Survey on datasets and methods for bio-literature ad hoc retrieval

Jinfeng Xiao, Peifeng Jing, Wenzhuang Chi

Abstract—Ad hoc retrieval is a frequent practice not only in people's daily life but also in research projects. Although in the general search domain many methods have been proposed to achieve good performance, the need for bio-literature ad hoc retrieval is yet to be well satisfied. In this report, we give an overview of the currently available resources for bio-literature ad hoc retrieval, including bio-literature databases to search through, extra data used to boost performance, labeled datasets for evaluation, text processing and ranking algorithms proved to be effective, and some ready-to-use tools. We also reviews some state-of-art information retrieval methods in the general domain. We show that the biomedical literature retrieval domain is rich in data while rather primitive in methodology, and thus has rich potential of future improvement.

1 Introduction

Surveying through related work is a fundamental component in most research projects. The first step of literature review on any topic is to formulate the information need into queries and use search engines to find relevant publications. A search engine usually returns a ranked list of documents based on relevance to each input query. This task is called *ad hoc retrieval*.

Given a query, the ultimate goal of an ad hoc retrieval engine is to make the returned list of documents, especially the top 10 to 20 documents, satisfy the user's actual information need behind the query as much as possible. There are three **challenges** for achieving this:

1) A comprehensive **database** which contains the documents that users are looking for, and supplementary datasets to assist the search if needed;
2) A mechanism for correct **interpretation** of queries and documents;
3) A **ranking function** which correctly reflects users' actual need.

The state-of-art progress in solving those challenges varys a lot across domains. In the general information retrieval (IR) domains like searching for web pages or news articles, researchers and developers have gone quite far in solving each of the four challenges after years of ongoing effort. The success and continuing growth of Google is an example of how solving those challenges can make a difference. Unfortunately, the ad hoc retrieval systems for the academia community are less developed and far from being satisfactory. For example, although Google's PageRank algorithm [60] was proposed 18 year ago and has been proved successful not only for web or citation search but also for system analysis in natural science [28], in **biomedical research** domain there is still no ready-to-use literature retrieval engines[1] that utilizes the citation networks in any way other than the citation count of each paper. Things in other research fields with less funding can be even worse. For example, while PubMed[2], the leading online public database for biomedical literature records [61], receive millions of dollars of funding from the U.S. government, IDEAS[3], the largest bibliographic database dedicated to Economics, can only rely on volunteers. The rest of this report will focus on the biomedical domain.

Transferring the more advanced ad hoc retrieval methods from the general domain into scientific domains is not a trivial task. Techniques known to be effective in the general domain can benefit or harm the performance of search engines in the biomedical domain, depending on implementation details [35]. One reason is that biomedical literatures include some domain-specific knowledge. Implementating general domain methods without handling such domain-specific knowledge with care tends to harm the performance on the biomedical corpus [34]. Previous works aiming at improving biomedical ad hoc retrieval have been attacking the 1st, 2nd and 4th challenges, especially on how to use the domain knowledge effectively, while little attention has been paid to the improvement in ranking functions.

The rest of this report is arranged as follows. Sections 2 to 4 review current progress in solving the three challenges in biomedical ad hoc retrieval, one section for each challenge. In each section we also discuss some related works in the general IR domain, and our thoughts about possible trans-domain applications. Finally Section 5 summarizes this report and briefly discusses the novelty of our ongoing project SetSearch.

1. This report does not consider systems whose ranking function is not revealed to the public. The only exception is Google Scholar, whose ranking function has been reversely engineered by third-party researchers [5].
2. https://www.ncbi.nlm.nih.gov/pubmed/
3. https://ideas.repec.org/

2 Available Data

In this section we review the datasets involved in the development and evaluation of bio-literature ad hoc retrieval, and compare the domain-specific data availability to that of the general IR domain. Such data can be categorized into three types:

1) Document corpus: The database that contains the documents users are searching for.
2) Auxiliary datasets: Datasets other than the document corpus that are used by search engines to assist search and improve performance.
3) Evaluation datasets: Golden standard datasets used to evaluate the performance of retrieval systems.

2.1 Document Corpus

The most popular document corpus of biomedical literature is MEDLINE[4], the U.S. National Library of Medicine premier bibliographic database that contains more than 23 million references to journal articles in life sciences with a concentration on biomedicine. Each document in MEDLINE is characterized by up to more than 60 fields[5]. The most important fields for ad hoc retrieval are:

- PubMed Unique Identifier (PMID). Each paper has its unique PMID.
- Title.
- Abstract.
- MeSH terms. More details to come in Section 2.2.
- Various "Date" fields, describing the timeline of the record.

Usually search systems use the title, abstract and/or MeSH terms to rank documents. The date fields can be either coded into the ranking function, or simply used to re-sort search results upon users' request. Some research papers [34], [35], [37] also use the date fields to rigorously define a subset of MEDLINE records.

PubMed is the most native engine for searching in MEDLINE. Although MEDLINE does not contain the full text of all its papers due to copyright issues, PubMed does provide links to the full text on publishers' websites. The general workflow of biomedical literature search is: 1) Search MEDLINE with PubMed; 2) Read the titles and abstracts of the first dozen of papers returned by PubMed; 3) For those truely relevant papers, follow the link to the full text and retrieve the full text from the publisher.

PubMed also provides a collection of 4.3 million full text articles, most of which have a corresponding entry in PubMed, archived at PubMed Central[6] (PMC). A subset of PMC articles, called the Open Access Subset, are made available to public under a Creative Commons or similar license that generally allows more liberal redistribution and reuse than a traditional copyrighted work. This full-text dataset can be used for certain research tasks [29], but MEDLINE is still much more popular in real biomedical literature search practices simply because it has much higher literature coverage despite that it does not contain full text.

