Journal of Biomedical Informatics 131 (2022) 104106

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

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

A comparative evaluation of biomedical similar article recommendation

Li Zhang [a], Wei Lu [a], Haihua Chen [b], Yong Huang [a], Qikai Cheng [a,*]

[a] School of Information Management, Wuhan University, Wuhan 430074, Hubei Province, China
[b] Department of Information Science, University of North Texas, Denton 76203, TX, USA

ARTICLE INFO

Keywords:
Biomedical article recommendation
Model evaluation
Text representation
BERT
Methodological comparison
Modeling strategy

ABSTRACT

Background: Biomedical sciences, with their focus on human health and disease, have attracted unprecedented attention in the 21st century. The proliferation of biomedical sciences has also led to a large number of scientific articles being produced, which makes it difficult for biomedical researchers to find relevant articles and hinders the dissemination of valuable discoveries. To bridge this gap, the research community has initiated the article recommendation task, with the aim of recommending articles to biomedical researchers automatically based on their research interests. Over the past two decades, many recommendation methods have been developed. However, an algorithm-level comparison and rigorous evaluation of the most important methods on a shared dataset is still lacking.

Method: In this study, we first investigate 15 methods for automated article recommendation in the biomedical domain. We then conduct an empirical evaluation of the 15 methods, including six term-based methods, two word embedding methods, three sentence embedding methods, two document embedding methods, and two BERT-based methods. These methods are evaluated in two scenarios: article-oriented recommenders and user-oriented recommenders, with two publicly available datasets: TREC 2005 Genomics and RELISH, respectively.

Results: Our experimental results show that the text representation models BERT and BioSenVec outperform many existing recommendation methods (e.g., BM25, PMRA, XPRC) and web-based recommendation systems (e.g., MScanner, MedlineRanker, BioReader) on both datasets regarding most of the evaluation metrics, and fine-tuning can improve the performance of the BERT-based methods.

Conclusions: Our comparison study is useful for researchers and practitioners in selecting the best modeling strategies for building article recommendation systems in the biomedical domain. The code and datasets are publicly available.

1. Introduction

The amount of scientific articles has been growing at an unprecedented rate in recent years. This phenomenal growth has caused locating relevant articles to become a non-trivial task in scientific research. Although academic search engines, such as Google Scholar and Microsoft Academic, and professional academic databases, such as PubMed and ACM Digital Library, have been developed for academic search, it is still a challenge for researchers, even senior researchers, to find appropriate literature.

Existing studies mainly use two strategies to help users access literature: retrieval and recommendation. The first strategy, such as the keyword-based retrievers, typically builds an inverse index to screen articles according to *keywords* given by users. Keyword-based retrievers have been very popular among academic search engines and academic databases. The second strategy usually automatically recommends the most similar articles to users based on their profiles or search histories.

Although both the article retrievers and automatic article recommenders aim to enhance the efficiency of accessing literature, their roles are largely different. As argued by Fiorini et al., the recommenders can be regarded as a complement to the retrievers [1]. Take PubMed's recommender as an example (see the "Similar Article" feature on the navigation page of a PubMed article): when a particular article within a list of articles is selected (clicked upon), this indicates to the system that the article better matches the user's information needs. The clicked information is recorded and will be used by PubMed's article recommender to suggest more articles to the user [2].

In addition to helping users access literature, article

* Corresponding author.
E-mail addresses: zlahu@foxmail.com (L. Zhang), weilu@whu.edu.cn (W. Lu), haihua.chen@unt.edu (H. Chen), yonghuang1991@whu.edu.cn (Y. Huang), chengqikai0806@163.com (Q. Cheng).

https://doi.org/10.1016/j.jbi.2022.104106
Received 10 December 2021; Received in revised form 27 May 2022; Accepted 28 May 2022
Available online 2 June 2022

Table 1
Overview of similar article recommendation approaches in biomedicine. *Article-oriented* represents whether the approach is an article-oriented recommender. *Supervised* represents whether the approach is supervised learning-based. *Method* represents whether the approach is mainly published as a novel recommendation method. *System* represents whether the approach is mainly published as a novel recommendation system or has (had) provided a system based on a novel method. *Code Available* represents whether the source code is publicly available.

Approaches	Article-oriented	Supervised	Method	System	Code Available	Key Features	Citation	Venue
PMRA	✓	×	✓	✓	×	An approach based on a probabilistic model, it is the underlying method of the "similar article" functionality of PubMed.	Lin and Wilbur (2007)	BMC Bioinformatics
PURE	×	✓	✓	✓	✓	An approach using content filtering on the set of articles that users can add/delete.	Yoneya and Mamitsuka (2007)	Genome Informatics
eTBLAST	✓	×	✓	✓	×	A web service aiming to find similar articles using weighted keywords and a text alignment algorithm.	Errami et al. (2007)	Nucleic Acids Res
PMRA-link	✓	×	✓	×	×	A graph-based method using PageRank and HITS on content-similarity networks.	Lin (2008)	BMC Bioinformatics
MScanner	×	✓	✓	✓	✓	An approach that can efficiently suggest articles to users using a Bayesian classifier.	Poulter et al. (2008)	BMC Bioinformatics
MedlineRanker	×	✓	×	✓	×	An approach using a Bayesian classifier with features extracted from nouns of article content.	Fontaine et al. (2009)	Nucleic Acids Res
PBC	✓	×	✓	×	×	An approach developed for full-text biomedical articles with similarity determined by bibliographic coupling.	Liu (2015)	PLOS ONE
XPRC	✓	×	✓	×	✓	An approach using term expansion based on PMRA.	Wei et al. (2016)	AMIA joint summits on translational science
Crow-rank	✓	✓	✓	×	×	An approach using a learning-to-rank model (SVMRank) and was trained on a crowd-sourcing corpus.	Lingeman and Yu (2016)	Arxiv
BioReader	×	✓	✓	✓	✓	An approach aiming to refine the article reading list for users with the training dataset consisting of two sets of articles.	Simon et al. (2019)	BMC Bioinformatics
LitSuggest	×	✓	×	✓	×	A web server providing not only biomedical article recommendations, but also many other useful services for users, such as searching results downloading/sharing and personalized digest delivery.	Allot et al. (2021)	Nucleic Acids Res

recommendation has also been applied to other applications. For example, in the biomedical field, article recommendation is being used for: credible datasets construction [3], entity recognition and relation extraction from biomedical articles [4,5], screening similar biomedical articles for systematic reviews [6,7], automatic Medical Subject Headings (MeSH)[1] indexing for biomedical articles [8–10], and biomedical article clustering [11–13].

In recent years, many recommendation methods or systems have been developed for biomedicine; for example, PubMed Related Article (PMRA) [14], Biomedical Research Article Distiller (BioReader) [15], and LitSuggest [16]. However, several questions come to mind, such as: what are the advantages and disadvantages of these different methods? Which method yields the best performance? Existing studies can not answer these questions since many of the existing methods are evaluated separately with different experimental settings or on non-standard datasets. In addition to these recommendation approaches, different text representation techniques have also been proposed to help understand human languages. For example, the pre-trained model (PTM) Bidirectional Encoder Representations from Transformers (BERT) [17,18] has outperformed the classical models [19,20] on many NLP tasks, such as text classification, information retrieval, sentiment analysis, and others, and might also be the most effective method among all the recommendation approaches. Therefore, other questions that naturally arise are how would text representation models compare to existing approaches in carrying out this task? and what are better modeling strategies for the article recommendation problem given that many modeling strategies have been developed?

To answer the above research questions, we conduct a *formal* evaluation and comparative study of various biomedical article recommendation methods. Before proceeding, we review existing studies with a similar purpose and summarize their contributions as well as limitations to highlight the significance of our study. According to our investigation, two studies are most relevant to our research. The first study presented an evaluation framework (CITREC) [21], which evaluated 35 similarity measures on a PubMed dataset based on a MeSH-based bibliometric indicator. However, the drawback of CITREC is that the MeSH-based indicator is not always reliable for judging article similarity because, for example, a recent gold-standard dataset [22] shows that some articles highly considered similar do not have any overlapping MeSH terms. Moreover, our statistics on the PubMed literature database show that 14% of articles do not have the MeSH metadata, and the number of MeSH terms assigned to biomedical articles varies over a large range. The second study used concept-based annotations on biomedical articles to determine the best-performing method [23]. However, the study only focused on articles with full text (many of the non-open access articles do not have the full text in reality), and only benchmarked three methods.

Our research differs from the existing studies in two aspects. Firstly, we evaluate all the methods with the same experimental settings on the same datasets: the Relevant Literature Search Consortium (RELISH) dataset [22] and the Text REtrieval Conference (TREC) 2005 Genomics dataset [24], which we will introduce in Section 4. Secondly, we also evaluate different text representation techniques in addition to these existing methods and systems. The text representation techniques we evaluate include word-level representation models (e.g., fastText [25], BioWordVec [26]), sentence-level representation models (e.g., InferSent [27], Sent2Vec [28]), document-level representation models (e.g., LDA [29], Doc2Vec [30]), and the BERT-based models (e.g., AllenAI's SPECTER [31], BioBERT [32]).

[1] see Appendix Table A1 for all abbreviations and acronyms

In summary, our contributions are threefold:

- We provide an evaluation of the article recommendation methods in the biomedicine domain. This evaluation of the 15 methods covers many existing methods and recommendation systems, as well as many text representation models that can be potentially adopted to address this problem. To the best of our knowledge, this is the first study providing such an evaluation for biomedical article recommendations. The code is available at https://github.com/carmanzhang/PSA.
- The evaluation results show that the BERT-based models significantly outperform many existing methods, e.g., PMRA. We provide data-side analysis for the best performers, including analysis of dataset bias and how fine-tuning improves the BERT-based models.
- We analyze the evaluation methods from an algorithmic point of view and compare their core modeling strategies. Through a joint analysis with the evaluation results, we highlight the characteristics of better modeling strategies.

The remaining sections are as follows: in Section 2, we review the most important recommendation methods and text representation models. In Section 3, we analyze these methods from an algorithmic perspective and compare their core modeling strategies. The experimental settings are described in Section 4, while the numerical results and the primary findings are presented in 5. We discuss several important aspects of our evaluation in Section 6. The conclusions are presented in Section 7.

2. Related works

In this section, we briefly review the existing and potential methods for biomedical article recommendations. Depending on the scenario where the recommendation methods are used, they are divided into two categories: article-oriented (AO) methods and user-oriented (UO) methods. The AO recommendation methods suggest candidate articles to a query article based on query-candidate similarity. The UO recommendation methods suggest candidate articles to a user based on the user's information needs, which are typically represented by two sets of articles, i.e., articles relevant/irrelevant to the information needs. In the following subsections, we review the most important AO and UO methods, which are itemized in Table 1. In addition, we also review several advanced text-processing techniques that can be potentially applied to biomedical article recommendations.

