Q1 (6%) Top-k Query

A real estate database contains information about 10 apartments available for sale, characterized by their ratings (a_1, out of 10) and prices (a_2, in thousands of dollars), as listed in the table below.

The database employs the following aggregation function (f):

$$f = 0.7 \times a_1 + 0.3 \times a_2$$

to rank these apartments, where higher scores indicate better options.

Apartment Data Table

Apartment	Rating (a_1)	Price (a_2)
A	10	5
B	1	9
C	3	3
D	7	8
E	9	6
F	6	1
G	2	2
H	5	7
I	4	10
J	8	4

Questions

a) Prepare two lists of apartments, sorted in descending order of a_1 and a_2, respectively.

b) Apply the Threshold Algorithm (TA) to determine the two best apartments according to the aggregation function f. Provide a step-by-step explanation of your process.

c) Apply the No Random Access (NRA) Algorithm to determine the two best apartments according to the aggregation function f. Provide a step-by-step explanation of your process.

d) Compare the TA and NRA solutions in terms of the number of iterations required. Which algorithm is more efficient in this case, and why?

Solution

a)

Rank	By Rating (a_1)	Rating	By Price (a_2)	Price
1	A	10	I	10
2	E	9	B	9
3	J	8	D	8
4	D	7	H	7
5	F	6	E	6
6	H	5	A	5
7	I	4	J	4
8	C	3	C	3
9	G	2	G	2
10	B	1	F	1

b) TA

Step	Accessed Items	f-score Calculations	Current Top-2	Threshold Calculation	Stop Condition
1	A (a_1=10), I (a_2=10)	f(A)=0.7×10+0.3×5=8.5 f(I)=0.7×4+0.3×10=5.8	A(8.5), I(5.8)	T=min(0.7×10+0.3×10, 0.7×10+0.3×10)=10	10 > 5.8 → Continue
2	E (a_1=9), B (a_2=9)	f(E)=0.7×9+0.3×6=8.1 f(B)=0.7×1+0.3×9=3.4	A(8.5), E(8.1)	T=min(0.7×9+0.3×9, 0.7×10+0.3×9)=9	9 > 8.1 → Continue
3	J (a_1=8), D (a_2=8)	f(J)=0.7×8+0.3×4=6.8 f(D)=0.7×7+0.3×8=7.3	A(8.5), E(8.1)	T=min(0.7×8+0.3×8, 0.7×10+0.3×8)=8	8 > 8.1 → Continue
4	D (a_1=7), H (a_2=7)	f(D)=7.3 (already calculated) f(H)=0.7×5+0.3×7=5.6	A(8.5), E(8.1)	T=min(0.7×7+0.3×7, 0.7×10+0.3×7)=7	7 ≤ 8.1 → **STOP**

Final top-2: A (8.5), E (8.1)

c) NRA

1. Initialize:
    - Seen: {}
    - Top-2: []
    - For each item, maintain lower and upper bounds
2. Access first items from both lists (A from a_1, I from a_2)
    - A: lower = upper = 8.5

- I: lower = upper = 5.8
- Top-2: [A, I]

3. Next items (E from a_1, B from a_2)
   - E: lower = upper = 8.1
   - B: lower = upper = 3.4
   - Top-2: [A, E]
4. Next items (J from a_1, D from a_2)
   - J: lower = 0.7×8 + 0.3×1 = 5.9, upper = 0.7×8 + 0.3×10 = 8.6
   - D: lower = 0.7×1 + 0.3×8 = 3.1, upper = 0.7×10 + 0.3×8 = 9.4
   - Check if any items can be discarded:
     - Worst score in top-2: 8.1
     - B (3.4), C, F, G, H cannot surpass 8.1
   - Top-2 remains [A, E]
5. Termination when top-2 are confirmed:
   - A and E have exact scores
   - No other items can surpass them

Final top-2: A (8.5), E (8.1)

d) Comparison

Algorithm	Iterations	Random Access	Efficient?
TA	3	Yes	More efficient here
NRA	3	No	Less efficient for few candidates

TA is more efficient in this case because the number of required accesses is low and random access is allowed, leading to earlier convergence.

Q2 (4%) Big Text Data

Suppose that a corpus with a dictionary of words {α, β, γ, δ} contains 3 documents, and the term frequencies (in brackets) for these documents are shown below.

Document Term Frequencies

Doc ID	Terms (frequency)
1	α(3), β(0), γ(2), δ(0)
2	α(1), β(0), γ(1), δ(0)
3	α(0), β(0), γ(1), δ(2)

Questions

a) Derive the tf-idf vectors for the three documents, based on the formulas discussed in the lecture notes (P. 26).

b) Consider the following string X, which is formed by concatenating terms α, β, and γ, i.e.,
$\mathrm{X} = \gamma\alpha\beta\gamma$
Suppose that X is used to query the documents above. Which of these documents should be ranked first, using the similarity function shown in P.22 of the lecture notes?

