Mining Biomedical Literature: An Open Source and Modular Approach

Hayda Almeida[1], Ludovic Jean-Louis[2], and Marie-Jean Meurs[1,3(✉)]

[1] Centre for Structural and Functional Genomics, Concordia University, Montreal, QC, Canada
[2] NetMail, Montreal, QC, Canada
[3] Université du Québec à Montréal, Montreal, QC, Canada
meurs.marie-jean@uqam.ca

Abstract. This paper presents the ongoing development of a full-text natural language search engine for biomedical literature. The system aims to provide search on the full-text content of documents belonging to a database composed of scientific articles, while allowing users to submit their search queries using natural language. Beyond the text content of articles, the system engine also utilizes article metadata, empowering the search by considering extra information from picture and table captions. User queries can be submitted to the system in natural language, releasing the user from the burden of translating their search needs into a query language.

Keywords: Natural language processing · Information retrieval · Full-text search · Document index · Natural language query · Search engine · Biomedical literature

1 Introduction

Scientific researchers and health care practitioners heavily rely on the retrieval of biomedical documents maintained in scientific databases to support their activities. Much effort has been put into improving the retrieval of bioliterature [14,15,30]. However, bioliterature search being essentially an information retrieval task, still imposes great challenges. PubMed (http://www.ncbi.nlm.nih.gov/pubmed) is one of the most popular scientific databases, and a substantial resource for biomedical professionals. It contains over 24 million records as of January 2016. In PubMed Central (PMC) (http://www.ncbi.nlm.nih.gov/pmc/) researchers have access to 3,7 million free full-article texts (Jan. 2016), which represent a fraction of all records maintained in PubMed.

The retrieval of documents in open literature databases is a critical step for biomedical research. The retrieved results can be used as input for a variety of tasks, such as data integration [17,18], literature curation [13], and literature triage [1,12]. However, the retrieval of relevant articles is challenging for researchers using these databases. With the goal of improving the search for

R. Khoury and C. Drummond (Eds.): Canadian AI 2016, LNAI 9673, pp. 168–179, 2016.
DOI: 10.1007/978-3-319-34111-8_22

open scientific literature, this work is an attempt to address two issues that scientific researchers can encounter when gathering relevant literature.

First, the search process in PubMed and PMC presents limitations. PubMed makes available a large amount of records, but its search engine retrieves articles by considering only the abstract content. The PMC search engine retrieves articles considering their full text content, but it only holds a portion of all PubMed records. Second, search requests utilized to retrieve information from these databases have to be translated into query language. As query language differs from natural language, not all users are comfortable enough to translate their search needs efficiently, which makes the task of retrieving relevant data even more difficult.

2 Related Work

We present here studies conducted towards improving and supporting document retrieval in open-access scientific literature databases. Also, we present a review of approaches developed to handle complex user queries, that aim to facilitate information search, better address search needs, and improve the retrieval results. Enhancing the document retrieval process will allow to provide more useful results to various research tasks relying on the input of scientific literature search.

2.1 Scientific Database Search

A variety of methods has been studied in an effort to improve document retrieval relevance of scientific databases. The approaches described here have evaluated the use of full-text articles, image captions, and annotations to enrich the search results, as well as techniques to re-rank retrieved documents. Many studies [4,8,11,22,26] reported that performing search in the full content of bio-literature documents, or in the article metadata, can improve the quality of the search results. The use of full-text articles to improve scientific literature retrieval was described in [8]. The authors aimed to extend the Medical Text Indexer (MTI) (http://ii.nlm.nih.gov/MTI), a tool that provides MeSH terms recommendations for experts working on the indexation of biomedical documents at the U.S. National Library of Medicine. In an evaluation conducted with a dataset composed of 500 articles from 17 journal issues, the authors claimed that the use of full-text articles yields improvement on recall of search results.

