

UNTIIA Lab at TREC 2021 - Clinical Trial*

Huyen Nguyen^{a,b}, Haihua Chen^{a,b}, Bhanu Prasad^{a,b}, Huanhuan Zhao^c,
Junhua Ding^{a,b}, Jiangping Chen^{b,*}, Ana Cleveland^b

^a*Intelligent Information Access Laboratory, University of North Texas*

^b*Department of Information Science, University of North Texas*

^c*Department of Bioinformatics, University of Texas at El Paso*

Abstract

This paper reports our participation in 2021 TREC Clinical Trial (CT) track based on the Elasticsearch information retrieval platform. We studied different query extraction and query expansion methods with both manual and automatic strategies for query construction. Our experiments on the 2016 Clinical Decision Support collection proved the effectiveness of the knowledge base mapping method for both query extraction and expansion. We proposed a novel query construction strategy to balance precision and accuracy: we retrieve clinical trials documents with a complete list of query terms first and then decrease the number of query terms used for searching additional documents to improve recall. We also investigated two transformer-based models for reranking: unsupervised and supervised learning. Pairs of query and candidate documents were encoded with the sentence-BERT in the unsupervised reranking model, and then their semantic similarity was compared using a Cross-encoder model. We also took advantage of BERT transformers for the supervised reranking model by finetuning the model on the ground truth of the 2016 Clinical Decision Support collection and then feeding it into the TFR-BERT model for reranking. Our experiment indicates that the unsupervised transformer reranking model outperformed the supervised learning model and achieved the highest performance

*This project was supported in part by the National Science Foundation under Grant 1852249, and in part by the National Security Agency under Grant H98230-20-1-0417.

*This is to indicate the corresponding author.

Email address: jiangping.chen@unt.edu (Jiangping Chen)

URL: <https://ia.ci.unt.edu/home> (Jiangping Chen)

among teams on a number of topics.

Keywords: Clinical Trial, Information Retrieval, ElasticSearch, BM25, BERT

1. Introduction

The TREC Biomedical Tracks, aiming to improve the speed at which treatments are developed and disseminated into clinical practice¹, has been running for 19 years at the Text REtrieval Conference. From 2003-2007, the TREC Genomics track focused on genomics researchers seeking relevant biomedical literature; from 2011-2012, the TREC Medical Records track focused on retrieving cohorts of patients from electronic health records (EHRs); from 2014-2016, the Clinical Decision Support track focused on clinicians looking for evidence-based full-text literature to support diagnosis, treatment, and testing decisions; from 2017-2020, the Precision Medicine track focused on oncologists looking for evidence-based treatment literature and clinical trials. This year, the newly introduced Clinical Trials (CT) track focuses on matching patients to relevant clinical trials.

Compared to the previous tracks, the CT track is more challenging. The topic text, which is a patient case description that simulates an admission statement in an EHR, is much longer (5-10 sentences) with non-standardized English texts. In other words, the topics provide more information but contain noise that may significantly reduce the accuracy of the retrieval results. There are more topics (75 topics) than in previous years, which will need more computational resources for getting the results.

Our team participated in TREC medical Tracks in the past [1, 2]. One of the shared purposes of our participation has been to train graduates in information retrieval and retrieval systems. We have seen the growing demands from industries on computational professionals who are able to develop or implement information retrieval systems. Other purposes of our participation in this year's

¹<http://www.trec-cds.org/>

TREC include: (1) to explore and implement a new information retrieval (IR) platform for the Lab; (2) to investigate effective query extraction and construction to handle lengthy and noisy topics. For this purpose, we consider both manual and automatic topic processing methods and compare their retrieval performance; and (3) to explore the performance of various reranking models, especially deep learning-based reranking models for information retrieval.

We chose to build a high-performance IR system using ElasticSearch, an efficient search platform supporting large-scale distributed search and a user-friendly interface for indexing and retrieval. ElasticSearch has been applied by other TREC participants and is becoming more and more popular among the TREC community. As for the topics, we investigated different automatic query extraction and query expansion methods to handle lengthy topics: knowledge-base mapping, keyword extraction, and named entity extractions were used for automatic query extraction, while knowledge-base mapping and pseudo-relevant feedback were for query expansion. We conducted a group of experiments to select the best parameters, such as text encoding models, number of expanded terms, type of entities. We proposed an iterative search strategy that could balance the precision and recall for the ranking. We implemented two BERT-based algorithms for reranking: one is an unsupervised learning model based on sentence-BERT [3], the other is a supervised learning model based on TFR-BERT [4]. The experiments show that the knowledge-base mapping method for query extraction and expansion generated the most effective queries. Our proposed search strategy produced a more inclusive candidate document list for reranking. Our experiments also show that unsupervised BERT Cross-encoder performs better than the TFR-BERT model for the reranking.

