

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

The Lecture Contains:

- ☰ Multiple attributes
- ☰ Aggregation
- ☰ Assumptions
- ☰ Example
- ☰ Fagin's algorithm (FA)
 - Example: $k = 1$
 - Analysis
- ☰ Threshold algorithm (TA)
 - Example: $k = 1$
 - Analysis

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Multiple attributes

- Each object in the database has m attributes
- Equivalently, m separate information sources
- Query involves all m attributes
- Attributes are independent
- Example: Object has <color, shape, size>
- Example query: Return all objects that are red, round, big
- Easily solved
- Even window queries can be handled efficiently

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Multiple attributes

- Each object in the database has m attributes
- Equivalently, m separate information sources
- Query involves all m attributes
- Attributes are independent
- Example: Object has <color, shape, size>
- Example query: Return all objects that are red, round, big
- Easily solved
- Even window queries can be handled efficiently
- Basic assumption is that query can be processed one attribute at a time

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Aggregation

- Query Q consists of m attributes as well
- For each attribute Q_j and each corresponding attribute O_{ij} of object O_i , there is a score of matching Q_j to O_{ij}
- Score of matching Q to O_i is an aggregate function f of m such individual scores

$$s(Q, O_i) = f(\forall_{j=1}^m s(Q_j, O_{ij}))$$

- NN query : Return k objects whose aggregate scores are the highest

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Assumptions

- Aggregate function f must be *monotonic*
- Denote $s(Q, O_i)$ by s_i and $s(Q_j, O_{ij})$ by s_{ij}

$$\forall_{j=1}^m s_{ij} \leq s_{kj} \implies s_i \leq s_k$$

- Example: sum, max
- Lists of scores of all objects for every Q_j is available
- List for Q_j is of the form $\{(O_{i_1j}, s_{i_1j}), (O_{i_2j}, s_{i_2j}), \dots, (O_{i_{mj}, s_{i_{mj}}})\}$
- Total size is $m \times n$
- Even ranked (sorted) lists are available

Prev topic

Next topic

Next page

Prev page

Example

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

[Prev topic](#)
[Next topic](#)
[Prev page](#)
[Next page](#)

Example

List ₁	List ₂	List ₃	Aggregate (sum)
$O_1, 24$	$O_3, 32$	$O_2, 18$	$O_4, 66$
$O_2, 22$	$O_4, 30$	$O_1, 16$	$O_3, 60$
$O_4, 20$	$O_2, 18$	$O_5, 14$	$O_2, 58$
$O_3, 18$	$O_5, 16$	$O_4, 12$	$O_1, 54$
$O_5, 14$	$O_1, 14$	$O_3, 10$	$O_5, 44$

- O_4 has the highest aggregate score, even though it is not the highest in any of the lists
- O_2 completely dominates O_5 , and hence, O_5 may be pruned
- Brute-force algorithm (BF) looks at all objects and scores
 - Running time: $O(mn)$
- Lists may not fit into memory
 - Number of disk page accesses may be excessive
 - Running time impractical

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Fagin's algorithm (FA)

- Access each of the m sorted lists in a sequential manner
 - This is sequential I/O
- Continue the sorted accesses till at least k objects have been "seen fully", i.e., all the m scores for these k objects have been retrieved
- For all objects seen so far, perform random accesses to complete retrieval of their m scores
- Compute the aggregates for all objects thus retrieved
- Output the top- k list

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen :
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : O_{11} ,
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42}, O_{13},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42}, O_{13}, O_{41},$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42}, O_{13}, O_{41}, O_{22}$
- Objects fully seen :

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42}, O_{13}, O_{41}, O_{22}$
- Objects fully seen : O_2

Prev topic
Next page

Next topic

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

- Objects seen : $O_{11}, O_{32}, O_{23}, O_{21}, O_{42}, O_{13}, O_{41}, O_{22}$
- Objects fully seen : O_2
- Stop sorted access
- Need to complete objects O_1, O_3, O_4
- Randomly access: $O_{12}, O_{31}, O_{33}, O_{43}$
- Compute aggregates of O_1, O_2, O_3, O_4
- Output O_4

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Due to monotonicity property
 - O_5 in earlier example

