An Experimentation Platform for Precision Medicine

Vincent Nguyen
vincent.nguyen@anu.edu.au
CSIRO Data61 & Australian National University
Canberra, ACT, Australia

Sarvnaz Karimi
Brian Jin
firstname.lastname@csiro.au
CSIRO Data61
Sydney, NSW, Australia

ABSTRACT

Precision medicine—where data from patients, their genes, their lifestyles and the available treatments and their combination are taken into account for finding a suitable treatment—requires searching the biomedical literature and other resources such as clinical trials with the patients’ information. The retrieved information could then be used in curating data for clinicians for decision-making. We present information retrieval researchers with an online system which enables experimentation in search for precision medicine within the framework provided by the TREC Precision Medicine (PM) track. A number of query and document processing and ranking approaches are provided. These include some of the most promising gene mention expansion methods, as well as learning-to-rank using neural networks.

CCS CONCEPTS

• **Information systems** → *Learning-to-rank*; *Query reformulation*; **Specialized information retrieval**;

KEYWORDS

Health informatics, Literature search, Domain-Specific Search

ACM Reference Format:
Vincent Nguyen, Sarvnaz Karimi, and Brian Jin. 2019. An Experimentation Platform for Precision Medicine. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR ’19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3331184.3331396

1 INTRODUCTION

Improving the treatment success of cancer patients relies on providing the right information to practising clinicians. While some of this information is published in the biomedical literature, searching among over 27 million MEDLINE abstracts, with new articles added each minute, makes it difficult if not impossible to know all the latest treatment options. Similarly, it is not straightforward to find clinical trials that a patient is eligible for. The goal of the TREC Precision Medicine (PM) track [13, 14] is to facilitate research in this domain by providing cancer patient scenarios (Figure 1) and their corresponding clinical trials[1] as well as treatment options in MEDLINE and abstracts from *The American Association for Cancer Research* (AACR) and the *American Society of Clinical Oncology* (ASCO). This track is in its third year, and 59 teams collectively participated in the past two years (32 teams with 258 runs in 2017 and 27 teams with 193 runs in 2018). These teams report a number of different techniques. Some are common, such as expanding genes and variants in the topics [6], and some are unique, such as creating a specialized knowledge graph for the task [17].

disease	Acute lymphoblastic leukemia
gene	ABL1, PTPN11
demographic	12-year-old male

Figure 1: A sample topic from TREC PM 2018 (Topic 1).

We provide a platform for researchers to experiment with some of the most popular query expansion and ranking methods for the PM track. We also implement learning-to-rank using the latest deep learning-based methods in text classification, including a language representation model called *Bidirectional Encoder Representations from Transformers* (BERT) [5] and *Universal Language Model Fine-tuning* (ULMFiT) [7]. Our aim is to facilitate experimentation within this area, and hence advance the field. Our system is built on top of the A2A[2] system [8] which is designed for experimentation within the TREC Clinical Decision Support track.

2 RELATED SYSTEMS

Literature on search for precision medicine is limited, with most relevant studies reported by the TREC PM 2017 and 2018 participants [13, 14]. These reports however often are work in progress and lack enough details on the methods and implementation details, making it difficult to reproduce the results.

There are other related systems in place. One is proposed by Koopman et al. [9] with a task-based search engine to assist in clinical search. Another platform is *EvALL* [1] where the output of different systems can be compared in the same setting. An information retrieval experimentation platform that uses *Domain Specific Language* (DSL) is proposed by Scells et al. [15] which can be used for systematic reviewing in the legal or medical domain.

Marshall et al. [11] release an open-source web-based system, *RobotReviewer*, which takes input biomedical articles or clinical trials and processes them for extraction and synthesis of evidence for the practice of Evidence-Based Medicine. We see our system

SIGIR ’19, July 21–25, 2019, Paris, France
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/3331184.3331396

[1] https://clinicaltrials.gov/

[2] https://www.vizie.csiro.au/trec-eval

as a complementary first step before RobotReviewer to find the articles to be processed for evidence.