In other studies [4,11], the image captions content was used instead of the articles full-text to support the document retrieval in scientific databases. The methods described by the authors were implemented in such a way that the query search in image captions is performed separately from the query search on the article text. In [15], Lu elaborated an overview of 28 free web-based systems for retrieval of general biomedical literature. All systems analyzed in this overview utilized PubMed or similar databases as data source. Among all approaches listed in [15], the most common ones used to improve the relevance of document retrieval were document clustering, and result re-ranking. In [25], the

authors presented a bioliterature search engine for data from PubMed and PMC. The approach also includes an annotation step, in which relevant entities are extracted from the article content, and used to support users on the search task.

Several studies have compared PubMed search results with Google Scholar search results [20,22]. In [22] the authors emphasize that PubMed searches only target article abstracts, while Google Scholar searches target the full-text content of documents. The reported results show that Google Scholar retrieves twice as many relevant documents than PubMed among the first result rank positions for clinical questions queries. In [20] the authors analyze search results provided by PubMed and Google Scholar on four clinical questions. The authors in [20] also show that Google Scholar results have better relevance than PubMed ones: the top 20 articles from Google Scholar articles tend to have a higher number of citations when compared with PubMed articles. In [26], the authors made use of full-text articles in an annotation task. The task aimed at curating GO terms from article content, and the authors addressed the importance of taking the full-text of articles into account when performing gene function curation. The study observed that article abstracts would contain 30 % of all GO terms found in a document, while the other article sections would contain 70 % of all GO terms annotated in a document.

2.2 Complex Query Processing

Handling complex queries in information retrieval tasks has been studied as a way of facilitating the search process for users. It is hence of critical interest to develop systems that allow users to submit natural language queries to search engines. Research has been conducted toward this goal [7,9,10,16] using query pre-processing, term suggestion, term expansion, and entity annotation. Facilitating literature search in scientific databases is a meaningful concern. Several studies [5,22,30] have demonstrated that searching PubMed can be a difficult and time-consuming task. PubMed users with highly specific search needs end up reformulating queries frequently [5]. Also, it has been noted that only a small number of clinical practitioners uses advanced options to generate PubMed queries [22]. Moreover, the majority of PubMed queries are submitted by inexperienced users [30], who have trouble expressing concepts with MeSH terms, and finally end up performing their search using natural language terms.

In [25], the authors used entity annotation to support handling complex user queries. The annotations are also used to label user search needs, and the different labels determine which document fields should receive a boost to improve result ranking. In [31], three different query expansion methods were applied in a search task handling patient clinical notes. In this work, query expansion was shown to increase recall, but decrease precision. The authors explained this as a possible result of noise introduced by Unified Medical Language System (UMLS) [3] annotations. In [9], the authors described an approach to handle natural language queries. Users submit search questions that are reformulated into a query composed of PubMed search terms and controlled vocabulary terms. After receiving the PubMed results for this search, GO terms are extracted from

the retrieved abstracts. All the complex query handling approaches described here presented some limitations. Even though [7,9,16] managed to process natural language queries, the search was restricted to article abstracts. The system described in [25], at the time our study was conducted, and to the best of our knowledge, has not been publicly released, and was last updated in 2011.

3 Methodology

We describe here the strategy implemented in bioMine to improve the retrieval of biomedical literature, and address the issues described in Sects. 1 and 2. bioMine combines two approaches. First, the full-text of journal articles is indexed, along with relevant article metadata. This task is handled by the *document indexation module*. Second, the queries are processed from their natural language format, as submitted by users, and enriched with biomedical terms provided by the UMLS Metathesaurus. This task is handled by the *complex query module*. The corpus of journal articles utilized in the development of bioMine is further described in Sect. 3.1. The *document indexation module* and the *complex query module* are described in more details in Sects. 3.2 and 3.3. bioMine and its modules are implemented in Java.