The bio-literature coverage of MEDLINE is high but not perfect though. The collection is formed by journal articles only, and whether to include a journal or not is decided by the Director of the National Library of Medicine, based on considerations of both scientific policy and scientific quality. It does not contain textbooks, conference papers, tutorials, online-only manuscripts, etc.

There have been efforts trying to expand the biomedical document corpus beyond MEDLINE. Google Scholar, another popular search engine for academia, searches against a much larger and diverse document corpus, including and not limited to journal articles, preprint archives, conference proceedings, and institutional repositories [27]. Web of Science[7], a comprehensive citation search engine supporting 256 disciplines, has a corpus of journals, books and conference proceedings. A recent work even includes less "academic" sources like newspaper articles and social media [23]. Expansion of the diversity of the corpus has two-sided effects though. A more diverse corpus is less likely to miss some documents that are of interest to the users, but may also lead to greater noise in its top results.

Despite the fact that MEDLINE contains only journal articles, it is still a popular dataset for biomedical IR research. By 2010 there were at least 28 IR tools built upon MEDLINE [52]. One reason of the popularity of MEDLINE can be that the dataset itself is well-documented and open to public. Even the daily updates of MEDLINE are released in a very timely manner. In contrast, none of the systems listed in the previous paragraph releases its corpus.

One distinct characteristic of MEDLINE, compared to corpora in the general IR domain, is that MEDLINE is in general more homogeneous and cleaner. The general IR community often works with less organized corpus, such as web pages, emails, news, and social media. Unlike MEDLINE which is well maintained and downloadable, many general-domain corpora are crawled from the web and thus can be messy. While searching in MEDLINE is basically processing structured plain text, searching in general corpus requires proper handling of various data types and structures. For example, ClueWeb [62] is a webpage database containing more than 733 million websites crawled from the Internet. Each webpage record in ClueWeb can contain heterogenous fields like its URL, text content, image content (not including other multimedia file types though), HTTP response headers and format files [3]. Presumably working on biomedical literature corpora does not brings as much headache in data acquisition, organization and formatting, thanks to MEDLINE.

4. https://www.nlm.nih.gov/pubs/factsheets/medline.html
5. https://www.nlm.nih.gov/bsd/mms/medlineelements.html
6. https://www.ncbi.nlm.nih.gov/pmc/
7. https://webofknowledge.com/

2.2 Auxiliary Datasets

As introduced in Section 1, the biomedical corpus has a lot of domain-specific knowledge, which should better be handled properly [34]. Auxiliary datasets are those data that can help the search engine utilize external knowledge for better performance.

2.2.1 MeSH Tree

MeSH is the National Library of Medicine's controlled vocabulary thesaurus. It consists of sets of terms naming descriptors in a 13-level hierarchical structure that permits searching at various levels of specificity. The hierarchy indicates "broader than/narrower than" relationships. Terms at shallower levels represent more general concepts. The deeper you go along the hierarchy, the more specific terms you will find. There are several ways to utilize MeSH terms:

- Use the MeSH Browser[8] to search for MeSH terms and their positions in the MeSH tree;
- Download the MeSH tree in various data formats and build it into your algorithm;
- Make use of the MeSH Term field of MEDLINE articles.

PubMed takes MeSH terms into consideration when searching for and ranking articles. There are also many other biomedical IR systems using MeSH, some of which will be discussed in the following sections.

2.2.2 Synonym Databases

Synonyms are pervasive in biomedical literature. Many genes, species and chemicals have more than one names, and those synonyms may or may not look similar (e.g. PTEN and MMAC1 refer the same gene [17]). Typically, biologists do not manually add synonyms to their queries, but they expect a good search engine to return documents with all synonyms considered, because those domain-specific synonyms are basically equivalent in any sense. Being able to recognize the synonyms is critical to search engines for a reasonable level of recall.

While in the general IR domain detection of synonyms is usually done automatically by named entity recognition (NER) techniques, in biomedical domain there are many trustable sources of synonym sets thanks to mass manual curation effort. Examples include Entrez Gene [54] (previously LocusLink [64]), UMLS [8], NCI Thesaurus [68], Disease Ontology [67], MEDIC [19], etc. For gene synonyms there are even species-specific databases like OMIM [30] for human, FlyBase [72] for flies, MGD [13] for mouse, and SGD [15] for yeast.

Synonym recognition has also been computationally attempted as biomedical NER (bioNER) challenges, an active area of biomedical text mining for more than ten years [17]. To push forward bioNER and bio-synonym recognition, a number of crowd-sourcing challenges have been hosted [39]. In particular, the BioCreative [53], [59], [38] workshop has repeated organized gene normalization tasks which are basically gene synonym recognition. The golden standard datasets used for evaluation of those tasks are chosen from the human curated databases listed in the previous paragraph. In principle bioNER has generated some synonym datasets based on those human curated databases, while in practice ad hoc retrieval systems tend to rely on human curated synonym sets and seldom use inferred ones.

