reason, models are often trained in the presence of highly imbalanced data, given that the average SR (especially in health care) includes a few dozen of publications [10, 13] while the amount of publications retrieved from bibliographic databases, either by means of continuous surveillance or ad hoc Boolean queries, is much larger. The class imbalance, among other factors, has been a persistent problem that affects the performance of classification models using the title and abstract of the publication [46].

To address this issue, active learning over a small subset of informative data has been argued to produce a better generalised model than one trained over more randomly selected data [47]. In the context of SRs, reviewers interactively train an eligibility assessment classifier by labelling publications that the classifier deems to be the most informative [48, 23, 11, 49]. Other works tackle the challenge of classification performance over unbalanced data by focusing on representing publications in a more informative manner; to do so, they consider additional publication aspects besides their title and abstract [50, 51], such as bibliographic information [46], citation

network and context [52], MeSH terms and UMLS concepts [48].

The evaluation of the proposed works is not uniform, not only because a common dataset is missing, but also because of the evaluation methodology they adopt. The performance of these works can be reported in terms of classification metrics, usually focusing on 'Precision' with 'Recall at 95%' or more [31, 28], 'Work Saved over Sampling at r% recall' [53], information retrieval metrics, and 'Yield' and 'Burden' for models performing active learning [11]. Moreover, instead of considering full automation, some works report their performance in a semi-automated setting, where human reviewers adopt models to facilitate their work [33].

3. The *ContReviews* System to Update *Living Evidences*

With the publication of new research, it is fundamental to keep *Living Evidences* up to date. This task involves the identification of the new publications that are relevant to each SR within the Living Evidence. This is more formally defined next. Let $\mathcal{P}$ be the domain of all scientific publications, and $\mathcal{R}$ be the domain of all SRs, where each SR $r \in \mathcal{R}$ is formally represented as a subset of publications; namely, the publications included in r ($r \subseteq \mathcal{P}$). In technical terms, for the scope of this work, an SR is just a collection of publications. The function Pubs returns all publications included in a subset $R \subseteq \mathcal{R}$ of SRs: $\mathsf{Pubs}(R) = \bigcup_{r \in R} r$. Let $\hat{R} \subseteq \mathcal{R}$ be a subset of SRs belonging to the same scientific domain (i.e., a *Living Evidence*), and $\hat{P} \subseteq \mathcal{P}$ be a set of *new* publications which are not included in any of the SRs in the *Living Evidence*; that is, $\mathsf{Pubs}(\hat{R}) \cap \hat{P} = \emptyset$. The task is to find the SR-specific subsets of new publications $\hat{P}_r \subseteq \hat{P}$ that are relevant to each SR in the *Living Evidence* $r \in \hat{R}$.

To tackle this task, *ContReviews* leverages an Academic Knowledge Graph (AKG) to facilitate the collection of new publications (i.e., $\hat{P}$ in the formulation above) and their structured properties, and a content-based recommendation model to suggest new publications to SRs (i.e., to find $\hat{P}_r$ for each SR $r \in \hat{R}$). The latter leverages the publication properties to describe both publications and SR profiles and, ultimately, to assess their relevance. To this end, *ContReviews* performs the following activities, which are illustrated in Figure 3 and are further explained in the following subsections:

- each publication and each SR is represented by means of multiple vectors, reflecting different publication properties, which map them to multiple feature spaces (Subsection 3.1);
- publications and SRs are matched, in each of these feature spaces, through a **property-specific likelihood of relevance** (Subsection 3.2);
- pairs of one publication and one SR are sampled from the *Living Evidence* and used to train a binary classification model, using their property-specific likelihoods of relevance as features (Subsection 3.3);
- this binary classification model is used as a relevance assessment function for the content-based recommendation model to infer the relevance of new publications to all SRs within the *Living Evidence* (Subsection 3.4).

3.1. Formal Representation of Publications and SRs

Both publications and SRs are formally described through publication properties. In the scope of this work, title, abstract, authors, and citations are considered publication properties.

However, this does not preclude that any set of publication properties can be adopted. As illustrated in Figure 3, each publication $p \in \mathcal{P}$ is represented by a set of vectors $v^{\pi}(p)$, one for each publication property π. Formally, the representation $v(p)$ of the publication $p \in \mathcal{P}$ is defined as the following set:

$$v(p) = \{v^{\pi_1}(p), \ldots, v^{\pi_n}(p) \mid \pi_n \in \Pi\} \tag{1}$$

where $\Pi = \{\pi_1, \ldots, \pi_n\}$ is the set of considered properties.

Similarly, each SR $r \in \mathcal{R}$ is represented by a set of vectors $v^{\pi}(r)$ that summarise all included publications. In this case, each $v^{\pi}(r)$ is obtained by averaging the vector representations of the publications included in it. For each property $\pi \in \Pi$:

$$v^{\pi}(r) = avg(\{v^{\pi}(p) \mid p \in Pubs(r)\}) \tag{2}$$

The specific construction of each vector $v^{\pi}(p)$ depends on whether the property π represents an entity (such as authors and citations) or a text (such as title and abstract).

For entities, binary vector representations based on vocabularies of entity instances extracted from all the SRs in the *Living Evidence* are used. For example, considering a vocabulary of n distinct author instances, a publication p is represented by a binary vector $v^{auths}(p)$ of length n, having 1 as the i-th element if p is authored by the i-th author in the vocabulary, and 0 otherwise.

Texts are represented by two vectors that may carry different aspects of the text. One vector corresponds to a 'bag-of-words' representation, based on 'Term-Frequency/Inverse-Document-Frequency' (Tf/Idf [54]). The second vector is based on contextual embeddings, obtained using the SciBERT [41] language model, fine-tuned for the task of updating a *Living Evidence* (as described in Section 5.3.3).