Solution

a)

$\mathrm{IDF}(t) = \ln\left(\frac{N}{df_t+1}\right) + 1$

- $df(\alpha) = 2 \rightarrow idf(\alpha) = \ln(3/(2+1)) + 1 = \ln(1) + 1 = 0 + 1 = 1$
- $df(\beta) = 0 \rightarrow idf(\beta) = \ln(3/(0+1)) + 1 = \ln(3) + 1 \approx 1.099 + 1 \approx 2.099$
- $df(\gamma) = 3 \rightarrow idf(\gamma) = \ln(3/(3+1)) + 1 = \ln(0.75) + 1 \approx -0.288 + 1 \approx 0.712$
- $df(\delta) = 1 \rightarrow idf(\delta) = \ln(3/(1+1)) + 1 = \ln(1.5) + 1 \approx 0.405 + 1 \approx 1.405$

TF-IDF vectors:

- **Doc 1:**
  α: 3 × 1 = 3
  β: 0 × 2.099 = 0
  γ: 2 × 0.712 = 1.424
  δ: 0 × 1.405 = 0
  → **[3, 0, 1.424, 0]**
- **Doc 2:**
  α: 1 × 1 = 1
  β: 0 × 2.099 = 0
  γ: 1 × 0.712 = 0.712
  δ: 0 × 1.405 = 0
  → **[1, 0, 0.712, 0]**
- **Doc 3:**
  α: 0 × 1 = 0
  β: 0 × 2.099 = 0
  γ: 1 × 0.712 = 0.712
  δ: 2 × 1.405 = 2.81
  → **[0, 0, 0.712, 2.81]**

b)

To determine which document should be ranked first using X = γαβγ as the query, I'll calculate the cosine similarity between the query vector and each document's tf-idf vector.

First, let me create the query vector for X = γαβγ:

- α occurs once: tf(α) = 1

- β occurs once: tf(β) = 1
- γ occurs twice: tf(γ) = 2
- δ occurs zero times: tf(δ) = 0

Converting to tf-idf:

- α: 1 × 1 = 1
- β: 1 × 2.099 = 2.099
- γ: 2 × 0.712 = 1.424
- δ: 0 × 1.405 = 0

Query vector: [1, 2.099, 1.424, 0]

Now calculating cosine similarity with each document:
Now calculating cosine similarity with each document:

- **Doc 1** [3, 0, 1.424, 0]:
  $\cos(q, d1) = \frac{1 \times 3 + 2.099 \times 0 + 1.424 \times 1.424 + 0 \times 0}{\sqrt{(1^2 + 2.099^2 + 1.424^2 + 0^2)} \times \sqrt{(3^2 + 0^2 + 1.424^2 + 0^2)}}$
  $= \frac{3 + 0 + 2.028}{\sqrt{7.434} \times \sqrt{11.028}}$
  $= \frac{5.028}{2.727 \times 3.321}$
  $= \frac{5.028}{9.056}$
  $= 0.555$
- **Doc 2** [1, 0, 0.712, 0]:
  $\cos(q, d2) = \frac{1 \times 1 + 2.099 \times 0 + 1.424 \times 0.712 + 0 \times 0}{\sqrt{(1^2 + 2.099^2 + 1.424^2 + 0^2)} \times \sqrt{(1^2 + 0^2 + 0.712^2 + 0^2)}}$
  $= \frac{1 + 0 + 1.014}{\sqrt{7.434} \times \sqrt{1.507}}$
  $= \frac{2.014}{2.727 \times 1.228}$
  $= \frac{2.014}{3.349}$
  $= 0.601$
- **Doc 3** [0, 0, 0.712, 2.81]:
  $\cos(q, d3) = \frac{1 \times 0 + 2.099 \times 0 + 1.424 \times 0.712 + 0 \times 2.81}{\sqrt{(1^2 + 2.099^2 + 1.424^2 + 0^2)} \times \sqrt{(0^2 + 0^2 + 0.712^2 + 2.81^2)}}$
  $= \frac{0 + 0 + 1.014 + 0}{\sqrt{7.434} \times \sqrt{8.403}}$
  $= \frac{1.014}{2.727 \times 2.899}$
  $= \frac{1.014}{7.906}$
  $= 0.128$

Since Doc 2 has the highest similarity score of 0.601, followed by Doc 1 with 0.555, and Doc 3 with 0.128, **Doc 2 should be ranked first**.