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Due to monotonicity property
 - O_5 in earlier example
- Comparison with brute-force (BF) algorithm
- Assume a disk page contains one object and its associated score
- BF does m random accesses and $m(n-1)$ sequential accesses
- Denote cost of random access by c_r and sequential access by c_s
- In earlier example
 - Database access cost of BF is $3c_r + 12c_s$
 - Cost of FA is $7c_r + 5c_s$
- FA does lesser sequential accesses but more random accesses

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Due to monotonicity property
 - O_5 in earlier example
- Comparison with brute-force (BF) algorithm
- Assume a disk page contains one object and its associated score
- BF does m random accesses and $m(n-1)$ sequential accesses
- Denote cost of random access by c_r and sequential access by c_s
- In earlier example
 - Database access cost of BF is $3c_r + 12c_s$
 - Cost of FA is $7c_r + 5c_s$
- FA does lesser sequential accesses but more random accesses
- " Seen" list is long leading to many random accesses

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Due to monotonicity property
 - O_5 in earlier example
- Comparison with brute-force (BF) algorithm
- Assume a disk page contains one object and its associated score
- BF does m random accesses and $m(n-1)$ sequential accesses
- Denote cost of random access by c_r and sequential access by c_s
- In earlier example
 - Database access cost of BF is $3c_r + 12c_s$
 - Cost of FA is $7c_r + 5c_s$
- FA does lesser sequential accesses but more random accesses
- "Seen" list is long leading to many random accesses
- Finding k "full" objects may need deep probing into the lists

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Threshold algorithm (TA)

- Access each of the m sorted lists in a sequential manner
- As soon as an object is seen in a list, (if necessary) complete its retrieval by performing random accesses to other $m - 1$ lists
- Compute its aggregate score
- Retain if in current top- k list
- Maintain a **threshold score**
- Threshold score is aggregate of m scores last seen by sorted accesses on each of the lists
- Stop when threshold score falls below k th score in top- k list
- Output the top- k list

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score

Prev topic

Next topic

Prev page

Next page

Example: k = 1

List ₁	List ₂	List ₃
O ₁ , 24	O ₃ , 32	O ₂ , 18
O ₂ , 22	O ₄ , 30	O ₁ , 16
O ₄ , 20	O ₂ , 18	O ₅ , 14
O ₃ , 18	O ₅ , 16	O ₄ , 12
O ₅ , 14	O ₁ , 14	O ₃ , 10

Sorted	Random	Top-k list	Threshold score	k th score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O ₁	(O ₁ , O ₂)	(O ₁ , 54)	24 + 22 + 18 = 64	54
O ₂	(O ₁ , O ₂)	(O ₂ , 40)	24 + 22 + 16 = 62	40
O ₃	(O ₁ , O ₂)	(O ₃ , 32)	24 + 22 + 18 = 64	32
O ₄	(O ₁ , O ₂)	(O ₄ , 20)	24 + 22 + 12 = 58	20
O ₅	(O ₁ , O ₂)	(O ₅ , 14)	24 + 22 + 10 = 56	14
O ₁	(O ₃ , O ₄)	(O ₁ , 54)	32 + 30 + 18 = 80	54
O ₂	(O ₃ , O ₄)	(O ₂ , 40)	32 + 30 + 16 = 78	40
O ₃	no action	(O ₃ , 32)	22 + 20 + 18 = 60	32
O ₄	no action	(O ₄ , 20)	22 + 20 + 12 = 54	20
O ₅	no action	(O ₅ , 14)	22 + 20 + 10 = 52	14

Prev topic

Next topic

Prev page

Next page

Example: k = 1

List ₁	List ₂	List ₃
<i>O</i> ₁ , 24	<i>O</i> ₃ , 32	<i>O</i> ₂ , 18
<i>O</i> ₂ , 22	<i>O</i> ₄ , 30	<i>O</i> ₁ , 16
<i>O</i> ₄ , 20	<i>O</i> ₂ , 18	<i>O</i> ₅ , 14
<i>O</i> ₃ , 18	<i>O</i> ₅ , 16	<i>O</i> ₄ , 12
<i>O</i> ₅ , 14	<i>O</i> ₁ , 14	<i>O</i> ₃ , 10

