Team Cat-Garfield at TREC 2018 Precision Medicine Track

Xuesi Zhou[1], Xin Chen[1], Jian Song[1], Gang Zhao[1], and Ji Wu[1]

Department of Electronic Engineering, Tsinghua University, Beijing, China
{zhouxs16, j-song17, chenx17}@mails.tsinghua.edu.cn
zhaogang12@126.com
wuji_ee@mail.tsinghua.edu.cn

Abstract. This paper describes the system designed for the TREC 2018 Precision Medicine Track by the Team Cat-Garfield from Tsinghua University. Our system is composed of two parts: the retrieval part aiming at high recall metric and the re-ranking part aiming at boosting infNDCG, P@10 and R-prec metrics via a heuristic scoring scheme. In order to introduce more external information into our systems, we developed a knowledge graph named TREC-KG, which plays a significant role in both the query expansion and the re-ranking part, and we utilized the qrels of TREC 2017 Precision Medicine Track as annotated data to train a classifier for precision medicine-relatedness of biomedical articles, named PM Classifier. In particular, we employed a hybrid method to score the precision medicine-relatedness of each retrieved article, combining the data-driven PM Classifier and a rule-based scoring scheme. The final results demonstrate that our systems are effective in both the biomedical article retrieval task and the clinical trial retrieval task, and the usage of the PM Classifier would bring performance improvement on R-prec metric in the biomedical article retrieval task.

Keywords: TREC 2018 · Information Retrieval · Precision Medicine

1 Introduction

Precision medicine is a term used to describe individualized treatment that encompasses the use of new diagnostics and therapeutics, targeted to the needs of a patient based on his/her own genetic, biomarker, phenotypic, or psychosocial characteristics[7].

Same as the track in 2017, TREC 2018 Precision Medicine Track focuses on the retrieval of biomedical articles and clinical trials which can provide useful precision medicine-related information to clinicians treating cancer patients. Participants are challenged with two tasks, the biomedical article retrieval task and the clinical trial retrieval task. The participants' retrieval systems are required to identify the biomedical articles addressing treatments in the perspective of precision medicine-relatedness, and the clinical trials for which the patient is eligible.

The track provides 50 topics consist of synthetic patient cases. A little different from 2017, the ”other” field is removed, and the 2018 track introduces several immunotherapy-related topics, which do not contain explicit genes and mutations, but rather description of biomarkers for immunotherapy.

In this work, we present our systems for the two tasks. For both the biomedical article retrieval task and the clinical trial retrieval task, our submitted systems share the same architecture, which consists of a retrieval part and a reranking part. In the retrieval part, we index the given document collections, and retrieve documents for each topic via Apache Lucene[1], mainly focusing on the recall performance. In the re-ranking part, we employed a scoring scheme to re-rank the retrieved documents. Different from the retrieval part, we use concepts extracted via MetaMap[2][1], in addition to the original text, to score the relevance between the retrieved documents and topics, mainly focusing on the precision performance. We hope that this manner, using different features and focusing on different goals, would bring complementarity between the two parts and improve the retrieval performance.

In addition, we utilized some external resources. We constructed a knowledge graph, referred as TREC Knowledge Graph (TREC-KG), for both the biomedical article task and the clinical trial task And we trained a classifier for precision medicine-relatedness of a article , referred as Precision Medicine Classifier (PM Classifier). The TREC-KG and the PM Classifier are detailed in Section 2.

2 Methods

We intuitively hope to improve system performance by introducing more external information, and we have tried two methods. We have built a knowledge graph, referred as TREC-KG, from several knowledge sources of interest. The TREC-KG was used for both the biomedical article task and the clinical trial task. We also utilize the qrels of TREC 2017 Precision Medicine Track as annotated data to train a classifier for precision medicine-relatedness of a article, referred as PM Classifier. The PM Classifier was used to improve the system performance of biomedical article retrieval task by evaluating the relevance of the article to ”treatment”.

Shown as Figure 1, the architecture of our systems consists of two parts, text based document retrieval and concept based document re-ranking.

For the text based document retrieval part, we expanded the queries incorporating the TREC-KG, and then retrieved candidate documents by Apache Lucene. Then for the concept based document re-ranking part, we built a reranker incorporating the TREC-KG and the PM Classifier, using concepts extracted via MetaMap as additional features of documents, and finally obtained the re-ranked candidate documents.