3 INDEXING

Our system provides the index of three collections: MEDLINE abstracts, AACR and ASCO abstracts, and clinical trials that were part of the TREC PM supplied data collections. All documents in the MEDLINE collection are stemmed with stopwords removed automatically at index time by Solr[3]. Medline abstracts documents have the following fields indexed: pmid (Pubmed ID), pmcid (Pubmed Central ID), title, abstract, article type, MeSH headings, article keywords and date published. Similarly, clinical trials are indexed with the following fields: brief title, minimum age, maximum age, official title, brief summary, detailed description, nct-id (clinical trial registry number), intervention type and intervention, inclusion and exclusion criteria, condition browse (MeSH keywords), referenced MEDLINE identifiers (if they exist) and primary outcome. Inclusion and exclusion criteria are extracted by using regular expressions and can be used to restrict the demographics of retrieved documents during query time. The age of patients are converted to age in days to avoid floating point and date arithmetic. Finally, the AACR and ASCO abstracts are indexed with the fields: id, meeting, title and abstract.

4 QUERY PROCESSING TECHNIQUES

We provide query expansion options as shown on the left-hand-side of Figure 2. Apart from standard query expansion techniques, such as pseudo-relevance feedback, expansion with domain-specific terminology from Unified Medical Language System (UMLS) metathesaurus, and expansion using semantically related terms determined with word embeddings of Wikipedia and MEDLINE, we provide gene and disease expansion as explained below.

Gene Expansion. Gene expansion uses genes identified by Metamap [2] to expand the topics using one of the four available options: (1) Expansion using Metamap, which expands the query using Metamap suggested concepts that are restricted to the following semantic types: Phenomenon or Process, Cell or Molecular Dysfunction, Molecular Biology Research Technique, Enzyme, *Amino Acid, Peptide, or Protein*, Gene or Genome, Biologically Active Substance, Pharmacologic Substance, Genetic Function, Organic Chemical, Neoplastic process, Molecular Function and Receptor. (2) Expansion using Wikipedia API[4]. (3) Expansion using the Human Gene Ontology[5] expands the abbreviated gene names with the first match found in the ontology. For instance, *cyclin-dependent kinase 4* is expanded from cdk4 through the use of the ontology.

Users can also use a combination of these options.

Disease Expansion. Disease expansion relies on Metamap to identify the disease names. They can then be expanded using one the three options:

- Metamap filtering, which uses UMLS concepts restricted to the semantics types of: Disease or Syndrome, Sign or Symptom, Pathologic Function, Anatomical Abnormality, Clinical Drug, Clinical Attribute and Neoplastic Process. Using these semantics types, we extract UMLS concepts using *MetamapLite* [3] which we denote as T_M, and extract terms using the Wikipedia API which we denote as T_W. We use the set intersection, $T_M \cup T_W$, in order to produce a final set of expansion terms.
- Semantic variation, in which disease mentions are expanded by finding semantically relevant words (at most three) with either Wikipedia, or MEDLINE word embeddings which are trained using Word2Vec.[6]

Demographic Attribute Expansion and Filtering. Clinical trials documents present demographic attributes transparently. As such, we are able to normalise demographic attributes found within queries to exact matches found within the clinical trial corpus. If *Normalize demographics* is chosen, queries containing strings indicating a child (e.g. *person at the age of 6*) will be expanded with the word *child*, and a query with the term *female* will be expanded with the terms *woman women*, and similarly for male with *man men*. The ages in the queries were normalised to be expressed in days to conform with the index. We use Solr's boolean query operators to exclude clinical trial documents that do not contain the appropriate demographic target group. An example of such a boolean query is:

```
-gender:male AND maximum_age:[0 TO 5110]
```

This operation excludes documents that are either for males or individuals over the age of 15 (5,510 days). Conversely, it will restrict the document result set to contain only patients that are both female and under the age of 15.

5 RANKING MODELS

We provide BM25 and language modelling ranking models from Apache Solr. For the case of BM25, we provide the option of tuning the *b* and *k1* parameters. Language model uses Dirichlet similarity.

Learning-to-rank. We have four different implementations of learning-to-rank (LTR) available:

1. SVM with Word Embedding: We use word embeddings created by Chiu et al. [4] from PubMed, where they released their best hyper-parameters that are empirically identified. Mean word embeddings are used to represent documents.
2. SVM with LETOR [12] features: Features are TF-IDF of each term in the query in each of the document facets, including title, abstract, and the full text (for clinical trials). We also use BM25 score of each facet, TF-IDF values, language model features (Dirichlet and Jelinek-Mercer) of each facet, and length of each facet.
3. ULMFit [7]: We use a language-model encoder for the generic WikiText-103 model and then fine-tune a Recurrent Neural Network (RNN) classifier using the language model encoder for LTR.
4. BERT [5]: The pre-trained bert-base-uncased model is fine-tuned with default BertConfig settings with a BERT classifier and a scaled loss function based on label frequency (more

[3] Solr version 6.6.0 http://lucene.apache.org/solr/

[4] https://pypi.python.org/pypi/wikipedia

[5] http://www.geneontology.org/ (accessed 16 Jan 2018)

[6] https://radimrehurek.com/gensim/models/word2vec.html

frequent label loss is scaled down, while less frequent label loss is scaled up).[7]

Training for the LTR models is based on the *training source* that the users select (2017 or 2018 topics). One limitation on this setting is its fixed set of fine-tuned parameters. In a later version of the system, we will add more options to make it more flexible.