3.2. Computing Property-specific Likelihoods of Relevance

A property-specific likelihood of relevance provides a sign of relevance between a publication and an SR from the perspective of a specific publication property. A set $\hat{R} \subseteq \mathcal{R}$ of SRs (that is, a *Living Evidence*) and a set of properties $\Pi = \pi_1, ...\pi_n$ are considered. For each publication $p \in \mathsf{Pubs}(\hat{R})$ one vector representation $v^{\pi}(p)$ for each property $\pi \in \Pi$ is obtained (Equation 1). Similarly, for each SR $r \in \hat{R}$, one vector representation $v^{\pi}(r)$ for each property $\pi \in \Pi$ is also obtained (Equation 2). For each SR $r \in \hat{R}$, a set of publications $S_r = P \cup N$ is sampled from $\mathsf{Pubs}(\hat{R})$, where P comprises a sample of publications included in r (i.e., relevant to r) and N comprises some publications randomly sampled from other SRs (i.e., irrelevant to r):

$$S_r = P \cup N \subseteq \{p \in \mathsf{Pubs}(r)\} \cup \{p \in \mathsf{Pubs}(\hat{R}) - \mathsf{Pubs}(r)\} \tag{3}$$

The **property-specific likelihoods of relevance** (with respect to a publication property $\pi \in \Pi$) between a publication p and a SR r, denoted as $\lambda_{\pi}(p, r)$, represents the likelihood that there exists a less relevant publication to r than p, considering the sample S_r. The following equation expresses this concept, where $|\cdot|$ denotes the cardinality of a set and *cos_sim* denotes the cosine similarity of two vectors:

$$\lambda_{\pi}(p, r) = \frac{|\{p' \in S_r : cos\_sim(v^{\pi}(p'), v^{\pi}(r)) < cos\_sim(v^{\pi}(p), v^{\pi}(r))\}|}{|S_r|} \tag{4}$$

To further explain the above concept, consider Figure 4.

The rationale behind using property-specific likelihoods of relevance is based on the observation that only a few publications, among the many existing, are relevant to an SR. Assuming

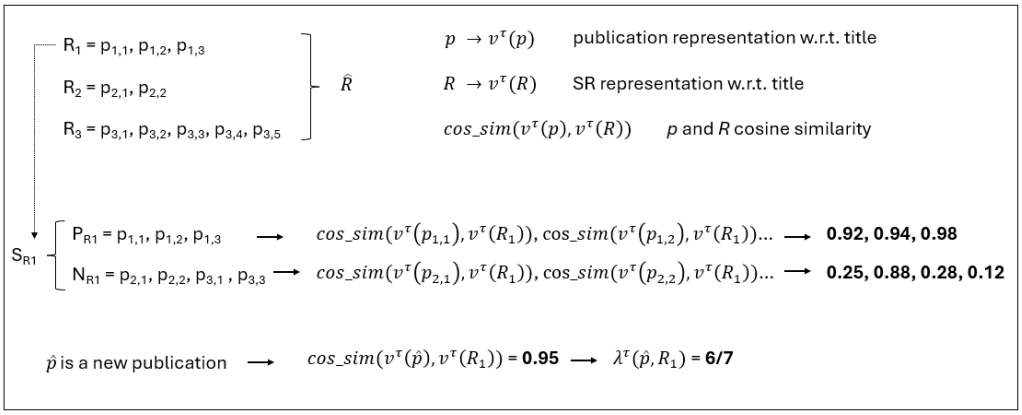

Figure 4: An example based on a *Living Evidence* with three SRs (i.e., $\hat{R} = \{R_1, R_2, R_3\}$), each one including some publications $p_{i,j}$. The publication property *title* is considered, which yields vector representations for publications (i.e., $v^\tau(p_{i,j})$) and SRs ($v^\tau(R_i)$). Focusing on R_1 as an example, the set S_{R_1} is constructed considering the set of included publications and a sample of the excluded ones. P_{R_1} and N_{R_1} yield the set of cosine similarities $cos\_sim(v^\tau(p_{i,j}), v^\tau(R_1))$. The values in P_{R_1} are all expected to be close to 1, because they are calculated for the publications included in R_1. Equivalently, the majority of values in N_{R_1} are expected to be lower, because they are calculated for the publications which are not included in R_1. To calculate the likelihood $\lambda_\tau(\hat{p}, R_1)$ that a new publication $\hat{p}$ is relevant to R_1 with respect to τ, the cosine similarity $cos\_sim(v^\tau(\hat{p}), v^\tau(R_1))$ is calculated. Say its value is 0.95, then $\lambda_\tau(\hat{p}, R_1) = 6/7$. In fact, there are 6 over 7 publications in S_{R_1} whose cosine similarity values with respect to R_1 are lower than 0.95.

that the relevant publications are 'closer' to the SR than the majority of publications (i.e., they have a higher cosine similarity in the corresponding feature spaces), the 'likelihood that a less close publication exists' provides a measure of relevance. Thus, to describe the relevance between a publication p and a SR r with respect to a set of publication properties $\Pi = \{\pi_1, ..., \pi_n\}$, the vector of property-specific likelihoods of relevance $\{\lambda_{\pi_1}(p, r), ..., \lambda_{\pi_n}(p, r)\}$ is used. In addition, the reason for using likelihoods of relevance, rather than bare similarity scores, is the fact that many SRs are related to similar research questions, or are compiled for the same research topic. Thus, some publications and SRs might be similar from the angle of certain properties just because of that, and the cosine similarity would not be significant. For example, 'smoking cessation' is a group of several SRs in the human behaviour domain: clearly the language used in these SRs is homogeneous, and potentially also authors and citations are so. Thus, publications from such subdomains might have high absolute similarity scores with all the 'smoking cessation' SRs, even if they are irrelevant. Likelihoods of relevance provide a statistical dimension for evaluating relevance, allowing the extent of similarity for relevant publications to be emphasised compared to irrelevant (but plausible) ones.

3.3. Training the relevance assessment function

The relevance assessment function is a model that infers the relevance of new publications to SRs in a *Living Evidence*. *ContReviews* implements this relevance assessment function as a binary classification model which uses property-specific likelihoods of relevance as features. This model is denoted as $\mathcal{M}_\Pi^{\hat{R}}$, where Π is a set of publication properties, and $\hat{R}$ is the *Living Evidence*.