[1] https://lucene.apache.org/

[2] https://metamap.nlm.nih.gov/

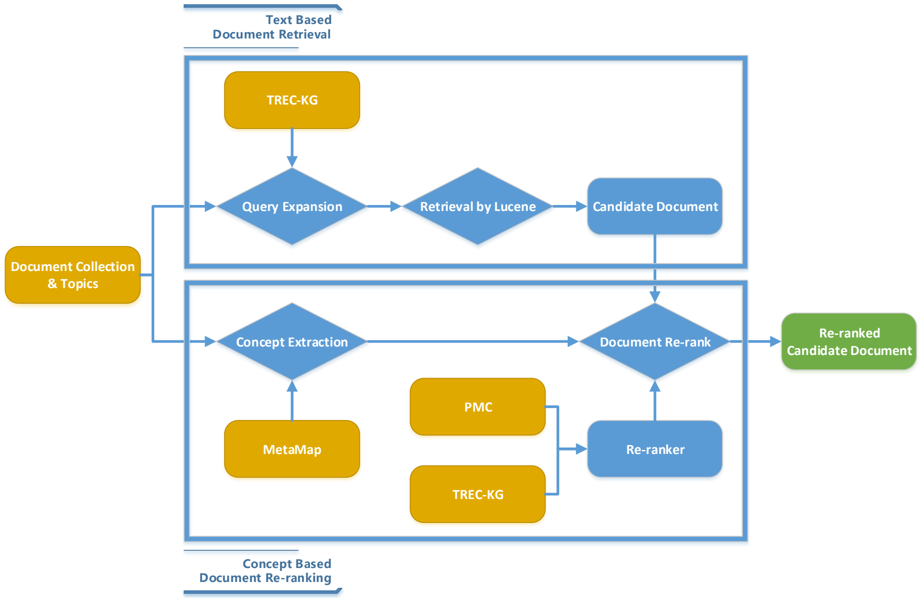


Fig. 1. Architecture of our system developed for TREC 2018 PM track. The yellow blocks indicate external resources or input. The green block indicates output. The blue blocks indicate intermediate operation or documents.

2.1 Construction of the TREC-KG

Knowledge Graph Design In order to construct a knowledge graph, the life cycle of the knowledge graph must be clearly defined. The life cycle of a knowledge graph can be roughly divided into the following three distinct stages:

1. Domain Knowledge Modelling
2. Knowledge Acquisition
3. Knowledge Integration and Identifier Normalization

Figure 2 depicts the construction of the TREC Knowledge Graph(TREC-KG). The medical knowledge bases in the leftmost part contain a variety of medical knowledge. The Downloading box in the Extraction Box corresponds to Knowledge Acquisition, the Parsing and Knowledge Extraction boxes correspond to Knowledge Integration.

Knowledge Modeling For the modelling of medical knowledge, different institutions have built a variety of knowledge bases for different purposes , such as Unified Medical System(UMLS) developed and maintained by the US National Library of Medicine[2], the Drug Gene Interaction Database developed by Kelsy C Cotto[5], the Catalogue of Somatic Mutations in Cancer(COSMIC) developed by the Sanger Institute[4] and so on. Each knowledge base makes use

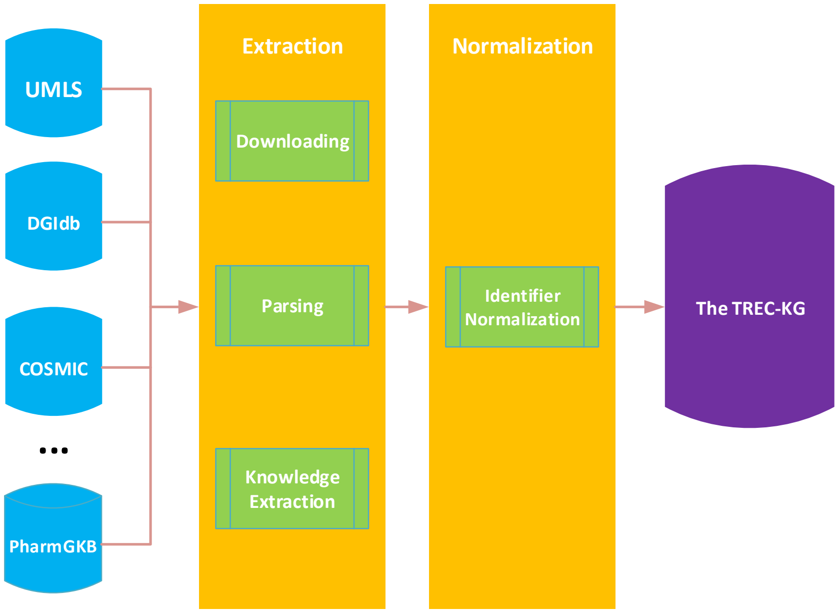


Fig. 2. Construction of the TREC-KG

of distinct methods to depict and model knowledge in the medical domain. The leftmost knowledge bases in the Figure 2 are the main sources of knowledge for the TREC-KG that we desired to build. In addition to these knowledge bases, we also incorporated Cancer Genome Interpreter(CGI)[9], CIViC[6], OncoKB[3] etc.

Knowledge Acquisition Knowledge acquisition is to extract desired knowledge which is relevant to the TREC 2018 Precision Medicine Track from vast amount of information contained in these medical knowledge bases. This phase corresponds to the Downloading in the Extraction Box of Figure 2.

Knowledge Integration and Normalization Medical knowledge from Knowledge Acquisition Phase contains massive amount of heterogeneous data due to dis-connectivity of various knowledge bases we used for the development of the TREC-KG. Therefore, Knowledge Integration is required to cope with the challenge of data heterogeneity induced by dis-connectivity of different knowledge bases. During the process of Knowledge Integration, triples in the format of (head, relation, tail) are utilized to represent medical knowledge, or rather, relationships between different entities including drugs, cancers and genes. Taking the diversity of formats adopted by individual medical knowledge resource into consideration, a variety of triple extraction interfaces should be designed specifically and carefully, respectively for each medical knowledge resource. The interface design phase corresponds to the Parsing in the Extraction Box. After the parsing interface design is completed, the design process will enter the Knowledge Extraction. During the Knowledge Extraction, triples of interest are extracted in accordance with the entity types.

Figure 3 gives an overview of the TREC-KG. This highly customized knowledge graph contains three kinds of entities: gene, cancer and drug.

The three kinds of entities contain different attributes. The attributes of a gene entity include gene name, synonyms, full name, authoritative name, and entrez id etc. The attributes of a cancer entity include synonyms, hypernyms, hyponyms, DOID, and ICD10 Code. And the attributes of a drug entity include synonyms, chembl id, generic name and brand name.