2.1. Article-oriented (AO) recommenders

The article-oriented recommenders are the most frequently encountered type of article recommenders. They can be found in many academic search engines and literature databases, such as Google Scholar, PubMed, ScienceDirect, ACM Digital Library, Semantic Scholar, and others. When a user clicks on a particular article, more articles similar to it will be suggested to the user on the web interface.

In biomedicine, the AO recommendation task was originally proposed by National Center for Biotechnology Information (NCBI) researchers to highlight the article recommendation problem. To address this problem, they developed PMRA [14]. By assuming that the topics of a document are represented by terms, PMRA uses Poisson distribution to model whether an article is related to a specific topic. The evaluation has shown that PMRA is statistically better than BM25. However, the method does not consider the semantic variation of terms (also known as "term mismatch"), which is a critical issue in information retrieval and recommendation [33–35]. Later, Lin [36] developed a graph-based recommender in which graph analysis algorithms [37,38] were used to re-rank the recommended articles of PMRA, and experiments showed that the graph-based re-ranking method improved the effectiveness of article recommendations.

In addition, a research team from NCBI and UC San Diego found that PMRA lowered the weight of terms that should be most directly related to an article's topics [39]. To mitigate this gap, the team proposed Extended PubMed Related Citation (XPRC). XPRC extended the original terms with five approximate terms using a skip-gram model [40], and evaluation results showed that XPRC outperformed PMRA on the TREC 2005 Genomics dataset.

Apart from the mentioned methods, a number of web servers have also been developed. PURE [41] and eTBLAST [42] are the two most well-known ones. PURE is a content-filtering-based recommender that can be reused by everyone with the standalone software package. eTBLAST searches for similar articles in two steps. First, a pool of 400 articles is gathered from PubMed using weighted keywords against all the background keywords obtained from the whole PubMed and, in a second step, the candidates are re-ranked by a sentence alignment algorithm.

2.2. User-oriented (UO) recommenders

The UO recommenders are very helpful for users because the two sets of articles (positives/negatives) should be more effective in capturing the user's information needs than a single query article. In addition, the recommenders can suggest articles dynamically by keeping track of the articles that are of interest to the user.

MScanner [43] is an early attempt. It uses all the PubMed articles as background information and trains a Bayesian classifier with the articles marked as interesting by the user. In addition, MScanner also provides easy-to-use web service. To make the service more efficient, it adopts MeSH terms and journal titles instead of the commonly used titles and abstracts for the recommendation. Much like MScanner, MedlineRanker [44] also adopts a Bayesian classifier. The main difference is that MedlineRanker uses more data: nouns in the title and abstract are selected and then are computed globally to obtain the weights of the terms.

BioReader [15] and LitSuggest [16] are the most recent attempts at UO recommenders. BioReader can refine the article reading list for a user from a large collection of biomedical articles. It first cleans the article content with a set of text mining techniques, such as stop word removal and word stemming, then uses the Mann–Whitney test to select the top representative terms from the established document-words matrix. These selected terms with term weights are further adopted to train a recommendation model. LitSuggest is a web-based recommendation system created by NCBI researchers with the aim of assisting biomedical researchers to meet their search needs. In comparison with BioReader, LitSuggest not only achieves better performance, but also offers many useful functionalities[2], e.g., model training and reuse, classification results downloading and sharing, and weekly digest delivery.

2.3. Text representation models

The methods mentioned above are based on term selection or term weighting. In other words, the accuracy of article similarity largely depends on how to select or weigh the most representative terms from article content. Such methods may be suboptimal because similar articles might be under-represented if they do not contain those critical terms. Biomedical knowledge discovery is a complex process, the same knowledge can be expressed by different terms/concepts and the meanings of a term may vary substantially in different contexts. In this regard, some shallow methods, such as the terms-based methods, can hardly reflect article similarity adequately.

Fortunately, the natural language processing techniques have made great progress recently. Many text representation models, such as BERT

[2] https://www.ncbi.nlm.nih.gov/research/litsuggest/

[17], have been developed and proven to be effective in many research tasks. Compared to the existing AO and UO recommenders, the text representation models can effectively capture the semantics of text, which may contribute to better recommendation approaches. However, the performance of the text representation models has not been explored in relation to this task.

To examine the performance of the text representation models, we consider several groups of text representation models as article recommenders: word-level representation models, sentence-level representation models, document-level representation models, and BERT-based representation models. In order to facilitate an in-depth comparative analysis, we will analyze the approaches from an algorithmic perspective and conduct an intensive empirical evaluation. The involved approaches, in addition to the mentioned text representation models, also cover some important AO and UO methods proposed previously.

3. Modeling strategy

In this section, to better understand the modeling strategy of different article recommenders, we first present a formal definition of the article recommendation task and then review two types of biomedical article recommenders based on their core modeling strategies: term-based recommenders and text representation-based recommenders.

3.1. Problem definition

Article recommendation aims to automatically suggest articles $x^c{}_i, \forall i \in [1, \ldots, N]$ to the query Q according to the query-candidate similarity r_i, where N represents the number of candidate articles involved in each run. The recommendation process can be formalized by Eq. 1, where the function φ represents a specific recommendation method.

$$r_i\left(x_i^c|Q\right) = \varphi\left(Q,\ x_i^c\right) \tag{1}$$

Depending on the recommendation scenario (AO or UO), the form of Q is different. In the AO recommendation, Q represents a query article x^q, i.e., $Q = x^q$, and the recommenders suggest the articles that are most relevant to it. However, the UO recommendation aims to recommend the most appropriate articles based on the user's information needs, which are usually represented by two sets of articles $Q = <\{x^{qpos}\}, \{x^{qneg}\}>$ that are of interest $\{x^{qpos}\}$ and of no interest $\{x^{qneg}\}$ to the user.

There are two main modeling strategies to address this problem. The first one focuses on how to extract key information from biomedical text, which has been extensively explored by existing studies. The second one focuses on how to effectively represent biomedical articles, such as the BERT models. Based on different modeling strategies, we divided the article recommendation approaches into two types: term-based recommenders and text representation-based recommenders, which we will discuss in the following subsections.

3.2. Term-based recommenders

Methods for term-based recommenders are developed based on two assumptions: (1) terms have different weights in delivering the core content of articles, and (2) the weighted terms can be used for the recommendations.

BM25 is one of the most popular methods under this category. In BM25, the terms that frequently occur in an article, but rarely occur in other articles, will be assigned a higher weight. As a simple but effective model, BM25 has been applied to many information retrieval and recommendation tasks.

Another term-based method is PMRA, which was developed for biomedicine in particular, using an elaborated weighting technique and tuned hyper-parameters on a large biomedical article repository. In PMRA, the weight of term t in an article x is represented by the following equation:

$$w_{t,x} = \frac{\sqrt{idf_t}}{1 + \left(\frac{\mu}{\lambda}\right)^{tf_t - 1} exp(-(\mu - \lambda)\cdot\ell)} \tag{2}$$

, where ℓ is the total number of terms in the article x, tf_t is the term frequency of t within x, and idf_t is the inverse document frequency of term t. Note that the parameters λ and μ denote the expected occurrence of a term when it is about the topic of x and not about the topic of the article, which can be determined by an extensive tuning process. With the weighting technique, the query-candidate similarity r can be calculated by the K exactly matched terms of the two articles, defined as

$$r(x^c|x^q) = \sum_{t=1}^{K} w_{t,x^c} * w_{t,x^q} \tag{3}$$

According to Eqs. 2 and 3, PMRA is very similar to BM25. Therefore, they also have similar advantages and disadvantages.

XPRC extends PMRA by supplementing the most similar terms for the terms of the query article. Through term expansion, XPRC is expected to establish more weighted connections between query and candidate articles. The modified similarity score is defined as:

$$w_{t,x^c} = \frac{\sqrt{idf_t}}{1 + \left(\frac{\mu}{\lambda}\right)^{p\sum_i tf_{t,i} - 1} exp(-(\mu - \lambda)\cdot\ell)} \tag{4}$$

, where $\sum_i tf_{t,i}$ represents the total frequency of the approximate terms of the original term t (including t) in the article x^c, and p is the ratio of the frequency of t in the query article x^q to the number of terms in x^q. As the weighting approach does not change, the similarity score of XPRC also can be calculated by Eq. 3.

The above comparative analysis shows that the three recommenders are different. However, the modeling strategies used in these recommenders are similar: weighting key terms. Although the modeling strategies are superior in efficiency, their limitations are also evident: semantic relatedness of terms and the position of terms are ignored. As we know, semantic information is critical for many NLP tasks, including recommendations. Note that, although XPRC integrates a word vector model to obtain term semantics, the semantics used are limited.

The user-oriented recommenders differ from the article-oriented recommenders in that the former first use a learning process to capture a user's preference from Q, and then apply the learned knowledge to prioritize candidate articles. The modeling strategy is formalized as follows:

$$\begin{cases} \sigma : Q \rightarrow T \\ \quad : [\boldsymbol{v_1},\ \ldots,\ \boldsymbol{v_M}]^\top \rightarrow [\iota_1,\ \ldots,\ \iota_M] \\ r(x^c|Q) = \sigma(x^c) \end{cases} \tag{5}$$

, where M is the total number of articles in Q, σ is the learnable function that tries to map the query (two sets of articles) to the real information need (the ground truth), $\boldsymbol{v}$ represents the extracted feature vector of an article (e.g., key terms) from Q, and T is the ground truth, with each ι representing whether or not the corresponding article is interesting to the user.

From Eq. 5 we know that the modeling strategies of the UO recommenders are similar to the AO recommenders as they are both grounded in key information weighting or selection. Therefore, both of them encounter the same challenges: semantic relatedness of terms and the position of terms being ignored.

The methods of term weighting and selection among different UO recommenders are slightly different. For example, MScanner chooses MeSH terms and journal titles as the key information instead of the more commonly used article abstracts. MedlineRanker selects nouns from part-of-speech annotations as the discriminative information. BioReader has more complex term weighting procedures. In BioReader, Term

Frequency-Inverse Document Frequency (TF-IDF) and the Mann–Whitney test are combined to weight and select the top representative terms.

3.3. Text representation-based recommenders

In the text representation-based recommenders, articles are represented by pre-trained embeddings, such as Word2Vec [40], and others, which can capture the semantic information of texts. Different from term-based modeling strategies, the text representation models do not require sophistical term selection. Instead, they focus on improving the quality of embedding that could effectively represent the core content of articles. Since every article is represented by a fixed-length embedding, it is straightforward to make recommendations based on the similarity of embeddings between articles. For AO recommendation, assuming that $\boldsymbol{e}^q$ and $\boldsymbol{e}^c$ are the embeddings of the query x^q and candidate article x^c, the semantic similarity r can be measured by the cosine similarity of $\boldsymbol{e}^q$ and $\boldsymbol{e}^c$ using Eq. 6.

$$r(x^c|x^q) = \frac{\boldsymbol{e}^q \cdot \boldsymbol{e}^c}{\|\boldsymbol{e}^q\| \cdot \|\boldsymbol{e}^c\|} \tag{6}$$

$$\begin{cases} \sigma : Q \rightarrow T \\ \quad : [\boldsymbol{e}_1, \ \ldots, \ \boldsymbol{e}_M]^\top \rightarrow [l_1, \ \ldots, \ l_M] \\ r(x^c|Q) = \sigma(x^c) \end{cases} \tag{7}$$

Likewise, by replacing the features of Eq. 5 with embeddings, the modeling strategy of the UO scenario recommendation can be formalized by Eq. 7.