$\mathcal{M}_\Pi^{\hat{R}}$ is trained with data from the entire *Living Evidence*. To obtain a supervised dataset for training, pairs of one publication $p \in \mathsf{Pubs}(\hat{R})$ and one SR $r \in \hat{R}$ are considered. Specifically,

each element in the training dataset is constructed by means of the following items:

- the property-specific likelihoods of relevance $\lambda_\pi(p, r)$, providing one feature for each publication property $\pi \in \Pi$;
- a binary label, which holds 1 if $p \in r$, and 0 otherwise

3.4. Inference

Let $\hat{R}$ be a *Living Evidence* and $\hat{P}$ a set of *new publications* such that $\hat{P} \cap \mathsf{Pubs}(\hat{R}) = \emptyset$. To score the relevance of the new publications to each SR in the *Living Evidence*, the model $\mathcal{M}_\Pi^{\hat{R}}$ introduced above is used. Specifically, for each pair of a new publication $p^* \in \hat{P}$ and an SR $r \in \hat{R}$, this inference step is achieved by doing the following:

- calculate the new publication's vector representations for each publication property, based on Equation 1, i.e., $\{v^{\pi_1}(p^*), ..., v^{\pi_n}(p^*)\}$;
- calculate the property-specific likelihoods of relevance, based on Equation 4, i.e., $\{\lambda_{\pi_1}(p^*, r), ..., \lambda_{\pi_n}(p^*, r)\}$;
- score the vector of property-specific likelihoods of relevance associated to the new publication p^* and the SR r by using the relevance assessment function model: $\lambda(p^*, r) = \mathcal{M}_\Pi^{\hat{R}}(\lambda_{\pi_1}(p^*, r), ..., \lambda_{\pi_n}(p^*, r))$.

$\lambda(p^*, r)$ provides the likelihood that p^* is relevant to r.

4. Experimental Settings

ContReviews is the first attempt, to our knowledge, to address the problem of *Living Evidence*. Common datasets and benchmarks for this problem, thus, are not available. Indeed, as discussed in Section 2, several researches have addressed the problem of SR updating, thus, several datasets are available; however, they are either too small to be significant in the *Living Evidence* space, too specific to an SR updating sub-task (i.e., query development, citation screening and abstract screening), or just too old [10]. Instead, to be realistic, *ContReviews* proposes an approach to evaluate the *Living Evidence* updating task which requires to assess several (i.e., thousands) SRs at once. For this reason, *ContReviews* is evaluated by using a new dataset of *Cochrane Reviews* (CR), which comprises 6,300+ SRs and 164,000+ included publications. This is a very large dataset in the specific context of updating SRs and *Living Evidences*, in fact, as previously outlined, the relevant works at the state of the art mostly consider one or more SRs.

4.1. Data Pre-processing

The publications in the CR dataset are matched with those maintained in OpenAlex, to gather the publication properties needed by *ContReviews*. This is necessary as the CR provide publication properties in text form, which require disambiguation. To this end, a selection of CR publication properties (title, journal, authors, volume, issue, pages, and DOI) is used to query OpenAlex publications through its APIs.[2] Furthermore, deduplication is applied when multiple

[2] `https://docs.openalex.org/#access`

Table 1: **Statistics about train and test datasets**: number of SRs and their included publications.

	Train dataset		Test dataset	
	CR-5	**CR-40**	**CR-5**	**CR-40**
Num. SRs	6395	699	6395	699
Num. subdomains	52	52	52	52
Tot num. publications	141240	45636	22955	4800
Min num. publications per SR	3	3	2	2
Avg num. publications per SR	22	65	3	6
Max num. publications per SR	596	363	10	10

OpenAlex publications match the same CR publication. Thus, a set of disambiguated publications described by title, abstract, citation network, and authors is obtained.

Two distinct datasets are generated over the CR—denoted as **CR-5** and **CR-40**—which correspond to all the SRs in the CR including, respectively, at least 5 publications and at least 40 publications. CR-5 and CR-40 are used to generate the training and test datasets. Specifically, CR-5 helps to evaluate *ContReviews* with small and large SRs, and CR-40 to compare *ContReviews* with state of the art methods. In fact, as previously discussed, the latter usually leverage one model per SR, requiring enough publications to support model training.

4.2. Train and Test Datasets

To train and test *ContReviews*, supervised datasets are generated from CR-5 and CR-40 (Table 1). The elements of each dataset are based on one publication, one SR and their label, which holds *true* if the publication is relevant to the SR (*positive publications*), and *false* otherwise (*negative publications)*. To discriminate which publications to use for training and which for testing, the date of publication is used: for each SR, the *most recent* 10% of included publications are used for testing, and the rest is used for training. These are all positive publications, because CR-5 and CR-40 comprise only relevant publications. In addition, some negative publications for each SR are artificially generated, by simply considering that publications included in one SR can be taken as negative publications for other SRs. It is worth noting that SRs in a *Living Evidence* belong to the same scientific domain, and there are several subdomains with homogeneous SRs. Moreover, to have enough differentiation, an additional source of negative publications is used, i.e. a set of health care related publications not included in any of the SRs in the CR (*external publications*).

The test dataset is restricted to have at least 2 and at most 10 positive publications per SR. In addition, 25 negative publications for each positive one are sampled from the external set of publications. The intention of this empirical setting is to simulate a realistic stream of new publications, where most of them are completely irrelevant to each SR in the CR dataset, and only a few of them are relevant for each SR.

The train dataset is restricted to having a maximum of 75 positive publications for each SR and 10 negative publications for each positive one. In contrast to the test dataset, where the negative publications are external to the CR dataset, the negative publications in the train dataset are sampled from other SRs within the CR dataset. This is specifically purposed to support the computation of property-specific likelihoods of relevance; in fact, to make them effective, their computation should involve a set of publications that are similar to each other. Hence, they are sampled from the CR dataset.