6 RE-RANKING USING CITATIONS

Clinical trials documents often cite MEDLINE articles. We implement a heuristic in order to utilise these relationships between clinical trial and MEDLINE articles. Given a query, if a MEDLINE article is referenced in a highly relevant or high scoring clinical trial, it receives a small boost based on the rank of that clinical trial document. We also applied the same boost such that clinical trials received a boost to their scores if they are linked to a high scoring MEDLINE document. These boosts were small and decreased exponentially with reciprocal rank:

$$S_d = S_d + b(R_d) \tag{1}$$

where S_d is the score of a document, and b is a boosting function that uses reciprocal ranking of a matched document in the second or paired document set,

$$b(R_d) = \frac{1}{exp(R_d)} \tag{2}$$

Merging search results using federated search. A user can choose to only search on clinical trials or only literature (a combination of MEDLINE, and ACCR & ASCO abstracts). However, if they choose both, our system can merge the results using federated search algorithms. We use the Generalized Document Scoring [10] (GDS) algorithm to achieve this. The GDS algorithm calculates the document score by determining the amount of overlap between the document and the query. This score is normalized to be between 0 (no overlap) and $\sqrt{2}$ (complete overlap). However, a drawback of using this method is that the query is treated as a bag-of-words (each term is given the same weight) and it does not distinguish important terms such as disease and gene mentions. In order to mitigate this, after applying GDS, we boost the scores of documents containing the disease name and gene mutation.

However, a limitation of GDS is that it can only be applied to one field at a time or the entire document at once, which is undesirable as the most important parts of the document are limited to only a few facets such as title, abstract and article keywords for MEDLINE documents, brief/official title, brief summary, detailed description and abstract for clinical trial and title and abstract for AACR & ASCO articles. We hence apply GDS to each field if they exist. Otherwise, we take the rank score detailed by the equation below as fallback:

$$RS(d, f) = 1 - \frac{rank_d}{|D| * 10} \tag{3}$$

where RS is the rank score function, d is a document in all retrieved documents D, f is the facet that doesn't exist on the current document and $rank_d$ denotes the rank of the document.

The facets for each document are normalized using weights; this ensures that when comparing collections with a different number of fields, for example, AACR & ASCO which do not have a keyword field while MEDLINE does, we are able to fairly compare between collections.

Alternatively, we use another more simple merging strategy called Randomized Round Robin (SRR). In the round robin merging algorithm, each document from each collection is interlaced into a final ranked list. This ensures that the final merged list will maintain the same ordering between documents of the same collection regardless of the differences in magnitude of scores between the collections. The variant, randomised round robin, will randomly select which collection to sample in the round robin merging process for each document.

7 EVALUATION METRICS

The system generates the results offline and creates an email notification. The execution time depends on the load of the system as well as what combination of the methods have been selected. For simple runs which use standard ranking models, it can be on the order of minutes. The retrieved results are evaluated using TREC standard scripts (*trec_eval* and *sample_eval*). Each search request generates results using the following metrics: infNDCG [16], infAP, iAP (inferred Average Precision), iP@10, NDCG, P@10 (Precision at rank 10), R-Prec (Recall-Precision), B-Pref and MAP (Mean Average Precision).

8 DEMONSTRATION SYSTEM

A screenshot of the experimental design page is shown in Figure 2. It shows a setting where a list of topics can be chosen as well as some of the implemented techniques.

At this point in time, the learning-to-rank models are fixed to our trained models. However, these will get expanded with the options of selecting between the features (for SVM LETOR), and changing hyperparameters for the models based on neural networks. Apart from the online system, the code for the system will be publicly available on a GitHub repository.

9 ACKNOWLEDGEMENT

Authors would like to acknowledge the contributions of Maciej Rybinski (CSIRO Data61) in improving the system post submission of this article. We also thank CSIRO Precision Health Future Science Platform (PH FSP) funding of the project. Vincent Nguyen is on CSIRO Research Office top-up PhD scholarship on the project and is also supported by the Australian Government Research Training Program Scholarship.

