CT4100 Information Retrieval: Assignment II

Jiarong Li 20230033, 1MAI1
Zhe Jing Chin 22221970, 1CSD1

The School of Computer Science
University of Galway
j.li11@nuigalway.ie, z.chin1@nuigalway.ie

1 Question 1

1.1 Requirements

Given a query that a user submits to an IR system and the top N documents that are returned as relevant by the system, devise an approach (high level algorithmic steps will suffice).

To suggest query terms to add to the query. Typically, we wish to give a large range of suggestions to the users capturing potential intended query needs, i.e., high diversity of terms that may capture the intended query context/content. Discuss the efficiency of your approach. (10 marks)

1.2 Approach

There are three main steps in our approach, we are starting with document cleaning to remove the stop-words and follow by scalar clustering to extract the correlated terms in each document. Lastly, we perform clustering to group the terms and return terms extracted from different groups for high-diversity suggestions.

1.2.1 Document Processing

We first remove the stop-words or function words in the documents with the tf-idf technique to remove the noise and keep only the meaningful terms. By doing this, we increase efficiency in terms of saving memory and decreasing computational time.

1.2.2 Correlated Terms Extraction

To extract the correlated terms, we perform scalar clustering. It includes the neighbourhood of a term when computing the correlation between terms. However, it's a challenge to figure out the optimal window size of the neighbourhood. The cosine similarity is computed between vectors while only high similarity terms are extracted as input for the next step.

We are performing an $\mathcal{O}(n)$ to loop through each top N documents returned by the system. In each document, we iterate through each and every vector to calculate the cosine similarity of one vs all. We could not skip vectors as the term in the first sentence is possibly correlated to the term in the last sentence, hence contributing to another $\mathcal{O}(n)$. Overall, we have a $\mathcal{O}(n^2)$ to extract the correlated terms with scalar clustering.

1.2.3 Clustering

Clustering is one of the unsupervised learning methods to group similar objects in the same group and separate not similar objects from other groups. We first apply the hierarchical clustering method to maximise the distance between each cluster by finding out the central seed of each sub-tree of the whole set of terms. We then apply the K-means method to minimise the distance between the terms within each cluster and maximise the distance between each cluster. Applying the clustering technique can help us extract the diversities of terms from different domains.

The advantage of applying K-means method is that K-means uses human judgement for evaluation which can improve the performance of searching. However, determining the size of K is crucial and difficult. If K is large, the cluster will be too tight to show the co-relations between the terms, whereas if K is small, the cluster will be too sparse to show the diversities between terms.

1.2.4 Terms Extraction from Clusters

To maintain high diversity while minimizing the number of terms to output, we narrow the suggestion to top 5 clusters based on the average correlation score of each cluster. Then we randomly pick 5 terms from each cluster, ergo returning maximum of 25 suggestions for query expansion.

2 Question 2

2.1 Requirements

Consider the following scenario: a company search engine is employed to allow people to search a large repository. All queries submitted to the system are recorded. A record that contains the id of the user and the terms in the query is stored. The order of the terms is not stored and neither is any timestamp. Each entry in this record is effectively and id and a set of terms. Any duplicate terms in a query is ignored.

The designers of the search engine, decide to use this information to develop an approach to make query term suggestions for users, i.e., at run time once a user an entered their queries terms, the system will suggest potential extra terms to add to the query.

Given the data available, outline an approach that could be adopted to generate these suggested terms. A brief outline is sufficient that captures the main ideas in your approach.

Identify advantages and disadvantages of your approach (briefly). (10 marks)

2.2 Approach

A brief outline of our approach would be to do a term cleaning with stop-words removal, then generating co-occurrence terms based on history log, lastly to return terms extracted from clusters and ranked lists.

2.2.1 Query pre-processing

We first apply stop-words removal to avoid the suggesting noise and keep only the meaningful terms within the query. We can save memory and decrease computational time with such query pre-processing.

2.2.2 Co-occurrence of terms

Then in the query analysis process, we find out the co-occurrence terms from the history log of the whole query set to find the most related terms as the term suggestion. Generally, the terms returned from this step should include 2 parties: (1) Terms searched by own-self. (2) Terms searched by others. However, for new users, it should only return terms searched by others.

2.2.3 Diversity of terms

We apply unsupervised clustering method to increase the diversity of term suggestions. We choose the top N terms from each cluster and return them to the user. As we mentioned in 2.2.2, the term returned should include the terms searched by own-self and include the terms searched by others. And for new users, it only returns the terms searched by other users.

2.3 Advantages

From the above approach, with the help of clustering, we are able to return a list of high diversity terms to the users such that they could explore in a broader area. Besides, the returned list is selected based on the ranked occurrence of previous searches, hence giving options for users to search based on the popularity of terms.

For existing users, they would be suggested terms that they had or had not searched previously, thus increasing the probability of exploring something they have not known before.

2.4 Disadvantages

On the contrary, since the system has no information about the previous query's context, it has a high probability to return completely non-related terms to the users.

Furthermore, it might be computationally expensive to get the co-occurrence of query terms. It is challenging to figure out the appropriate clusters either.