In addition, positive and negative publications in both datasets are enriched with the property-specific likelihoods of relevance required by the relevance assessment function. Calculating them involves obtaining publication vector representations for all the publication properties considered by the model. The evaluation setting considers representations of texts (i.e., title and abstract) based on bag-of-words and embeddings, and representations of entities (i.e., authors and citations) based on binary vectors. To obtain binary vector representations of the citation and author, the information extracted from OpenAlex is used to construct vocabularies of citations and authors. To obtain vector representations based on bag-of-words of title and abstract, texts are lowercased, stop-words removed, and tokenised over unigrams. All unigrams obtained from all publications in all the SRs are used to construct a vocabulary of terms, so that publications can be represented as vectors of Tf-Idf scores based on their title and abstract. In addition, to obtain vector representations for the publication based on embedded titles and abstracts, a fine-tuned version of SciBERT [41] is used. SciBERT is pre-trained on a corpus of 1.14 million scientific papers and is therefore closer to the health care context than more general language models. SciBERT was fine-tuned to predict publication relevance to SRs, through a multi-label classifier.

4.3. Training the Relevance Assessment Function

A binary classification model based on LightGBM [55, 56] is used as a relevance assessment function. The LightGBM model is trained on the dataset introduced above, which contains pairs of one publication and one SR, their likelihoods of relevance representing features, and a binary label representing relevance. As discussed, to support the calculation of likelihoods of relevance, the training dataset holds 10 negative records for each positive one: to avoid any issue related to class unbalancing, negative publications are downsampled prior to training the binary classifier.

5. Evaluation

The evaluation results are reported in Table 2 as descriptive statistics of the best precision with recall of 95% or higher, computed over all the SRs in the Cochrane Reviews (CR) test dataset introduced above.

5.1. Baseline

As previously mentioned, evaluation benchmarks are not available in the context of *Living Evidences*; therefore, two baseline methods are developed and evaluated on the CR datasets. Let r be an SR, and p be a publication whose abstract is denoted as p^a. Moreover, let $e(\cdot)$ be a function that returns the embedding of the input text. The first baseline method, as proposed in [10], evaluates the cosine similarity between an abstract embedding (i.e., $e(p^a)$) and the average embedding of the abstracts included in an SR (that is, $avg(\{e(p_i^a), p_i \in r\})$). Specifically, this cosine similarity is used as a measure of the relevance of p to r. The second baseline method shadows the abstract screening methods proposed in several state of the art studies [28, 32, 35, 37]: a binary classification model per SR is trained—based on the LightGBM algorithm [55]—that uses the embeddings of abstracts included in an SR (i.e., $\{e(p_i^a), p_i \in r\}$) as model features.

To assess the evaluation, the classification metrics of precision with recall of (at least) 95% are used, as proposed in [28, 32, 35, 37]. The rationale behind this is twofold: in one way, the priority for SR updating models is to capture *all* the relevant publications (that is, the recall should be as high as possible); on the other hand, although precision can be sacrificed in the name of high recall, it is still desirable to achieve good precision to lower the number of the irrelevant

Table 2: Average precision and recall over all the SRs. Recall above 95% is specifically considered. The standard deviation refers to precision.

	Dataset	Avg recall	Avg precision	Std Dev
classification@eABST (pt)	CR-40	0.981	0.115	0.202
classification@eABST (ft)	CR-40	0.982	0.211	0.356
similarity@eABST (pt)	CR-40	0.981	0.065	0.092
similarity@eABST (ft)	CR-40	0.981	0.227	0.372
ContReviews@lGBM	CR-40	1	0.974	0.086
ContReviews@lGBM	CR-5	1	0.981	0.086

publications that are recommended to reviewers and, ultimately, to reduce the amount of manual efforts.

The descriptive statistics of the evaluation metrics for all the SRs in the CR dataset are reported in Table 2: *similarity@eABST* refers to the first baseline method, using cosine similarities between the embeddings of publications and SRs; *classification@eABST* refers to the second baseline method, where abstract screening is considered. The baseline methods using both the pre-trained SciBERT language model and its fine-tuned version introduced in Subsection 4.2, are evaluated. The notations *pt* and *ft* are used to refer to embeddings based, respectively, on the pre-trained SciBERT and its fine-tuned version. Note that the *classification@eABST* baseline results are only reported for the *CR-40* dataset, in fact, *CR-5* would have to few included publications per SR to train the abstract screening models.

5.2. Main Evaluation Results

ContReviews evaluation results are reported in Table 2 (*ContReviews@lGBM*). They show that *ContReviews* achieves average precision above 97.4% with recall of 100% with both datasets. Instead, both the baseline methods (*similarity@eABST* and *classification@eABST*) achieve recall greater than 98% as *ContReviews* does, but at the price of precision being significantly lower. As discussed, lower precision rates correspond to an higher amount of irrelevant publications recommended to reviewers, triggering additional human labour. In addition, *ContReviews* shows better standard deviation over SR precision values, meaning that the average precision is more regular compared to the baseline methods. Note that the latter have much higher standard deviation, except *similarity@eABST (pt)* which, however, achieves the lowest precision.

Table 3 reports the statistics on the precision gap between *ContReviews* and the baseline methods. It shows that for a small fraction of SRs (the *% worst* column is less than 4%) the baseline methods perform better than *ContReviews* in terms of precision with a recall grater than 95%. However, *ContReviews* still achieves precision values above 86% (*Avg precision* column), though the gap with the baseline method is consistent (*Avg gap* column is above 11%). Moreover, shifting the threshold for recall from 95% to 97%, *ContReviews* beats the baseline methods for all SRs.

Figure 5 shows descriptive statistics about the results summarised in Table 3. Specifically, it uses candlesticks to display the precision values for each SR, and represent them for both *ContReviews@LightGBM* (*precision_y*) and the best baseline method (*precision_x*). In addition, their gap (*delta_p*) is also shown. The candlesticks show that the interquartile range of both models (that is, the precision points between the first quartile and the third quartile) collapses to the median value; and that the SRs achieving better precision with the baseline method are outliers.

Table 3: Statistics about the gap in precision between the baseline methods and *ContReviews*, with the CR-40 dataset. *% worst* of all the SRs perform better with the baseline methods; for them *ContReviews* achieve average precision as of *Avg precision* with a gap to baseline methods as of *Avg gap*.