3.1 Corpus Description

The search engine described here was developed using the open-access scientific articles provided by PubMed and PMC. The articles are part of the PubMed Baseline Database (BD) files, and the PMC Open Access (OA) Subset repository (http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/). PubMed and PMC are open access databases managed by the U.S. National Institutes of Health's National Library of Medicine (NIH/NLM). PubMed holds life science journal articles, citations, and books. PMC contains the digital archive of biomedical and life sciences journal literature. The PMC OA Subset holds part of the complete PMC collection, in which all documents are available under the Creative Commons (https://creativecommons.org/about/license/) license. As of January 2016, PubMed BD files contain over 24,350,000 entries, with publication years since 1809. The PMC OA Subset contains over 1,200,000 journal articles, with publication years since 1973. PubMed BD documents are available in XML format, while PMC OA documents are available in NXML format. These file formats are standardized according to two different Document Type Definitions (DTD) managed by the Journal Article Tag Suite (JATS) (http://jats.nlm.nih.gov/). In total, 25,403,053 documents from PubMed BD and PMC OA were used to generate the bioMine search index. To generate the index, a set of specific XML tags was extracted from all files, and used to represent each document entry. The list of tags utilized in our experiments is detailed in Sect. 3.2.

3.2 Document Indexation Module

The bioMine indexation module is built based on the open-source search platform Solr (http://lucene.apache.org/solr/). When indexing documents with Solr, a

Table 1. Indexed fields in bioMine, and their availability in PubMed BD and PMC OA

Field	PubMed BD	PMC OA
1 Article title	✓	✓
2 Journal title	✓	✓
3 Abstract	✓	✓
4 Body section titles	✗	✓
5 Body full content	✗	✓
6 Author names	✓	✓
7 Reference authors	✓	✓
8 Reference title	✗	✓
9 Reference IDs	✗	✓
10 Object captions	✗	✓
11 PMCID	✓	✓
12 PMID	✓	✓
13 Article keywords	✗	✓
14 Publication year	✓	✓

document is considered as a set of key/value pairs where the keys are index *fields* and the values represent the indexed content. Since the files provided by PubMed BD and PMC OA datasets are XML documents, we built a parser to process each file, and populate the index set of fields. The bioMine parser retrieves tags from many XML document sections, using their content to semantically represent each article in the index. The XML tags used for the bioMine document indexation are considered as bioMine document fields. Table 1 shows the list of chosen fields, and the availability of fields according to document provenance.

The content extracted from tags in XML and NXML documents are kept as is. Granularity is an important factor when indexing documents, since it defines how many index entries will be associated with each XML document. For instance, each section of an article could be assigned an individual index entry. Since bioMine goal is to support the discovery of articles, we choose to have one index entry for each article instead. The bioMine engine indexes the full content of an article body. This content, in addition to the other document fields, provides a semantic representation of a document content in the index. Searches of journal articles in bioMine are handled by the complex query module, which is further explained in Sect. 3.3.

3.3 Complex Query Module

User queries submitted to bioMine are handled by the complex query module. The bioMine query module accepts and processes queries submitted in natural language. In addition, bioMine queries are separated in types according to the perceived user need, and are processed under different strategies to help improve the result relevance. To identify the user needs, and label a query, this module performs a first syntactic analysis in the user query. This step assigns one of

the three bioMine *types* to the user query. Each query type utilizes a different search strategy. After being assigned a type, queries can be expanded with UMLS Metathesaurus terms. The query expansion step assigns to queries the UMLS concepts related to the user query terms. We describe hereafter the query type labelling and generation strategies, and the query expansion step.