The rest of the paper is organized as follows: Section 2 presents the experimental design and the methodologies implemented for the 2021 TREC CT track. Section 3 describes the runs submitted. Section 4 presents the evaluation results and discussion. The paper concludes with a summary and an outlook of future research.

2. Methodology

Like our previous TREC participation [1] [2], the whole process of matching potential patients (queries) to clinical trials (the documents) includes four stages in this study: Collection Indexing, Query Processing, Document Retrieval, and Reranking. However, we applied different strategies from our previous participation at each stage. The four stages are depicted in Figure 1 and will be explained below.

2.1. Collection Indexing

The TREC 2021 clinical trial track document collection comprises 3.7 million clinical trial documents in XML format. Besides, we made use of the TREC 2016 clinical decision support track test collection to develop our retrieval system. The two collections were separately indexed using ElasticSearch². As for the 2021 test document collection, all fields are indexed. However, we concatenated all content fields together for our retrieval convenience, including brief title, official title, brief summary text block, detailed description text block, primary outcome description, and secondary outcome description. We hope that the recall could be maximized by including all content fields. A similar indexing process was applied to the 2016 document collection.

2.2. Topic processing

TREC 2021 clinical trial track consists of 75 topics in XML format. Their format is very different from the topics of previous precision medicine tracks. TREC precision medicine 2019 and 2020 are structured into predetermined information types, such as disease, genetic variants, demographic information (in topics 2019), and disease, genetic variants, the proposed treatment in topics 2020. In contrast, topics of this year are in the format of admission notes which are long and narrative; they also contain many acronyms and abbreviations.