Different kinds of entities are related to each other through different relationships. For instance, the relationship between a gene entity and a drug entity involves a variety of interactions, such as inhibitors, blockers and so on. After the extraction of triples, MetaMap is applied for entity linking, or rather, mapping the literal form of entities to Concept Unique Identifier(CUI). This is exactly what Identifier Normalization does.

Finally, there are 211,743 gene-drug triples, 71,388 gene-cancer triples, 4,062 cancer-drug triples, 211,743 drug-gene triples, 7,275 cancer-gene triples and 3,824 drug-cancer triples in our TREC-KG.

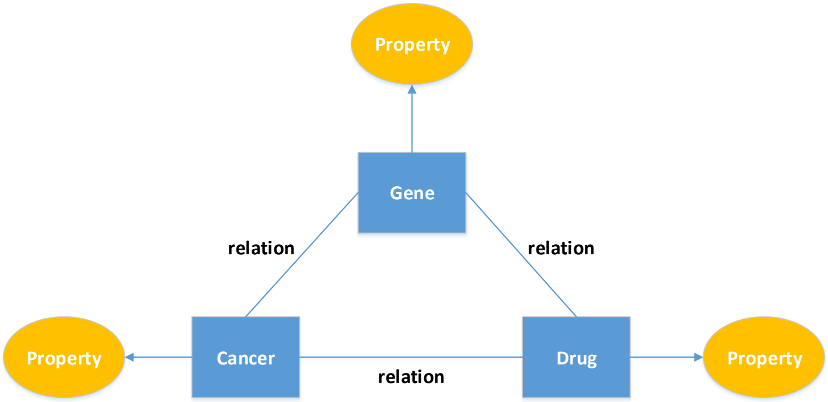


Fig. 3. Overview of the TREC-KG

2.2 Implementation of the PM Classifier

Due to the requirement that we should pay more attention to the precision medicine-relatedness of the retrieved documents, we have to consider whether a retrieved document focuses on precision medicine. Because the standards to determine whether a document is precision medicine-related is implicit, fuzzy and hard to express via human language, it is reasonable to introduce data-driven methods to tackle it. We cast the problem as a text classification problem–given a retrieved document, determine whether it focuses on precision medicine. The training corpus is the qrels of TREC 2017 Precision Medicine Track, about 20,000 retrieved biomedical articles were annotated with ”PM” for ”Precision Medicine” and ”Not PM” for ”Not Precision Medicine”.

Different from the typical form of text classification where the sentence to be classified is in literal form, we converted original sentences into sequences of CUIs (Unique Concept Identifier) in UMLS via MetaMap and conducted classification on the converted ”sentence”, or rather, sequence of CUIs. During the preprocessing of text, we only focus on the 1500 most frequently CUIs.

In order to conduct the text classification task, we employed the simple Convolutional Neural Network (TextCNN)[11] proposed by Ye Zhang and Byron Wallace in 2015, as well as the model, that is a little more complex, Text Recurrent Neural Network (TextRCNN)[8], which constructs the representation of text using a convolutional neural network and captures contextual information with the recurrent structure.

In the light of the intuition that the CNN-based model TextCNN and the RNN-based model TextRCNN may capture different aspects of semantics in text, which are essential to determine the true class of a sentence or CUI sequence, we combined the two models by concatenating the feature representations obtained after max-pooling in TextCNN and TextRCNN. The concatenated representations were then passed into a fully-connected softmax layer to determine the final class. We called the combined model as ”TextRCNN-CNN” model.

The word embeddings used in these two models were randomly initialized at the beginning of the training process. We tried different size of word embeddings, including 50, 100 and 200. The experiments showed that the word embeddings of size 50 were the best option.

For TextCNN and TextRCNN, we implemented our models using convolution kernels of 2, 3, 5, 7 with 15 feature maps each. For TextRCNN, we set the state size of the BiLSTM to 50, the same as the word embedding size. The number of hidden units of the full-connected softmax layer layer is 32. We trained our models with Adam optimizer using the learning rate of 0.001.

In order to alleviate the effect of overfitting, we divided the qrels into five groups with respect to different topics, referred as blue, yellow, green, red and black. During the training process, we trained different models using each group as test set and the other groups as training set to avoid over-fitting for specific diseases. The groups are detailed in Table 1.

Table 2 shows the performance of the PM Classifier implemented by TextCNN , TextRCNN and TextCNN-CNN respectively.

According to Table 2, the PM Classifier implemented by TextRCNN-CNN achieved the best performance on the groups of yellow, green, black, and best average performance over the five groups. We finally ensembled these TextRCNN-CNN models with majority voting strategy as our PM Classifier.

Table 1. Description of the qrels groups used to train the PM Classifier.

Group	**Num. of Articles**	**Topics**
Blue	4,298	1, 9, 10, 20, 21, 25, 27 and 30
Yellow	3,931	4, 13, 14, 23, 26 and 29
Green	3,856	2, 3, 11, 12, 15 and 19
Red	3,731	5, 6, 7, 8, 22 and 24
Black	2,873	16, 17, 18 and 28

Table 2. Performances of the PM Classifier implemented by different models on different test set (In terms of AUROC).

	Blue	Yellow	Green	Red	Black	Avg.
TextCNN	0.700	0.764	0.697	**0.669**	0.710	0.708
TextRCNN	**0.727**	0.766	0.676	0.656	0.756	0.716
TextRCNN-CNN	0.691	**0.803**	**0.709**	0.653	**0.796**	**0.730**

2.3 Text Based Document Retrieval

Indexing of Biomedical Articles We indexed both MEDLINE articles and articles published in AACR/ASCO Proceedings via Lucene. The indexing strategy varies for each article collection because the articles in the latter are only available in text format.