8. https://meshb.nlm.nih.gov/search

2.2.3 Entity Datasets

Instead of synonym datasets, the greatest contribution of bioNER to ad hoc retrieval is probably a couple of entity databases. A biomedical *entity* can be roughly understood as a string with a specific meaning. An entity is described by two fields: the type of the entity, and a unique ID which distinguishes its meaning from the other entities. Entity recognition can help ad hoc retrieval in four ways:

- Synonym recognition. This can be done with either a synonym database as discussed in the last section, or an entity database. For example, in an entity database where "PTEN" and "MMAC1" are both mapped to the same type "gene" and the same entity ID, these two completely different strings are regarded as exactly the same.
- Term disambiguation. For example, the term "AR" can mean either a gene, a chemical element, or the initials of a person. A bag-of-words search engine has no way to distinguish those meanings from the context. However, such disambiguation task will become trivial if the engine has a pre-processed text corpus where each "AR" has entity information associated with it.
- Entity-aware smart search. A search engine can become more intelligent by coding entity information into its ranking function. More details to come in Sections 3.3 and 4.
- Entity-aware user interface. More details to come in Section 5.

A good example of integrating bioNER and search is PubTator [76]. It utilizes entities of five types: genes, diseases, species, chemicals and mutations. Each the five entity datasets is itself an individual project [77], [45], [46], [75], [74] and downloadable from the PubTator server. PubTator did not explore much about how to use entities to enhance search other than highlighting the entities in the results, because its main functionality was not to search but to assist biocuration. Nevertheless, its entity datasets can be potentially used as auxiliary datasets for other search-oriented systems. There are also other biomedical entity sets (e.g. TREC 2007 Genomics Track [36] defined a dataset with 14 entity types: antibodies, biological substances, cell or tissue types, diseases, drugs, genes, molecular functions, mutations, pathways, proteins, strains, signs or symptoms, toxicities, and tumore types), but to the best of our knowledge, PubTator is the only entity dataset that has a reasonably high

coverage AND provides a large entity-labeled MEDLINE corpus so that it can be easily implemented into foreign search systems.

The construction of PubTator was guided by biomedical databases as listed in Section 2.2. Those databases give well-defined terminology sets, which provide unique advantages to bioNER. NER in general domain has to find more clever ways to extract knowledge. Recent work has generated some good database for the general IR community as well. For example, DBpedia is a database that contains more than 1.95 million entities, which consists by a comprehensive knowledge base, including persons, places, music albums etc. The knowledge in DBpedia is extracted from Wikipedias structured data and it rebuild the knowledge in a better structure that is suitable for interacting with IR systems. In order to ensure the query performance, DBpedias knowledge is built with RDF tuples and stored in MySQL and Virtuoso database [2]. And another popular public accessible knowledge base is Freebase [9]. Freebase contains more than 125 million tuples, 4000 types and 7000 properties. Its data also is extracted from Wikipedia and other online knowledge bases. To be suitable for being a high-performance knowledge base, Freebase has following great features: Scalable, HTTP, JSON based API, Lightweight, Large and Diverse Data Set, and Complete Normalization. YAGO [70] is another lightweight but high-performance knowledge base. It is scalable and can extract data from other knowledge bases. Currently in contains 900,000 entities and more than 5 million facts. Among the knowledge base there are several types of relations, such as TYPE, SUBCLASSOF, and MEANS. These relations can leverage an adaptive knowledge base which is very suitable for user-defined entity queries. In addition, YAGOs knowledge is mainly from Wikipedia and WordNet, plus YAGOs defined relations among the knowledge set, all of these features leverage a very high accuracy of YAGO.

2.3 Evaluation Datasets

A document corpus with relevance labels with respect to a series of queries is required for evaluation of the performance of search engines. Different search tasks require different evaluation datasets. Based on the importance and popularity of MEDLINE as discussed in Section 2, this report only covers the evaluation of ad hoc retrieval on MEDLINE.

The only reasonably recent, large and reliable MEDLINE evaluation dataset is from TREC Genomics Tracks 2004 and 2005 [35], [37], crowd-sourcing workshops for enhancing biomedical ad hoc retrieval and text categorization. The corpus is a 10-year subset of MEDLINE, consisting of 4,591,008 records published between 1994 and 2003. The dataset include 100 queries carefully designed to reflect biologists' information need at that time. For each query, a possibly relevant document collection was constructed by pooling the top results from workshop participants. Human judges with biological background levels from undergraduate to PhD then labeled each document in the collection as relevant or not. Documents not in the collection are by default irrelevant. Those labels are the golden standard for calculating evaluation metrics like the mean average precision (MAP), precision@10, etc. The corpus, query set, and relevance labels are all freely downloadable.

There exist other MEDLINE-based evaluation datasets. For example, the evaluation dataset from TREC Genomics Track 2003 [34] contains a 1-year subset of MEDLINE and 100 2-keyword queries in the form of "gene + disease". Unfortunately that dataset is of rather low quality [33] and thus the document set has been removed from the track holder's website. The OHSUMED dataset [32] contains a subset of 270 journals in MEDLINE from 1987 to 1991, and relevance judgement for 106 queries. This dataset was still used in 2010 for evaluation of system evaluation [87].

TREC, or Text Retrieval Conference in full, also runs other tracks, many of which are in the general IR domain. One of the most widely used TREC tracks is its Web Track [22], [49], [80]. This track is for IR challenges in the web search domain. Its document collection was built on ClueWeb09 [16], which is 25TB in size and contains roughly 1 billion web pages crawled from the Internet between 1993 and 2006. TREC itself is still active, and this year there are eight tracks running. Seven of them are in the general IR domain, and the rest is about precision medicine. Unfortunately the Genomics Track closed in 2007, and so did the release of new datasets for evaluation of biomedical ad hoc retrieval systems.

The advance of the general IR field compared to the biomedical IR field can be reflected from the crowd-source tasks and available evaluation datasets. For example, more than 10 years ago the general IR community already worked on crowd-source tasks on cross-language queries [41], [11], while in the biomedical IR field there has been no any sign of such attempt till now.