Model	Threshold	% worst	*Avg precision*	Avg gap
classification@eABST (ft)	0.95	3.15%	0.867	-0.113
similarity@eABST (ft)	0.95	3.78%	0.870	-0.106
classification@eABST (ft)	0.97	0%	na	na
similarity@eABST (ft)	0.97	0%	na	na

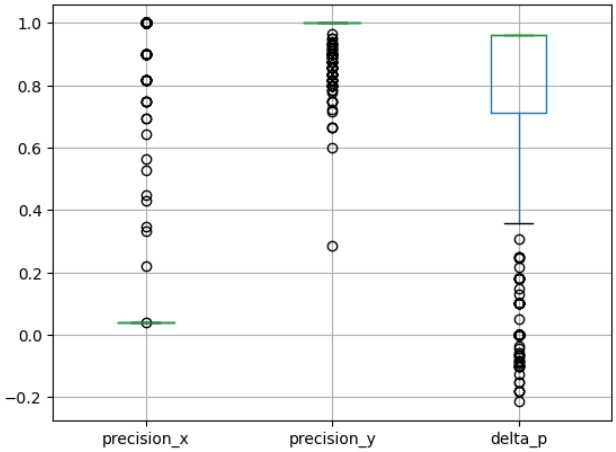

Figure 5: Candlesticks of precision with recall of 95%, for *similarity@eABST (ft)* (leftmost), *ContReviews@LightGBM* (center) and their gaps (rightmost). The box extends from the first quartile to the third quartile of the precision points, with a line at the median; and the whiskers extend from the box to the farthest data point lying within 1.5x the inter-quartile range from the box. Outlier points are those past the end of the whiskers.

Candlesticks provide a good intuition of the *ContReviews* performance compared to baseline methods. To formally confirm this intuition, a statistical significance test is used to verify the "null hypothesis" that both samples of precision values are drawn from the same population. Rejecting this null hypothesis means that there is a significant difference between the means, that is, they are not different by chance, and so it is the observed difference in performance between the baseline methods and *ContReviews*. Specifically, the Kolmogorov-Smirnov[3] two-sided test rejected the null hypothesis, hence, the means of the baseline method and *ContReviews* are drawn from different populations, and their difference is statistically meaningful.

Finally, *ContReviews* shows good adaptation to large datasets with small SRs, i.e., CR-5 holds 6000+ SRs with 22 publications included on average, and some of them have a few publications included. The chart in Figure 6 shows the trend of precision with recall of 100% for SRs with up to 50 publications included, being above 86%.

[3]Python function 'ks_2samp', `https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ks_2samp.html`

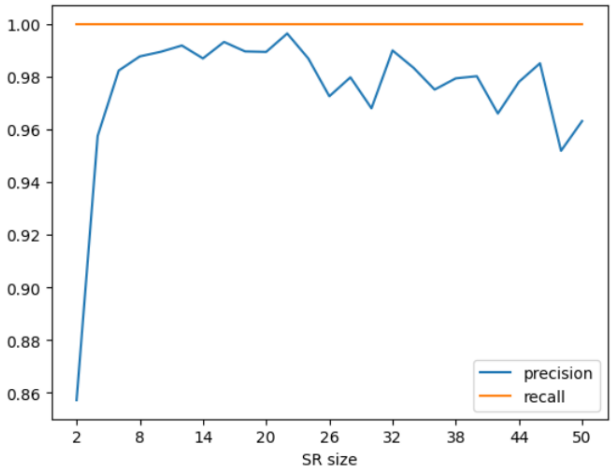

Figure 6: Precision with recall of 100% obtained by our *ContReviews* with the CR-5 dataset; data points are plot for SRs with 50 publications included at max, in bins of size 5.

5.3. Ablation Studies

To assess the evaluation results introduced above, the following aspects are considered: (i) using property-specific likelihoods of relevance, instead of bare cosine similarities; (ii) using multiple publication vector representations, instead of a single one based on the publications' abstract; (iii) using alternative embedding models. To this aim, the **single property model** is introduced, that is, a simplified version of *ContReviews* based on a single publication property. This single property model is based on one likelihood of relevance, which corresponds to the considered property. This likelihood of relevance is directly used to assess the relevance, i.e., no relevance assessment function is used.

5.3.1. Using property-specific likelihoods of relevance

To assess the importance of using property-specific likelihoods of relevance, two models are compared: one (*ContReviews@eABST (ft)* in Table 4) is the single property model based on abstracts; the other one is the baseline method *similarity@eABST* reported in Table 2. These two models differ only because the former leverages the property-specific likelihood of relevance, while the latter uses the absolute cosine similarity. *ContReviews@eABST (ft)* largely outperforms *similarity@eABST* (with or without fine-tuning), that is, using property-specific likelihoods of relevance is effective. In addition, the evaluation results for other single property models (i.e., *sABST*, *sTITL*, *sCITA*, *sAUTH*) are also reported in Table 4: they all outperform the baseline methods in Table 2, showing the effectiveness of property-specific likelihoods of relevance.

5.3.2. Using multiple vector representations

The effectiveness of using multiple vector representations is also evaluated, by comparing *ContReviews* using all the available publication properties (i.e., *ContReviews@lGBM (ft)* in Table 2) to single property models (i.e., *ContReviews@eABST (ft)* in Table 4). The former achieves the best results on all the evaluation tests, showing that using an ensemble of vector representations is effective. However, single property models, especially those using abstracts (*ContReviews@sABST* and *ContReviews@eABST (ft)*), are more computationally efficient and achieve

Table 4: Evaluation results for models using one single property, in terms of precision with recall of 0.95 or greater. The notion of *P(R) is used*. *ContReviews@feat* uses one single property, without a relevance assessment model. *feat* can be embeddings of abstracts (*eABST*), bag-of-words representation of abstracts (*sABST*), embeddings of titles (*eTITL*) or binary representations of citations (*sCITA*) and authors (*sAUTH*).