For MEDLINE articles in xml format, we indexed five fields, including ”PMID”, ”ArticleTitle”, ”AbstractText”, ”MeshHeadingList” and ”ChemicalList”. And in particular, the gender and age information in the ”MeshHeadingList” was indexed separately according to Table 3.

Table 3. Age description in biomedical articles.

Age(years)	**Description**
0-1	Infant
2-6	Child, Preschool
7-12	Child
13-18	Adolescent
19-24	Young Adult
25-44	Adult
45-64	Middle Aged
65-79	Aged
>80	Aged, 80 and over

In addition, we indexed the annotations of Disease, Gene and Mutation subject to each article, which are provided by PubTator[10].

For the AACR/ASCO Proceedings articles, we were only able to index a subset of the fields for MEDLINE articles, including Article ID, ArticleTitle and AbstractText, because these articles were only available in text format.

Indexing of Clinical Trials Similar to the indexing of biomedical articles, we indexed several fields of clinical trials in xml format via Lucene. We did a preprocessing for each trial because there were many fields which are not related to the retrieval task. In the pre-processing, we extracted useful fields and separated the Inclusion Criteria and Exclusion Criteria from the Eligibility field. Finally, we indexed the fields NCT ID, Title, Brief Title, Brief Summary, Detailed Description, Study Type, Intervention Type, Inclusion Criteria, Exclusion Criteria, Healthy Volunteers, Keywords and MeSH Terms. And in particular, the gender and age information in the fields of ”Gender” and ”Min/Max Age” was indexed separately according to Table 4.

Table 4. Description subject to age in Scientific Abstracts.

Age(years)	**Description**
0-24	Young
25-44	Adult
45-64	Middle
65-80	Aged
>80	Old

Query Expansion Figure 4 presents the framework of our query expansion strategy. We paid more attention to the fields of disease and gene of the topic to conduct query expansion.

For the disease field, we expanded the disease name with its synonyms, hypernyms and hyponyms via the TREC-KG. While for the gene field, it contains two aspects of information, gene name and amino acids related to the variant. For gene name, the literal form of gene name provided in the topic is not sufficient enough to retrieve relevant articles or trials due to the diversity of the gene name representations. We expanded the gene name with its synonyms, full names, related drugs which interact with the focused gene, together with corresponding interaction. For each amino acid (20 in total), we expanded it with its three-letter abbreviation coupling with single-letter abbreviation.

In particular, as for the immunotherapy-related topics, we simply expand the biomarker appearing in the gene field with their synonyms and relevant keywords representing descriptions in the topic.

Biomedical Article Retrieval We completed this part by 1) retrieval based on TF-IDF feature via Lucene and 2) post-processing.

For each topic, we retrieve biomedical articles according to the following three steps:

1. Topic Expansion: We expanded each topic by using our query expansion results. We expanded the disease with its synonyms, hyponyms and MeSH_ID,

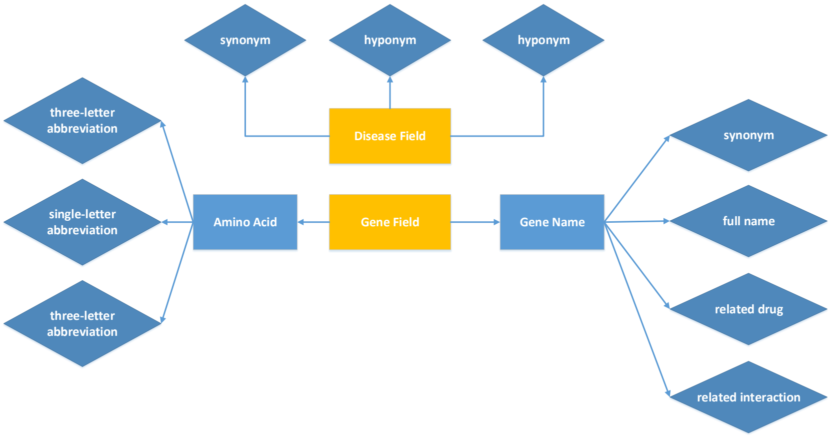


Fig. 4. Framework of Query Expansion

expanded the gene with its full name, synonyms and Entrez_ID, and expanded the variant with its synonyms and PubTator_ID. In particular, for the immunotherapy-related topics, we expanded the biomarker with its synonyms and several keywords.

2. Generate Query: We encoded the disease, gene and variant as three additive disjunctive Boolean query with a clause, noted as Q_d ,Q_g and Q_v. We combined Q_d ,Q_g and Q_v into a Boolean query Q_1 according to Table 5. In addition, we generated a boost query Q_2 subject to precision medicine-relatedness by using words in Table 6.
3. Retrieval via Lucene: We used the combination of Q_1 and Q_2 for the retrieval.

We reserved the top 150,000 results for each topic, noted this list as C_0.

Table 5. Generating Q_1 for biomedical article retrieval task.

Additive disjunctive query	Weight
$Q_d \vee Q_g \vee Q_v$	4.5
$Q_d \vee Q_g$	4.0
$Q_d \wedge Q_g \vee Q_v$	5.0
$Q_d \wedge (Q_g \vee Q_v)$	6.0
$Q_d \wedge Q_g \wedge Q_v$	9.0

However, the TF-IDF features can not reveal the topics of the articles well. We address this problem by post-processing including the following three aspects:

1. Matching Age & Gender: For each result in C_0, we attempted to match the age and gender between the topic and the article, and the award weights for age and gender were 1.2 and 1.5 respectively.