3 Query/Document Interpretation

This section reviews how search engines process the document corpus and the query before calculating relevance scores. Documents and queries are by themselves nothing more than strings, and every search engine has to somehow interpret those strings in ways that reflect users' need as much as possible.

The simplest interpretation is the bag-of-words model, where each document or query is represented by a integer count vector (whose length is the size of the vocabulary set) indicating the number of occurrence of each unique word in the document or query. Such vector has some variants, for example a binary vector indicating whether the words exist or not, or a numeric vector after smoothing over the integer count vector. Such a model is easy to understand and implement.

The greatest disadvantage of the bag-of-words model is that it loses all information about the relative position

of the words in the context. Although some local co-occurrence information like phrases can be captured by adding n-grams to the bag-of-words model, it is almost impossible to capture mid-to-long range interaction between words. There are many works in the general IR domain for capturing such mid-to-long range interactions, but the biomedical IR is much less active. Microsoft's Literome [63] is an interesting work in discovering pairwise interactions between genetic entities from text, but the queries must contain exactly two gene identifiers and thus it cannot meet many types of ad hoc retrieval needs. In general biomedical ad hoc retrieval engines are now still using bag-of-words models.

There are ways to improve the performance of bag-of-words models though, especially by utilizing biomedical domain-specific knowledge. Query expansion, tokenization and normalization are some of the query/document interpretation techniques that have been proved effective. In the following two subsections, we first introduce methods developed by the general IR community, and then follow by biomedical domain-specific techniques. General IR methods can usually be directly applied to biomedical corpora, while the domain-specific techniques are also important to the performance.

3.1 Query Expansion

3.1.1 General IR Methods

Vocabulary problems [26], [14] has been long-standing challenges to the IR community. Current information retrieval systems often take in short and inherently ambiguous input queries, which can lead to discrepancies between the users' information need and the highly ranked documents. The precision of the retrieval system will then be harmed. On the other hand, the recall can be decreased due to miss of relevant documents, which may contain synonyms instead of the exact query terms.

Several approaches have been proposed to overcome such vocabulary problems. Examples include spelling error correction [71], [21], stemming [78], query segmentation [6], [7] and query expansion [14]. Query expansion is considered as the most widely studied and promising methodology to improve the effectiveness of document ranking. The fundamental philosophy of query expansion is to conduct search with enriched queries which contains additional relevant terms and thus has less ambiguity [14], [47].

Many retrieval tasks may benefit from the query expansion techniques, such as question answering, multimedia information retrieval, information filtering and cross-language information retrieval [14]. Nevertheless, query expansion may not be a good idea for some web search cases. It has been observed by Broder [12] that web queries can be categorized as informational, navigational and transactional. Query expansion is more beneficial to informational queries because users could use accurate words and phrases to describe what they are looking for. On the other hand, it is possible that queries drift after expansion if users are looking for particular URLs (navigational queries) or perform some web-mediated activity (transactional queries) [14].

The initial representative method of query expansion is relevance feedback, including excite relevance feedback, pseudo relevance feedback and indirect relevance feedback. The basic idea of relevance feedback is to involve users in the retrieval process to improve performance. Users mark the returned documents as relevant or irrelevant with original query. The system will compute a better representation of the information need based on the user feedback and search documents utilizing this better expanded query. [14], [47], [81], [85], [10], [18] The Rocchio algorithm is the classic algorithm for implementing relevance feedback, which utilizes users feedback and improves the recall of the retrieval system [55]. Relevance feedback relies on explicit feedback from users, but only a small portion of users are willing to provide their feedback [42], [69] Pseudo-relevance feedback later became more popular and widely used because it automates the relevance feedback process [73], [84]. The automation is based on the assumption that the highly ranked documents are likely to be relevant and can be used to improve search. Another feedback approach assumes that more clicks on a link indicates more relevant to the query. Using the clickstream data is an indirect way to collect implicit feedback in large quantities on the web search, and such feedback is assumably more useful feedback than pseudo-relevance feedback [55].

The relevance feedback models heavily use the term frequency information in the top retrieved documents. It may introduce noise into the expanded query and causes query drift which hurts the effectiveness of the retrieval. To overcome this, some recent works [81], [85], [10], [18] utilize external high-quality datasets, such as Wikipedia and Freebase, to improve query expansion and retrieval. Those external datasets provide more features for query expansion which enhance the relevance of selected additional terms not only with unigrams and phrases, but with semantics (including entities, topics, etc.) as well.

3.1.2 Biomedical Domain-Specific Methods

The development of query expansion techniques in the biomedical IR domain is to some extent in a reverse order. Unlike in the general domain where inference came before the utilization of external datasets, in the biomedical IR domain the most intuitive idea and also the first thing to try for query expansion is to use the existing biomedical term databases as listed in Section 2.2. Such method is referred to as "domain-specific query expansion" in the rest of this report.

TREC Genomics Tracks 2003-05 [34], [35], [37] established the fact that domain-specific query expansion is critical to the performance of biomedical ad hoc retrieval engines. The best performers of all three years expand the queries with some of the biomedical vocabulary sets

described in Section 2. The query expansion strategies of the best performers are:

- The best run in 2005 [40] used LocusLink (now Entrez Gene) and AcroMed (now expired) to add gene synonyms into the query and then manually removed some inaccurate names. It also included lexical variants of the query terms into the expanded query. More about lexical variants will be discussed in the next section.
- The best run in 2004 [25] performed query expansion by sequential application of reference database feedback and pseudo-relevance feedback. Each LocusLink and MeSH record was indexed as one document. In the reference database feedback step, the best matched database record of each database were retrieved against the original query, and terms were extracted from the two best records. Not only synonyms but also words from summary sentences were added to the query. Then in the pseudo-relevance feedback step, the expanded query from the last step was submitted against the target corpus, and Rocchio feedback [66] was applied to expand the query preceding the final search.
- The best run in 2003 [44] used the MeSH tree and lexical variants to define synonyms, and rewarded exact matches higher than synonyms. It also expanded the query with additional general key words such as genetics, gene expression, sequence, etc.