²<https://www.elastic.co/>

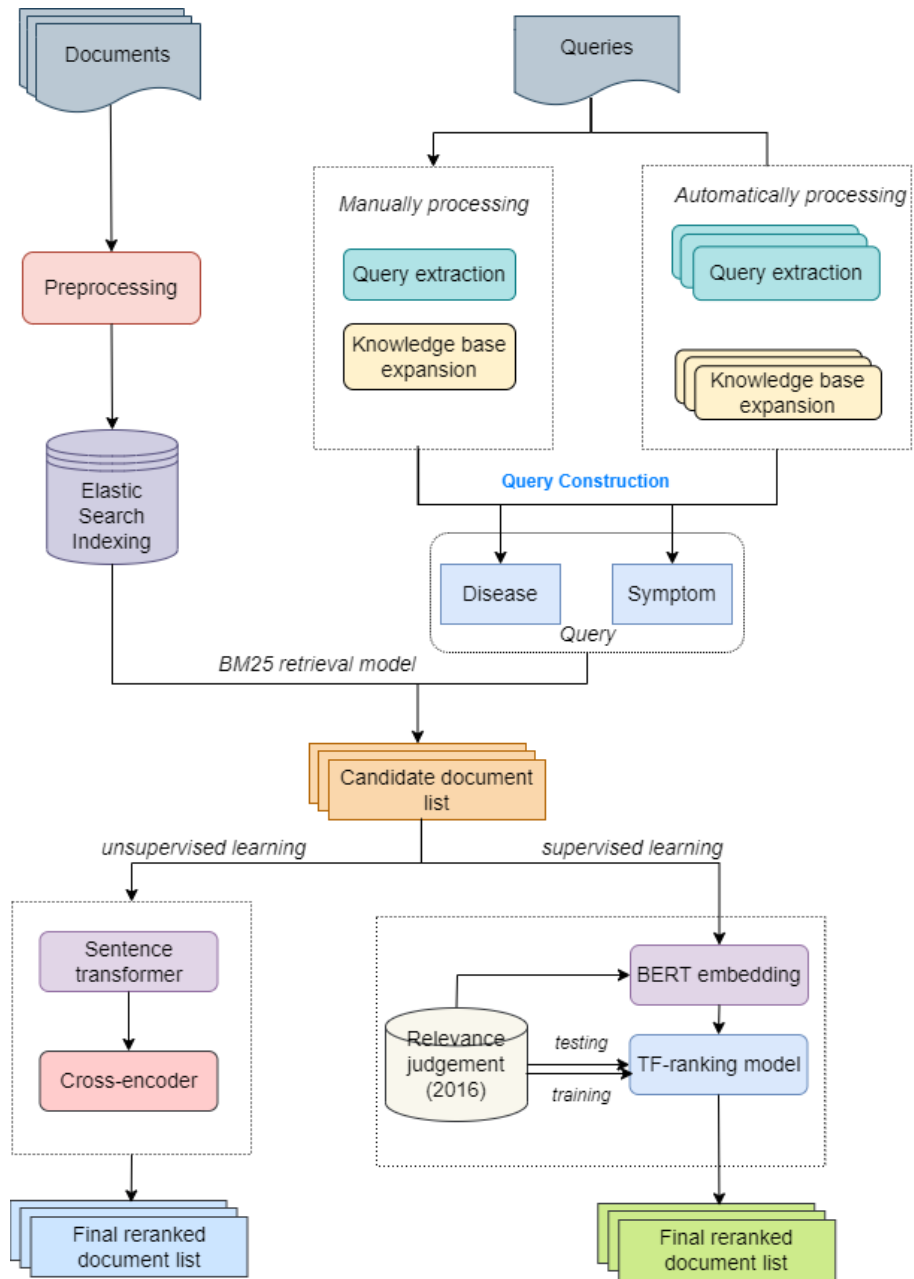


Figure 1: General Architecture of UNT IIA IR System

Therefore, query processing is necessary to extract the important information from topics for retrieval.

The topics include a good deal of demographic information such as age, gender, and physical exams. They may also contain patients' past medical history, family history, and labs and diagnostics studies. By studying the topics, we select terms representing disease and symptom as keywords for retrieval. These terms are mentioned in the physical exams and diagnostics studies. Generally, our topic processing is for the following purposes: (1) to extract the most important clinical trial terms for searching (i.e., disease and symptom information is predetermined), (2) to expand the selected terms with their synonyms and broader terms to increase recall. Figure 2 presents our topic processing strategies and methods.

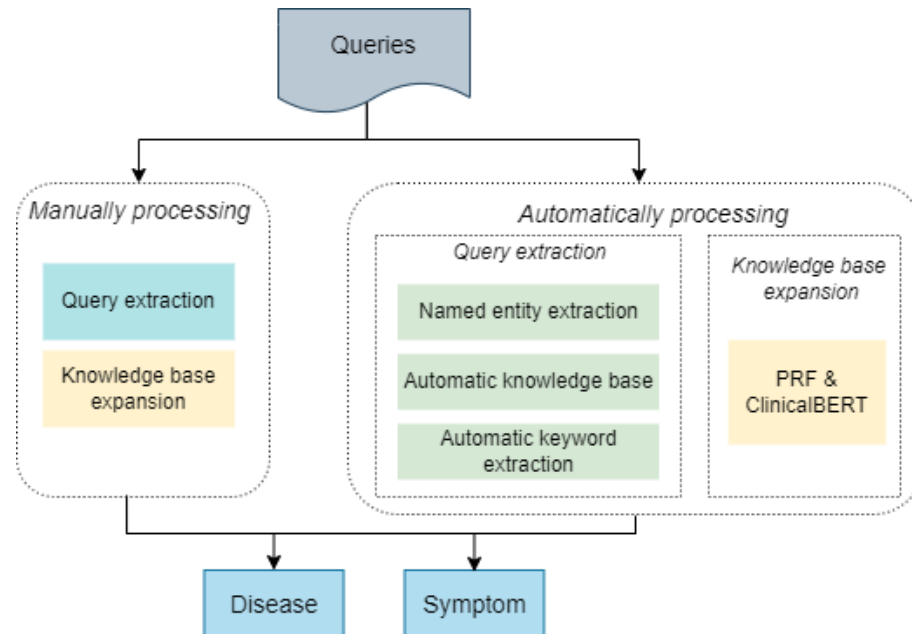


Figure 2: Query processing methods

2.2.1. Manual topic processing

Lexigram³ was used to extract candidate query lists that include disease and symptom. Lexigram is an online tool built on a knowledge graph, covering clinical entities such as drugs, diseases, and symptoms. Then, by referring to other available resources such as UMLS, Wikipedia, etc., and using our knowledge, we refined the candidate lists and extracted more terms from topics.

2.2.2. Automatic topic processing

Automatic query processing includes two steps: query extraction and query expansion.