Table 6. Treatment-related Keyword list.

prevent	prophylaxis	prophylactic	prognosis	prognoses
prognostic	outcome	survival	survive	treatment
therapy	therapeutic	personalized	efficacy	

2. Punishment for Cell Line: For each result in C_0, we penalized it if there is a MeSH term including "Cell Line, Tumor". The punishment weight was 0.91.
3. Award for the AACR/ASCO Proceeding articles. The award weight was 2.2.

All weights above were set by grid searching on the qrels of the TREC 2017 PM Track. Finally, we ranked C_0 by the new scores, and reserved the top 20,000 results for each topic.

Clinical Trial Retrieval Similarly, we completed this part by 1) retrieval based on TF-IDF feature via Lucene and 2) post-processing.

We retrieve clinical trials according to the following three steps:

1. Topic Expansion: Mainly the same as the retrieval for biomedical articles. In particular, we used "solid tumor" as a expansion for all diseases.
2. Generate Query: We generated Q_1 and Q_2 according to Table 7 and 6, just the same as the retrieval for biomedical articles.
3. Retrieval via Lucene: We used the combination of Q_1 and Q_2 for the retrieval

We reserved the top 20,000 results for each topic, noted as C_0.

Table 7. Generating Q_1 for clinical trial retrieval task.

Additive disjunctive query	**Weight**
$Q_d \vee Q_g \vee Q_v$	7.5
$Q_d \vee Q_g$	6.0
$Q_d \wedge Q_g \vee Q_v$	3.5
$Q_d \wedge (Q_g \vee Q_v)$	4.0
$Q_d \wedge Q_g \wedge Q_v$	12.0

Our post-processing includes two aspects:

1. Matching Age & Gender: The same as the retrieval for biomedical articles and the weights for age and gender are 7.5 and 5 respectively.
2. Punishment for EXCLUSION_CRITERIA: We retrieved in the field of EXCLUSION_CRITERIA with Q_1 and reserved a ranked list C_e. And then we penalized the score of articles both in C_e and C_0 with punishment weight of 0.01.

All weights above were set by grid searching on the qrels of the TREC 2017 PM Track. Finally, we ranked C_0 by the new scores, and reserved the top 20,000 results for each topic.

2.4 Concept Based Document Re-ranking

Re-ranking of Biomedical Articles For the biomedical article retrieval task, we built our re-ranker based on the concepts extracted via MetaMap in addition to the original text. Along with the evaluation method, we implemented our re-ranker in a heuristic way, respectively scoring each candidate document in three dimensions – disease, gene(and variant) and PM.

Our main idea to improve the retrieval performance is introducing more external information, specifically more features and more annotated data. In addition to the text features used in the text based document retrieval part, we used the concepts extracted via MetaMap from each document as additional features. We treated the qrels of the TREC 2017 PM Track as additional annotated data, and the PM Classifier results can be seen as a supplement to the PM dimension.

Our re-ranker, shown as Figure 5, consists of four parts, namely Not PM Filter and three matchers corresponding to the three dimensions of disease, gene(and variant) and PM.

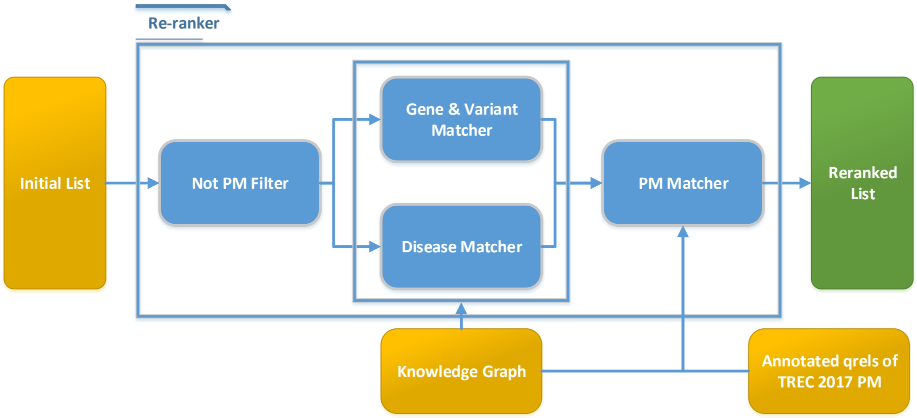


Fig. 5. Architecture of our re-ranker. The yellow blocks indicate external resources or input. The green block indicates output. The blue blocks indicate the re-ranker.

We first filtered out the articles which are definitely not precision medicine-related by a simple Not PM Filter. The Not PM Filter first filtered out articles without any concepts of cancer, and required the remaining articles to contain concepts of gene or treatment-related keywords indicated in Table 6.

We built three matchers, corresponding to three dimensions of disease, gene(and variant) and PM, to score the relevance between the given topic and the retrieved articles. We heuristically set the weights for the matching of disease, gene, variant and precision medicine-relatedness.

For a given topic, we used the query expansion results of disease and gene(and variant) as text features, and the concepts extracted via MetaMap as concept features. In addition, for the gene field in the immunotherapy-related topics, we use the expanded biomarkers and keywords instead of original text as text features, and the corresponding concepts as concept features. These keywords were divided into 1-3 groups, representing different kinds of information of the description in the topic. Taking the gene field of topic 25 as an example, the original text, the biomarkers and the keywords are shown as Table 8.