Eqs. 6 and 7 indicate that the modeling strategy of text representation-based recommenders mainly relies on embeddings. In other words, the quality of embeddings will largely determine the performance of the recommenders. Recently, many different embedding techniques, such as word embeddings, sentence embeddings, document embeddings, and BERT embeddings, have been proposed, either in the general domain or in the biomedical domain. All of them could be used to capture semantic information from biomedical articles.

3.3.1. Word embeddings

The word embedding models capture the semantic information at the word level. The models first map an article x to a word embedding matrix $\boldsymbol{W}^x = [\boldsymbol{e}_1, \ \ldots, \ \boldsymbol{e}_\ell]^\top$ with each row representing the embedding of a specific word, and then use an average pooling technique to compress the matrix into a single d-dimensional embedding $\boldsymbol{e}^x = \frac{1}{|\boldsymbol{W}^x|}\sum_{\boldsymbol{e}\in\boldsymbol{W}^x}\boldsymbol{e}$. fastText [25] and BioWordVec [26] are the representative models of this group. Based on the assumption that words fit well within their own context, the models learn word embedding through predicting the context words surrounding the given words with the skip-gram model [40]. The training objective is to maximize the following log-likelihood:

$$\sum_{i=1}^{S}\sum_{c\in\mathscr{C}_i}\log p(t_c|t_i) \tag{8}$$

, where the context $\mathscr{C}_i$ is the set of words surrounding t_i, and S is the number of training samples. The probability of observing a context word t_c given t_i will be computed by their word embeddings.

The word embedding-based approaches have the capability to learn word semantics to avoid the term mismatch issue in the term-based approaches. However, since they leverage the pooling technique, the word position is ignored in this process. Without the position information, the models are unable to learn global information [45,46].

3.3.2. Sentence or document embeddings

The sentence or document embedding models generate article representation for the whole document, and differ from the word embedding models, which are at the word level. Sent2Vec [28] and InferSent [27] are the two most outstanding models among all the sentence embedding models. Sent2Vec expands word embedding into the sentence level. The training strategy of Sent2Vec is similar to word embedding models. However, the main difference is that Sent2Vec considers n-gram (n consecutive words) embeddings. Eq. 9 shows how to derive the Sent2Vec embedding from a list of n-gram embeddings $R(x)$ of article x.

$$\boldsymbol{e}^x = \frac{1}{|R(x)|}\sum_{\boldsymbol{e}\in R(x)}\boldsymbol{e} \tag{9}$$

Since n-gram partially considers word position, Sent2Vec is likely to yield more meaningful article embeddings. Compared to Sent2Vec, InferSent works in a different way; it is a supervised model trained on the Stanford Natural Language Inference (SNLI) [47] dataset. The training framework accepts two inputs (paired sentences from SNLI) and maps them to embeddings e_1 and e_2 with an encoder network, then the composed embeddings $[e_1 \oplus e_2 \oplus |e_1 - e_2| \oplus e_1 * e_2]$ are passed into a fully connected network to obtain more effective predictions, which are used for backpropagation and supervised learning. Since training on SNLI requires a high-level understanding of language and involves reasoning about the semantic relationships within sentences, the model can yield high-quality embeddings.

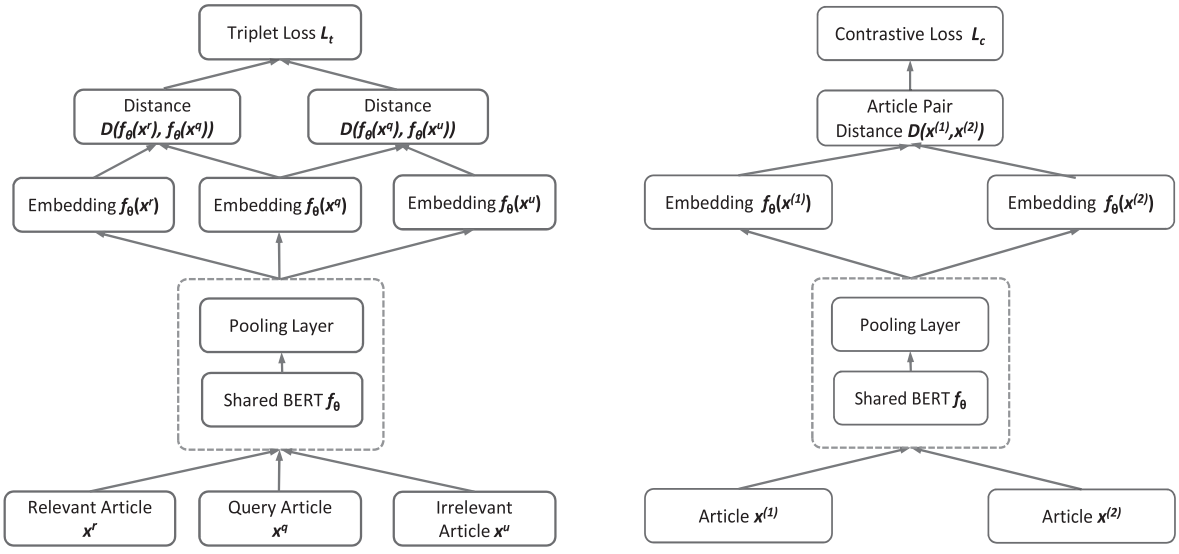
Another group of methods represents articles with document-level embeddings, such as LDA and Doc2Vec. In the LDA model, articles are represented by topical distribution, and a particular topic is represented by a set of weighted words. Therefore, as a bag-of-word model, LDA suffers from the same issue as the term-based methods. In terms of Doc2Vec, the training approach is highly dependent on the method of learning word embeddings. Doc2Vec initializes the document embeddings randomly and uses the averaged word vectors to update the document embeddings. This training approach, therefore, makes it difficult for Doc2Vec to learn global information.

3.3.3. BERT-based recommender

BERT is the state-of-the-art language representation model developed based on multi-layer bidirectional Transformer encoders [48]. The Transformer encoder is an attention structure that can effectively capture semantic information from texts. Furthermore, the Transformer encoder also considers word position: the position is encoded as a part of the input for capturing in-depth semantics. Therefore, the limitations in the previous models, such as term mismatch and ignoring word position, could be largely resolved with BERT.

Fine-tuning BERT with a siamese network Applying BERT to the recommendation task is straightforward: we make the recommendation based on the similarity between a query and a candidate article (both of them are represented by embeddings) by using Eq. 6 or Eq. 7. However, the BERT models are pre-trained on a generic corpus, meaning that they can hardly achieve the optimal performance for a specific task. As another part of BERT, fine-tuning has proven to be a promising technique to obtain more effective models [48]. Therefore, it is worthwhile to explore BERT fine-tuning for the biomedical article recommendation task. However, there is no one-size-fits-all method for fine-tuning, so we need to design a fine-tuning strategy for different tasks. To tune a BERT recommender, the BERT model should accept more than one input (i.e., query and candidate articles). A commonly adopted approach is to combine them as a single input and feed them to the network. However, this manner will result in a very high computational overhead. To overcome this issue, we used Sentence-BERT (SBERT) [18], a siamese network architecture built on top of BERT. SBERT can efficiently learn high-quality text representation from logically related inputs (query and candidates in this task) with the input individually processed by a siamese network. With the SBERT architecture, we elaborate on how to fine-tune BERT for the two recommendation circumstances, respectively.

Tuning BERT for the AO scenario In this recommendation scenario,



(a) Fine-tuning BERT with the triplet loss for article-oriented recommendation

(b) Fine-tuning BERT with the contrastive loss for user-oriented recommendation

Fig. 1. Fine-tuning BERT for better article recommenders. The triplet loss and the contrastive loss are adopted for fine-tuning BERT in the article-oriented and user-oriented recommendation scenarios, respectively.

we used the triplet loss to tune the BERT model, where a valid training instance is a triplet: a query article x^q, and two kinds of candidates x^c, i.e., relevant candidates and the irrelevant candidates, denoted by x^r and x^u, respectively. Assuming that f is a BERT network with parameters θ, and $f_\theta(*)$ is the function that can project articles to embeddings, the training objective with the triplet loss will, as formalized in Eq. 10, try to minimize $D(x^q, x^r)$ and maximize the $D(x^q, x^u)$, where $D(x^q, x^r)$ represents the distance between query article x^q and relevant article x^r, $D(x^q, x^u)$ is the distance between query article x^q and irrelevant article x^u, $\| * \|_2$ represents the Euclidean distance, and α is a margin between the positive and negative pairs. Fig. 1a demonstrates how to fine-tune a BERT model with the triplet loss; the triplet input is first encoded to embeddings, then $D(x^q, x^r)$ and $D(x^q, x^u)$ are calculated by the Euclidean distance on the embeddings. Last, the loss will punish the model when the distance between x^q and x^u is less than the distance between x^q and x^r by at least α.

$$\begin{aligned}\mathscr{L}_t(x^q, x^r, x^u) = &\max\left(D^2(x^q, x^r) - D^2(x^q, x^u) + \alpha, 0\right) \\ &= \max(\|\mathrm{f}_\theta(x^q) - \mathrm{f}_\theta(x^r)\|_2^2 \\ &-\|\mathrm{f}_\theta(x^q) - \mathrm{f}_\theta(x^u)\|_2^2 + \alpha, 0)\end{aligned} \tag{10}$$

Tuning BERT for the UO scenario In the UO scenario, the fine-tuning process should learn the user's information preferences from the relevant and irrelevant articles. In this regard, we used the contrastive loss $\mathscr{L}_c$ to minimize the distance between the positives while maximizing that distance between the negatives. This is in line with our intuition because similar articles should be closer to each other than to the irrelevant ones. The contrastive loss used in this article is formalized