	CR-40		CR-5	
Model	Mean - P(R)	Std dev - P(R)	Mean - P(R)	Std dev - P(R)
ContReviews@eABST (ft)	0.964 (0.981)	0.11 (0.036)	na	na
ContReviews@sABST	0.944 (0.981)	0.131 (0.036)	0.919 (0.995)	0.175 (0.02)
ContReviews@sTITL	0.875 (0.981)	0.216 (0.364)	0.87 (0.995)	0.232 (0.19)
ContReviews@eTITL (ft)	0.953 (0.981)	0.132 (0.036)	0.881 (0.995)	0.222 (0.02)
ContReviews@sCITA	0.517 (0.981)	0.470 (0.036)	0.623 (0.995)	0.461 (0.02)
ContReviews@sAUTH	0.191 (0.981)	0.349 (0.036)	0.173 (0.995)	0.334 (0.02)

comparable average results (though their standard deviation is considerably higher, suggesting that they are less regular across all the SRs).

5.3.3. Embeddings

The evaluation results reported in this section are based on three embedding models. The first model, SciBERT [41], is pre-trained on a corpus of 1.14 million scientific papers, making it more closely aligned with the context of Cochrane Reviews than more general language models. The second model is fine-tuned to predict the relevance of publications to SRs, using the training dataset described in Subsection 4.2. It specifically employs a SciBERT encoder with a multi-label classifier built upon it.

A third model based on LongFormer [57], and fine-tuned for semantic similarity using the same network setup and training approach as Specter [58], was considered. The rationale behind this third model is twofold. In one way, it has been empirically observed that SciBERT, as well as other embedding models based on BERT, often truncates the CR abstracts. This is due to the maximum input sequence length being significantly lower than the average abstract length, which exceeds 2000 tokens, as seen in the CR dataset. In contrast, Longformer supports longer input sequences. On the other hand, as the property-specific likelihoods of relevance are determined through cosine similarity, an embedding model designed to optimise a loss function based on cosine similarity becomes appealing.

These embedding models are evaluated for their ability to determine the relevance of new publications to SRs, by using the test dataset introduced in Subsection 4.2 (results are presented in Table 5). To this aim, two relevance assessment functions are considered: the first one is a binary classification model using the input sequence embeddings as features (i.e., LightGBM, similarly to the previous evaluations); the second one simply uses the cosine similarity between the SR's embeddings and the publications embeddings and a threshold for classification. In addition, to obtain abstract embeddings, two methods have been evaluated, respectively denoted as *pooler* and *average* in Table 5: the first considers the token trained to represent the entire input sequence; the second averages the embeddings of all the input sequence tokens.

It can be observed that almost all the fine-tuned models achieve consistent performance: the baseline method, the fine-tuning task, and the method to obtain sentence embeddings do not seem to be important to determine the quality of the obtained embeddings, at least in regard to the considered downstream task. Thus, it can be concluded that fine-tuning over SRs data is the most important factor for obtaining high-quality embeddings. Moreover, the model based on

Table 5: Evaluation results for different embedding models.

		LightGBM		**Cosine similarity**	
Model	Method	Precision	Recall	Precision	Recall
SciBERT	Pooler	0.073	0.979	0.046	0.979
pre-trained	Average	0.141	0.979	0.213	0.979
SciBERT	Pooler	0.209	0.979	0.213	0.979
fine-tuned	Average	**0.213**	0.979	**0.218**	0.979
LongFormer	Pooler	0.134	0.976	0.072	0.976
pre-trained	Average	0.129	0.979	0.058	0.979
LongFormer	Pooler	**0.212**	0.979	**0.212**	0.979
fine-tuned	Average	0.167	0.979	0.091	0.979

LongFormer is more computationally expensive compared to SciBERT, due to the larger size of the baseline method, and to the more complex training approach.

6. Conclusions and Future Work

In this work, *ContReviews* is proposed to address the challenge of updating *Living Evidences*, that is, keeping large collections of Systematic Reviews (SRs) up to date as soon as new publications are available. *ContReviews* is specifically proposed in the context of health care, where *Living Evidences* are crucial.

ContReviews leverages an academic knowledge graph (i.e., OpenAlex [17]) to identify all the most recent publications, and a content-based recommendation model to match them to the SRs in a *Living Evidence*. This approach demonstrated to adapt well to the *Living Evidence* updating challenge, where thousands of SRs must be kept current. On the contrary, traditional approaches are tailored to a specific SR, and applying them to the *Living Evidence* problem would require excessive efforts.

The reported experiments show high average precision (that is, 98.1%) with high recall (that is, 100%) across all SRs in the test *Living Evidence*. In contrast, models at the state of the art depend on SR-specific data and, to achieve high recall, need to sacrifice precision aggressively.

Future work will focus on improving the current model from a computational efficiency perspective. More sophisticated embedding methods might help to obtain more informative and denser representations for publication properties. Finally, additional validation is needed for *Living Evidences* in different domains (i.e., non-health care), as they have different characteristics.

Acknowledgements

This work was partially supported by MUR for the Department of Excellence DISCo at the University of Milano-Bicocca. Moreover, we acknowledge with thanks the Cochrane Collaboration, which provided data about studies included in thousands of their systematic reviews.

References

[1] R. J. Piper, How to write a systematic literature review: a guide for medical students, National AMR, fostering medical research 1 (2013) 1–8.

[2] J. Higgins, J. Thomas, J. Chandler, M. Cumpston, T. Li, M. Page, V. Welch (Eds.), Cochrane Handbook for Systematic Reviews of Interventions version 6.3 (updated February 2022), John Wiley & Sons, 2022, chapter 1: Starting a review. https://training.cochrane.org/handbook/current/chapter-01#section-1-1.

[3] A. P. Siddaway, A. M. Wood, L. V. Hedges, How to do a systematic review: a best practice guide for conducting and reporting narrative reviews, meta-analyses, and meta-syntheses, Annual review of psychology 70 (2019) 747–770.

[4] S. McDonald, S. Turner, M. J. Page, T. Turner, Most published systematic reviews of remdesivir for covid-19 were redundant and lacked currency, Journal of clinical epidemiology 146 (2022) 22–31.

[5] K. G. Shojania, M. Sampson, M. T. Ansari, J. Ji, S. Doucette, D. Moher, How quickly do systematic reviews go out of date? a survival analysis, Annals of internal medicine 147 (4) (2007) 224–233.