Query Type. The complex query module labels user queries with one of the three following types: *Keyword Query* (K_Q), *Open Question Query* (O_Q) or *Statement Query* (S_Q). The three query types are defined based on common user search needs, and they are used as an attempt to increase the recall of documents that are relevant to a given query. bioMine uses the query types in order to guide the search engine towards prioritizing documents having query terms in specific document fields that are searched differently according to the query type. The query labelling process analyzes query terms for syntactic cues that can indicate the user search intent. The syntactic cues can be the presence of punctuation, or stop-words (the stop-word list used in bioMine query module is composed by a combination of an English stop-word list, and PubMed stop-word list [19]). A K_Q is a user query formulated without stop-words. K_Q are submitted to the search engine as is. K_Q example: *enzyme structure function.* An O_Q is a user query that contains interrogative cues (question words or question mark). O_Q are normalized before submitted to the search engine, by having removed the interrogative cues, and stop-words (if any present). O_Q example: *what is the relationship between the structure of an enzyme and its function?* A S_Q is a user query that contains stop-words, but no interrogative cues. S_Q are normalized by having the stop-words removed before being submitted to the search engine. S_Q example: *the relationship between enzyme structure and function.*

Query Type Generation Strategies. Each query type is associated to a different strategy to generate a bioMine query. The query strategies aim to better address the user search needs. They determine: the document fields considered in a search; if boost weights are assigned to certain document fields; and the relevance of search terms that are found if search terms are considered separately or sequentially (query term search is concerned with the presence of search terms in specific fields, while phrase term search looks for search terms sequentially in specific fields).

Boost weights are used to prioritize documents in which the search terms are found in the specific (boosted) document fields. The three query types processed by bioMine, and their strategies are as described in the following table. For all strategies, field **14** is only included in the search if a query has a numeric term of 2 or 4 characters length; and fields **11** and **12** are only included if a query has a numeric term of more than 4 characters length.

	Fields	Phrase terms	Boost?	Query terms	Boost?
K_Q *strategy*	**1**, **6**	✓	✓	✓	✗
	3, **5**	✓	✗	✓	✓
	10, **13**	✗	✗	✓	✗
	14	✗	✗	✓	✗
	11, **12**	✗	✗	✓	✗
O_Q *strategy*	**1**, **10**	✓	✗	✓	✓
	3, **5**	✓	✓	✓	✓
	6, **13**	✗	✗	✓	✗
	14	✗	✗	✓	✗
	11, **12**	✗	✗	✓	✗
S_Q *strategy*	**1**, **5**	✓	✓	✓	✓
	3, **10**	✓	✗	✓	✓
	6, **13**	✗	✗	✓	✗
	14	✗	✗	✓	✗
	11, **12**	✗	✗	✓	✗

Query Expansion. After the query terms are pre-processed during the query type labelling step, they are expanded with UMLS concepts. To perform this expansion step, we utilize the open-source tool MetaMap [2] (https://metamap.nlm.nih.gov). MetaMap is an UMLS Metathesaurus annotator system. The tool is capable of processing natural language input, extracting or mapping a given text content to UMLS concepts. The query, already processed in the type labelling step, is sent to a MetaMap instance, which annotates UMLS concepts related to the query, if any are found. The MetaMap annotations found are added to the query terms (if the original terms do not overlap with the annotation terms).

4 Experimental Evaluation and Preliminary Results

The evaluation of IR systems is commonly performed with the use of reference judgments [27]. Reference judgments are a mapping of queries and correct response documents. Some previous studies reviewed in Sect. 2 made use of reference judgment collections of less than a thousand documents [26,31]. These works had either an annotated collection, or experts available to annotate one. Creating a reference judgment collection can be costly, and even unfeasible, in tasks handling large datasets, since usually annotations are done manually by specialists. Since the dataset used in this work is considerably large, generating a reference judgement with manual annotations would be an effortful and expensive task.

4.1 Evaluation Without Reference Judgments

Previous work [23,29] dedicated effort to develop methods to evaluate IR systems without reference judgments. The authors suggested comparing results of similar systems, and the documents retrieved for a given query. The evaluation methods should consider, for example, presence or absence of retrieved documents, and

their ranking position in the result list. The tasks related to biomedical document retrieval described in Sect. 2 lacked enough information about their evaluation methods. In [25] the authors described a full search engine similar to bioMine, but did not present which evaluation was adopted. [10,22,31] have not provided detailed information about, for instance, the indexation process or the document relevance computation, preventing the presented work to be reproduced. On top of this, at the time this study was conducted, the systems developed in these works appeared not to be available as open source software. This makes it difficult to carry on a fair comparison between results obtained by these approaches, and results obtained by bioMine.

