FIRE 2017 IRLeD track submission for Task - 1

G. V. Sandeep

Shikhar Bharadwaj

Birla Institute of Technology & Science, Pilani

Hyderabad Campus.

Working of The Model

Type: Automatic

Description:

1. Read all the documents word wise by splitting them using nltk word tokenizer.
2. Convert all words into lowercase. Ignore all those words with length less than 4 and ignore all the stop words.
3. For each word, find its frequency in each document and also check whether it occurs in the set of keywords or not. Thus, for each word we will have many tuples associated in the format (A,B) where:
    a. A = frequency of the word in the document.
    b. B = whether that word exists is the set of keywords or not.
        i. 1 - if exists only in the document
        ii. 2 - if exists in both the document as well as the associated set of keywords.
4. Since now we have all words and their associated list of tuples we will calculate the threshold frequency above which the word the is likely to be present in set of keywords. Eg: abduction [(3,0), (2,0), (4,1), (5,1)] - then the threshold would be 4
5. While computing the threshold we give weightage to some words using POS tagging. For that purpose we have initially found out what Part of Speech is more likely to be a catchphrase from the given training set. After normalising the POS weight to be in a range of [0,1], we multiply the POS weight by a coefficient and subtract this value from the threshold.

6. Once we have a threshold for each word, we iterate over all the documents again.
7. For each document we pick up one word from it, calculate its frequency in the document and compare it with the threshold for that word. If the frequency is greater than the threshold of the word than the word is retained in the document or else it is dropped. Thus at the end of this iteration we have a set of shortened documents.
8. From this shortened document we find a list of top 10 popular words by noting the frequency of each word. After getting the popular words we make a set S of sentences from the document that contain any one of these popular words.
9. Now we filter the document even further by retaining only S. From this set we filter the words that are not present more than once in the document.
10. Finally after filtering all useless words we get a list of keywords that are scored by summing TF IDF, number of occurrences in S and word frequency scores. This score is normalised to be in a range of [0,1].

Limitations:

- Doesn’t generate keyphrases as the information of position is lost during preprocessing but rather generates keywords
- Since it is an extractive model, the model cannot predict keywords which are not present in the document itself.
- Assumes that the most important keywords necessary to identify a document are the ones that are most frequent in the document.