Query extraction. Three different automatic information extraction methods, including automatic knowledge base (KB) extraction, keyword extraction, and named entity recognition (NER), were applied to compare the performance. The automatic KB extraction method identifies terms contained in knowledge bases such as UMLS and DrugBank. Inspired by [5], [6], [7] that achieved good performance in TREC 2019, 2016, and 2015, respectively, we used Metamap [8] for extracting terms related to disease and symptom. We specified two semantic types: Sign or Symptom, and Disease or Syndrome.

As for automatic keyword extraction, we used Yake! [9], an unsupervised model based on text statistic features to select the most important words from the text. We experimented with different numbers of terms (k= 15, 14, 13, 12, 11, 10) and found that k=12 generated the highest-quality query lists.

BERN [10], a named entity recognition (NER) model, was implemented to identify the initial candidate entities. BERN is based on BioBERT [11], a pre-trained biomedical language representation model for biomedical text mining, to initially identify biomedical entities. The BERN model then removed the overlapping entities between entity types by applying multiple pre-defined rules. We used BERN to extract disease and symptom entities from the topics.

³<https://www.lexigram.io>

We conducted a series of experiments with these methods on the 2016 test collection and, we found that the automatic KB method achieved the highest recall. Therefore, we selected the automatic KB method for implementation on 2021 Clinical Trial topics.

Query expansion. To enrich our queries with their variants, we performed two query expansions: Metamap automatic KB, and Pseudo Relevance Feedback (PRF). When extracting UMLS terms, Metamap also returns their synonyms and broad terms if they exist in UMLS. Queries extracted by this method are used for the first run. We also implemented PRF on top of ClinicalBERT [12] for query expansion. As shown in figure 3, using queries extracted manually (no query expansion), we first retrieved a set of documents and then selected titles of the top 20 documents for implementing PRF. All the three automatic information extraction methods mentioned previously were tested using the 2016 test collection for finding the best methods. We extracted all possible terms using automatic KB and NER methods, and retained a maximum of 10 terms from each document title by using automatic keyword extraction. Extracted terms from titles and queries were encoded with ClinicalBERT word embeddings [13]. ClinicalBERT was trained on top of the BERT model but further finetuned on MIMIC clinical notes; it yields a better performance on some Clinical NLP tasks [13]. Finally, we measured the similarity of extracted terms and queries using cosine similarity, in which we calculated the distance between document and query vector representations. We selected the top two terms from each document title with the highest similarity scores as expanded queries. Expanded queries obtained with this method were used for the fourth run.

2.3. Document Retrieval

Accumulative Retrieval. We developed an optimal retrieval method, called accumulative retrieval, to balance precision and recall. We assumed that the more topic terms found in the document, the more relevant is the document to the query. Our ranking starts with a complete list of query terms to guarantee

Algorithm 1 The Accumulative Retrieval.

- 1: **Input:** term list (m terms) of the initial query T_1 , term list (n terms) of the query expansion T_2 .
 - 2: **Step 1:** retrieve documents using T_1 with Boolean “AND” strategy
 - 3: **for** iteration = $m, m - 1, \dots, 2$ **do**
 - 4: number of terms = m , query = t_1 AND t_2 AND $\dots t_m$
 - 5: Retrieve the documents with the query
 - 6: Append the retrieved documents to the document list
 - 7: **end for**
 - 8: **Step 2:** retrieve documents using $T_1 + T_2$ with Boolean “AND” strategy
 - 9: **for** iteration = $m + n, m + n - 1, \dots, 2$ **do**
 - 10: number of terms = $m + n$, query = t_1 AND t_2 AND $\dots t_{(m+n)}$
 - 11: Retrieve the documents with the query
 - 12: Append the retrieved documents to the document list
 - 13: **end for**
 - 14: **Step 3:** retrieve documents using T_1 with Boolean “OR” strategy
 - 15: **for** iteration = $m, m - 1, \dots, 2$ **do**
 - 16: number of terms = m , query = t_1 OR t_2 OR $\dots t_m$
 - 17: Retrieve the documents with the query
 - 18: Append the retrieved documents to the document list
 - 19: **end for**
 - 20: **Step 4:** retrieve documents using $T_1 + T_2$ with Boolean “OR” strategy
 - 21: **for** iteration = $m + n, m + n - 1, \dots, 2$ **do**
 - 22: number of terms = $m + n$, query = t_1 OR t_2 OR $\dots t_{(m+n)}$
 - 23: Retrieve the documents with the query
 - 24: Append the retrieved documents to the document list
 - 25: **end for**
 - 26: Remove duplication of the retrieved document list
 - 27: **Output:** top 1000 document for reranking.
-