4.2 Pseudo-judgments Evaluation

Pseudo-judgements are evaluation collections automatic generated by IR systems, with little or none human influence. For pseudo-judgments, the top K results retrieved by a search are considered as the most relevant, and further evaluated. In [6,21] the authors investigated the use of pseudo-relevance judgments to evaluate IR systems, by using a pool of results. The authors in [6] described a method that consists in generating a set of similar queries, retrieving relevant documents for all similar queries, and finally evaluating the ranked results against human relevance judgments. According to the authors, the system and the human's ranking were correlated. In [21], the authors described an effort to generate pseudo-relevance judgments instead of human relevance judgments using a pool of search results retrieved by a variety of IR systems.

4.3 bioMine Evaluation

Considering both [6,21] works, we suggest a comparable evaluation method, that uses pseudo-judgments, and sets of annotated queries. Queries and their corresponding relevant documents were obtained from the mycoCLAP [24] database. Biocurators working on mycoCLAP have searched extensively biomedical literature databases. They have evaluated several thousands of scientific articles to characterize fungal enzymes having specific properties, and finally map an article with a mycoCLAP enzyme entry. Within all articles mapped to mycoCLAP entries, 9 documents belong to the PMC OA. The great majority of the other documents belong to PubMed BD. We utilized the user search queries generated by mycoCLAP biocurators to retrieve: the 9 documents in mycoCLAP that can be found in PMC OA, as well as 10 randomly selected documents in mycoCLAP that can be found in PubMed BD. Our goal is to evaluate the document retrieval performance for article journals containing full-text, as well as abstract only. Our evaluation dataset, as shown in Table 2, is then composed by a set of 19 mappings of queries and correct response documents.

4.4 Evaluation Metric and Preliminary Results

To compute bioMine performance evaluation, we utilize the Mean Reciprocal Rank (MRR) score, that was previously applied in information retrieval

Table 2. List of queries and correct response documents

mycoCLAP ID	PID	$Q_{\#}$	Biocurator query
AMY13A_CRYFL	PMC3068306	Q_1	Alpha-amylase from Cryptococcus flavus activity characterization
BGL3C_ASPFU	PMC3312866	Q_2	Aspergillus fumigatus beta-glucosidase purification and characterization
MAN5A_ASPNG	PMC2780388	Q_3	Characterization of GH5 beta-mannanase enzyme from Aspergillus niger
MLG16B_ASPFU	PMC3092853	Q_4	Characterization of GH16 beta-glucanase from Aspergillus fumigatus
PGX28B_FUSOX	PMC3180650	Q_5	Purification and characterization of an exo- polygalacturonase from Fusarium oxysporum
PMO9D_PHACH	PMC3223205	Q_6	Phanerochaete chrysosporium GH61 purification and characterization
RHA78E_EMENI	PMC3312857	Q_7	Purification and characterization of an alpha- L-rhamnosidase from Aspergillus nidulans
XYN11A_LEUGO	PMC2291056	Q_8	Xylanase characterization from Leucoagaricus gongylophorus
XYN11B_TRIRE	PMC2702311	Q_9	Recombinant expression and characterization of xylanase from Trichoderma reesei
ZAX43C_PENPU	PMID20562284	Q_{10}	Bifunctional alpha-L-arabinofuranosidase/xylobiohydrolase from Penicillium purpurogenum
MSD47S_ASPPH	PMID10215597	Q_{11}	Enzymatic properties alpha-mannosidase Aspergillus saitoi
CBH6A_MAGOR	PMID20709852	Q_{12}	Characterization of Magnaporthe oryzae cellobiohydrolase
ABF51A_ASPAW	PMID9758835	Q_{13}	Substrate specificity of alpha-L- arabinofuranosidase from Aspergillus awamori
CHI18B_CANAL	PMID7708682	Q_{14}	Cloning and characterization Candida albicans chitinase
AGL13B_CANAL	PMID1400249	Q_{15}	Characterization of Candida albicans maltase
GAN53A_HUMIN	PMID12761390	Q_{16}	Beta-1,4-galactanases from Humicola insolens and Myceliophthora thermophila
BGN5A_NEOSP	PMID12427996	Q_{17}	Neotyphodium sp beta-1,6-glucanase expression and characterization
EBG16A_FLAVE	PMID21653698	Q_{18}	Purification of endo-beta-1,3-galactanase from Flammulina velutipes
XYL3A_ASPOR	PMID9872754	Q_{19}	Aspergillus oryzae beta-xylosidase optimum pH and temperature