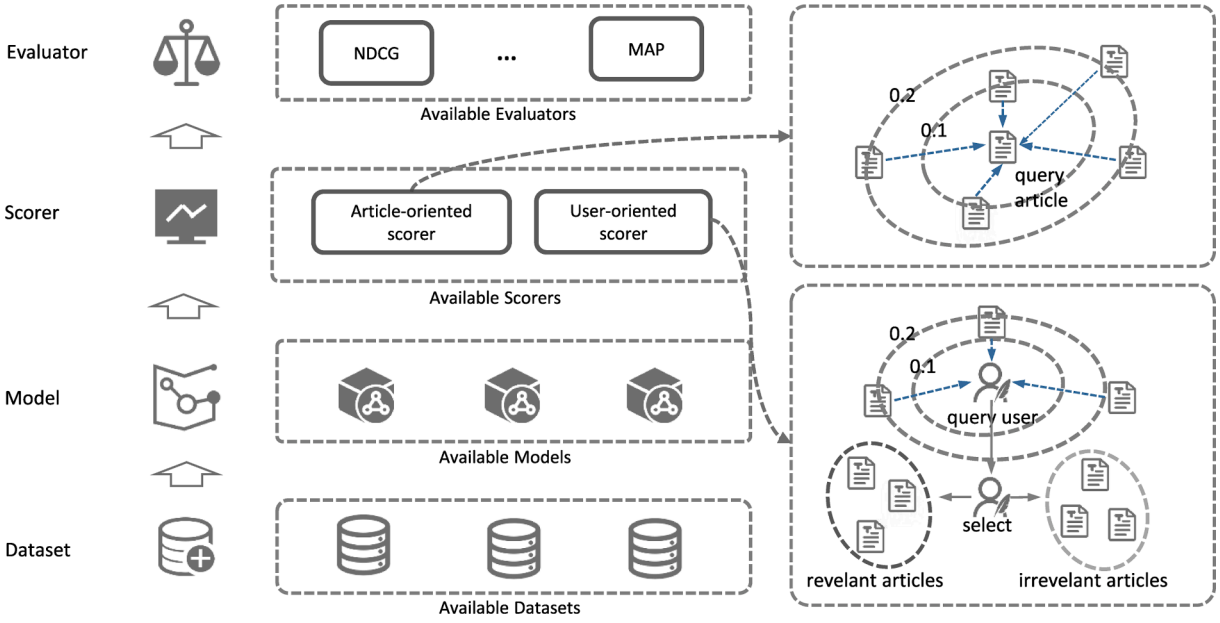


Fig. 2. The evaluation workflow for biomedical article recommendation.

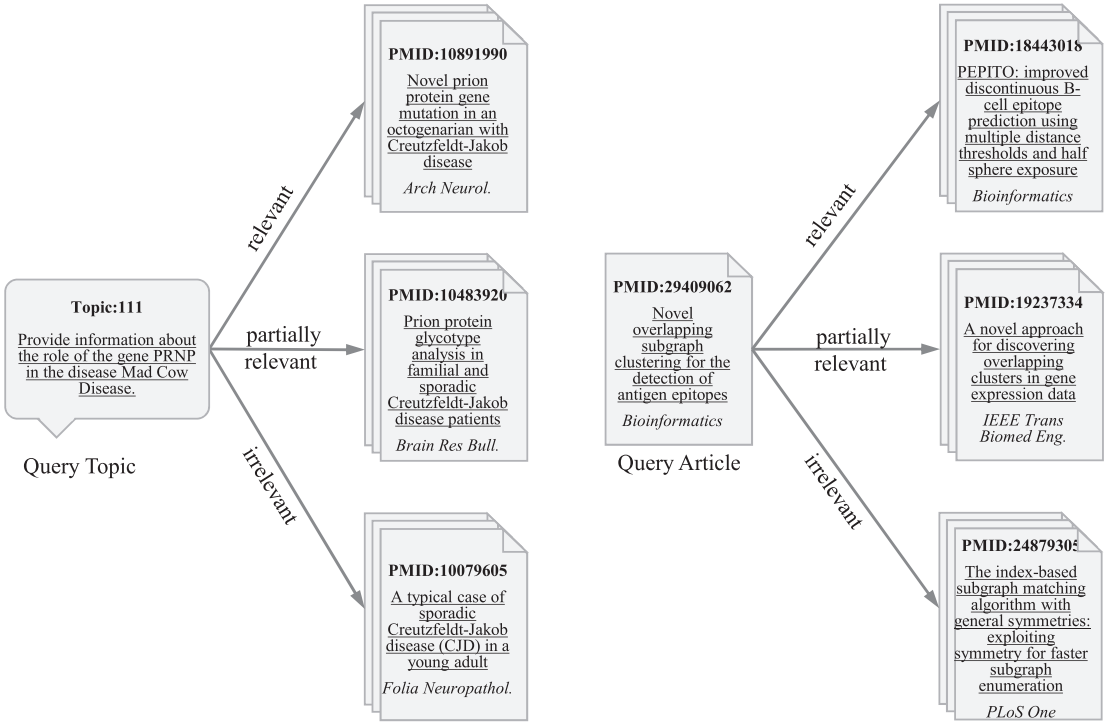


Fig. 3. The structure of the evaluation datasets.

in Eq. 11, where $x^{(1)}, x^{(2)}$ are the paired input (here, we do not use the symbols x^q, x^r, x^u as there are no explicit query articles in the UO recommendation scenario), and y indicates the label regarding whether two articles are related; 1 means the distance should be reduced and vice versa. The tuning process for this scenario is depicted in Fig. 1b. Likewise, articles are fed to the network individually to derive text embeddings, then the embedding distances are calculated to fine-tune the model.

$$\mathscr{L}_c(x^{(1)}, x^{(2)}) = \frac{1}{2}[y \cdot D^2(x^{(1)}, x^{(2)}) - (1-y) \cdot \{\max(\alpha - D(x^{(1)}, x^{(2)}), 0)\}^2] \tag{11}$$

4. Evaluation workflow and experimental setup

In this section, we describe the evaluation workflow and the experimental settings.

4.1. Evaluation workflow

We evaluate the recommendation methods with the workflow showing in Fig. 2. The workflow consists of four stages: dataset, model, scorer, and evaluator. We specify the datasets $DS = \{DS_i\}_{i=1}^{\ell_d}$ (where ℓ_d is the number of datasets) in the first stage. The training stage aims to develop several recommendation models based on the datasets. Assuming the evaluation models are $MD = \{MD_i\}_{i=1}^{\ell_m}$, where ℓ_m denotes the number models; this stage will train some MD on DS if necessary. In the third stage, the evaluation models act as scorers and make inferences on the test dataset $DS^{(t)} = \{DS_i^{(t)}\}_{i=1}^{\ell_d}$. Regarding the recommendation scenarios $SC = \{SC_i\}_{i=1}^{\ell_s}$, we adapted the workflow so that it can be used to evaluate the methods in two scenarios: article-oriented and user-oriented scenarios ($\ell_s = 2$, accordingly). In the last stage, the performance is evaluated based on the predictions of the previous stage. After all the stages are finished, the evaluation metrics of the models MD on the datasets DS and in the two recommendation scenarios SC can be obtained.

4.2. Evaluation datasets

TREC 2005 Genomics The dataset was initially developed for testing retrieval experiments using defined topics and similarity judgments and was later adopted for testing article recommendation approaches [14]. The topics are characterized by descriptive sentences (see Fig. 3a), e.g., *Provide information about the role of the gene PRNP in the disease Mad Cow Disease*, which not only reflect the real information needs of biomedical researchers but also give the reason why two particular articles under the same topic are similar. The similarity judgments are on three levels: relevant, partially relevant, and irrelevant. However, due to the graded similarity that is established between an article and a specific topic, it is impossible to conduct an evaluation for the article recommendation task without repurposing the topic-to-article structured dataset to the article-to-article structured dataset [24]. To this end, we select partial articles under each topic as queries and leave the others as candidates.

RELISH RELISH is a large dataset aimed at benchmarking biomedical similar article recommenders specifically. It was curated via crowdsourcing, with more than 1,500 biomedical scientists from various research areas participating in the annotation process. Collectively, over 180,000 biomedical articles have been included. In RELISH, a query article is associated with multiple candidates, and each candidate is tagged with one of the three similarity scores: relevant, partially relevant, and irrelevant (see Fig. 3b for the structure of RELISH). Regarding the quality, the dataset has been rigorously evaluated. For example, the authors show that there is no systematical bias observed among annotators with different levels of background, and the scores judged by different annotators are quite stable [22].

To benchmark the learnable AO methods (e.g., BERT), we split the

Table 2
Experimental results of article-oriented article recommenders on the RELISH dataset

Method Group	Method	MAP@5	MAP@10	MAP@15	NDCG@5	NDCG@10	NDCG@15	AVG.
-	Random	79.33	77.22	75.41	80.70	77.67	76.40	77.79
Term-Based Method	XPRC	84.34	81.98	80.59	85.32	82.43	81.78	82.74
	BM25	88.91	86.72	84.54	89.48	87.39	86.21	87.21
	PMRA	90.30	87.57	85.75	90.95	88.40	87.45	88.40
Word Embedding	fastText	85.75	82.81	81.79	86.79	83.79	83.12	84.01
	BioWordVec	89.84	86.51	84.67	89.90	86.67	85.53	87.19
Sentence Embedding	InferSent	85.21	82.16	80.41	86.56	83.31	82.35	83.33
	WikiSentVec	87.92	85.23	83.40	88.65	85.74	84.81	85.96
	BioSentVec	90.76	88.10	86.16	90.05	87.76	86.89	88.29
Document Embedding	LDA	85.44	82.66	80.36	86.51	82.91	81.31	83.20
	Doc2Vec	86.23	84.74	83.39	86.55	84.70	84.09	84.95
BERT	BioBERT	88.14	85.81	83.90	88.97	86.29	85.10	86.37
	SPECTER	92.27	90.00	88.36	91.47	89.12	88.42	89.94
BERT with fine-tuning	BioBERT	**94.11**	**92.10**	**90.64**	92.85	90.72	89.93	91.73
	SPECTER	93.76	91.65	90.39	**93.40**	**91.20**	**90.52**	**91.82**

queries of the two datasets into the standard training/validation/test sets following the ratio of 8:1:1. However, benchmarking UO methods requires a learnable model to capture the user's information preferences. To facilitate this process, we split *the candidate articles under each query* into the training/validation/test folds with the same ratio of 8:1:1 for training, validation, and evaluation.

4.3. Methods in the comparative study

The comparative study covers 15 methods, which are divided into two categories: term-based recommenders and text representation-based recommenders. The first category includes three AO methods and three UO methods developed for biomedical article recommendations. Among these, PMRA and MScanner have been used to provide literature recommendations for a wide range of biomedical researchers. Note that we eliminated a recently created term-based method, LitSuggest [16], from the comparison because the implementation details are not provided. The second category corresponds to the text representation-based methods that can serve as both the AO and UO recommenders. In this category, we evaluate nine text representation models, including two word embedding methods, three sentence embedding methods, two document embedding methods, and two BERT-based methods, which make recommendations based on different levels of the semantics of articles. All the evaluation methods used in this article are itemized as follows.