[6] J. Elliott, R. Lawrence, J. C. Minx, O. T. Oladapo, P. Ravaud, B. Tendal Jeppesen, J. Thomas, T. Turner, P. O. Vandvik, J. M. Grimshaw, Decision makers need constantly updated evidence synthesis (2021).

[7] J. H. Elliott, T. Turner, O. Clavisi, J. Thomas, J. P. T. Higgins, C. Mavergames, R. L. Gruen, Living systematic reviews: An emerging opportunity to narrow the evidence-practice gap, PLoS Medicine 11 (2014).

[8] J. Thomas, A. Noel-Storr, I. Marshall, B. Wallace, S. McDonald, C. Mavergames, P. Glasziou, I. Shemilt, A. Synnot, T. Turner, et al., Living systematic reviews: 2. combining human and machine effort, Journal of clinical epidemiology 91 (2017) 31–37.

[9] I. Marshall, B. Wallace, Toward systematic review automation: a practical guide to using machine learning tools in research synthesis, Systematic Reviews 8 (12 2019). doi:10.1186/s13643-019-1074-9.

[10] W. Kusa, O. E Mendoza, M. Samwald, P. Knoth, A. Hanbury, Csmed: Bridging the dataset gap in automated citation screening for systematic literature reviews, Advances in Neural Information Processing Systems 36 (2024).

[11] M. Miwa, J. Thomas, A. O'Mara-Eves, S. Ananiadou, Reducing systematic review workload through certainty-based screening, Journal of Biomedical Informatics 51 (2014) 242–253. doi:https://doi.org/10.1016/j.jbi.2014.06.005.
URL https://www.sciencedirect.com/science/article/pii/S1532046414001439

[12] T. Turner, J. Elliott, B. Jeppesen, J. Vogel, S. Norris, R. Tate, S. Green, The australian living guidelines for the clinical care of people with covid-19: What worked, what didn't and why, a mixed methods process evaluation, PLOS ONE 17 (2022) e0261479. doi:10.1371/journal.pone.0261479.

[13] I. Chalmers, M. Enkin, M. Keirse, Preparing and updating systematic reviews of randomized controlled trials of health care, The Milbank quarterly 71 (3) (1993) 411—437. doi:10.2307/3350409.
URL https://doi.org/10.2307/3350409

[14] H. Bastian, P. Glasziou, I. Chalmers, Seventy-five trials and eleven systematic reviews a day: how will we ever keep up?, PLoS medicine 7 (9) (2010) e1000326.

[15] P. Garner, S. Hopewell, J. Chandler, H. MacLehose, E. A. Akl, J. Beyene, S. Chang, R. Churchill, K. Dearness, G. Guyatt, C. Lefebvre, B. Liles, R. Marshall, L. Martínez García, C. Mavergames, M. Nasser, A. Qaseem, M. Sampson, K. Soares-Weiser, Y. Takwoingi, L. Thabane, M. Trivella, P. Tugwell, E. Welsh, E. C. Wilson, H. J. Schünemann, When and how to update systematic reviews: consensus and checklist, BMJ 354 (2016). arXiv:https://www.bmj.com/content/354/bmj.i3507.full.pdf, doi:10.1136/bmj.i3507.
URL https://www.bmj.com/content/354/bmj.i3507

[16] I. Shemilt, A. Arno, J. Thomas, T. Lorenc, C. Khouja, G. Raine, K. Sutcliffe, I. Kwan, K. Wright, A. Sowden, et al., Cost-effectiveness of microsoft academic graph with machine learning for automated study identification in a living map of coronavirus disease 2019 (covid-19) research, Wellcome Open Research 6 (210) (2021) 210.

[17] J. Priem, H. Piwowar, R. Orr, Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts (2022). doi:10.48550/ARXIV.2205.01833.
URL https://arxiv.org/abs/2205.01833

[18] S. Raza, C. Ding, News recommender system: a review of recent progress, challenges, and opportunities, Artificial Intelligence Review (2022) 1–52.

[19] M. J. Pazzani, D. Billsus, Content-based recommendation systems, in: The adaptive web: methods and strategies of web personalization, Springer, 2007, pp. 325–341.

[20] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.

[21] H. Scells, G. Zuccon, B. Koopman, A comparison of automatic boolean query formulation for systematic reviews, Information Retrieval Journal 24 (2021) 3–28.

[22] S. Wang, H. Li, G. Zuccon, Mesh suggester: A library and system for mesh term suggestion for systematic review boolean query construction, in: Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, 2023, pp. 1176–1179.

[23] A. D'Ambrosio, H. Grundmann, T. Donker, An open-source integrated framework for the automation of citation collection and screening in systematic reviews, arXiv preprint arXiv:2202.10033 (2022).

[24] S. Wang, H. Scells, B. Koopman, G. Zuccon, Can chatgpt write a good boolean query for systematic review literature search?, arXiv preprint arXiv:2302.03495 (2023).