tasks [28]. The Reciprocal Rank (RR) score is the inverse rank position ($\frac{1}{POS}$) of a correct response document retrieved by a system. It evaluates to 1 in case the correct response document is ranked at first position in a search result list. For an evaluation considering a set of queries and correct response documents, the MRR can be utilized to compute an average among all RR scores. The search queries listed in Table 2 are the search terms as provided by users searching for scientific literature. We submitted these search queries as is to bioMine. The user search queries are internally processed by the complex query module, and finally submitted to bioMine search index. For the sake of comparison, we submitted the same user search queries to PubMed BD and PMC OA search engines, with default configurations. All queries mapped to a PMID were searched in bioMine and PubMed BD, while the queries mapped to a PMCID were searched in PMC OA and bioMine. The first 20 ranked results were taken into account from each ranked result list provided by bioMine, and PubMed BD or PMC OA. A RR score was computed for each ranked result list, considering the correct response document provided by the biocurators. Finally, we computed the MRR for all

Table 3. Queries submitted to bioMine, PubMed BD, PMC OA, and correct response document ranking

$Q_{\#}$	PID rank	bioMine rank	bioMine RR score	$Q_{\#}$	PID rank	bioMine rank	bioMine RR score
Q_1	3	2	0.500	Q_{10}	2	1	1.000
Q_2	1	20	0.050	Q_{11}	N/A	7	0.143
Q_3	1	2	0.500	Q_{12}	1	1	1.000
Q_4	2	8	0.125	Q_{13}	2	1	1.000
Q_5	2	13	0.077	Q_{14}	1	1	1.000
Q_6	9	1	1.000	Q_{15}	2	1	1.000
Q_7	2	5	0.200	Q_{16}	1	N/A	0.000
Q_8	1	17	0.059	Q_{17}	N/A	1	1.000
Q_9	1	10	0.100	Q_{18}	1	N/A	0.000
				Q_{19}	1	1	1.000
Total # of queries = 19				**MRR = 0.513**			

queries submitted to bioMine. According to our results, queries mapped to a PMCID always retrieved the correct response document, either using bioMine or PMC OA search. The correct response document was ranked higher in bioMine search compared to the PMC OA search for Q_1 and Q_6, while Q_3 and Q_7 presented the correct document in similar ranking between bioMine and PMC OA search results, with only few positions of difference in the result list. When looking into results of queries mapped to a PMID, we observe that in some cases the first 20 rank did not have the expected document. For PubMed, this issue occurred in search results for queries Q_{11} and Q_{17}. While PubMed did not retrieve the correct response document among the top 20 results, bioMine was able to retrieve it in a fairly high ranked position. For bioMine, this issue occurred in search results of queries Q_{16} and Q_{18}, however for all other documents, we observe that bioMine retrieved the correct response document in a higher ranked position than PubMed. The MRR score for all 19 queries submitted to bioMine is 0.513, which demonstrates that the system is capable of retrieving the correct response document ranked at first position approximately half of the time (Table 3).

