Information Organization Midterm Report(November 22, 2023)

Name: XIONG ZHIPENG **Student ID:** 44231536

QUESTION1: N=52000

Term	Apricot	Apple	Applemango	Appendix	Apply	and
Document frequency	194	511	10	86	2540	50970
rank	2500	1000	50000	6000	200	10

1) How a very frequent term like “and” in the above is called?

Words that appear frequently in a text and do not carry much meaning are known as "stop words".

2) Compute IDF score of each term in the table.

As we know, the formula for calculating IDF is:$IDF = \log\left(\frac{N}{df}\right)$ where N is the total number of the documents and the df is the document frequency of the term.

Term	apricot	apple	applemango	appendix	apply	and
IDF	$\log\left(\frac{52000}{194}\right) = 2.4282$	$\log\left(\frac{52000}{511}\right) = 2.0076$	$\log\left(\frac{52000}{10}\right) = 3.7160$	$\log\left(\frac{52000}{86}\right) = 2.7815$	$\log\left(\frac{52000}{2540}\right) = 1.3112$	$\log\left(\frac{52000}{50970}\right) = 0.0087$

3) Suppose that an inverted index having non-positional posting lists is constructed using this dictionary, where each posting consists of document ID and its term frequency, and the postings are sorted by the descending order of term frequencies. Show an efficient method to return the top-100 documents ranked by TF-IDF score, for the Boolean query below, where ‘|’ means OR, ‘&’ means AND, and ‘^’ means NOT. List up the names of the methods you used.

$$(\text{^}\boldsymbol{applemango} \mid \boldsymbol{apple})\&\boldsymbol{apply}\&\boldsymbol{apricot}$$

Answer within 200 words.

a) **Inverted Indexing:** Retrieve the data structure that maps words to sets of documents.
b) **Boolean Retrieval:** Get the posting list for apply&apricot and (^applemango | apple) first. And then intersect the two posting-lists, get the (^applemango | apple)&apply&apricot.
c) **TF-IDF Scoring:** calculate TF-IDF score using the term frequency and the result of (2)
d) Return top 100 documents by TF-IDF score

4) Estimate a function $F(r)$ that approximately computes term frequency $F(r)$ from its rank r.

$\mathrm{F(r)} = K/r$ K may equal to 50000 because the most common term is 'and' whose document frequency is 50970.

5) How the distribution of (3) is called?

Zipf's law

6) Show a space-efficient format of the dictionary, utilizing front coding, recoding their frequencies and pointers to their posting lists.

5apple5apply10applemango7appendix → 5app*le2ly7lemango5endix

Or:

apple 4y 5mango 3endix

QUESTION2: N=10000

Term	Collection frequency	Document frequency	Frequency in d_1	Frequency in d_2	Frequency in d_3
fresh	4000	2000	5	10	50
fish	2000	500	10	20	10
sushi	1000	200	10	0	20

1) **We want to rank the terms in the document $d_1, d_2, and\ d_3$ by importance, using TF-IDF score. Calculate the TF-IDF score of each term, using ltn.lnc, namely logarithmic term frequency $1 + \log_{10}(tf_{t,d})$, IDF $log_{10}\frac{N}{df_t}$ document frequency. Here, use $log_{10} 2 = 0.30$ and $log_{10} 5 = 0.70$**

Term	d_1	d_2	d_3
fresh	Ltn $= 1 + \log_{10} 5$ $= 1.7$ idf $= log_{10}\frac{10000}{2000}$ $= 0.7$	Ltn $= 1 + \log_{10} 10$ $= 2$ idf $= log_{10}\frac{10000}{2000}$ $= 0.7$	Ltn $= 1 + \log_{10} 50$ $= 2.7$ idf $= log_{10}\frac{10000}{2000}$ $= 0.7$

	$\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 1.7 * 0.7}$ $\boldsymbol{= 1.19}$	$\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 2 * 0.7}$ $\boldsymbol{= 1.4}$	$\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 2.7 * 0.7}$ $\boldsymbol{= 1.89}$
Fish	$\boldsymbol{Ltn}$ $\boldsymbol{= 1 + \log_{10} 10}$ $\boldsymbol{= 2}$ $\boldsymbol{idf}$ $\boldsymbol{= log_{10} \frac{10000}{500}}$ $\boldsymbol{= 1.3}$ $\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 2 * 1.3}$ $\boldsymbol{= 2.6}$	$\boldsymbol{Ltn}$ $\boldsymbol{= 1 + \log_{10} 20}$ $\boldsymbol{= 2.3}$ $\boldsymbol{idf}$ $\boldsymbol{= log_{10} \frac{10000}{500}}$ $\boldsymbol{= 1.3}$ $\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 2.3 * 1.3}$ $\boldsymbol{= 2.99}$	$\boldsymbol{Ltn}$ $\boldsymbol{= 1 + \log_{10} 10}$ $\boldsymbol{= 2}$ $\boldsymbol{idf}$ $\boldsymbol{= log_{10} \frac{10000}{500}}$ $\boldsymbol{= 1.3}$ $\boldsymbol{TF}$ $\boldsymbol{-\ IDF\ weight}$ $\boldsymbol{= ltn * idf}$ $\boldsymbol{= 2 * 1.3}$ $\boldsymbol{= 2.6}$
sushi	$\boldsymbol{Ltn}$ $\boldsymbol{= 1 + \log_{10} 10}$ $\boldsymbol{= 2}$	$\boldsymbol{Ltn = 0}$ $\boldsymbol{idf}$ $\boldsymbol{= log_{10} \frac{10000}{200}}$ $\boldsymbol{= 1.7}$	$\boldsymbol{Ltn}$ $\boldsymbol{= 1 + \log_{10} 20}$ $\boldsymbol{= 2.3}$