Table 8. The text features of gene dimension for topic 25. Keywords were divided into two groups.

Original text	**Biomarkers**	**Keywords**
high serum LDH levels	LDH, lactate dehydrogenase	[high], [serum]

For the dimension of disease, we scored the relevance by matching the text feature and concept feature of disease field between the given topic and the retrieved articles. The matching results were divided into three categories, ”match”, ”more general” and ”mismatch”. The ”more general” basically indicated that the disease in the topic is malignant whereas the article uses a more general description that does not distinguish between benign and malignant. We gave a penalty for the ”more general” results, and an extra bonus when the text or concept matches the topic in the article title.

For the dimension of gene(and variant), we adopt different strategies depending on whether the topic is immunotherapy-related.

For the non immunotherapy-related topics, we scored each gene(and variant) given in the topic separately. The matching results were only divided into ”match” and ”mismatch”. The ”match” for variant required that the matching result for the corresponding gene is also ”match”. And similar to the Disease Matcher, we gave an extra bonus when the text or concept matches the topic in the article title. Finally, we selected the highest score among the genes(and variants) as the score of gene dimension of the article.

For the immunotherapy-related topics, we scored each article by matching the biomarkers and keywords in the article title or abstract. The matching rules are:

1. If each keywords group has at least one keyword matched in a single sentence, we treat the sentence ”keyword matched”.
2. If at least one biomarker matches in the sentence, we treat the sentence ”biomarker matched”.
3. If there is one sentence in the article both ”keyword matched” and ”biomarker matched”, we treat the article ”exact matched”.
4. If there is no sentence both ”keyword matched” and ”biomarker matched”, while there are sentences ”keyword matched”, we treat the article ”partial matched”.
5. If there is no sentence ”keyword matched”, we treat the article ”mismatched”.

We finally score the gene dimension of the article according to the matching result.

For the dimension of PM, we built a ruled-based PM Matcher which mainly depends on matching the retrieved articles with manually summarized keywords and information obtained from the TREC-KG, such as cancer-related drugs. We tried to determine whether a retrieved biomedical article focuses on precision medicine in a hybrid manner, combining the data-driven PM Classifier and the rule-based PM Matcher.

We manually summarized several keywords lists, shown as Table 9. In addition, we extracted a related drug list for each topic via the TREC-KG. For each retrieved article, we first divided it into several parts, including title, MeshHeading List, Chemical List, background, objective, conclusion, and unspecified, by the xml structure or simply matching the corresponding part name. The PM Matcher then tried to match each category of keywords with each part of the article, and match the related drug list with the parts of title, MeshHeading List, Chemical List and conclusion of the article. And then the PM Matcher scored the article in the PM dimension based on these matching results. Note that, the PM Matcher would severely penalize the article if the Negative Keywords matched.

We then tried to integrate the PM Classifier results into the score for the PM dimension in two ways. One is giving an additional score to the articles classified as ”PM” in addition to the score given by the PM Matcher. The other is that 1) if the PM Classifier classifies the article as ”PM”, we give a score to the article for the PM dimension, not considering the PM Matcher results any more, 2) if the PM Classifier classifies the article as ”Not PM”, we score the article according to the scoring strategy of the PM Matcher.

Table 9. Keywords lists used in the PM Matcher

Category	**Keywords**
Intervention Keywords	treatment, therapy, therapeutic, prevent, theranostic, prophylaxis, prophylactic, prognosis, prognoses, prognostic
Final Indicator Keywords	complete response, partial response, stable disease, progressive disease, overall survival, progression-free survival, disease control rate, objective response rate
Supplementary Keywords	patient-specific, response, personalized, personalization, individual, survival, survive, outcome, efficacy
Negative Keywords	pathophysiological, database, detect, identification, identify, pathway, diagnostic, diagnosis, cell line

Finally, for each article, if it contains MeshHeading List field with age or gender information, we would filter it out if it doesn’t match the ”demographic” of the topic. And for other cases, we didn’t consider the ”demographic”.

Re-ranking of Clinical Trials The re-ranker for the clinical trial retrieval task was similar to that for the biomedical article retrieval task, we also constructed a scoring scheme for the dimensions of disease, gene(and variant) and PM. The main difference is that the PM classifier is no longer used in the clinical trial retrieval task and we did some field mapping due to the different xml structure of the clinical trials, as shown in Table 10.

Table 10. Mapping of xml fields from clinical trials to biomedical articles.

Clinical Trial fields	**Scientific Abstract fields**
official_title brief_title	ArticleTitle
mesh_term_list keyword_list	MeshHeadings
detailed_description brief_summary	Abstract

We also considered the fields of ”study_type”, ”intervention_type” and ”primary_purpose”. For example, if the field of ”study_type” contains ”intervention”, the field of ”intervention_type” contains ”drug” or the fields of ”genetic” and ”primary_purpose” contain ”treatment”, we would give a bonus score.

3 Results

3.1 Evaluation Results of Biomedical Article Retrieval Task

We submitted 5 runs for the biomedical article retrieval task. The difference between these runs is mainly the ways of using the PM Classifier and the corresponding weight. Table 11 shows a description of the 5 runs, and Table 12 shows our evaluation results, with the best and median results for the biomedical articles retrieval task.

The BASE run achieved the best performance on the metrics of infNDCG and P@10, while the PBPK run achieved the best performance on the metric of R-prec. These results indicate that the usage of the PM Classifier leads to performance loss on the metrics related to precision, but may bring gains on the metrics related to recall.