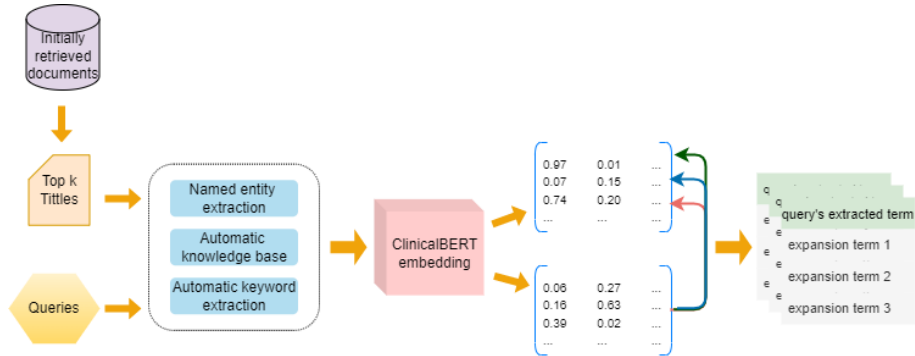


Figure 3: Pseudo Relevance Feedback (PRF) with ClinicalBERT word embeddings

the best precision. Then, gradually decreasing the number of query terms used for searching. Then, newly retrieved documents were added to improve the recall. The process ends until we hit 1000 retrieved documents. Algorithm 1 presents our retrieval method.

Retrieval Model. In each step in Algorithm 1, Boolean model was combined with our accumulative retrieval method to retrieve documents which were then ranked with the BM25 model. As we observed, the BM25 model has been used as a baseline by most TREC teams.

2.4. Reranking

Our study leveraged the transformer – a deep learning model– to develop two reranking models: unsupervised learning and supervised learning. The transformer adopts the attention mechanism for allowing to input sequential data in parallel. Recently, the transformer models have proven their incomparable performance in many NLP tasks. In the unsupervised learning model, we used *cross-encoder/ms-marco-MiniLM-L-6-v2*, a pre-trained sentence embedding model built on top of BERT transformer [3] to encode queries and documents, and further trained on MS MARCO, a large-scale information retrieval dataset. Each pair of the query and the document was encoded into 384-dimension dense vector space and then simultaneously passed to a transformer network, called Cross-Encoder. The model returned a score between 0

and 1, indicating the similarity between the query and document. We used this reranking model for the fourth run to rerank the initially retrieved documents returned by the second run.

We used the relevance judgment of 2016 Clinical Decision Support to train our supervised reranking model. Like topics of this year, topics of the 2016 collection are curated EHR admission notes; therefore, the model trained on that dataset is likely applicable to this year’s task. We implemented TFR-BERT [4] for this reranking model. BERT embedding model was used to represent query-document pairs in this model. Query-document pairs were concatenated to input into the BERT model as sentence pairs with the format: *[CLS] query text [SEP] document text [SEP]*. The hidden units of the CLS token were fed into the TF-ranking model to finetuning using pointwise, pairwise, and listwise losses [4].

3. Experiment and Results

3.1. Run Description

We submitted results from 5 experiments or runs. Our final runs are summarized as follows:

– UNTIARUN1 (Baseline): Automatic knowledge-base extraction on topics and expansion with BM25. We used Metamap to extract the terms from the topics and obtained their synonyms and broader terms from UMLS ⁴. Finally, the retrieved documents were ranked with BM25 scores.

– UNTIARUN2: Manual knowledge-base extraction and expansion with BM25. We extracted and expanded queries based on external knowledge bases such as UMLS, Lexigram for searching. Similarly, BM25 was used to rank our retrieved documents initially.

– UNTIARUN3: Combining manual query extraction and ClinicalBERT-PRF query expansion. Queries extracted manually with knowledge bases were

⁴<https://uts.nlm.nih.gov/uts/umls/home>

used as input of our ClinicalBERT-PRF query expansion model. Finally, we retrieved documents for this run using manually-extracted terms and PRF-expanded terms. Again, BM25 was used to rank our retrieved documents.

– UNTIARUN4: Unsupervised learning reranking model. Our BERT Cross-encoder reranking model reranked documents retrieved from UNTIARUN2.