	idf $= log_{10} \frac{10000}{200}$ $= 1.7$ TF $- IDF\ weight$ $= ltn * idf$ $= 2 * 1.7$ $= 3.4$	TF $- IDF\ weight$ $= ltn * idf$ $= 0$	idf $= log_{10} \frac{10000}{200}$ $= 1.7$ TF $- IDF\ weight$ $= ltn * idf$ $= 2.3 * 1.7$ $= 3.91$

2) **We want to measure similarities among $d_1, d_2, and\ d_3$ by cosine similarity.**

   a) **Show the term frequency vectors of $d_1, d_2, and\ d_3$ on the three term frequency is $1 + \log_{10}(tf_{t,d})$, and length-normalized.**

Word	d_1				d_2				d_3			
	ltn	idf	wt	N'lized	ltn	idf	wt	N'lized	ltn	idf	wt	N'lized
Fresh	1.7	0.7	1.19	0.27	2	0.7	1.4	0.42	2.7	0.7	1.89	0.37
Fish	2	1.3	2.6	0.59	2.3	1.3	2.99	0.91	2	1.3	2.6	0.51
sushi	2	1.7	3.4	0.77	0	1.7	0	0	2.3	1.7	3.91	0.77

	d_1	d_2	d_3
$W_{fresh}^2 + W_{fish}^2 + W_{sushi}^2$	19.7361	10.9	25.62
$\sqrt{W_{fresh}^2 + W_{fish}^2 + W_{sushi}^2}$	4.44	3.3	5.06

b) Compute the cosine values for each pair of the three vectors, and determine which document pair is most similar.

$$\boldsymbol{Similarity_{(d_1,d_2)} = \sum_{i}^{3} W_{d_{1i}} * W_{d_{2i}}}$$

$$\boldsymbol{= 1.19 * 1.4 + 2.6 * 2.99 + 3.4 * 0 = 9.44}$$

$$\boldsymbol{Similarity_{(d_1,d_3)} = \sum_{i}^{3} W_{d_{1i}} * W_{d_{3i}}}$$

$$\boldsymbol{= 1.19 * 1.89 + 2.6 * 2.6 + 3.4 * 3.91 = 22.3031}$$

$$\boldsymbol{Similarity_{(d_2,d_3)} = \sum_{i}^{3} W_{d_{2i}} * W_{d_{3i}}}$$

$$\boldsymbol{= 1.4 * 1.89 + 2.99 * 2.6 + 0 * 3.91 = 10.42}$$

$$\boldsymbol{Similarity_{(d_1,d_3)} > Similarity_{(d_2,d_3)} > Similarity_{(d_1,d_2)}}$$

(d_1, d_3) pair is most similar.

QUESTION3:

Document	1	2	3	4	5	6	7	8	9	10	11	12
Alice	4	0	2	1	3	1	2	5	3	2	1	2
Bob	3	2	5	3	4	0	3	3	0	2	3	1
SysA	3	1	6	1	0	1	3	6	2	4	1	0
SysB	5	1	4	3	3	1	5	6	0	3	0	3

1) **Define that Alice and Bob agree on document I if s(Alice,i)≥3 and s(Bob,i) ≥3 or s(Alice,i)<3 and s(Bob,i)<3. Otherwise, there have no agreement on i. Now, how you evaluate the level of agreement between Alice and Bob?**

I think it is appropriate to evaluate the level of agreement of between Alice and Bob using the proportion of documents on which they agree. The percentage is 7/12.

2) **Suppose that information retrieval system SysA and SysB are tested on query "seasonal greetings" and they returned ranking on documents 1-12, as shown in the same table. Also, define the average score of document i as avg(i)=[s(Alice,i)+s(Bob,i)]/2. Now suppose that document I is relevant to the query if avg(i) ≥3. Using this relevance, compute the precision, recall, and F1-score of SysA and SysB.**

Document	1	2	3	4	5	6	7	8	9	10	11	12
avg	3	1	3	2	3	0	2	4	1	2	2	1

sysA	3	1	6	1	0	1	3	6	2	4	1	0
sysB	5	1	4	3	3	1	5	6	0	3	0	3

If the score of the system ≥3, the document is considered retrieved.

sysA retrieve the relevant document 1,3, 8;

For sysA, the retrieved documents are: 1, 2, 3, 4, 6, 7, 8, 9, 10, 11.