5 Conclusion and Ongoing Work

Literature search on open access scientific databases is a task that supports many activities in the life sciences and biomedical domain, but it can still be challenging. In this paper, we described the ongoing development of bioMine, a bioliterature search engine that aims to facilitate the retrieval of scientific literature. bioMine is an attempt to address two main issues. First, while the indexed content from PubMed BD, one of the most popular scientific databases, records only the abstract content of documents, bioMine offers the possibility of searching for literature using also the full-text of scientific articles, obtained

from the PMC OA. Second, while open access databases require users to perform searches using a query language, bioMine provides the possibility of using natural language queries thanks to its complex query module. Despite the large size of the indexed corpus, bioMine is still in its infancy, and much work is needed to enhance its performance. For instance, further analysis is currently being carried on to improve the retrieval of full text documents.

Reproducibility. To ensure full reproducibility and comparisons between systems, bioMine is publicly released as an open source software in the following repository: https://github.com/BigMiners/bioMine.

References

1. Almeida, H., Meurs, M.-J., Kosseim, L., Butler, G., Tsang, A.: Machine learning for biomedical literature triage. PLOS ONE **9**(12), 12 (2014)
2. Aronson, A.R., Lang, F.-M.: An overview of MetaMap: historical perspective and recent advances. J. Am. Med. Inform. Assoc. **17**(3), 229–236 (2010)
3. Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. **32**(suppl 1), D267–D270 (2004)
4. Divoli, A., Wooldridge, M.A., Hearst, M.A.: Full text and figure display improves bioscience literature search. PLOS ONE **5**(4), e9619 (2010)
5. Dogan, R.I., Murray, G.C., Névéol, A., Lu, Z.: Behaviour, Understanding PubMed User Search Behaviour through Log Analysis. Database, 2009:bap018 (2009)
6. Efron, M., Winget, M.: Query polyrepresentation for ranking retrieval systems without relevance judgments. J. Am. Soc. Inf. Sci. Technol. **61**(6), 1081–1091 (2010)
7. Fontelo, P., Liu, F., Ackerman, M.: askMEDLINE: a free-text, natural language query tool for MEDLINE/PubMed. BMC Med. Inform. Decis. Mak. **5**(1), 5 (2005)
8. Gay, C.W., Kayaalp, M., Aronson, A.R.: Semi-automatic Indexing of Full Text Biomedical Articles. In: AMIA Annual Symposium Proceedings, vol. 2005, p. 271. American Medical Informatics Association (2005)
9. Gobeill, J., Gaudinat, A., Pasche, E., Vishnyakova, D., Gaudet, P., Bairoch, A., Ruch, P.: Deep Question Answering for Protein Annotation. Database, 2015:bav081 (2015)
10. Griffon, N., Chebil, W., Rollin, L., Kerdelhue, G., Thirion, B., Gehanno, J.-F., Darmoni, S.J.: Performance evaluation of Unified Medical Language System synonyms expansion to query PubMed. BMC Med. Inform. Decis. Mak. **12**(1), 12 (2012)
11. Hearst, M.A., Divoli, A., Guturu, H., Ksikes, A., Nakov, P., Wooldridge, M.A., Ye, J.: BioText search engine: beyond abstract search. Bioinformatics **23**(16), 2196–2197 (2007)
12. Hirschman, L., Burns, G.A.P.C., Krallinger, M., Arighi, C., Cohen, K.B., Valencia, A., Wu, C.H., Chatr-Aryamontri, A., Dowell, K.G., Huala, E., et al.: Text Mining for the Biocuration Workow. Database, 2012:bas020 (2012)
13. Howe, D., Costanzo, M., Fey, P., Gojobori, T., Hannick, L., Hide, W., Hill, D.P., Kania, R., Schaeffer, M., St Pierre, S., et al.: Big data: the future of Biocuration. Nature **455**(7209), 47–50 (2008)
14. Hunter, L., Cohen, K.B.: Biomedical language processing perspective: what is beyond PubMed? Mol. Cell **21**(5), 589 (2006)