There are many other pieces of work exploring query expansion for biomedical ad hoc retrieval. For example, Lu, Fang and Zhai (2009) [51] proposed two different strategies to extend a standard language modeling approach for gene synonym query expansion, and claimed that most previous strategies could be more or less covered by their general strategies. Entity-aware search systems like DeepLife [23] can handle synonyms naturally with its entity database, as discussed in Section 2.2.3.

Despite those explorations by the research community, the query expansion mechanisms of Google Scholar and PubMed, arguably the two most popular literature search engines among biologists, is far from satisfactory. Google Scholar simply does not automatically expand queries. PubMed expand queries into synonyms without proper weighting, and thus query terms with more synonyms tend to dominate the top returned results.

3.2 Document Pre-Processing

3.2.1 General IR Methods

All search engines have to pre-process its corpus from text strings into something (usually a vector) to feed into their ranking functions. Techniques in the general IR domain include:

- Stemming: To reduce derived words to their word stem, base or root form. It can unify difference tenses of the same verb, single/plural forms of the same noun, etc.
- Stop word removal: To remove common but meaningless words, e.g. a, the, etc. This step will reduce the size of the post-indexing corpus, but is not absolutely necessary since stop words will be penalized with the inverse document frequency (idf) term encoded in most ranking functions.
- Tokenization: To break down the text into tokens which are small units of meaningful text. The most trivial method for English is to break at each white space. However, to break "White House" into "White" and "House" is nonsense. Correct tokenization requires some intelligence. NER is an obvious way to improve tokenization: Once a system recognized "White House" as an entity, it will not try to break them apart.
- Normalization: To create equivalent classes of tokens despite superficial differences in their character sequences. This is basically to group synonyms together.

3.2.2 Biomedical Domain-Specific Methods

In the biomedical domain, while stemming and stop word removal can be done similar to the general domain, tokenization and normalization requires proper handling of domain-spedific knowledge. Normalization can be done with synonym or entity databases as discussed in Section 2. Tokenization is more tricky. For example, "MIP-1-alpha", "MIP-1alpha", "(MIP)-1alpha" and "MIP-1 alpha" refer to the same gene [43], and it is the tokenization strategy that determines whether those lexical variances can be correctly matched. In the TREC Genomics 2005 ad hoc retrieval task, the best run [40] was likely to benefit from its clever tokenization. The best run used Okapi BM25 ranking function [4], carefully designed tokenization, and domain-specific query expansion. The first runner up [1] used a more complicated ranking strategy inspired by a semi-supervised learning algorithm Alternating Structure Optimization, a trivial tokenization strategy, and domain-specific query expansion. It is thus possible that the more carefully designed tokenization brought more benefit than the more advanced ranking strategy. Jiang and Zhai (2007) [43] also confirmed in their systematic study that tokenization could significantly affect the biomedical ad hoc retrieval accuracy.

3.3 Entities in Search

3.3.1 General IR Methods

Search algorithms have moved beyond the term-based information retrieval, and the entity-based document and query representation is recently developed. Entity-based retrieval models can help improve query expansion technologies, collaborate with term-based model in the latent space, and compute relevance in the entity space to improve ranking accuracy.

There is work on query expansion using knowledge bases for text-centric information retrieval [20]. External knowledge graphs, such as Wikipedia and Freebase, could

help introduce a better corpus for pseudo relevance feedback [81], [85] and generate query expansion more effectively. Wikipedia and Freebase can be considered as large structured document collections. Entities in those databases are specified explicitly, and all the corresponding information and connections related to the specific entity can be used as sources for pseudo-relevance feedback. Moreover, there are entity-query feature expansion (EQFE) that extract richer learning features by cooperating text and entity of the document and query [65]. It links entity and terms using entity-linking tools and the ranking function is improved to rank based on entities instead of terms.

Entities can also collaborate with the latent space learning models and improve the relevance effectiveness. Latent Entity Space (LES) models the relevance between queries and document through high-dimensional latent entities in contrast to the traditional term space [50]. Each dimension of the latent entity space corresponds to one entity, and both queries and documents are mapped to the latent space accordingly. The relevance between query and documents is then estimated by the vector space model in latent entity space. The main advantage of LES is its capability to capture semantics of queries and documents. EsdRank [80] uses external semi-structured data, including vocabularies, terms and entities, as objects. It considers objects as in a latent space and projects query and documents to the latent space. The objects are the source to connect queries and documents, which improve ranking functions.

Building entity-based text representation is a more recent trend. There are works on linking entities to query and documents to improve term-based ranking accuracy with entity-based retrieval models [31], [24], [56], [57], [58]. The bag-of-entities model represents queries and documents by their entity annotations and then rank documents with the given query in the entity space [82]. Explicit Semantic Ranking (ESR) further improves the ranking in the entity space and rank documents based on their semantic connections from the knowledge graph [83].