– UNTIARUN5: Supervised learning reranking model. Our TFR-BERT reranking was trained on the relevance judgment of the 2016 Clinical Decision Support track. Similar to UNTIARUN4, we used documents that were achieved from UNTIARUN2 for the reranking task in this run.

3.2. Result

Our results are summarized in this section. Table 1 and Figure 4 present results of our five submissions in terms of reciprocal rank, P@10, and NDCG@10, in comparison to the averages of the TREC medians. The result indicates that UNTIARUN4 outperforms other runs, and other runs perform quite poorly. Note that we applied an unsupervised reranking model for UNTIARUN4.

However, compared to the TREC official median (manual runs) in each topic, our system performs well in a number of topics. Figure 5 and figure 6 show our performance on NDCG@10 compared to the TREC official median and best, respectively. For example, our UNTIARUN4 performs better than the TREC official median of NDCG@10, reciprocal rank, and P@10 in a number of topics (14 topics, 28 topics, and 22 topics, respectively) (figure 5). Our results also indicate that we achieve the highest NDCG@10, reciprocal rank, and P@10 scores for certain topics; for example, our UNTIARUN4 yields the highest reciprocal rank score for 19 topics, the highest NDCG@10 for 4 topics, and the highest P@10 for 3 topics (figure 6).

4. Discussion

Our outperformed RUN4 indicates that our unsupervised learning reranking model is more effective than the supervised one. It significantly improves the initial retrieved document rank by 0.26, 0.16, and 0.27 on Reciprocal rank, P@10,

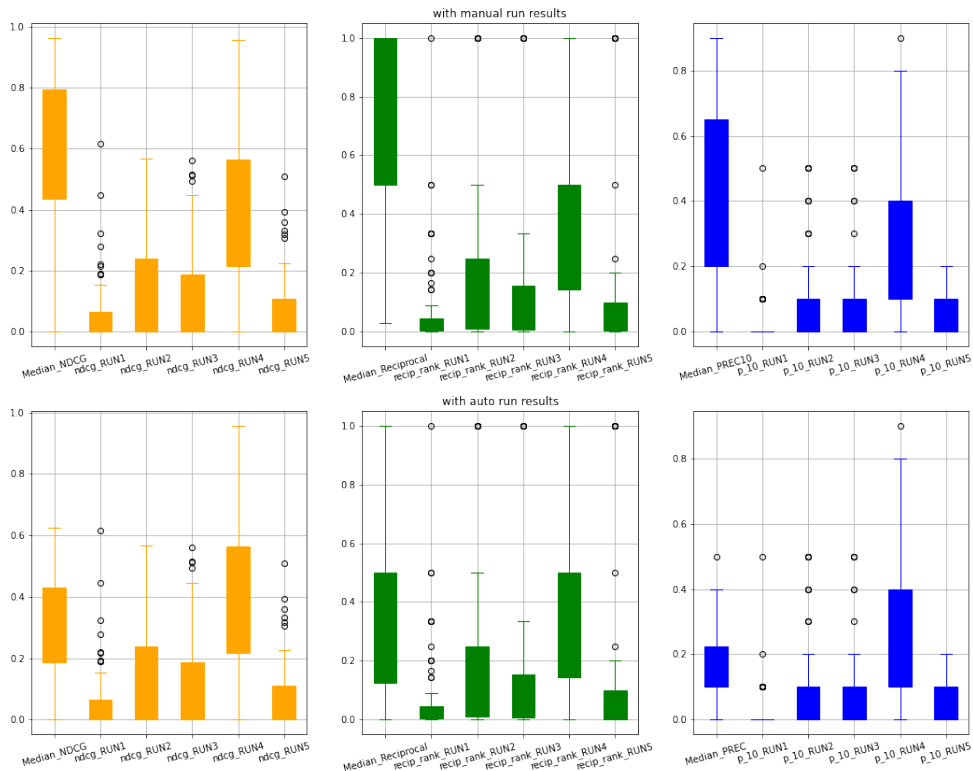


Figure 4: All submissions compared to the official median NDCG@10, reciprocal rank, and P@10 of manual runs (upper), and automatic runs (lower).

and NDCG@10, respectively. In contrast, the supervised reranking model failed in this reranking task and even worsened the initial results (UNTIARUN2). The model was trained on the relevance judgments of the 2016 Clinical Decision Support track. Even though the topics of this track are in a similar format to topics of this year, documents are entirely different, i.e., scientific articles rather than clinical trials. The distinctions between training data and testing data could be one possible reason for the declined performance. In the future, we