15. Lu, Z.: PubMed and Beyond: A Survey of Web Tools for Searching Biomedical Literature. Database, 2011:baq036 (2011)
16. Lu, Z., Wilbur, W.J., McEntyre, J.R., Iskhakov, A., Szilagyi, L.: Finding query suggestions for PubMed. In: AMIA Annual Symposium Proceedings, vol. 2009, p. 396. American Medical Informatics Association (2009)
17. Morris, B.D., White, E.P.: The EcoData retriever: improving access to existing ecological data. PLOS ONE **8**(6), e65848 (2013)
18. Mudunuri, U.S., Khouja, M., Repetski, S., Venkataraman, G., Che, A., Luke, B.T., Girard, F.P., Stephens, R.M.: Knowledge and theme discovery across very large biological data sets using distributed queries: a prototype combining unstructured and structured data. PLOS ONE **8**(12), e80503 (2013)
19. National Center for Biotechnology Information. PubMed [Table, Stopwords] (2005)
20. Nourbakhsh, E., Nugent, R., Wang, H., Cevik, C., Nugent, K.: Medical literature searches: a comparison of PubMed and Google Scholar. Health Inf. Libr. J. **29**(3), 214–222 (2012)
21. Ravana, S.D., Rajagopal, P., Balakrishnan, V.: Ranking retrieval systems using pseudo relevance judgments. Aslib J. Inf. Manage. **67**(6), 700–714 (2015)
22. Shariff, S.Z., Bejaimal, S.A.D., Sontrop, J.M., Iansavichus, A.V., Haynes, R.B., Weir, M.A., Garg, A.X.: Retrieving clinical evidence: a comparison of PubMed and google scholar for quick clinical searches. J. Med. Internet Res. **15**(8), e164 (2013)
23. Spoerri, A.: Using the structure of overlap between search results to rank retrieval systems without relevance judgments. Inf. Process. Manage. **43**(4), 1059–1070 (2007)
24. Strasser, K., McDonnell, E., Nyaga, C., Wu, M., Wu, S., Almeida, H., Meurs, M.-J., Kosseim, L., Powlowski, J., Butler, G., et al.: mycoCLAP, the Database for Characterized Lignocellulose-active Proteins of Fungal Origin: Resource and Text Mining Curation Support. Database, 2015:bav008 (2015)
25. Thomas, P., Starlinger, J., Vowinkel, A., Arzt, S., Leser, U.: GeneView: a comprehensive semantic search engine for PubMed. Nucleic Acids Res. **40**(W1), W585–W591 (2012)
26. Van Auken, K., Schaeffer, M.L., McQuilton, P., Laulederkind, S.J.F., Li, D., Wang, S.-J., Hayman, G.T., Tweedie, S., Arighi, C.N., Done, J., Mller, H.-M., Sternberg, P.W., Mao, Y., Wei, C.-H., Lu, Z.: BC4GO: A Full-text Corpus for the BioCreative IV GO Task. Database, 2014:bau074 (2014)
27. Voorhees, E.M.: Variations in relevance judgments and the measurement of retrieval effectiveness. Inf. Process. Manage. **36**(5), 697–716 (2000)
28. Voorhees, E.M., et al.: The TREC-8 question answering track report. In: TREC, vol. 99, pp. 77–82 (1999)
29. Wu, S., Crestani, F.: Methods for ranking information retrieval systems without relevance judgments. In: Proceedings of the 2003 ACM Symposium on Applied Computing, pp. 811–816. ACM (2003)
30. Yoo, I., Mosa, A.S.M.: Analysis of PubMed user sessions using a full-day PubMed query log: a comparison of experienced and nonexperienced PubMed users. JMIR Med. Inform. **3**(3), e25 (2015)
31. Zeng, Q.T., Redd, D., Rindflesch, T., Nebeker, J.: Synonym, topic model and predicate-based query expansion for retrieving clinical documents. In: AMIA Annual Symposium Proceedings, vol. 2012, p. 1050. American Medical Informatics Association (2012)