3.3.2 Biomedical Domain-Specific Methods

NER work in the biomedical IR field, for example PubTator [76], is usually guided and evaluated by biomedical terminology databases as listed in Section 2.2. Since those databases serve as well-defined ground truth, biomedical search engines tend to use those databases directly instead of relying on inferred entities. The domain-specific synonym finding and tokenization methods can be used to derive more comprehensive entity datasets from the biomedical databases for better retrieval performance.

4 Ranking Functions

This section reviews popular ranking functions for biomedical retrieval tasks.

Okapi BM25 [4] has long been popular among biomedical literature search engines. The best ad hoc retrieval runs in TREC Genomics Tracks 2004-05 both used Okapi weighting. An adapted version of Okapi BM25 is also the scoring scheme for PubMed, according to PubMed's online documentation[9]. PubMed's adaptation to Okapi BM25 adds a field weight term to favor hits in titles over hits in abstracts, and a date weight term to favor recent publications. This adapted BM25 was the only ranking function for PubMed by January 2017.

Learning-to-rank [48] is another popular class scoring methods, which combines multiple features (and Okapi BM25 can be one of the features) and then uses machine learning on labeled corpus to find the optimal weights for the features. Some time in 2017 between January and April, PubMed added a new learning-to-rank component and started applying that to re-rank the top articles retrieved by its adapted Okapi BM25. PubMed's learning-to-rank algorithm combines 179 features, most of which are computed from the query-document term pairs. The weights of the features were trained with relevance data extracted from the anonymous and aggregated PubMed search logs over an extended period of time. There are also research papers applying learning-to-rank to biomedical ad hoc retrieval [79], [87].

Some effort have been spent on diversifying the search results [79], [86]. The goal of this branch of work is to return as many aspects of information relevant to the query as possible in a small number of top results. The assumption is that users do not like a term or aspect to dominate the first page of results, if the query itself is multi-facet.

5 Discussion

This report reviews currently available datasets and methods for biomedical literature ad hoc retrieval, with comparison to similar tasks in the general IR domain. The review is organized around three challenges for ad hoc retrieval: databases, query/document interpretation, and ranking functions. The general IR domain is years ahead of the biomedical IR domain in many aspects, while the biomedical domain has the advantage of a large amount of well-defined terminologies. Various domain-specific methods have been developed to utilize domain-specific knowledge to assist search. Being rich in data and rather primitive in query/document interpretation and ranking, there is large room for future work in the biomedical literature search domain.

References

[1] R. K. Ando, M. Dredze, and T. Zhang. TREC 2005 genomics track experiments at IBM watson. *TREC 2005*. http://trec.nist.gov/pubs/trec14/papers/ibm-tjwatson.geo.pdf.

[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. *The semantic web*, pages 722–735, 2007.

9. https://www.ncbi.nlm.nih.gov/books/NBK3827/#pubmedhelp.Consumer_Healt

[3] K. Balog, P. Serdyukov, and A. P. de Vries. Overview of the TREC 2010 entity track. *TREC 2010*. http://trec.nist.gov/pubs/trec19/papers/ENTITY.OVERVIEW.pdf.

[4] M. Beaulieu, M. Gatford, X. Huang, S. Robertson, S. Walker, and P. Williams. Okapi at trec-5. *NIST SPECIAL PUBLICATION SP*, pages 143–166, 1997.

[5] J. Beel and B. Gipp. Google scholars ranking algorithm: an introductory overview. In *Proceedings of the 12th International Conference on Scientometrics and Informetrics (ISSI09)*, volume 1, pages 230–241. Rio de Janeiro (Brazil), 2009.

[6] M. Bendersky, W. B. Croft, and D. A. Smith. Joint annotation of search queries. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 102–111. Association for Computational Linguistics, 2011.

[7] S. Bergsma and Q. I. Wang. Learning noun phrase query segmentation. In *EMNLP-CoNLL*, volume 7, pages 819–826, 2007.

[8] O. Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl 1):D267–D270, 2004.

[9] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM, 2008.

[10] W. C. Brandão, R. L. Santos, N. Ziviani, E. S. Moura, and A. S. Silva. Learning to expand queries using entities. *Journal of the Association for Information Science and Technology*, 65(9):1870–1883, 2014.

[11] M. Braschler and C. Peters. Cross-language evaluation forum: Objectives, results, achievements. *Information retrieval*, 7(1):7–31, 2004.

[12] A. Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36, pages 3–10. ACM, 2002.

[13] C. J. Bult, J. T. Eppig, J. A. Kadin, J. E. Richardson, J. A. Blake, M. G. D. Group, et al. The mouse genome database (mgd): mouse biology and model systems. *Nucleic acids research*, 36(suppl 1):D724–D728, 2008.

[14] C. Carpineto and G. Romano. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44(1):1, 2012.

[15] J. M. Cherry, C. Adler, C. Ball, S. A. Chervitz, S. S. Dwight, E. T. Hester, Y. Jia, G. Juvik, T. Roe, M. Schroeder, et al. Sgd: Saccharomyces genome database. *Nucleic acids research*, 26(1):73–79, 1998.

[16] C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 web track. *TREC 2009*. http://trec.nist.gov/pubs/trec18/papers/WEB09.OVERVIEW.pdf.

[17] A. M. Cohen and W. R. Hersh. A survey of current work in biomedical text mining. *Briefings in bioinformatics*, 6(1):57–71, 2005.

[18] J. Dalton, L. Dietz, and J. Allan. Entity query feature expansion using knowledge base links. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 365–374. ACM, 2014.

[19] A. P. Davis, T. C. Wiegers, M. C. Rosenstein, and C. J. Mattingly. Medic: a practical disease vocabulary used at the comparative toxicogenomics database. *Database*, 2012:bar065, 2012.