Table 1: Our runs' results in comparison to the TREC average of the median

Submission	Reciprocal	P@10	NDCG@10
UNTHIARUN1	0.0736	0.0253	0.0551
UNTHIARUN2	0.1690	0.0893	0.1389
UNTHIARUN3	0.1399	0.0667	0.1060
UNTHIARUN4	0.4243	0.2520	0.4043
UNTHIARUN5	0.1176	0.0360	0.0739
AVG_Median (manual)	0.7213	0.4573	0.6212
AVG_Median (auto)	0.2942	0.1613	0.3040

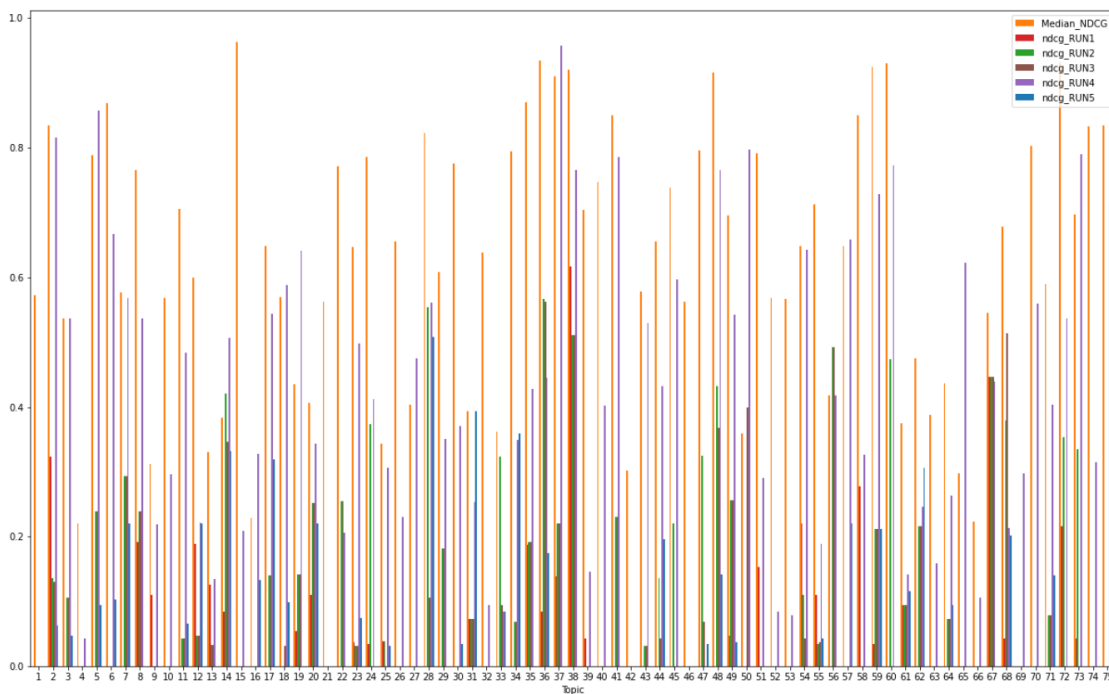


Figure 5: NDCG@10 of all the submitted runs compared to the official median NDCG@10 of manual runs.

will further investigate our supervised ranking model by training it on the clinical trial task of the TREC 2019 precision medicine track since the documents