4.3.1. Term-based recommenders

BM25 Although many advanced computational models have been developed in recent decades, BM25 remains a strong baseline in information retrieval or recommendation. For comparison, we used the default hyper-parameters $k1 = 1.5, b = 0.75$, and $\epsilon = 0.25$.

PMRA PMRA is a probability model proposed by [14]. It has been integrated into PubMed as an important feature to power users' searching experience (see "Similar Article" in the navigation page of a PubMed article). For comparison, we used the optimized parameters $\lambda = 0.022$ and $\mu = 0.013$ suggested by Lin and Wilbur [14].

XPRC Extended from PMRA, XPRC expands terms with an additional five similar terms using pre-trained word vectors. To replicate this method, we used BioWordVec [26] as the pre-trained word vectors.

MScanner MScanner is a Bayesian classifier-enabled article recommender. The web service embedded in MScanner can prioritize articles efficiently by using MeSH terms and journal titles.

MedlineRanker Different from MScanner, MedlineRanker uses nouns from the title and abstract to build the recommendation model.

BioReader The method uses document-word matrices and the Mann–Whitney test to select the top representative terms, then uses supervised learning algorithms to build the recommendation model. We replicated BioReader based on the author's implementation. Note that multiple learning algorithms are provided in their implementation. We reported the performance of BioReader with the support vector machine (SVM) implementation because the setup shows better performance than the other three top performers on RELISH.

4.3.2. Text representation-based recommenders

Word Embedding One of the basic methods for measuring article similarity is averaging over word embeddings. Here, we considered two popular pre-trained word embeddings: fastText [25] and BioWordVec [26]. In terms of the experimental settings, we used 300d fastText and 200d BioWordVec.

Sentence Embedding We evaluated the two sentence embeddings, e.g., InferSent [27] and Sent2Vec [28], which are popular models for solving many biomedical problems, such as evidence-based clinical data mining [49] and biomedical literature understanding [50]. Note that we considered two versions of Sent2Vec, referred to as BioSentVec (trained on the PubMed corpus) and WikiSentVec (trained on the Wikipedia corpus).

Document Embedding We evaluated LDA [29] and Doc2Vec [30] for this task as both can generate document-level embeddings for articles of arbitrary length. To train the models, 2% of the PubMed Central articles (approx. 48 k full-text articles) were randomly selected as the training corpus, and the number of topics was set to 64 for both models.

BERT Embedding We considered two pre-trained BERT models: AllenAI's SPECTER [31] and BioBERT [32], which have been widely used in many academic text processing tasks. SPECTER was trained on a massive amount of academic articles (Semantic Scholar open corpus[3]), with citation relationship integrated to enhance its ability in downstream tasks, such as scholarly recommendation. BioBERT is a domain-specific model, which was pre-trained on large-scale biomedical data. To efficiently tune these models, we adopted the SBERT architecture (see Fig. 1) to speed up the training/inference process. In terms of the parameter settings, we set the maximum epochs to 3, batch size to 16, and the learning rate was set to 1e-5 as suggested by Sun et al. [51]. The first 1,500 training steps were used for warming up the model. We evaluated the model every 3,000 steps and saved the best model when the validation loss reached the minimum. We used the maximum input length of 200 due to our GPU memory constraints. It should be pointed out that the mapping function σ (see Eq. 5 and Eq. 7) is indispensable for the development of the UO methods, here we adopted SVM as the mapping function. Note that we did not fine-tune the BERT models on the TREC dataset because it is hard to obtain high-quality training samples due to the topic-article structure of the dataset.

[3] https://api.semanticscholar.org/corpus

Table 3
Experimental results of article-oriented article recommenders on the TREC Genomics dataset.

Method Group	Method	MAP@5	MAP@10	MAP@15	NDCG@5	NDCG@10	NDCG@15	AVG.
-	Random	31.54	30.74	29.28	43.43	44.66	43.86	37.25
Term-Based Method	XPRC	49.33	47.31	45.18	59.21	58.46	58.06	52.93
	BM25	46.48	44.53	41.89	58.18	56.68	55.25	50.50
	PMRA	47.83	45.40	42.38	59.50	57.64	55.85	51.43
Word Embedding	fastText	50.05	47.18	44.61	60.81	57.96	56.47	52.85
	BioWordVec	50.89	48.57	46.25	61.28	59.64	58.77	54.23
Sentence Embedding	InferSent	48.16	45.00	42.45	58.46	56.32	55.15	50.92
	WikiSentVec	55.04	52.06	49.30	64.19	61.72	60.38	57.11
	BioSentVec	**56.53**	**53.46**	**50.78**	**65.74**	**63.31**	**62.28**	**58.68**
Document Embedding	LDA	38.59	38.05	36.23	51.46	51.11	50.19	44.27
	Doc2Vec	43.49	41.77	39.30	54.67	53.10	51.73	47.34
BERT	BioBERT	52.75	48.92	45.97	63.35	60.64	58.67	55.05
	SPECTER	54.71	50.85	48.41	62.84	60.87	60.13	56.30

Table 4
Experimental results of user-oriented recommenders on the RELISH dataset.

Method Group	Method	MAP@5	MAP@10	MAP@15	NDCG@5	NDCG@10	NDCG@15	AVG.
-	Random	78.14	76.32	75.72	80.73	77.65	76.71	77.55
Term-Based Method	MScanner	87.19	84.92	83.73	87.16	84.48	83.21	85.12
	MedlineRanker	88.69	86.33	85.32	88.10	85.60	84.36	86.40
	BioReader	87.84	85.65	90.48	88.69	85.18	87.02	87.48
Word Embedding	fastText	88.88	86.73	85.23	88.35	85.79	84.13	86.52
	BioWordVec	89.24	87.17	86.00	88.59	86.04	84.58	86.94
Sentence Embedding	InferSent	89.17	87.11	86.36	88.57	86.05	84.93	87.03
	WikiSentVec	90.09	87.97	86.83	89.16	86.81	85.55	87.74
	BioSentVec	**91.03**	**89.15**	88.16	89.89	87.63	86.65	88.75
Document Embedding	LDA	86.22	83.70	83.43	86.46	83.51	82.86	84.36
	Doc2Vec	88.29	85.89	84.64	87.99	85.12	83.62	85.93
BERT	BioBERT	89.56	87.01	86.17	89.71	87.38	86.70	87.76
	SPECTER	90.65	88.49	87.54	90.52	88.66	87.78	88.94
BERT with fine-tuning	BioBERT	90.81	88.59	88.04	**90.81**	**88.88**	**88.20**	**89.22**
	SPECTER	90.91	88.66	**88.23**	90.66	88.74	88.09	**89.22**

Table 5
Experimental results of user-oriented recommenders on the TREC Genomics dataset.

Method Group	Method	MAP@5	MAP@10	MAP@15	NDCG@5	NDCG@10	NDCG@15	AVG.
-	Random	19.50	20.17	16.97	25.84	27.21	25.12	22.47
Term-Based Method	MScanner	40.25	39.23	37.91	46.30	47.04	46.95	42.95
	MedlineRanker	51.80	47.92	45.67	53.97	52.98	52.10	50.74
	BioReader	52.71	52.47	50.95	58.68	60.85	59.70	55.89
Word Embedding	fastText	54.24	53.05	52.14	61.50	61.18	61.28	57.23
	BioWordVec	57.77	55.00	52.92	**64.78**	64.51	63.06	59.67
Sentence Embedding	InferSent	51.40	50.61	49.11	56.47	57.88	57.63	53.85
	WikiSentVec	55.74	53.53	52.80	60.59	59.94	60.64	57.21
	BioSentVec	**59.95**	**58.88**	**57.05**	64.67	**65.82**	**65.73**	**62.02**
Document Embedding	LDA	45.90	45.90	43.66	51.92	54.72	52.81	49.15
	Doc2Vec	47.96	46.93	45.95	51.55	52.87	52.57	49.64
BERT	BioBERT	52.88	53.06	51.19	55.25	58.72	58.90	55.00
	SPECTER	55.98	53.30	51.13	62.36	59.47	58.45	56.78

4.4. Evaluation metrics

We used the standard ranking metrics MAP and NDCG for performance assessment as article recommendation is a typical ranking problem. In the two evaluation datasets, the similarities are graded into three levels, we followed two existing studies [14,39] to transform the three levels to the corresponding similarity scores: 0, 1, and 2, and reported the top-N performance of MAP and NDCG in percentages with N set to $[5, 10, 15]$. Note that we considered the relevant and partially relevant levels as the same similarity score in calculating MAP (i.e., similarity scores are 1), as prior studies [14,39] did. However, we considered them separately in calculating NDCG because the two similarity levels will lead to different cumulative gains.

5. Results and analysis

Tables 2–5 present the evaluation results of all the methods in the article-oriented and user-oriented scenarios, and on the TREC Genomics and the RELISH datasets, respectively. We summarized our observations and conducted the analysis from five aspects.

5.1. Term-based modeling strategies

In Table 2 and Table 3, we observed that BM25 outperformed several text-representation models (e.g., LDA, Doc2Vec), and even showed comparable performance with the BERT-based models (e.g., BioBERT, SPECTER) on RELISH. This is not surprising because BM25 is a strong baseline and has been broadly adopted in information retrieval and recommendation systems.

In terms of other term-based AO recommenders, PMRA outperformed BM25 on both datasets, which is aligned with previous findings [14,39]. As per the methodologies compared in Section 3, the advanced term weighting technique and the exhaustively tuned hyperparameters enable PMRA to be a better performer than BM25 on biomedical articles.

We also found that XPRC did not perform well compared to PMRA on RELISH. However, it showed better performance than PMRA on the TREC dataset. By analyzing the extended terms from the two datasets, we found that the extended terms for the TREC dataset were slightly more discriminative than the extended terms for RELISH. Such differences might be the reason that made XPRC perform better on the TREC dataset.

When further looking at Table 2 and Table 3, we observed the performance gaps between the term-based methods and the text representation-based models on the RELISH dataset are smaller than those on the TREC dataset. This difference might also be caused by the domain coverage of the datasets. Since the TREC dataset only covers the Genomics domain (vs. RELISH, which covers the full spectrum of biomedicine domains), the candidate articles, regardless of whether they are relevant or irrelevant articles, should have many genomics-related terms in common. This makes the term-based methods less discriminative.

Table 4 and Table 5 show the results of the user-oriented benchmarks. The findings of the benchmarks are similar to that of the article-oriented benchmarks. For example, the three web-based systems (MScanner, MedlineRanker, and BioReader) are mainly inferior to many text representation models, which suggests that the advanced representation techniques can help build better recommendation systems, although some of the web-based recommendation systems have already received good feedback from users. As discussed in the *Modeling Strategy* section, the three recommendation systems are all grounded in term weighting/selecting, and indeed they have the same limitations (e.g., term mismatch and missing term positions) as the evaluated article-oriented recommenders. Additionally, among the three recommenders, MScanner performed the worst and BioReader showed better results in general than the others on both datasets. The reason MScanner did not perform well is that, in order to quickly return recommendations from the massive number of PubMed articles, only the journal titles and MeSH terms were used, meaning that limited knowledge was used from the input for the recommendation. Regarding BioReader, it embedded a set of feature engineering techniques to select the most significant terms, which leads to better performance.

