Assignment 3

Submitted by

Swati Verma

MT19073

Question 1

Assumptions:

1. No metadata has been removed from any of the documents.

Preprocessing steps:

1. Words in all the documents are converted to lowercase.
2. Punctuations have been removed from all the documents.
3. Expansion such as don't to do not has not been handled because the library was not working on my system.
4. Stopwords have been removed from all the documents using nltk library.
5. To convert surface words to root word lemmatization is used instead of stemming because lemmatization gives meaningful words.
6. All the above steps are done both for documents and query which is given by the user.
7. Different variations of idf have been used in this question.

Methodology:

1. The file is read and after tokenization and lemmatization a dictionary is created which contains document id and tokenized document list as key value pair.
2. A tf_dictionary is made which consists of each term as key and a dictionary of doc_id and term frequency of that term as value. Basically this dictionary contains the term frequency of each term for each document respectively.
3. A dataset where relevance score is given has been read and a dictionary called gd_final is created which contains doc_id and relevance score as key value pairs. Gd values have been normalized by dividing each gd value by the sum of all the gd values so that the relevance score can lie between 0 and 1. This dictionary will act as a champion list.

4. To make a global champion list two lists are created namely high list and low list. A high list is created using the top r documents which have the highest tf values for each term. And a low list consists of all the remaining documents which have lower tf values for each term.
5. For query processing I have used g(d)+ cosine_similariy between query terms and each document. Initially, the high list is considered for query processing. If top k relevant documents are found using high list then top k documents are returned, else lower list is also considered and top k relevant documents are returned.

Question 2

Assumptions:

1. No metadata has been removed from any of the documents.
2. To compute dcg, summation of (rel(i)/log base2(i+1)) is used.

Methodology:

1. The file containing urls has been read and a list of lists is created where each list contains tokens of each line of url.
2. To find max dcg score the urls are sorted on the basis of the first column which is the relevance score. Dcg is calculated by using the formula mentioned in assumption and total number of combinations is computed.
3. To compute dcg and ndcg different functions are defined.
4. After that ndcg was calculated for top 50 urls and the whole dataset.
5. At the end precision recall curve was plotted where precision and recall was computed after each url retrieval. The curve obtained looks like sawtooth.