Sorted	Random	Top- <i>k</i> list	Threshold score	<i>k</i> th score
		Φ	∞ + ∞ + ∞ = ∞	−∞
<i>O</i> ₁₁	(<i>O</i> ₁ , 24)	(<i>O</i> ₁ , 24)	24 + ∞ + ∞ = ∞	24
<i>O</i> ₁₂	(<i>O</i> ₂ , 22)	(<i>O</i> ₂ , 22)	22 + ∞ + ∞ = ∞	22
<i>O</i> ₁₃	(<i>O</i> ₃ , 18)	(<i>O</i> ₃ , 18)	18 + 24 + 22 = 64	18
<i>O</i> ₁₄	(<i>O</i> ₄ , 20)	(<i>O</i> ₄ , 20)	20 + 22 + 18 = 60	20
<i>O</i> ₁₅	(<i>O</i> ₅ , 14)	(<i>O</i> ₅ , 14)	14 + 22 + 18 = 54	14
<i>O</i> ₂₁	(<i>O</i> ₂ , 22)	(<i>O</i> ₂ , 22)	22 + 20 + 18 = 60	22
<i>O</i> ₂₂	(<i>O</i> ₄ , 20)	(<i>O</i> ₄ , 20)	20 + 22 + 18 = 60	20
<i>O</i> ₂₃	(<i>O</i> ₃ , 18)	(<i>O</i> ₃ , 18)	18 + 20 + 22 = 60	18
<i>O</i> ₂₄	(<i>O</i> ₅ , 14)	(<i>O</i> ₅ , 14)	14 + 20 + 22 = 56	14

Prev topic

Next topic

Prev page

Next page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 24)$	$24 + \infty + \infty = \infty$	24
O_{12}	O_{11}, O_{13}	$(O_2, 22)$	$22 + \infty + \infty = \infty$	22
O_{13}	O_{11}, O_{12}	$(O_4, 20)$	$24 + 22 + 20 = 66$	20
O_{21}	O_{11}, O_{13}	$(O_3, 18)$	$24 + 22 + 18 = 64$	18
O_{22}	O_{12}, O_{13}	$(O_5, 14)$	$20 + 18 + 14 = 52$	14
O_{23}	O_{11}, O_{12}	$(O_3, 18)$	$20 + 14 + 18 = 52$	18
O_{31}	O_{12}, O_{13}	$(O_1, 14)$	$14 + 18 + 16 = 48$	14
O_{32}	O_{11}, O_{13}	$(O_4, 12)$	$12 + 18 + 16 = 46$	12
O_{33}	O_{12}, O_{13}	$(O_5, 10)$	$12 + 14 + 16 = 42$	10

Prev topic

Next topic

Prev page

Next page

Example: k = 1

List ₁	List ₂	List ₃
<i>O</i> ₁ , 24	<i>O</i> ₃ , 32	<i>O</i> ₂ , 18
<i>O</i> ₂ , 22	<i>O</i> ₄ , 30	<i>O</i> ₁ , 16
<i>O</i> ₄ , 20	<i>O</i> ₂ , 18	<i>O</i> ₅ , 14
<i>O</i> ₃ , 18	<i>O</i> ₅ , 16	<i>O</i> ₄ , 12
<i>O</i> ₅ , 14	<i>O</i> ₁ , 14	<i>O</i> ₃ , 10

Sorted	Random	Top- <i>k</i> list	Threshold score	<i>k</i> th score
		Φ	∞ + ∞ + ∞ = ∞	−∞
<i>O</i> ₁₁	<i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₁ , 54)	24 + 32 + 18 = 74	24
<i>O</i> ₁₂	<i>O</i> ₁₁ , <i>O</i> ₁₃	(<i>O</i> ₂ , 54)	22 + 30 + 16 = 68	22
<i>O</i> ₁₃	<i>O</i> ₁₁ , <i>O</i> ₁₂	(<i>O</i> ₃ , 50)	18 + 30 + 18 = 66	18
<i>O</i> ₁₄	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₄ , 60)	20 + 18 + 22 = 60	20
<i>O</i> ₁₅	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₅ , 60)	14 + 16 + 30 = 60	14
<i>O</i> ₁₆	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₁ , 54)	24 + 32 + 18 = 74	24
<i>O</i> ₁₇	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₂ , 54)	22 + 30 + 16 = 68	22
<i>O</i> ₁₈	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₃ , 50)	18 + 30 + 18 = 66	18
<i>O</i> ₁₉	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₄ , 60)	20 + 18 + 22 = 60	20
<i>O</i> ₂₀	<i>O</i> ₁₁ , <i>O</i> ₁₂ , <i>O</i> ₁₃	(<i>O</i> ₅ , 60)	14 + 16 + 30 = 60	14