5.2. Text representation-based modeling strategies

Although some term-based methods show decent performance, they are suboptimal compared to several text representation-based models (e.g., Sent2Vec, BERT). For example, in Table 2, there are four text representation-based models that outperformed PMRA on RELISH with the maximum margin being 3.82%. In Table 3, there are five text representation-based models that outperformed XPRC on the TREC dataset with the maximum margin being 5.75%. In Table 4, there are six text representation-based models that outperformed BioReader on the RELISH dataset with the maximum margin being 1.74%. In Table 5, there are five text representation-based models that outperformed BioReader on the TREC dataset with the maximum margin being 6.13%. The superiority of Sent2Vec and the BERT models can be explained by the appropriate modeling strategy of the representation models. By using neural networks and advanced techniques – e.g., attention mechanics, word position encoding, skip-gram, and n-gram word encoding – the models can learn semantics very well from global information of articles while considering word position, which largely resolved the issues faced by term-based methods. The superiority also indicates that the in-depth capture of semantics from articles is an essential characteristic of better modeling strategies.

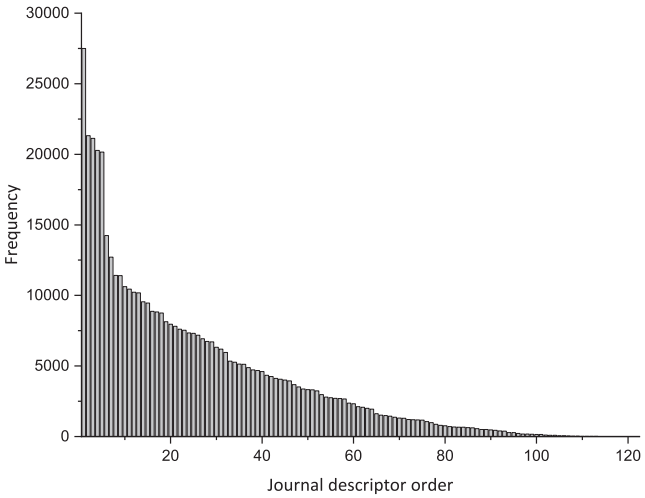


Fig. 4. The distribution of journal descriptor (JD) frequency in RELISH. JDs were detected from the test set of RELISH using the JDI tool. The horizontal axis represents JD orders ranging from [1, 122], and the vertical axis represents journal descriptor frequency (the number of articles falling into a particular journal descriptor).

Additionally, we also have several findings by comparing different types of text representation models. First, LDA and Doc2Vec show the worst performance overall. As we know, LDA is essentially a bag-of-word model, which makes LDA suffer from the same issue as the term-based methods. Doc2Vec updates article embeddings with averaged word vectors; this training technique makes it difficult for Doc2Vec to learn global information. Second, InferSent achieved comparable performance with LDA and Doc2Vec on the two datasets. This is surprising as this model has been shown to generalize well on many tasks. A deeper analysis shows that the poor performance may be caused by the training dataset. InferSent was trained on SNLI, which is a dataset consisting of image captions from the web. Therefore, the domain knowledge between SNLI and biomedical literature differs significantly, and such a knowledge gap could explain why InferSent did not perform well on the biomedicine datasets. Third, the word embeddings, such as fastText and BioWordVec, achieved more moderate results than BERT and Sent2Vec in most scenarios. This conclusion is coincident with existing studies [52], as using word embeddings with the pooling technique has inherent limitations; e.g., the information loss issue and not taking account of term positions.

Furthermore, the fine-tuned BERT models (e.g., SPECTER, BioBERT) outperformed many strong baselines on RELISH. For example, the fine-tuned SPECTER improved PMRA by 3.42% in terms of the *AVG.* metric in Table 2). Such remarkable improvement indicates that the cutting-edge BERT models might be the optimal methods for the biomedical similar article recommendation task.

5.3. Data aspect modeling strategies

In this subsection, we present additional findings from the data perspective that are also critical for building a better article recommender.

First, the domain-specific models, such as BioSentVec and BioWordVec, outperformed their generic equivalents (e.g., InferSent, WikiSentVec, and fastText). This conclusion was strongly supported by the BioSentVec model, which outperformed most methods and even outperformed the original BERT models on the TREC dataset, as shown in Table 3 and Table 5. We believe this is due to the divergence of domain knowledge learned in these models. BioSentVec and BioWordVec were trained specifically on biomedicine datasets (e.g., PubMed literature data, and clinical notes from the MIMIC-III Clinical

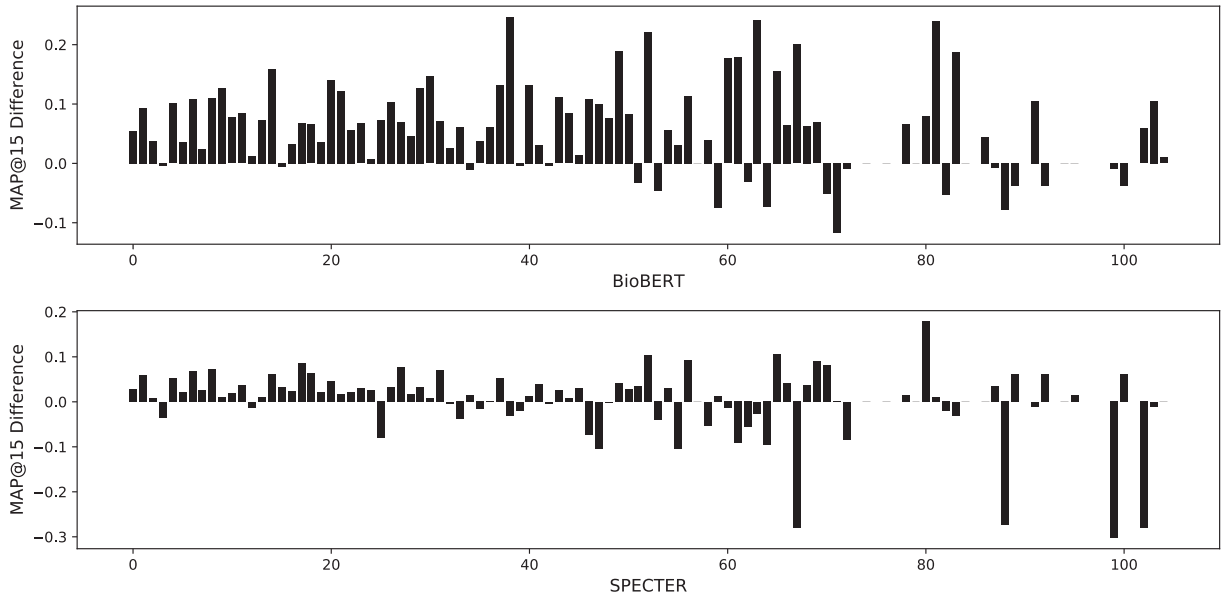


Fig. 5. Performance comparison (MAP@15) of the BERT models after/before fine-tuning, the horizontal axis represents the journal descriptor orders shown in Fig. 4, and the vertical axis is the performance gaps between fine-tuned model and the original model. Bars above the horizontal axis suggest fine-tuning has a positive effect on the BERT models, and vice versa.

database), enabling them to learn more domain knowledge (e.g., biomedical entities/concepts) than their generic equivalents [53,54]. The above conclusion can also be demonstrated by the InferSent model, which performs well on a variety of tasks [27]. However, our evaluation results indicate that InferSent fails to achieve the expected performance as the other sentence-embedding models. The reason, as aforementioned, is that SNLI contains little biomedicine knowledge. Also, the performance gap between MScanner and MedlineRanker can prove this inference. The two models are very similar except for the information used – MedlineRanker uses the full abstract and it supposedly has obtained more domain knowledge than MScanner.

Second, we found SPECTER outperformed BioBERT in all the scenarios, although both SPECTER and BioBERT have learned considerable domain knowledge[4]. The main difference between SPECTER and BioBERT is that SPECTER incorporated citation relationship to improve document-level representations [31], indicating the effectiveness of the citation information in determining article relevance [55]. The above-discussed findings demonstrate that integrating more knowledge from data is also a useful modeling strategy for enhancing article recommenders.

5.4. *Fine-tuning the pre-trained models for performance improvement*

As can be seen from Table 2 and Table 4, BERT with fine-tuning significantly outperformed the original BERT models on RELISH, meaning that fine-tuning the pre-trained models is an effective strategy for performance improvement, aligned with the conclusion from [53]. To better understand how fine-tuning improved the pre-trained models, we used the Journal Descriptor Indexing (JDI) tool [56] to decompose the test instances of RELISH into the individual research disciplines of biomedicine. Under this experimental setting, we then examined the difference in model performance between the fine-tuned BERT and the original BERT models across disciplines[5].

The JDI tool was developed by the National Library of Medicine (NLM) for categorizing biomedical text. JDI has been successfully used in many applications, such as automatic indexing of biomedical articles [57] and author name disambiguation [58]. For an article, JDI can index it with a ranked list of Journal Descriptors (JDs), which correspond to biomedicine disciplines. In this research, we extracted the top three disciplines for each PubMed article and aggregated the articles into their respective disciplines. After this step, the RELISH test set was decomposed to 122 discipline-specific datasets (see Fig. 4) containing varying numbers of articles. Then, we evaluated the performance of the BERT recommenders for each JD (discipline).

Fig. 5 shows the performance variances of the BERT models via fine-tuning, where the vertical bars represent the performance gaps between the fine-tuned models and the original models, while the bars over the horizontal axis indicate a positive effect of fine-tuning. The horizontal axis is the discipline order, with discipline ranked by frequency (the number of articles) in descending order (see Fig. 4). From Fig. 5, we found that most negative bars of MAP@15 appear with higher JD orders, while, for smaller orders, the bars are mainly above the horizontal axis. When looking at the JD distribution shown in Fig. 4, we draw the conclusion that fine-tuning improves the performance of larger disciplines but might not affect the performance of small disciplines. In some situations, fine-tuning might even distort the model performance of small disciplines. In other words, whether fine-tuning can improve the BERT models or not largely depends on the size of the available training samples [53]. We refer to this as *performance bias* as it shows unbalanced performance improvements across disciplines. The performance bias is critical for recommenders in the production environment because recommenders with the issue will offer diverse searching experiences for users in different research areas.

[4] A portion of SPECTER's training data is biomedical articles.

[5] Note that we did not provide the same analysis for the TREC dataset as it only focuses on the Genomics discipline while RELISH contains the full spectrum of biomedicine disciplines. In addition, it is also hard to obtain meaningful training samples for fine-tuning from the TREC dataset, as clarified in Section 4.