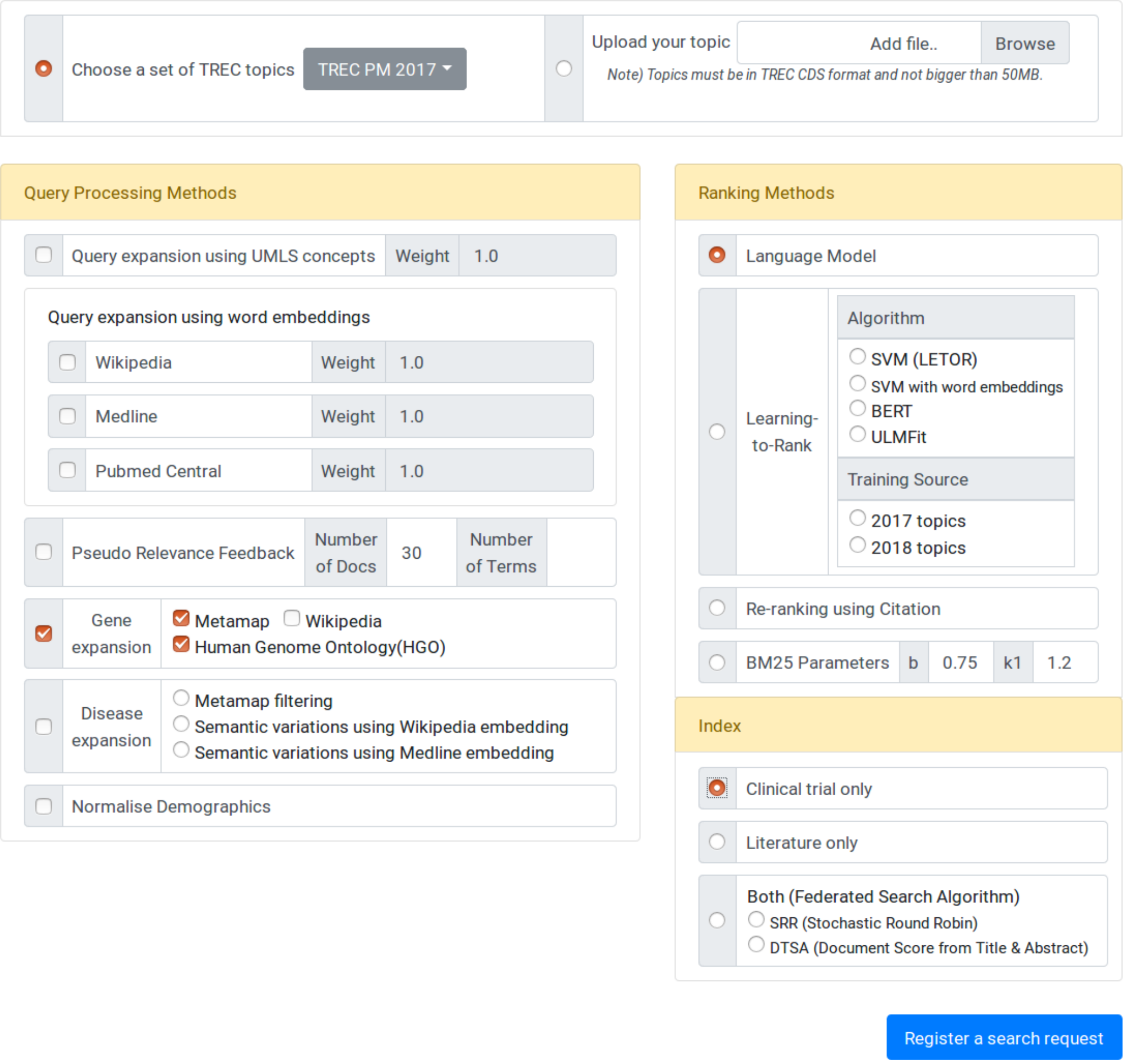
REFERENCES

[1] E. Amigó, J. Carrillo-de Albornoz, M. Almagro-Cádiz, J. Gonzalo, J. Rodríguez-Vidal, and F. Verdejo. 2017. EvALL: Open Access Evaluation for Information Access Systems. In *SIGIR.* 1301–1304.