[20] L. Dietz, A. Kotov, and E. Meij. Utilizing knowledge bases in text-centric information retrieval. In *Proceedings of the 2016 ACM on International Conference on the Theory of Information Retrieval*, pages 5–5. ACM, 2016.

[21] H. Duan and B.-J. P. Hsu. Online spelling correction for query completion. In *Proceedings of the 20th international conference on World wide web*, pages 117–126. ACM, 2011.

[22] O. Egozi, S. Markovitch, and E. Gabrilovich. Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems (TOIS)*, 29(2):8, 2011.

[23] P. Ernst, A. Siu, D. Milchevski, J. Hoffart, and G. Weikum. Deeplife: An entity-aware search, analytics and exploration platform for health and life sciences. *ACL 2016*, page 19, 2016.

[24] P. Ferragina and U. Scaiella. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1625–1628. ACM, 2010.

[25] S. Fujita. Revisiting again document length hypotheses TREC-2004 genomics track experiments at patolis. *TREC 2004*. http://trec.nist.gov/pubs/trec13/papers/patolis.geo.pdf.

[26] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971, 1987.

[27] J. Giles. Science in the web age: Start your engines. *Nature*, 438(7068):554–555, 2005.

[28] D. F. Gleich. Pagerank beyond the web. *SIAM Review*, 57(3):321–363, 2015.

[29] H. Gurulingappa, B. Müller, M. Hofmann-Apitius, and J. Fluck. Information retrieval framework for technology survey in biomedical and chemistry literature. In *TREC*. Citeseer, 2011.

[30] A. Hamosh, A. F. Scott, J. S. Amberger, C. A. Bocchini, and V. A. McKusick. Online mendelian inheritance in man (omim), a knowledgebase of human genes and genetic disorders. *Nucleic acids research*, 33(suppl 1):D514–D517, 2005.

[31] F. Hasibi, K. Balog, and S. E. Bratsberg. Entity linking in queries: Tasks and evaluation. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pages 171–180. ACM, 2015.

[32] W. Hersh, C. Buckley, T. Leone, and D. Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *SIGIR94*, pages 192–201. Springer, 1994.

[33] W. Hersh and E. Voorhees. Trec genomics special issue overview. *Information Retrieval*, 12(1):1–15, 2009.

[34] W. R. Hersh and R. T. Bhupatiraju. TREC 2003 genomics track overview. *TREC 2003*. http://trec.nist.gov/pubs/trec12/papers/GENOMICS.OVERVIEW3.pdf.

[35] W. R. Hersh, R. T. Bhupatiraju, L. Ross, P. Johnson, A. M. Cohen, and D. F. Kraemer. TREC 2004 genomics track overview. *TREC 2004*. http://trec.nist.gov/pubs/trec13/papers/OVERVIEW13.pdf.

[36] W. R. Hersh, A. Cohen, L. Ruslen, and P. Roberts. TREC 2007 genomics track overview. *TREC 2007*. http://trec.nist.gov/pubs/trec16/papers/GEO.OVERVIEW16.pdf.

[37] W. R. Hersh, A. Cohen, J. Yang, R. T. Bhupatiraju, P. Roberts, and M. Hearst. TREC 2005 genomics track overview. *TREC 2005*. http://trec.nist.gov/pubs/trec14/papers/GEO.OVERVIEW.pdf.

[38] L. Hirschman, M. Colosimo, A. Morgan, and A. Yeh. Overview of biocreative task 1b: normalized gene lists. *BMC bioinformatics*, 6(1):S11, 2005.

[39] C.-C. Huang and Z. Lu. Community challenges in biomedical text mining over 10 years: success, failure and the future. *Briefings in bioinformatics*, 17(1):132–144, 2016.

[40] X. Huang, M. Zhong, and L. Si. York university at TREC 2005: Genomics track. *TREC 2005*. http://trec.nist.gov/pubs/trec14/papers/yorku-huang2.geo.pdf.

[41] M. Iwayama, A. Fujii, N. Kando, and A. Takano. Overview of patent retrieval task at ntcir-3. In *Proceedings of the ACL-2003 workshop on Patent corpus processing-Volume 20*, pages 24–32. Association for Computational Linguistics, 2003.

[42] B. J. Jansen, A. Spink, and T. Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information processing & management*, 36(2):207–227, 2000.

[43] J. Jiang and C. Zhai. An empirical study of tokenization strategies for biomedical information retrieval. *Information Retrieval*, 10(4-5):341–363, 2007.

[44] M. Kayaalp, A. R. Aronson, S. M. Humphrey, N. C. Ide, L. K. Tanabe, L. H. Smith, D. Demner-Fushman, R. F. Loane, J. G. Mork, and O. Bodenreider. Methods for accurate retrieval of MEDLINE citations in functional genomics. *TREC 2003*. https://mor1.nlm.nih.gov/pubs/pdf/2003-trec-mk.pdf.

[45] R. Leaman, R. I. Doğan, and Z. Lu. Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics*, page btt474, 2013.

[46] R. Leaman, C.-H. Wei, and Z. Lu. tmchem: a high performance approach for chemical named entity recognition and normalization. *Journal of cheminformatics*, 7(1):S3, 2015.

[47] H. Li, J. Xu, et al. Semantic matching in search. *Foundations and Trends® in Information Retrieval*, 7(5):343–469, 2014.

[48] T.-Y. Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

[49] X. Liu and H. Fang. Latent entity space: a novel retrieval approach for entity-bearing queries. *Information Retrieval Journal*, 18(6):473–503, 2015.