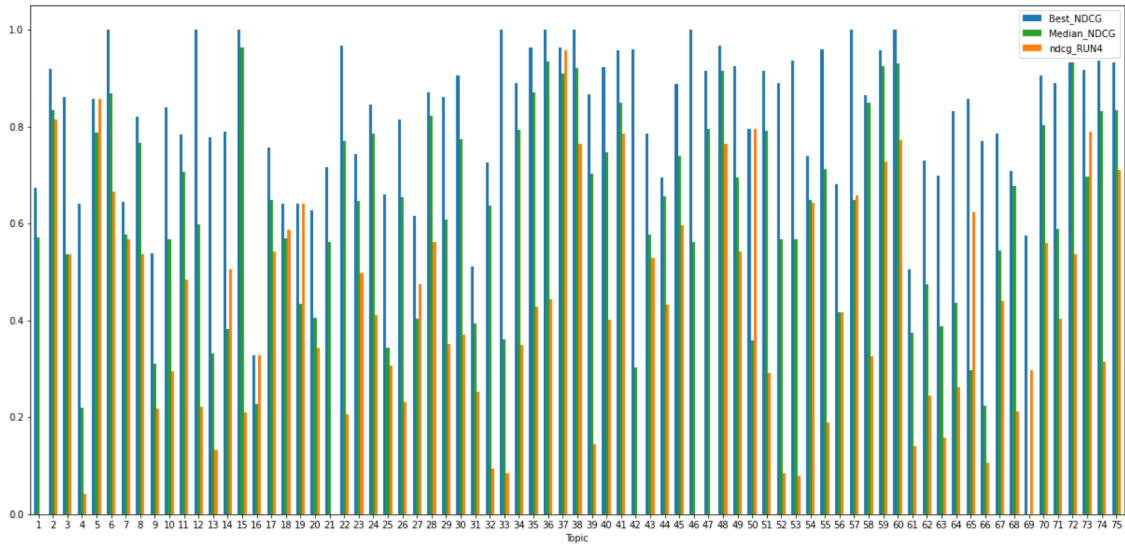


Figure 6: NDCG@10 of our best run compared to the official median, and best NDCG@10 of the manual runs.

are comparable with this year’s documents.

The results also prove that our initial ranking model (i.e., BM25) is very limited. We used this ranking model for three first runs, which performed poorly, even for the UNTIARUN2, in which queries were manually processed. However, the performance improved substantially by using the top 2000 documents from the UNTIARUN2 for reranking with the UNTIARUN4, meaning that the BM25 model failed to include and/or rank the most relevant documents in the top 1000 retrieved documents.

We did not achieve the performance that we expected. The IR system failed in topics 1, 9, 12, 21, 42, 46, and 47 with very low recalls. We would like to investigate the causes by observing processed queries of these topics in the future.

5. Conclusion

This paper presents the methods and results of our participation in the TREC 2021 Clinical Trials track. First, we successfully implemented a new platform for our IR system; it proves suitable for use next year. Further, the results indicate that one of our reranking models performed effectively. This run exceeded the official TREC medians (automatic runs) and contributed the highest performance in many topics. However, the results showed that our topic processing methods are not effective as we expected. We will need to conduct more analysis to understand issues during the experiments. In the future, we would like to investigate other topic processing methods that can handle long topics better. We also want to improve our supervised reranking model.

Acknowledgements

The authors would like to thank Payam Kelich, Ph.D. student at the University of Texas at El Paso, for assisting us in configuring the ElasticSearch platform and the experiment environment. The authors would also like to thank Dustin Ho, Jesus Guillen, and Zachary Tian for analyzing the topics.

References

- [1] L. K. Viswavarapu, J. Chen, A. D. Cleveland, J. Philbrick, Unt medical information retrieval at trec 2016., in: TREC, 2016.
- [2] L. K. Viswavarapu, J. Chen, A. D. Cleveland, H. Chen, Unt precision medicine information retrieval at trec 2017., in: TREC, 2017.
- [3] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, arXiv preprint arXiv:1908.10084.
- [4] S. Han, X. Wang, M. Bendersky, M. Najork, Learning-to-rank with bert in tf-ranking, arXiv preprint arXiv:2004.08476.

- [5] G. M. Di Nunzio, S. Marchesin, M. Agosti, Exploring how to combine query reformulations for precision medicine., in: TREC, 2019.
- [6] H. Gurulingappa, L. Toldo, C. Schepers, A. Bauer, G. Megaro, Semi-supervised information retrieval system for clinical decision support., in: TREC, 2016.
- [7] J. Palotti, A. Hanbury, Tuw@ trec clinical decision support track 2015, Tech. rep., Vienna University of Technology Vienna Austria (2015).
- [8] A. R. Aronson, F.-M. Lang, An overview of metamap: historical perspective and recent advances, *Journal of the American Medical Informatics Association* 17 (3) (2010) 229–236.
- [9] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, A. Jatowt, Yake! keyword extraction from single documents using multiple local features, *Information Sciences* 509 (2020) 257–289.
- [10] D. Kim, J. Lee, C. H. So, H. Jeon, M. Jeong, Y. Choi, W. Yoon, M. Sung, J. Kang, A neural named entity recognition and multi-type normalization tool for biomedical text mining, *IEEE Access* 7 (2019) 73729–73740.
- [11] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics* 36 (4) (2020) 1234–1240.
- [12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805.
- [13] E. Alsentzer, J. R. Murphy, W. Boag, W.-H. Weng, D. Jin, T. Naumann, M. McDermott, Publicly available clinical bert embeddings, arXiv preprint arXiv:1904.03323.