Figure 6 (a)-(d) demonstrate the impact of different ways of using the PM Classifier on R-prec metric for the biomedical article retrieval task. The strategies behaved differently across 50 topics. It appears that the PM Classifier trained on the qrels is not capable of capturing the implicit and fuzzy standards to determine whether a retrieved biomedical article is precision medicine-related across all the 50 topics.

It is evident that the former three strategies, PBAH, PBH and PBL, have a similar impact on R-prec metric of the retrieval system. In Figure 6 (a)-(c), there are more dark grey columns representing the decline of R-prec metric

Table 11. Our submitted runs for the biomedical article retrieval task.

BASE	Baseline, not using the PM Classifier
PBAH	Using the PM Classifier results with an additional high weight for all the re-ranked articles
PBH	Using the PM Classifier results with an additional high weight, but not affecting the top10 articles of the BASE
PBL	Using the PM Classifier results with an additional low weight, but not affecting the top10 articles of the BASE
PBPK	Incorporating the PM Classifier results with the PM Matcher, giving a boost score if the PM Classifier classifies the article as ”PM” or according to the scoring strategy of the PM Matcher

Table 12. Evaluation results from our systems with the best and median results for the biomedical article retrieval task. Bold is our best.

	infNDCG	P@10	R-prec
Best	0.7135	0.8660	0.4461
Median	0.4291	0.5460	0.2672
MSIIP_BASE	**0.5621**	**0.6704**	0.3216
MSIIP_PBAH	0.5297	0.6540	0.2979
MSIIP_PBH	0.5459	0.6680	0.2999
MSIIP_PBL	0.5462	0.6680	0.3008
MSIIP_PBPK	0.5577	0.6360	**0.3257**

than green columns representing the improvement of R-prec metric, which indicates that these strategies deteriorate R-prec metric on average. However, the PBPK strategy, incorporating the PM Classifier results with the rule based PM Matcher, achieved an opposite result that the number of green columns is greater than that of dark grey columns, which indicates that the PBPK strategy can boost performance on R-prec metric.

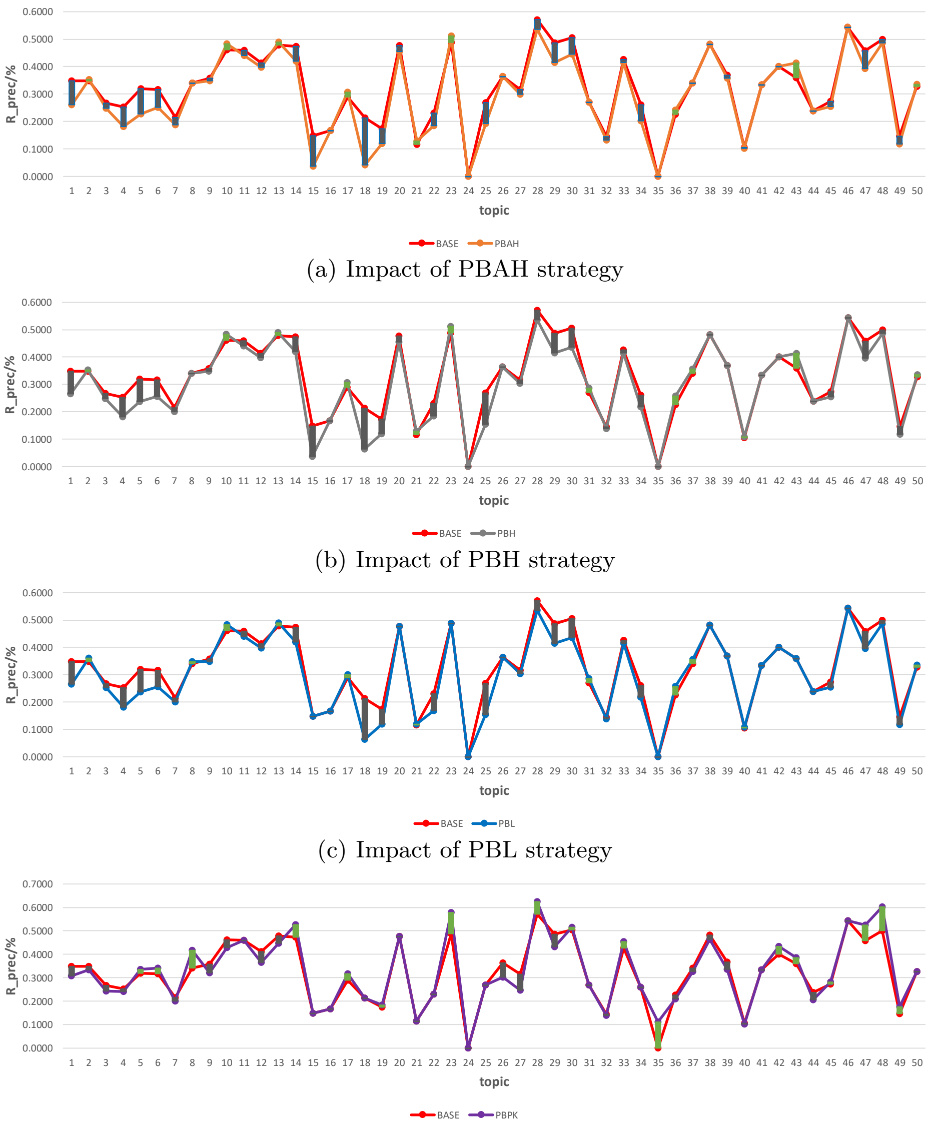
3.2 Evaluation Results of Clinical Trial Retrieval Task

We submitted 5 runs for the clinical trials retrieval task. Table 13 compares the different methods we used, and Table 14 shows our evaluation results, with the best and median results for the clinical trials retrieval task.

