

## Round 1 results — Run UB\_NLP\_RUN\_1 submitted from UB\_NLP

### Run Description

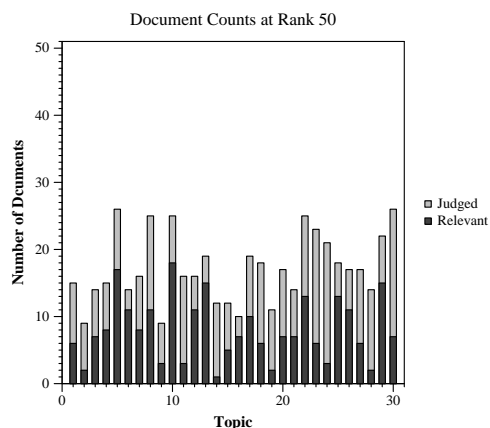
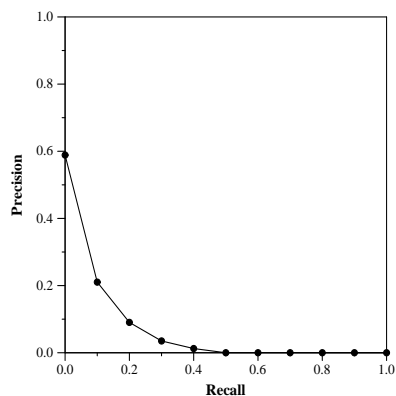
We have implemented a composite traditional IR + state of the art NLP approach to develop our system. Below are the details

1. We have trained a NER on the NCBI disease corpus using bioBERT, and used the NER to capture disease names from document abstracts, based on which we created out initial "disease-document" index.
2. We generated extracts from the document body and indexed them using BM25 model. This is our second level index.
3. Finally, we perform question answering using BERT based on the extracts that's returned by step 2
4. Query preprocessing: We generate 2 kinds of queries: i) tokenized bag of words query ii) contextual query tokenized bag of words query: a) We tokenize a query and remove stopwords. If there is a mention of disease in the query, then we add the alternate names of the disease to he tokens. Example: query->"What is the incubation period of COVID-19?" tokens-> "incubation", "period", "COVID-19" enriched tokens-> "incubation", "period", "COVID-19", "ncovid", "ncov", "novel", "coronavirus", "19-covid" contextual query: a) We detect disease name(s) in the query and replicate the same query n times based on the number of alternate names the disease has in our data. Example query->"What is the incubation period of COVID-19?" replicated queries->"What is the incubation period of COVID-19?", "What is the incubation period of ncovid?", "What is the incubation period of novel coronavirus?", "What is the incubation period of 19-covid?", "What is the incubation period of ncov?"
5. Retrieving results: i)We maintain separate indexes for separate kinds of disease. We have different index for MERS, SARS and NCOVID. Based on the disease that is detected in the query, we filter out the non relevant documents. ii)We query the BM25 model for the disease using the query tokens(bag of words) and retrieve the top 4000 extracts iii)Based on the BM25 ranking of the extracts, we perform BERT question answering on the top 500 extracts. Note that extracts != document. A document is broken down into extracts of 256 words. iv) The BERT question answering outputs logits for each position of the sequence. Higher the logit more the probability of the extract containing the answer to the query. We rank extracts by the logit value of the starting word of the answer. A document score is generated by taking the max of all the extracts of the document for which a logit value is present. We rank the documents in decreasing score. In order to make the document count 1000, we add the remaining documents(until the count is 1000) returned by BM25, after scaling their scores with respect to the min logit score.

Summary Statistics	
Run ID	UB_NLP_RUN_1
Topic type	automatic
Contributed to judgment sets?	yes

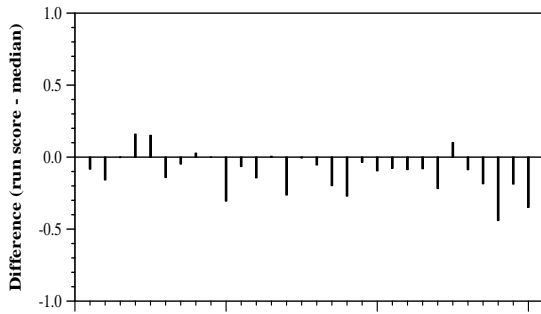
Overall measures	
Number of topics	30
Total number retrieved	30000
Total relevant	2352
Total relevant retrieved	744
MAP	0.0574
Mean Bpref	0.2214
Mean NDCG@10	0.2453

Document Level Averages	
	Precision
At 5 docs	0.3800
At 10 docs	0.2733
At 15 docs	0.2556
At 20 docs	0.2417
At 30 docs	0.2022
R-Precision	
Exact	0.1290

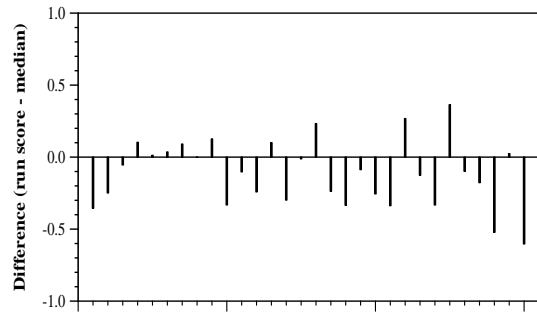


Round 1 results — Run UB\_NLP\_RUN\_1 submitted from UB\_NLP

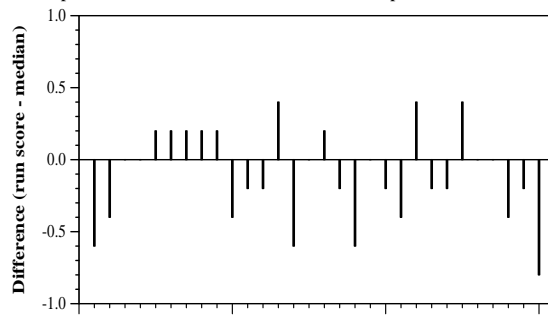
---



Per-topic difference from median bpref for all Round 1 runs



Per-topic difference from median NDCG@10 for all Round 1 runs



Per-topic difference from median P@5 for all Round 1 runs