CMPE 493 TERM PROJECT FINAL PRESENTATION

Çağrı ÇİFTÇİ
Karahan ŞAHİN
İbrahim Özgürcan ÖZTAŞ

Information Retrieval of/during Covid-19

→ WHAT WE HAVE DONE SO FAR

1. Preprocessing Update
   - Lemmatization
   - Stemming (Porter’s Algorithm) (REMOVED!)
   - In addition to title and abstract, we have tried to use body
2. New document model (Okapi BM25) addition to TF-IDF
3. Query expansion
4. Clustering (K-Means) using feature selection
5. Ranking Fusion

Updated Preprocessing

→ We used body of documents in addition to title and abstract.

→ We fixed contractions and used WordNet Lemmatization.

→ We removed Porter’s stemming algorithm to be able to apply query expansion properly.

New Document Model (BM-25)

→ Addition to TF-IDF , we used BM-25 document model to find out the actual value of a word in either a query or document.

→ We used TF-IDF and BM-25 models individually. Also, we used several different ranking fusion methods to combine their scores.

→ We've realized that although BM25 model is more sensitive towards to both tf and idf scores, the model gives lower score while ranking

Query Expansion

→ While processing the query, we did “part-of-speech tagging” for disambiguating the word senses of given token. Then we extended our given query with the retrieved synonyms from WordNet.

- Initially, we have planned to use UMLS corpus for domain specific query expansion but we have encountered **license issues** regarding UMLS corpus, thus we’ve discarded it.
- For MeSH data, we considered that the categorical labels would be dysfunctional for our query modeling

→ For query expansion, we have also tried to used the narrative parts of the query.

- This has resulted with the larger queries with misleading tokens.

Clustering (K-Means)

→ We’ve selected initial centroids to gather data around randomly.

→ Then, we’ve taken the average of the data vectors within the clusters separately, which leads us to the new centroids.

→ KEY: Since **averaging** document **vectors** <u>increase</u> the # of dimensions a document has, we’ve limited the **feature size as 20** and we’ve selected **the highest 20** after any iteration.

→ This has decreased the complexity within the acceptable time limits.

Ranking Fusion

→ We combined TF-IDF and BM-25 scores using following ranking fusion functions:

→ Reciprocal Ranking Fusion (RRF) ====>

$$RR(d_i) = \sum_{q \in Rankings} \frac{1}{rank_q(d_i)}$$

→ Comb Methods developed by Belkin et al. [22]

$$score_{CombSUM}(d) = \sum_{m \in D_m} score(d), \qquad score_{CombANZ}(d) = \frac{1}{\sum_{m \in D_m:\, d \in top_m(1000)} (1)} \sum_{m \in D_m} score(d),$$

$$score_{CombMNZ}(d) = \sum_{m \in D_m:\, d \in top_m(1000)} 1 \cdot \sum_{m \in D_m} score(d)$$

Status of Claimed Improvements

1. Combine different ranking functions (Alpha) DONE!
   a. BM25 ADDED! and TF-IDF
   b. RRF and Comb Functions ADDED!
2. Inject neural networks: BERTs (Epsilon) NOT DONE!
3. Clustering:(Gamma) DONE!
   a. Feature Selection ADDED!
4. Add Question/Answer Model: BioBERT(Delta) NOT DONE!
5. Term Expansion(Beta) DONE!

Error Analysis

1. Document vectors
    a. Sparse vectors → Memory Allocation problem
    b. Solution: Implemented vectors as dictionaries
2. Index comparison
    a. Searching docs with compared indexes → Time Complexity Issue
    b. Solution: Added mapper dictionaries
3. Initial vector structure: vectors as dictionaries
    a. Disabled the use of external libraries → Complicated coding process
    b. Solution: Self-implemented classes, which increased learning rate of the course materials

Test results for the final version

⟶ The table below represents the initial results.

⟶ Our FINAL results are in here.

	Cosine Similarity
Mean Average Precision (MAP)	0.2874
Precision of top 10 results (P@10)	0.6240
Normalized Discounted Cumulative Gain (NDCG)	0.7345

Randomization Test Result

⟶ We have implemented a randomization test module which is available to all users.

⟶ Due to its time complexity and combinatorial complexity, we have tested only one combinatorial case of the results in our analysis:

===> Randomization Test between TF-IDF and TF-IDF Clustered

- P_MAP: 0.000999000999
- P_P@5: 0.847185462095
- P_P@10: 0.371628435679
- P_NDCG: 0.000999000999

Resources

→ [1]Belkin, N.J.; Kantor, P.; Fox, E.A.; Shaw, J.A. Combining the evidence of multiple query representations for information retrieval. Inf. Process. Manag. 1995, 31, 431–448. [CrossRef]

→ Analyzing the selection of the best k in RRF for our implementation: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.150.2291&rep=rep1&type=pdf

→ Feature evaluation: Chen, Jimmy, and William Hersh. ”A Comparative Analysis of System Features Used in the TREC-COVID Information Retrieval Challenge.” medRxiv (2020)

THANKS!