IR Mini Project

Contextual Query Processing

Debangshu Bhattacharya (MDS201910)

Swaraj Bose (MDS201936)

Avirup Chakraborty (MDS201908)

Ipsita Ghosh (MDS201913)

1 Overview

In this project, we have considered an approach to use contextual information of the query to enhance the cosine similarity based search engine of a vector space model. Based on the predicted context of the query, we find out the documents which are relevant with respect to that context. Then, we simply modify our query using Rocchio feedback mechanism. We compare this method with original cosine similarity based model and see the improvement in Mean Average Precision (MAP).

2 Summary

We consider the vector space model and the TF-IDF representation of documents. As the benchmark, we take the cosine similarity for scoring documents w.r.t queries. The main idea of the project is to improve this scoring using contextual information of query and documents. We first take a corpus of documents, labelled w.r.t their contexts. From these documents, we extract relevant words automatically (after removing non relevant words from them). We claim that these words contain the contextual information of those documents. We next take a separate corpus of unlabelled documents containing instances from each of the considered contexts. We pretend we do not know the true context of these contexts. We only use the true context to evaluate the search engine performance. Mean average precision is used to compare the existing method with our proposed method. Here, the true contexts of the documents have been used in the aforementioned evaluation.

In the course of this project we thought of the word "air" which can have multiple meanings depending on the context (illustrated below) and we restrict our context space to 4 contexts. The contexts and the different implications of the word "air" in those contexts are provided below:

- **Rank**: AIR stands for All India Rank.
- **Nature**: The air we breathe
- **Broadcast**: AIR stands for All India Radio, or in a way like "...the news on air tonight"
- **Flight**: air force, airplanes, air crafts, Air India

We construct a toy dataset, and we preprocess the documents in our corpus to remove stopwords, and special characters. We then proceed to extract the relevant words from our training set of documents. Subsequently, we find the probabilities of occurrence of these words over the various contexts. With the created matrix of words and their probabilities of occurrence in the different contexts taken into consideration, we next find the context of each document belonging to the corpus of documents on which we want to evaluate the performance of our proposed method (test set). To predict the context for these documents, we take every word in the document and find its probability of occurrence over various contexts (if the word exists in the probability matrix). Now, for all the words in the document, we add the probabilities of occurrence over all terms for each individual context, and finally assign the context which has the highest average of these probabilities. In case we fail to assign a context to a document, the idea is to proceed with cosine similarity without any modifications.

For a given query, we first wish to predict its context. This is done using the same probability matrix and following the same method as described above. Once we have the context of the query, we immediately know which are the documents which belong to the same context, and hence we know the relevant documents w.r.t the query. Similarly, we know the documents which are not relevant w.r.t the query. We use Rocchio feedback based on the above to modify our initial query vector using the information on the relevant and non relevant documents. We then score documents based on the cosine similarity score of the modified query and the corpus of documents at our disposal.

3 Results

We are retrieving top 3 documents for each individual query. The average precision (AP) scores are computed with a relevance level of 3. For modifying query with respect to context using Rocchio feedback, we use α = 1, β = 0.8, and γ = 0.2 in our experiment. The APs have been calculated for 6 different queries and eventually the MAP based on them for both the cosine similarity method of scoring and our proposed method of scoring. The results have been tabulated below:

Query	Predicted Context	AP (Cosine similarity)	AP (Proposed method)
IIT cut off this year	Rank	1.00	1.00
current atmosphere pressure	Nature	0.667	1.00
air radio jockey	Broadcast	1.00	1.00
computer science toppers this year	Rank	1.00	1.00
members of Air India club	Flight	0.00	0.667
available luxury flight tickets	Flight	0.111	0.389

Further, the Mean Average Precision for the two methods are as follows:
MAP (Cosine similarity) = 0.630
MAP (Our method) = 0.843

4 Conclusion

We can see from the table above that our method is significantly outperforming vanilla cosine similarity method with an MAP rise from 0.630 to 0.843, which is approximately a 33.81% increase. We are able to correctly predict the contexts in each case. For example, let us take the query "members of Air India club". Unaltered cosine similarity was unable to retrieve any documents from our corpus, but our proposed method having correctly detected that the context of the query is "Flight", manages to retrieve relevant documents, which is reflected by the AP scores.

5 Future Works

- Here, we have taken a single training set consisting of a few documents to extract words which are relevant w.r.t a context, and have not added anything to the list of the words post this. We could further explore the idea of adding words from future documents which we encounter, if we can classify them to a context with a high probability. The words from these documents can also be extracted and added to our existing word-probability matrix for various contexts.
- Since we have considered a toy dataset, we have used basic context extraction algorithm (after removing non-relevant words according to our judgement). We can also incorporate the use of NLP/ML algorithms to extract the contexts if sufficient data is available.

References

[1] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, England. Online edition (c) 2009 Cambridge UP.