Prev topic
Next page

Next topic

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	∞
O_3	O_{12}, O_{13}	$(O_3, 32)$	$32 + \infty + \infty = \infty$	∞
O_2	O_{12}, O_{13}	$(O_2, 30)$	$30 + \infty + \infty = \infty$	∞
O_4	O_{12}, O_{13}	$(O_4, 18)$	$18 + \infty + \infty = \infty$	∞
O_5	O_{12}, O_{13}	$(O_5, 16)$	$16 + \infty + \infty = \infty$	∞
O_1	O_{12}, O_{13}	$(O_1, 14)$	$14 + \infty + \infty = \infty$	∞

Prev topic

Next topic

Prev page

Next page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{13}	O_{11}, O_{12}	$(O_2, 30)$	$22 + \infty + \infty = \infty$	∞
O_{12}	O_{13}, O_{11}	$(O_3, 32)$	$18 + \infty + \infty = \infty$	∞
O_{11}	O_{12}, O_{13}	$(O_4, 30)$	$20 + \infty + \infty = \infty$	∞
O_{12}	O_{13}, O_{11}	$(O_5, 16)$	$14 + \infty + \infty = \infty$	∞
O_{13}	O_{11}, O_{12}	$(O_1, 14)$	$24 + \infty + \infty = \infty$	∞
O_{11}	O_{12}, O_{13}	$(O_2, 18)$	$22 + \infty + \infty = \infty$	∞
O_{12}	O_{13}, O_{11}	$(O_3, 18)$	$32 + \infty + \infty = \infty$	∞
O_{13}	O_{11}, O_{12}	$(O_4, 12)$	$30 + \infty + \infty = \infty$	∞
O_{11}	O_{12}, O_{13}	$(O_5, 14)$	$16 + \infty + \infty = \infty$	∞

Prev topic
Next page

Next topic

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_3	O_{11}, O_{12}	$(O_3, 32)$	$32 + \infty + \infty = \infty$	∞
O_2	O_{11}, O_{12}	$(O_2, 22)$	$22 + \infty + \infty = \infty$	∞
O_4	O_{11}, O_{12}	$(O_4, 20)$	$20 + \infty + \infty = \infty$	∞
O_5	O_{11}, O_{12}	$(O_5, 14)$	$14 + \infty + \infty = \infty$	∞
O_1	O_{12}, O_{13}	$(O_1, 14)$	$14 + \infty + \infty = \infty$	∞
O_3	O_{12}, O_{13}	$(O_3, 10)$	$10 + \infty + \infty = \infty$	∞
O_2	O_{12}, O_{13}	$(O_2, 18)$	$18 + \infty + \infty = \infty$	∞
O_4	O_{12}, O_{13}	$(O_4, 12)$	$12 + \infty + \infty = \infty$	∞
O_5	O_{12}, O_{13}	$(O_5, 16)$	$16 + \infty + \infty = \infty$	∞
O_1	O_{12}, O_{13}	$(O_1, 16)$	$16 + \infty + \infty = \infty$	∞
O_3	O_{12}, O_{13}	$(O_3, 18)$	$18 + \infty + \infty = \infty$	∞
O_2	O_{12}, O_{13}	$(O_2, 22)$	$22 + \infty + \infty = \infty$	∞
O_4	O_{12}, O_{13}	$(O_4, 20)$	$20 + \infty + \infty = \infty$	∞
O_5	O_{12}, O_{13}	$(O_5, 14)$	$14 + \infty + \infty = \infty$	∞

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_3	O_{21}, O_{22}	$(O_2, 60)$	$22 + 32 + 10 = 74$	60
O_{12}	O_{11}, O_{13}	$(O_1, 54)$	$24 + 22 + \infty = 74$	60
O_{13}	O_{11}, O_{12}	$(O_1, 54)$	$24 + 30 + \infty = 74$	60
O_{21}	O_{22}, O_{23}	$(O_2, 60)$	$20 + 30 + 10 = 70$	60
O_{22}	O_{21}, O_{23}	$(O_2, 60)$	$24 + 20 + 10 = 70$	60
O_{23}	O_{21}, O_{22}	$(O_2, 60)$	$20 + 20 + 10 = 70$	60

Prev topic

Next topic

Prev page

Next page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{12}	O_{11}, O_{13}	$(O_1, 54)$	$24 + 32 + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{12}	O_{11}, O_{13}	$(O_1, 54)$	$24 + 32 + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60

