IR-TP Project individual report Group -9

K Yogeshwara Krishna 19EC10032

Worked on Part 2a queries TF-IDF and cosine similarity.

Task and Design:

- Read and parsed queries from the file path. Tokenized it, removed stop words, punctuation marks and performed lemmatization.
- Calculated Term Frequency for every term by counting instances in each processed document. Calculated Document Frequency by checking the length of the postings list in the inverted index.
- Built TF and IDF of the tokens in each query. Using dictionaries for TF helped in an average constant time lookup. Defaultdict() helped to keep multiple keys.
- Built the TF-IDF vector with ltc, lpc and apc conventions using numpy vectors and functions for faster performance.
- Checked if any terms inside a log function becomes zero or not, to avoid that error.
- Built a function for cosine similarity, applied it on the above vectors between queries and documents using the formula:

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2} \sqrt{\sum\limits_{i=1}^{n} B_i^2}}$$

and ranked them based on the following types

- Lnt.ltc (n means just 1)
- Lnc.lpc
- Anc.apc

Where the conventions have these formulas:

l (logarithm)	$1 + \log(\mathrm{tf}_{t.d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2 + w_2^2 + \ldots + w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N - \mathrm{df}_t}{\mathrm{df}_t}\}$		

- Also built a function to write files into a csv file, due to multiple such usages.
- Used Garbage collection to delete non relevant dictionaries and other variables created during intermediate steps.

Challenges:

- The vector size was too large for our local computer's memory: We lower cased the tokens to reduce the vocabulary size, makes sense intuitively too.