[50] X. Liu, P. Yang, and H. Fang. Entity came to rescue-leveraging entities to minimize risks in web search. Technical report, DTIC Document, 2014.

[51] Y. Lu, H. Fang, and C. Zhai. An empirical study of gene synonym query expansion in biomedical information retrieval. *Information Retrieval*, 12(1):51–68, 2009.

[52] Z. Lu. Pubmed and beyond: a survey of web tools for searching biomedical literature. *Database*, 2011:baq036, 2011.

[53] Z. Lu, H.-Y. Kao, C.-H. Wei, M. Huang, J. Liu, C.-J. Kuo, C.-N. Hsu, R. T.-H. Tsai, H.-J. Dai, N. Okazaki, et al. The gene normalization task in biocreative iii. *BMC bioinformatics*, 12(8):S2, 2011.

[54] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 39(suppl 1):D52–D57, 2011.

[55] C. D. Manning, P. Raghavan, H. Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[56] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8. ACM, 2011.

[57] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM, 2007.

[58] D. Milne and I. H. Witten. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 509–518. ACM, 2008.

[59] A. A. Morgan, Z. Lu, X. Wang, A. M. Cohen, J. Fluck, P. Ruch, A. Divoli, K. Fundel, R. Leaman, J. Hakenberg, et al. Overview of biocreative ii gene normalization. *Genome biology*, 9(2):S3, 2008.

[60] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[61] M. V. Plikus, Z. Zhang, and C.-M. Chuong. Pubfocus: semantic medline/pubmed citations analytics through integration of controlled biomedical dictionaries and ranking algorithm. *BMC bioinformatics*, 7(1):424, 2006.

[62] J. Pomikálek, M. Jakubícek, and P. Rychlỳ. Building a 70 billion word corpus of english from clueweb. In *LREC*, pages 502–506, 2012.

[63] H. Poon, C. Quirk, C. DeZiel, and D. Heckerman. Literome: Pubmed-scale genomic knowledge base in the cloud. *Bioinformatics*, 30(19):2840–2842, 2014.

[64] K. D. Pruitt and D. R. Maglott. Refseq and locuslink: Ncbi gene-centered resources. *Nucleic acids research*, 29(1):137–140, 2001.

[65] H. Raviv, O. Kurland, and D. Carmel. Document retrieval using entity-based language models. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 65–74. ACM, 2016.

[66] J. J. Rocchio. Relevance feedback in information retrieval. 1971.

[67] L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946, 2012.

[68] N. Sioutos, S. de Coronado, M. W. Haber, F. W. Hartel, W.-L. Shaiu, and L. W. Wright. Nci thesaurus: a semantic model integrating cancer-related clinical and molecular information. *Journal of biomedical informatics*, 40(1):30–43, 2007.

[69] A. Spink and T. Saracevic. Interaction in information retrieval: selection and effectiveness of search terms. *JASIS*, 48(8):741–761, 1997.

[70] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM, 2007.

[71] K. Toutanova and R. C. Moore. Pronunciation modeling for improved spelling correction. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 144–151. Association for Computational Linguistics, 2002.

[72] S. Tweedie, M. Ashburner, K. Falls, P. Leyland, P. McQuilton, S. Marygold, G. Millburn, D. Osumi-Sutherland, A. Schroeder, R. Seal, et al. Flybase: enhancing drosophila gene ontology annotations. *Nucleic acids research*, 37(suppl 1):D555–D559, 2009.

[73] E. M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR94*, pages 61–69. Springer, 1994.

[74] C.-H. Wei, B. R. Harris, H.-Y. Kao, and Z. Lu. tmvar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*, page btt156, 2013.

[75] C.-H. Wei, H.-Y. Kao, and Z. Lu. Sr4gn: a species recognition software tool for gene normalization. *PloS one*, 7(6):e38460, 2012.

[76] C.-H. Wei, H.-Y. Kao, and Z. Lu. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, page gkt441, 2013.

[77] C.-H. Wei, H.-Y. Kao, and Z. Lu. Gnormplus: an integrative approach for tagging genes, gene families, and protein domains. *BioMed research international*, 2015, 2015.

[78] P. Willett. The porter stemming algorithm: then and now. *Program*, 40(3):219–223, 2006.

[79] J. Wu, J. X. Huang, and Z. Ye. Learning to rank diversified results for biomedical information retrieval from multiple features. *Biomedical engineering online*, 13(2):S3, 2014.

[80] C. Xiong and J. Callan. Esdrank: Connecting query and documents through external semi-structured data. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 951–960. ACM, 2015.

[81] C. Xiong and J. Callan. Query expansion with freebase. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pages 111–120. ACM, 2015.

[82] C. Xiong, J. Callan, and T.-Y. Liu. Bag-of-entity representation for ranking. In *Proceedings of the sixth ACM International Conference on the Theory of Information Retrieval (ICTIR 2016)*, pages 181–184, 2016.

[83] C. Xiong, R. Power, and J. Callan. Explicit semantic ranking for academic search via knowledge graph embedding. In *ACM WWW*, 2017.

[84] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM, 1996.

[85] Y. Xu, G. J. Jones, and B. Wang. Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 59–66. ACM, 2009.

[86] X. Yin, X. Huang, and Z. Li. Promoting ranking diversity for biomedical information retrieval using wikipedia. In *European Conference on Information Retrieval*, pages 495–507. Springer, 2010.

[87] H. Yu, T. Kim, J. Oh, I. Ko, S. Kim, and W.-S. Han. Enabling multi-level relevance feedback on pubmed by integrating rank learning into dbms. *BMC bioinformatics*, 11(2):S6, 2010.