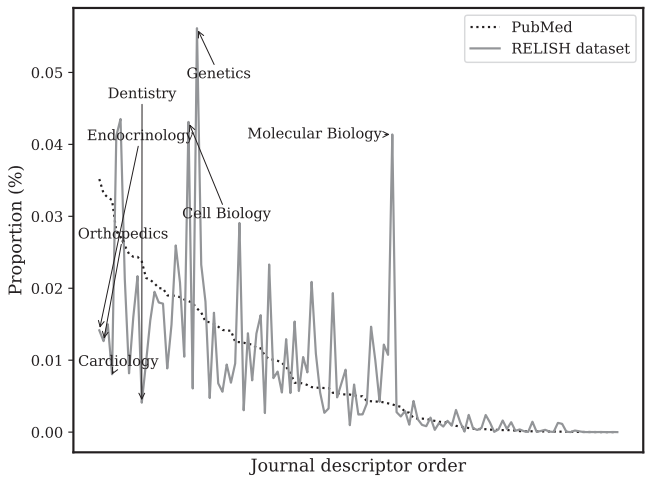


Fig. 6. Journal descriptor (discipline) distribution (%) in RELISH and the whole PubMed. Different from the x-axis of Fig. 4, the x-axis represents the ranked JD of the whole PubMed literature database. Compared to PubMed's JD distribution, RELISH's distribution shows clear deviations from the real distribution.

5.5. Dataset bias analysis

Inspired by the performance bias, we further investigated whether there is a significant bias in the RELISH dataset in terms of biomedicine disciplines. The creators of the RELISH dataset found that RELISH may have a slight over-representation of those publications related to the *high-throughput omics technologies* [22]. They used a qualitative approach, i.e., word cloud, to put more emphasis on *diversity* instead of *numerical distribution*. However, how the bias is distributed across all biomedicine disciplines remains unclear. Uncovering the disciplinary bias of the RELISH dataset should be important as it can help others to recognize the limitations of RELISH and, more importantly, to better understand in which disciplines the RELISH-based article recommenders may perform suboptimally.

To quantify the disciplinary bias, we compared the JD distribution of RELISH to that of PubMed. We used the JDI tool to extract the disciplines from all the PubMed articles[6]. The discipline distribution is shown in Fig. 6; we can see that some disciplines, such as Genetics, Cell Biology, and Molecular Biology, pinpointed in this plot, show significant deviations from the background distribution, which is aligned with the conclusion in [22]. In addition to the three disciplines, our analysis also uncovers the bias issue in more disciplines, such as Endocrinology, Orthopedics, Cardiology, and Dentistry.

6. Discussion

6.1. Better modeling strategies

Our evaluation shows that the recommendation methods with different modeling strategies achieved various levels of performance. However, several common findings can still be identified. These findings collectively highlighted the characteristics of better modeling strategies.

The term-based methods developed fine-grained term weighting/selection techniques, e.g., term weighing in PMRA and noun selection in MedlineRanker. These measures are indeed helpful for improving the effectiveness of recommenders. However, the modeling strategies of the group of methods face the same limitations: semantic relatedness and the positions of terms are not seriously considered. The limitations restrict the term-based methods from deeply mining the relationship between articles, while this aspect is critical for achieving better recommendation performance. Fortunately, the text representation-based models can handle such issues appropriately. With advanced techniques – e.g., attention mechanics and word position encoding, the modeling strategies enable the recommenders to capture the semantics of articles.

Additionally, another helpful modeling strategy learned from our analysis is to integrate more knowledge from data. BioSentVec and BioWordVec with more integrated domain knowledge outperformed their equivalents (WikiSentVec and fastText) trained on the generic-domain corpus. SPECTER with citation relationship integrated also outperformed another BERT model. The findings demonstrate that, in addition to improving recommendations from an algorithmic perspective, incorporating more knowledge from a data perspective is also a valuable modeling strategy that can effectively boost biomedical article recommenders.

6.2. Method contributions and potential value for future works

This article evaluated a variety of biomedical article recommendation methods, covering many existing approaches and additional text representation models, and spanning two recommendation scenarios. In our evaluation, we demonstrated that many text representation models can be used to develop effective recommenders. We thoroughly analyzed the evaluation methods and compared their limitations and strengths from an algorithmic perspective.

Furthermore, we demonstrated that fine-tuning can improve the BERT models in both article-oriented and user-oriented recommendation scenarios. The tuned BERT models outperformed existing approaches by remarkable margins (e.g., approximately 3.4% improvements over PMRA on RELISH); such huge improvements and recommendation methods can have at least two implications for future works. First, the promising methods may benefit worldwide biomedical scientists if integrated into PubMed. A query analysis for PubMed showed that there were approximately 2.5 million users accessing PubMed and 3 million searches issued on a working day in 2017 [59]. As such, improving article recommendations for PubMed is becoming a crucial topic given the considerable number of users. Second, because the fine-tuning technique and most text representation models are generic, they can be effortlessly scaled to other academic digital libraries/literature databases to power the literature access experience for a wide range of researchers beyond biomedicine.

Additionally, the methods benchmarked here are helpful for identifying better modeling strategies, e.g., mining more semantics and integrating more domain (or externally compiled) knowledge, which would be of great interest for many other research problems and applications relying on an understanding of biomedical articles, such as screening biomedical articles for systematic reviews [6,7], biomedical article clustering [11–13], automatic MeSH indexing [8–10], and data curation in biomedicine [3]. The modeling strategies are valuable for building more effective methods used in these tasks.

6.3. Future improvements

The RELISH dataset is of high quality in terms of annotation accuracy. However, in terms of discipline distribution, clear deviations were found. This issue may result in poor recommendation performance for the under-represented disciplines. Therefore, future works can shift more attention to building a large unbiased dataset for this task.

This study investigated similar article recommendations primarily from a semantic perspective. However, the similarity may not be entirely determined by a single perspective/factor. Other perspectives are also worth exploring; for example, whether an article is cited by another one, whether both articles are published in the same journal or

[6] The 2019 baseline version, which contains nearly 30 million articles

written by the same author. Such perspectives offer new interpretations for article similarity beyond the content information.

We argue that another way to improve this task is to integrate *user intelligence* [1]. When the query article and a certain candidate article are frequently viewed by a large group of users, it may imply that the query and the candidate articles are highly related. In this sense, future works can consider mining user intelligence from user behavior data, such as query logs [60], to power article recommendations.

6.4. Limitations

The evaluation presented in this article was only carried out in the offline mode (on evaluation datasets); thus, it may not reflect the actual performance of the recommendation methods in realistic scenarios. The actual performance can be measured by conducting a large-scale A/B test in the online mode. Unfortunately, it is hard for us to conduct such experiments on a mature literature system. Despite this limitation, we believe this work still makes valuable contributions. The intensive evaluation and the in-depth analysis of the recommendation approaches will provide insights for future studies.

7. Conclusion

This study evaluated 15 article recommendation methods in biomedicine. The evaluation methods include not only existing methods but also advanced text representation techniques, such as BERT. The evaluation results showed that many text representation models outperformed the existing recommendation methods and systems. In addition to the empirical evaluation, we also compared these methods and analyzed their limitations from an algorithmic perspective. All these efforts helped us to identify better modeling strategies for biomedical article recommendations. Furthermore, we provided data-aspect analysis, e.g., dataset bias in terms of discipline distribution. The analysis is helpful for others to better understand the evaluation datasets and the best-performing methods, which will eventually benefit article recommendation research. In the future, we intend to develop an effective method/criterion that can be used for online article recommendations and performance evaluation.

Declaration of Competing Interest

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

Acknowledgment

This work was supported by the National Key Research and Development Program of China [No.2019YFB1404702]. The authors are also very grateful to the editor and the anonymous reviewers for their constructive comments and suggestions.

Appendix A. Appendix: A

.

Table A1
Abbreviations and acronyms used in this article.

PMRA	PubMed Related Article
XPRC	Extended PubMed Related Citation
BioReader	Biomedical Research Article Distiller
TF-IDF	Term Frequency-Inverse Document Frequency
SVM	Support Vector Machine
PTM	Pre-Trained Model
BERT	Bidirectional Encoder Representations from Transformer
SBERT	Sentence BERT
TREC	Text Retrieval Conference
RELISH	Relevant Literature Search Consortium
AO	Article-oriented
UO	User-oriented
JD	Journal Descriptor
JDI	Journal Descriptor Indexing
MeSH	Medical Subject Headings
NLM	National Library of Medicine
NCBI	National Center for Biotechnology Information
SNLI	Stanford Natural Language Inference