Our best system is MSIIP_TRIAL1, indicating that the query expansion results and the modified matching methods for the immunotherapy-related topics were helpful. However, for the bonus score in the fields of ”study_type”, ”intervention_type” and ”primary_purpose”, we didn’t find any contribution to the evaluation results.

4 Conclusion

In this paper, we described our retrieval system for the TREC 2018 Precision Medicine Track. The track consists of two tasks, the biomedical article retrieval



(a) Impact of PBAH strategy

(b) Impact of PBH strategy

(c) Impact of PBL strategy

(d) Impact of PBPK strategy

Fig. 6. Impact of the PM Classifier on R-prec metric for the biomedical article retrieval task. The dark grey columns indicate performance loss and the green columns indicate performance improvement.

Table 13. Our submitted runs for the clinical trial retrieval task.

MSIIP_TRIAL1	Mainly the same as MSIIP_TRIAL3, but not considering the fields of ”study_type”, ”intervention_type” and ”primary_purpose”
MSIIP_TRIAL2	Mainly the same as MSIIP_TRIAL5, but not using the query expansion results, not considering ”study_type”, ”intervention_type” and ”primary_purpose” fields.
MSIIP_TRIAL3	Baseline, containing all our methods
MSIIP_TRIAL4	Mainly the same as MSIIP_TRIAL5, but not using the query expansion results.
MSIIP_TRIAL5	Mainly the same as MSIIP_TRIAL3, but for the immunotherapy-related topics, employing the same matching strategy as other topics

Table 14. Evaluation results from our runs with the best and median results for the clinical trials retrieval task. Bold is our best.

	infNDCG	P@10	R-prec
Best	0.7559	0.7620	0.5767
Median	0.4297	0.4680	0.3268
MSIIP_TRIAL1	**0.5504**	**0.6260**	**0.4294**
MSIIP_TRIAL2	0.5304	0.5700	0.4066
MSIIP_TRIAL3	0.5428	0.6160	0.4272
MSIIP_TRIAL4	0.5317	0.5980	0.4076
MSIIP_TRIAL5	0.5500	0.6220	0.4294

task and the clinical trial retrieval task. We submitted five runs for each task. In the light of the intuition that introducing external information would bring performance improvement, we constructed the TREC-KG to integrate knowledge of interest from several external knowledge sources, and trained the PM Classifier for precision medicine-relatedness using the qrels of the TREC 2017 PM Track as training corpus. Our system architecture consists of a text-based retrieval part and a concept-based re-ranking part, incorporating the TREC-KG and the PM Classifier. In particular, to score the precision medicine-relatedness of a biomedical article, we employed a hybrid method, combining our proposed data-driven PM Classifier with the rule-based PM Matcher.

Our experiments verify the effectiveness of our system on both the two tasks, and that the usage of the PM Classifier would boost R-prec metric on the biomedical article retrieval task.

References

1. Aronson, A.R.: Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In: Proceedings of the AMIA Symposium. p. 17 (2001)
2. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research **32**(suppl_1), D267–D270 (2004)
3. Chakravarty, D., Gao, J., Phillips, S., Kundra, R., Zhang, H., Wang, J., Rudolph, J.E., Yaeger, R., Soumerai, T., Nissan, M.H., et al.: Oncokb: a precision oncology knowledge base. JCO precision oncology **1**, 1–16 (2017)

4. Forbes, S., Beare, D., Bindal, N., Bamford, S., Ward, S., Cole, C., Jia, M., Kok, C., Boutselakis, H., De, T., et al.: Cosmic: high-resolution cancer genetics using the catalogue of somatic mutations in cancer. Current protocols in human genetics **91**(1), 10–11 (2016)
5. Griffith, M., Griffith, O.L., Coffman, A.C., Weible, J.V., McMichael, J.F., Spies, N.C., Koval, J., Das, I., Callaway, M.B., Eldred, J.M., et al.: Dgidb: mining the druggable genome. Nature methods **10**(12), 1209 (2013)
6. Griffith, M., Spies, N.C., Krysiak, K., McMichael, J.F., Coffman, A.C., Danos, A.M., Ainscough, B.J., Ramirez, C.A., Rieke, D.T., Kujan, L., et al.: Civic is a community knowledgebase for expert crowdsourcing the clinical interpretation of variants in cancer. Nature genetics **49**(2), 170 (2017)
7. Jameson, J.L., Longo, D.L.: Precision medicine—personalized, problematic, and promising. Obstetrical & Gynecological Survey **70**(10), 612–614 (2015)
8. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: AAAI. vol. 333, pp. 2267–2273 (2015)
9. Tamborero, D., Rubio-Perez, C., Deu-Pons, J., Schroeder, M.P., Vivancos, A., Rovira, A., Tusquets, I., Albanell, J., Rodon, J., Tabernero, J., et al.: Cancer genome interpreter annotates the biological and clinical relevance of tumor alterations. Genome medicine **10**(1), 25 (2018)
10. Wei, C.H., Kao, H.Y., Lu, Z.: Pubtator: a web-based text mining tool for assisting biocuration. Nucleic acids research **41**(W1), W518–W522 (2013)
11. Zhang, Y., Wallace, B.: A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv preprint arXiv: 1510.03820 (2015)