	Relevant	Nonrelevant
Retrieved	TP=3	FP=7
Not Retrieved	FN=1	TN=1

Precision P = TP/(TP+FP) = 3/10 = 0.3

Recall R = TP/(TP+FN) =3/4 = 0.75

F1-score = 2 * (Precision * Recall) / (Precision + Recall) = 2 * (0.3*0.75)/(0.3+0.75) ≈ 0.42857

sysB retrieve the relevant document 1, 3, 5, 8:

For sysB, the retrieved documents are: 1, 2, 3, 4, 5, 6, 7, 8, 10, 12.

	Relevant	Nonrelevant
Retrieved	TP=4	FP=6
Not Retrieved	FN=0	TN=2

Precision P = TP/(TP+FP) = 4/10 = 0.4

Recall R = TP/(TP+FN) = 4/4 = 1

F1-score = 2 * (Precision * Recall) / (Precision + Recall) = 2*(0.4 * 1)/(0.4 + 1) ≈ 0.57143

3) Now determine which system in SysA or SysB is producing a

ranking that is closer to the ranking by the average score avg(i) of the 12 documents. Here, consider using precision@K, and recall@K, with K varying from 1 to12. Here, precision (resp. recall@K) with respect to a ranking is the precision (resp.recall) when documents ranked higher or equal to K are regarded as releveant.

According to the average of the F1-Score, the sysA is better.

For SysA, the ordered documents are 3, 8, 10, 1, 7, 9, 2, 4, 6, 11, 5, 12.

Ranking:6, 6, 4, 3, 3, 2, 1, 1, 1, 1, 0, 0

if K > 6, there isn't relevant document.

K	Retrieved	Relevant Retrieved	Precision@K	Recall@K
1	3	1	1	0.1
2	3,8	2	1	2/6≈0.33
3	3,8,10	3	1	3/5=0.6
4	3,8,10,1	3	3/4=0.75	3/3=1
5	3,8,10,1,7	2	2/5=0.4	2/2=1
6	3,8,10,1,7,9	2	2/6 ≈ 0.33	1

Average Precision = (1+1+1+3/4+2/5+2/6)/6≈0.7472

Average Recall=(0.1+1/3+3/5+1+1+1)/6≈0.6722

Average F1-Score=2*0.7472*0.6722/(0.7472+0.6722) ≈0.7077

For SysB, the ordered documents are 8, 1, 7, 3, 4, 5, 10, 12, 2, 6, 9, 11

Ranking: 6, 5, 5, 4, 3, 3, 3, 3, 1, 1, 0, 0

if K > 6, there isn't relevant document.

K	Retrieved	Relevant Retrieved	Precision@K	Recall@K
1	8	1	1	0.1
2	8, 1	2	1	0.25
3	8, 1, 7	3	1	3/8 = 0.375
4	8, 1, 7, 3	4	1	1
5	8, 1, 7, 3, 4	3	3/5=0.6	1
6	8, 1, 7, 3, 4,5	1	1/6 ≈ 0.1667	1

Average Precision=(1+1+1+1+3/5+1/6)/6≈0.7944

Average Recall=(0.1+0.25+3/8+1+1+1)/6≈0.62208

Average F1-Score= 2*0.7944*0.62208/(0.7944+0.62208)=0.6977

Discuss measurement of similarities of ranking other than the method used in previous question(within 100 words).

Besides the Jaccard index which measures the proportion of shared items to total items in two ranked lists, there is Kendall's Tau, which is a measure of the correspondence between two rankings. Values close to 1 indicate that the rankings are similar, while values close to -1 indicate that one ranking is the reverse of the other. Kendall's Tau is calculated based on the number of pairwise disagreements between two ranking lists.

QUESTION4:

1) **Explain where the following notions are used in the top-5 result.**

   a) **Snippet.**

   Snippet is a short summary of the document, which is designed so as to allow the user to decide the documents' relevance. In the image, the black words, such as "2-7 Hibikino, Wakamatsu-ku…", following the URL are snippet.

   b) **Metadata**

   Metadata is data that provides information about other data. In the

image, the green words, such as “www.waseda.jp › top › access › kitakyushu-campus” , following the URL, are Metadata.

c) Dynamic summary

Dynamic summary is a customized summary that is generated for a specific query. Dynamic summaries display one or more “windows” on the document, aiming to present the pieces that have the most utility to the user in evaluating the document with respect to their information need. In the image, the bold words-“Kitakyushu” and “station” are the results using Dynamic summary.

2) Argue what ranking strategy could been used to generate this top-5 results.

When searching for a particular query, the search engine first filters for documents that contain all the relevant keywords. It then ranks them based on various factors such as term frequency, inverse document frequency, and the presence of the query terms in important parts of the document like the title or URL. If the query terms appear close together in a document, it considers term proximity as well. In the case of web pages, the engine also uses PageRank to determine the overall importance or popularity of each page. In summary, the search engine uses strategies like Term Frequency, Inverse Document Frequency (IDF), Precision, Recall, F1-score, Term Proximity, PageRank, and Weight to deliver the best results for the user's query.