References

[1] N. Fiorini, R. Leaman, D.J. Lipman, Z. Lu, How user intelligence is improving pubmed, Nat. Biotechnol. 36 (10) (2018) 937–945.
[2] N. Tran, P. Alves, S. Ma, M. Krauthammer, Enriching PubMed related article search with sentence level co-citations, Amia Annu Symp Proc (2009) 650–654.
[3] R. Islamaj, W.J. Wilbur, N. Xie, N.R. Gonzales, N. Thanki, R. Yamashita, C. Zheng, A. Marchler-Bauer, Z. Lu, PubMed Text Similarity Model and its application to curation efforts in the Conserved Domain Database, Database 2019 (2019) 1–13. https://doi.org/10.1093/database/baz064.
[4] J. Li, Y. Sun, R.J. Johnson, D. Sciaky, C.H. Wei, R. Leaman, A.P. Davis, C. J. Mattingly, T.C. Wiegers, Z. Lu, BioCreative V CDR task corpus: a resource for chemical disease relation extraction, Database, Oxford University Press, 2016. https://doi.org/10.1093/database/baw068.
[5] R. Islamaj, R. Leaman, S. Kim, D. Kwon, C.-H. Wei, D.C. Comeau, Y. Peng, D. Cissel, C. Coss, C. Fisher, et al., NLM-Chem, a new resource for chemical entity recognition in PubMed full text literature, Sci. Data 8 (2021) 1–12.
[6] B.C. Wallace, T.A. Trikalinos, J. Lau, C. Brodley, C.H. Schmid, Semi-automated screening of biomedical citations for systematic reviews, BMC Bioinf. 11 (2010) 1–11.
[7] X. Ji, A. Ritter, P.-Y. Yen, Using ontology-based semantic similarity to facilitate the article screening process for systematic reviews, J. Biomed. Inform. 69 (2017) 33–42.
[8] Y. Mao, Z. Lu, MeSH Now: automatic MeSH indexing at PubMed scale via learning to rank, J. Biomed. Semant. 8 (2017) 1–9.
[9] X. Xun, K. Jha, Y. Yuan, Y. Wang, A. Zhang, MeSHProbeNet: a self-attentive probe net for MeSH indexing, Bioinformatics 35 (19) (2019) 3794–3802.
[10] S. Peng, R. You, H. Wang, C. Zhai, H. Mamitsuka, S. Zhu, DeepMeSH: deep semantic representation for improving large-scale MeSH indexing, Bioinformatics 32 (12) (2016) i70–i79.
[11] G.. Jun, W. Feng, J. Zeng, H. Mamitsuka, S. Zhu, Efficient semisupervised MEDLINE document clustering with MeSH-semantic and global-content constraints, IEEE Trans. Cybern. 43 (4) (2012) 1265–1276.
[12] W.B.A. Karaa, A.S. Ashour, D. Ben Sassi, P. Roy, N. Kausar, N. Dey, Medline text mining: an enhancement genetic algorithm based approach for document clustering, in: Appl. Intell. Optim. Biol. Med., Springer, 2016, pp. 267–287.
[13] K.W. Boyack, C. Smith, R. Klavans, A detailed open access model of the PubMed literature, Sci. Data 7 (1) (2020) 1–16.
[14] J. Lin, W.J. Wilbur, PubMed related articles: a probabilistic topic-based model for content similarity, BMC Bioinf. 14 (2007) 1–14, https://doi.org/10.1186/1471-2105-8-423.
[15] C. Simon, K. Davidsen, C. Hansen, E. Seymour, M.B. Barnkob, L.R. Olsen, BioReader: a text mining tool for performing classification of biomedical literature, BMC Bioinf. 19 (2019) 165–170.
[16] A. Allot, K. Lee, Q. Chen, L. Luo, Z. Lu, LitSuggest: A web-based system for literature recommendation and curation using machine learning, Nucl. Acids Res. 49 (W1) (2021) W352–W358.
[17] J. Devlin, M.-W. Chang, K. Lee, K. 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 vol. 1, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186.
[18] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992.
[19] L. Gao, Z. Dai, T. Chen, Z. Fan, B. Van Durme, J. Callan, Complement lexical retrieval model with semantic residual embeddings, Eur. Conf. Inf. Retr. (2021) 146–160.
[20] C. Bhagavatula, S. Feldman, R. Power, W. Ammar, Content-based citation recommendation, in: Proc. 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. vol. 1 (Long Pap., Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 238–251. https://doi.org/10.18653/v1/N18-1022.
[21] Bela Gipp, Norman Meuschke, Mario Lipinski, CITREC: An Evaluation Framework for Citation-Based Similarity Measures based on TREC Genomics and PubMed Central, in: Proceedings of the iConference, iSchools, 2015.
[22] P. Brown, A.-C. Tan, M.A. El-Esawi, T. Liehr, O. Blanck, D.P. Gladue, G.M. F. Almeida, T. Cernava, C.O. Sorzano, A.W.K. Yeung, et al., Large expert-curated database for benchmarking document similarity detection in biomedical literature search, Database (Oxford) 2019 (2019) 1–67, https://doi.org/10.1093/database/baz085.
[23] L. Jael, G. Castro, R. Berlanga, A. Garcia, In the pursuit of a semantic similarity metric based on UMLS annotations for articles in PubMed Central Open Access, J. Biomed. Inform. 57 (2015) 204–218.
[24] W. Hersh, A. Cohen, J. Yang, R.T. Bhupatiraju, P. Roberts, M. Hearst, Trec 2005 genomics track overview, in: Proc. TREC, 2005. https://trec.nist.gov/pubs/trec14/papers/GEO.OVERVIEW.ps.
[25] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguistics 5 (2017) 135–146.
[26] Y. Zhang, Q. Chen, Z. Yang, H. Lin, Z. Lu, BioWordVec, improving biomedical word embeddings with subword information and MeSH, Sci. Data 6 (1) (2019) 1–9.
[27] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes, Supervised learning of universal sentence representations from natural language inference data, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 670–680.
[28] M. Pagliardini, P. Gupta, M. Jaggi, Unsupervised learning of sentence embeddings using compositional n-gram features, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 528–540.
[29] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet Allocation, in: T.G. Dietterich, S. Becker, Z. Ghahramani (Eds.), Adv. Neural Inf. Process. Syst. 14 [Neural Information Syst. Nat. Synth. NIPS 2001, December 3-8, 2001, Vancouver, Br. Columbia, Canada], MIT Press, 2001, pp. 601–608. https://proceedings.neurips.cc/paper/2001/hash/296472c9542ad4d4788d543508116cbc-Abstract.html.
[30] Q. V Le, T. Mikolov, Distributed Representations of Sentences and Documents, in: Proc. 31th Int. Conf. Mach. Learn. 2014, Beijing, China, 21-26 June 2014, JMLR.org, 2014, pp. 1188–1196. http://proceedings.mlr.press/v32/le14.html.
[31] A. Cohan, S. Feldman, I. Beltagy, D. Downey, D. Weld, SPECTER: Document-level Representation Learning using Citation-informed Transformers, in: Proc. 58th Annu. Meet. Assoc. Comput. Linguist., Association for Computational Linguistics, Online, 2020, pp. 2270–2282. https://doi.org/10.18653/v1/2020.acl-main.207.
[32] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.
[33] C. Carpineto, G. Romano, A survey of automatic query expansion in information retrieval, Acm Comput. Surv. 44 (1) (2012) 1–50.
[34] L. Nie, H. Jiang, Z. Ren, Z. Sun, X. Li, Query expansion based on crowd knowledge for code search, IEEE Trans. Serv. Comput. 9 (5) (2016) 771–783.
[35] J. Singh, A. Sharan, A new fuzzy logic-based query expansion model for efficient information retrieval using relevance feedback approach, Neural Comput. Appl. 28 (9) (2017) 2557–2580.
[36] J. Lin, PageRank without hyperlinks: Reranking with PubMed related article networks for biomedical text retrieval, BMC Bioinformatics 9 (2008) 1–12.
[37] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: Bringing order to the web, Stanford InfoLab (1999).
[38] J.M. Kleinberg, Hubs, Authorities, and communities, ACM Comput. Surv. 31 (1999).
[39] W. Wei, R. Marmor, S. Singh, S. Wang, D. Demner-Fushman, T.-T. Kuo, C.-N. Hsu, L. Ohno-Machado, Finding related publications: extending the set of terms used to assess article similarity, AMIA Summits Transl. Sci. Proc. 2016 (2016) 225.
[40] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, in: Y. Bengio, Y. LeCun (Eds.), 1st Int. Conf. Learn. Represent. ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Work. Track Proc., 2013. http://arxiv.org/abs/1301.3781.
[41] T. Yoneya, H. Mamitsuka, Pure: a pubmed article recommendation system based on content-based filtering, Genome Inf. 18 (2007) 267–276.
[42] M. Errami, J.D. Wren, J.M. Hicks, H.R. Garner, eTBLAST: a web server to identify expert reviewers, appropriate journals and similar publications, Nucl. Acids Res. 35 (2007) W12–W15.
[43] G.L. Poulter, D.L. Rubin, R.B. Altman, C. Seoighe, MScanner: A classifier for retrieving Medline citations, BMC Bioinf. 9 (2008) 1–12.
[44] J.F. Fontaine, A. Barbosa-Silva, M. Schaefer, M.R. Huska, E.M. Muro, M.A. Andrade-Navarro, MedlineRanker: Flexible ranking of biomedical literature, Nucl. Acids Res. 37 (2009) 141–146. https://doi.org/10.1093/nar/gkp353.
[45] H. Gholamalinezhad, H. Khosravi, Pooling Methods in Deep Neural Networks, a Review, ArXiv Prepr. ArXiv2009.07485 (2020).
[46] N. Akhtar, U. Ragavendran, Interpretation of intelligence in cnn-pooling processes: a methodological survey, Neural Comput. Appl. 32 (3) (2020) 879–898.
[47] S.R. Bowman, G. Angeli, C. Potts, C.D. Manning, A large annotated corpus for learning natural language inference, in: L. Màrquez, C. Callison-Burch, S. Jian, D. Pighin, Y. Marton (Eds.), Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, The Association for Computational Linguistics, 2015, pp. 632–642.
[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is All you Need, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S.V.N. Vishwanathan, R. Garnett (Eds.), Adv. Neural Inf. Process. Syst. 30 Annu. Conf. Neural Inf. Process. Syst. 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.
[49] Q. Chen, J. Du, S. Kim, W.J. Wilbur, Z. Lu, Deep learning with sentence embeddings pre-trained on biomedical corpora improves the performance of finding similar sentences in electronic medical records, BMC Med. Inform. Decis. Mak. 20 (2020) 1–10.
[50] A. Allot, Q. Chen, S. Kim, R. Vera Alvarez, D.C. Comeau, W.J. Wilbur, Z. Lu, LitSense: making sense of biomedical literature at sentence level, Nucl. Acids Res 47 (2019) W594–W599.
[51] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: China Natl, Conf. Chinese Comput. Linguist., 2019, pp. 194–206.
[52] N.S Tawfik, M.R Spruit, Evaluating sentence representations for biomedical text: Methods and experimental results, J. Biomed. Inform. 104 (2020) 103396.
[53] H. Chen, J. Chen, J. Ding, Data evaluation and enhancement for quality improvement of machine learning, IEEE Trans. Reliab. 70 (2) (2021) 831–847.
[54] H. Chen, W. Lei, J. Chen, W. Lu, J. Ding, A comparative study of automated legal text classification using random forests and deep learning, Inf. Process. Manag. 59 (2) (2022) 102798.
[55] R.L. Liu, Passage-based bibliographic coupling: An inter-article similarity measure for biomedical articles, PLoS ONE 10 (10) (2015) 1–22.
[56] S.M. Humphrey, C.J. Lu, W.J. Rogers, A.C. Browne, Journal descriptor indexing tool for categorizing text according to discipline or semantic type, in: AMIA Annual Symposium Proceedings, American Medical Informatics Association, 2006, p. 960.
[57] A. Névéol, S.E. Shooshan, S.M. Humphrey, J.G. Mork, A.R. Aronson, A recent advance in the automatic indexing of the biomedical literature, J. Biomed. Inform. 42 (5) (2009) 814–823.
[58] D. Vishnyakova, R. Rodriguez-Esteban, K. Ozol, F. Rinaldi, Author Name Disambiguation in MEDLINE Based on Journal Descriptors and Semantic Types, in: S. Ananiadou, R. Batista-Navarro, K.B. Cohen, D. Demner-Fushman, P. Thompson (Eds.), Proc. Fifth Work. Build. Eval. Resour. Biomed. Text Mining, BioTxtM@COLING 2016, Osaka, Japan, December, 2016, The COLING 2016 Organizing Committee, 2016, pp. 134–142. https://aclanthology.org/W16-5115/.
[59] N. Fiorini, D.J. Lipman, Z. Lu, Cutting edge: towards PubMed 2.0, Elife 6 (2017).
[60] Z. Lu, W. Kim, W.J. Wilbur, Evaluation of query expansion using mesh in pubmed, Inform. Retriev. 12 (1) (2009) 69–80.