[25] S. Wang, H. Scells, A. Mourad, G. Zuccon, Seed-driven document ranking for systematic reviews: A reproducibility study, in: European Conference on Information Retrieval, Springer, 2022, pp. 686–700.
[26] S. Wang, H. Scells, M. Potthast, B. Koopman, G. Zuccon, Generating natural language queries for more effective systematic review screening prioritisation, arXiv preprint arXiv:2309.05238 (2023).
[27] S. Wang, H. Scells, B. Koopman, G. Zuccon, Neural rankers for effective screening prioritisation in medical systematic review literature search, in: Proceedings of the 26th Australasian Document Computing Symposium, 2022, pp. 1–10.
[28] I. J. Marshall, T. A. Trikalinos, F. Soboczenski, H. S. Yun, G. Kell, R. Marshall, B. C. Wallace, In a pilot study, automated real-time systematic review updates were feasible, accurate, and work-saving, Journal of Clinical Epidemiology 153 (2023) 26–33.
[29] I. J. Marshall, B. Nye, J. Kuiper, A. Noel-Storr, R. Marshall, R. Maclean, F. Soboczenski, A. Nenkova, J. Thomas, B. C. Wallace, Trialstreamer: A living, automatically updated database of clinical trial reports, Journal of the American Medical Informatics Association 27 (12) (2020) 1903–1912.
[30] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. P. Hsu, K. Wang, An overview of microsoft academic service (mas) and applications, in: Proceedings of the 24th International Conference on World Wide Web, 2015, pp. 243–246.
[31] Ö. Kart, A. Mestiashvili, K. Lachmann, R. Kwasnicki, M. Schroeder, Emati: a recommender system for biomedical literature based on supervised learning, Database 2022, baac104 (12 2022). arXiv:https://academic.oup.com/database/article-pdf/doi/10.1093/database/baac104/47779573/baac104.pdf, doi:10.1093/database/baac104.
URL https://doi.org/10.1093/database/baac104
[32] X. Qin, J. Liu, Y. Wang, Y. Liu, K. Deng, Y. Ma, K. Zou, L. Li, X. Sun, Natural language processing was effective in assisting rapid title and abstract screening when updating systematic reviews, Journal of Clinical Epidemiology 133 (2021) 121–129. doi:https://doi.org/10.1016/j.jclinepi.2021.01.010.
[33] S. Perlman-Arrow, N. Loo, N. Bobrovitz, T. Yan, R. K. Arora, A real-world evaluation of the implementation of nlp technology in abstract screening of a systematic review, medRxiv (2022). arXiv:https://www.medrxiv.org/content/early/2022/02/25/2022.02.24.22268947.full.pdf, doi:10.1101/2022.02.24.22268947.
URL https://www.medrxiv.org/content/early/2022/02/25/2022.02.24.22268947
[34] A. Bannach-Brown, P. Przybyła, J. Thomas, A. Rice, S. Ananiadou, J. Liao, M. M.R., Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error, Systematic Reviews (2019). doi:10.1186/s13643-019-0942-7.
[35] W. Kusa, A. Hanbury, P. Knoth, Automation of citation screening for systematic literature reviews using neural networks: A replicability study, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvåg, V. Setty (Eds.), Advances in Information Retrieval, Springer International Publishing, Cham, 2022, pp. 584–598.
[36] G. Kontonatsios, S. Spencer, P. Matthew, I. Korkontzelos, Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews, Expert Systems with Applications: X 6 (2020) 100030. doi:https://doi.org/10.1016/j.eswax.2020.100030.
URL https://www.sciencedirect.com/science/article/pii/S2590188520300093
[37] R. van Dinter, C. Catal, B. Tekinerdogan, A multi-channel convolutional neural network approach to automate the citation screening process, Applied Soft Computing 112 (2021) 107765. doi:https://doi.org/10.1016/j.asoc.2021.107765.
URL https://www.sciencedirect.com/science/article/pii/S1568494621006864
[38] G. Kontonatsios, S. Spencer, P. Matthew, I. Korkontzelos, Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews, Expert Systems with Applications: X 6 (2020) 100030.
[39] J. Devlin, M.-W. Chang, K. Lee, K. N. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2018, pp. 4171–4186.
[40] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).
[41] I. Beltagy, K. Lo, A. Cohan, Scibert: A pretrained language model for scientific text, arXiv preprint arXiv:1903.10676 (2019).
[42] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing (2020). arXiv:arXiv:2007.15779.
[43] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks (2019). arXiv:1908.10084.
[44] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: E. P. Xing, T. Jebara (Eds.), Proceedings of the 31st International Conference on Machine Learning, Vol. 32 of Proceedings of Machine Learning Research, PMLR, Bejing, China, 2014, pp. 1188–1196.

URL https://proceedings.mlr.press/v32/le14.html
[45] A. Grover, J. Leskovec, node2vec: Scalable feature learning for networks (2016). arXiv:1607.00653.
[46] B. K. Olorisade, P. Brereton, P. Andras, The use of bibliography enriched features for automatic citation screening, Journal of biomedical informatics 94 (2019) 103202.
[47] D. Cohn, L. Atlas, R. Ladner, Improving generalization with active learning, Machine learning 15 (1994) 201–221.
[48] B. C. Wallace, T. A. Trikalinos, J. Lau, C. Brodley, C. H. Schmid, Semi-automated screening of biomedical citations for systematic reviews, BMC bioinformatics 11 (1) (2010) 1–11.
[49] K. Hashimoto, G. Kontonatsios, M. Miwa, S. Ananiadou, Topic detection using paragraph vectors to support active learning in systematic reviews, Journal of Biomedical Informatics 62 (2016) 59–65. doi:https://doi.org/10.1016/j.jbi.2016.06.001.
URL https://www.sciencedirect.com/science/article/pii/S1532046416300442
[50] J. Portenoy, J. D. West, Constructing and evaluating automated literature review systems, Scientometrics 125 (3) (2020) 3233–3251.
[51] C. W. Belter, Citation analysis as a literature search method for systematic reviews, Journal of the Association for Information Science and Technology 67 (11) (2016) 2766–2777.
[52] J. Hou, X. Wang, J.-J. Dubois, R. B. Rice, A. Haddock, Y. Wang, Extreme systematic reviews: A large literature screening dataset to support environmental policymaking, Proceedings of the 31st ACM International Conference on Information & Knowledge Management (2022).
[53] A. Cohen, W. Hersh, K. Peterson, P.-Y. Yen, Reducing workload in systematic review preparation using automated citation classification, Journal of the American Medical Informatics Association 13 (2) (2006) 206–219. doi:https://doi.org/10.1197/jamia.M1929.
URL https://www.sciencedirect.com/science/article/pii/S1067502705002367
[54] G. Salton, Introduction to modern information retrieval, McGraw-Hill (1983).
[55] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, Advances in neural information processing systems 30 (2017).
[56] J. H. Friedman, Greedy function approximation: A gradient boosting machine., Annals of Statistics 29 (2001) 1189–1232.
URL https://api.semanticscholar.org/CorpusID:39450643
[57] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, arXiv preprint arXiv:2004.05150 (2020).
[58] A. Cohan, S. Feldman, I. Beltagy, D. Downey, D. S. Weld, Specter: Document-level representation learning using citation-informed transformers, arXiv preprint arXiv:2004.07180 (2020).