Prev topic

Next topic

Prev page

Next page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{21}	no action	$(O_3, 60)$	$22 + 32 + 18 = 72$	60
O_{42}	O_{41}, O_{43}	$(O_4, 66)$	$22 + 32 + 18 = 72$	60
O_{41}	no action	$(O_4, 66)$	$20 + 32 + 18 = 70$	60
O_{43}	no action	$(O_4, 66)$	$20 + 32 + 18 = 70$	60

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{21}	no action	$(O_3, 60)$	$22 + 32 + 18 = 72$	60
O_{42}	O_{41}, O_{43}	$(O_4, 66)$	$22 + 30 + 18 = 70$	66
O_{41}	no action	$(O_4, 66)$	$20 + 32 + 18 = 70$	66
O_{43}	no action	$(O_4, 66)$	$20 + 30 + 18 = 68$	66

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{21}	no action	$(O_3, 60)$	$22 + 32 + 18 = 72$	60
O_{42}	O_{41}, O_{43}	$(O_4, 66)$	$22 + 30 + 18 = 70$	66
O_{13}	no action	$(O_4, 66)$	$22 + 30 + 16 = 68$	66

Prev topic

Next topic

Prev page

Next page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{21}	no action	$(O_3, 60)$	$22 + 32 + 18 = 72$	60
O_{42}	O_{41}, O_{43}	$(O_4, 66)$	$22 + 30 + 18 = 70$	66
O_{13}	no action	$(O_4, 66)$	$22 + 30 + 16 = 68$	66
O_{41}	no action	$(O_4, 66)$	$20 + 30 + 16 = 66$	66

Prev topic

Next topic

Next page

Prev page

Example: $k = 1$

List ₁	List ₂	List ₃
$O_1, 24$	$O_3, 32$	$O_2, 18$
$O_2, 22$	$O_4, 30$	$O_1, 16$
$O_4, 20$	$O_2, 18$	$O_5, 14$
$O_3, 18$	$O_5, 16$	$O_4, 12$
$O_5, 14$	$O_1, 14$	$O_3, 10$

Sorted	Random	Top- k list	Threshold score	k^{th} score
		Φ	$\infty + \infty + \infty = \infty$	$-\infty$
O_{11}	O_{12}, O_{13}	$(O_1, 54)$	$24 + \infty + \infty = \infty$	54
O_{32}	O_{31}, O_{33}	$(O_3, 60)$	$24 + 32 + \infty = \infty$	60
O_{23}	O_{21}, O_{22}	$(O_3, 60)$	$24 + 32 + 18 = 74$	60
O_{21}	no action	$(O_3, 60)$	$22 + 32 + 18 = 72$	60
O_{42}	O_{41}, O_{43}	$(O_4, 66)$	$22 + 30 + 18 = 70$	66
O_{13}	no action	$(O_4, 66)$	$22 + 30 + 16 = 68$	66
O_{41}	no action	$(O_4, 66)$	$20 + 30 + 16 = 66$	66

Stop; Output O_4

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Dominated by threshold score
- Monotonicity property

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

[Prev topic](#)[Next topic](#)[Prev page](#)[Next page](#)

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Dominated by threshold score
- Monotonicity property
- **Instance optimal**
- For any algorithm that does not make random guesses and any sorted list, cost of TA has the same order as the best algorithm
 - Best for any instance (any database, any k , etc.)
- FA is optimal with a high probability

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Dominated by threshold score
- Monotonicity property
- Instance optimal
- For any algorithm that does not make random guesses and any sorted list, cost of TA has the same order as the best algorithm
 - Best for any instance (any database, any k , etc.)
- FA is optimal with a high probability
- Comparison
- In earlier example
 - Cost of TA is $11c_r + 4c_s$

Module 8: Miscellaneous Topics

Lecture 37: Aggregation

Prev topic

Next topic

Prev page

Next page

Analysis

- Correct
- Any object not seen cannot be in top- k list
- Dominated by threshold score
- Monotonicity property
- Instance optimal
- For any algorithm that does not make random guesses and any sorted list, cost of TA has the same order as the best algorithm
 - Best for any instance (any database, any k , etc.)
- FA is optimal with a high probability
- Comparison
- In earlier example
 - Cost of TA is $11c_r + 4c_s$
- Good when m is small; suffers many random accesses when m is large