[2] A. Aronson and F. Lang. 2010. An overview of MetaMap: Historical perspective and recent advances. *Journal of the American Medical Informatics Association* 17, 3 (2010), 229–236.

[3] A. Aronson, W. Rogers, and D. Demner-Fushman. 2017. MetaMap Lite: an evaluation of a new Java implementation of MetaMap. *Journal of the American Medical Informatics Association* 24, 4 (2017), 841–844.

[7] We use and modify the code provided in: https://github.com/huggingface/pytorch-pretrained-BERT.



● Choose a set of TREC topics TREC PM 2017 ▾

○ Upload your topic Add file.. Browse

Note) Topics must be in TREC CDS format and not bigger than 50MB.

Query Processing Methods

□ Query expansion using UMLS concepts Weight 1.0

Query expansion using word embeddings

□ Wikipedia Weight 1.0

□ Medline Weight 1.0

□ Pubmed Central Weight 1.0

□ Pseudo Relevance Feedback Number of Docs 30 Number of Terms

☑ Gene expansion ☑ Metamap □ Wikipedia ☑ Human Genome Ontology(HGO)

□ Disease expansion ○ Metamap filtering ○ Semantic variations using Wikipedia embedding ○ Semantic variations using Medline embedding

□ Normalise Demographics

Ranking Methods

● Language Model

○ Learning-to-Rank
Algorithm ○ SVM (LETOR) ○ SVM with word embeddings ○ BERT ○ ULMFit
Training Source ○ 2017 topics ○ 2018 topics

○ Re-ranking using Citation

○ BM25 Parameters b 0.75 k1 1.2

Index

● Clinical trial only

○ Literature only

○ Both (Federated Search Algorithm) ○ SRR (Stochastic Round Robin) ○ DTSA (Document Score from Title & Abstract)

Figure 2: System interface showing some of the options provided for the users.

[4] B. Chiu, G. Crichton, A. Korhonen, and S. Pyysalo. 2016. How to Train good Word Embeddings for Biomedical NLP. In *ACL Workshop on Biomedical Natural Language Processing*. 166–174.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805

[6] T. Goodwin, M. Skinner, and S. Harabagiu. 2017. UTD HLTRI at TREC 2017: Precision Medicine Track. In *TREC*. Gaithersburg, MD.

[7] J. Howard and S. Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *The 56th Annual Meeting of the Association for Computational Linguistics*. Melbourne, Australia, 328–339.

[8] S. Karimi, V. Nguyen, F. Scholer, B. Jin, and S. Falamaki. 2018. A2A: Benchmark Your Clinical Decision Support Search. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. Ann Arbor, MI, 1277–1280.

[9] B. Koopman, G. Zuccon, and J. Russell. 2017. A Task-oriented Search Engine for Evidence-based Medicine. In *SIGIR*. Shinjuku, Tokyo, Japan, 1329–1332.

[10] P. Li, P. Thomas, and D. Hawking. 2013. Merging Algorithms for Enterprise Search. In *ADCS*. 42–49.

[11] I. Marshall, J. Kuiper, E. Banner, and B. Wallace. 2017. "Automating Biomedical Evidence Synthesis: RobotReviewer. In *ACL*. Vancouver, Canada, 7–12.

[12] T. Qin, T-Y Liu, J. Xu, and H. Li. 2010. LETOR: A Benchmark Collection for Research on Learning to Rank for Information Retrieval. *Information Retrieval* 13, 4 (2010), 346–374.

[13] K. Roberts, D. Demner-Fushman, E. Voorhees, W. Hersh, S. Bedrick, and A. Lazar. 2018. Overview of the TREC 2018 Precision Medicine Track. In *TREC*. Gaithersburg, MD.

[14] K. Roberts, D. Demner-Fushman, E. Voorhees, W. Hersh, S. Bedrick, A. Lazar, and S. Pant. 2017. Overview of the TREC 2017 Precision Medicine Track. In *TREC*. Gaithersburg, MD.

[15] H. Scells, D. Locke, and G. Zuccon. 2018. An Information Retrieval Experiment Framework for Domain Specific Applications. In *SIGIR*. Ann Arbor, MI, 1281–1284.

[16] E. Yilmaz and J.A. Aslam. 2006. Estimating Average Precision with Incomplete and Imperfect Judgments. In *CIKM*. 102–111.

[17] X. Zhou, X. Chen, J. Song, G. Zhao, and J. Wu. 2018. Team Cat-Garfield at TREC 2018 Precision Medicine Track. In *TREC*. Gaithersburg, MD.