Promotion Analysis in Multi-Dimensional Space

Tianyi Wu (UIUC)
Dong Xin (Microsoft Research)
Qiaozhu Mei (University of Michigan)
Jiawei Han (UIUC)

Outline

- Introduction
- Query execution algorithms
- Spurious promotion
- Experiment
- Conclusion

Outline

- Introduction
- Query execution algorithms
- Spurious promotion
- Experiment
- Conclusion

Promotion analysis: introduction

- Formulate and study a useful function
  - Promotion analysis through ranking
  - General goal: promote a given object by leveraging subspace ranking
- Motivating example
  - A marketing manager of a book retailer
  - Basic fact
    - Book sales: 30^{th} out of 100 other retailers
    - Not particularly interesting!
  - After promotion analysis, he discovered:
    - Ranked 1st in the {college students, science and technology} area
    - Further advertising and marketing decisions
- Another example: person promotion

Promotion query

Observation

Global rank	May not be interesting	**Local rank**	Can be more interesting
Full-space	Compare to all other objects in all aspects	**Subspaces**	Compare objects in certain areas
Low cost	Single SQL query	**High cost**	Many subspaces

THE PROMOTION QUERY PROBLEM

Given: an object (e.g., product, person)

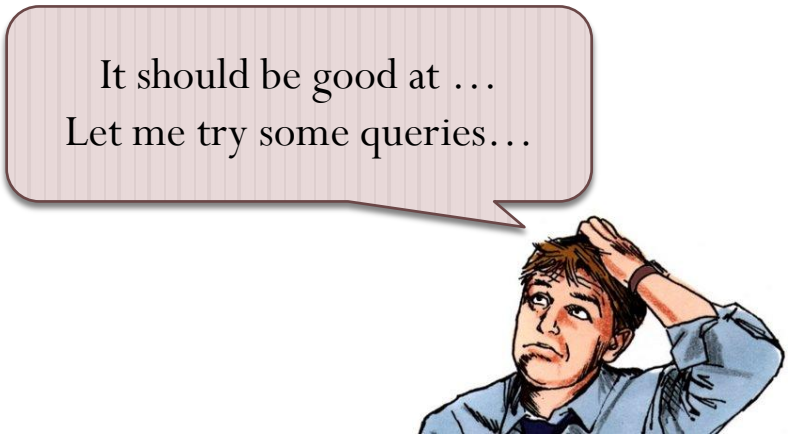
Goal: discover the most interesting subspaces where the object is highly ranked

Subspace rank: why interesting

- Discover merit and competitive strengths
  - *E.g., a bestselling car model among hybrid cars*
- Enhance image
  - *E.g., fortune 500 company*
- Facilitate decision making
  - *E.g., marketing plan that focuses on college students*
- Deliver specific information
  - *E.g.,"top-3 university in biomedical research"vs."top-20 university"*
- Extensively practiced in marketing
  - *Market segmentation*
  - *Customer targeting and product positioning*

Challenges

- Current systems
  - Given a condition, find top-*k* objects
  - Sophisticated early termination and pruning algorithms
- Promotion query: not well-supported
  - User: manual search and navigation
  - Trial-and-error
- Computationally expensive
  - The rank measure: holistic
  - A blow-up of subspaces

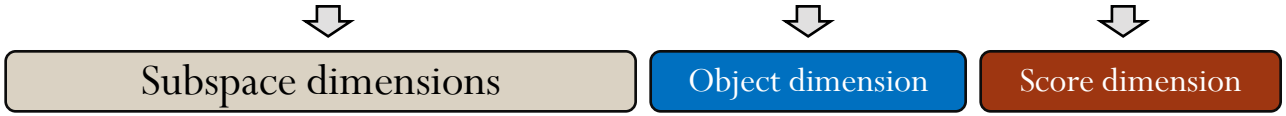


Promotion analysis

Multidimensional data model

- Fact table

Location	Time	Object	Score
Lyon	July	T	0.5
Chicago	July	T	0.8
Chicago	August	S	1.0
Chicago	July	S	1.0
Lyon	August	V	0.3
Chicago	August	V	0.6
Chicago	July	V	0.7

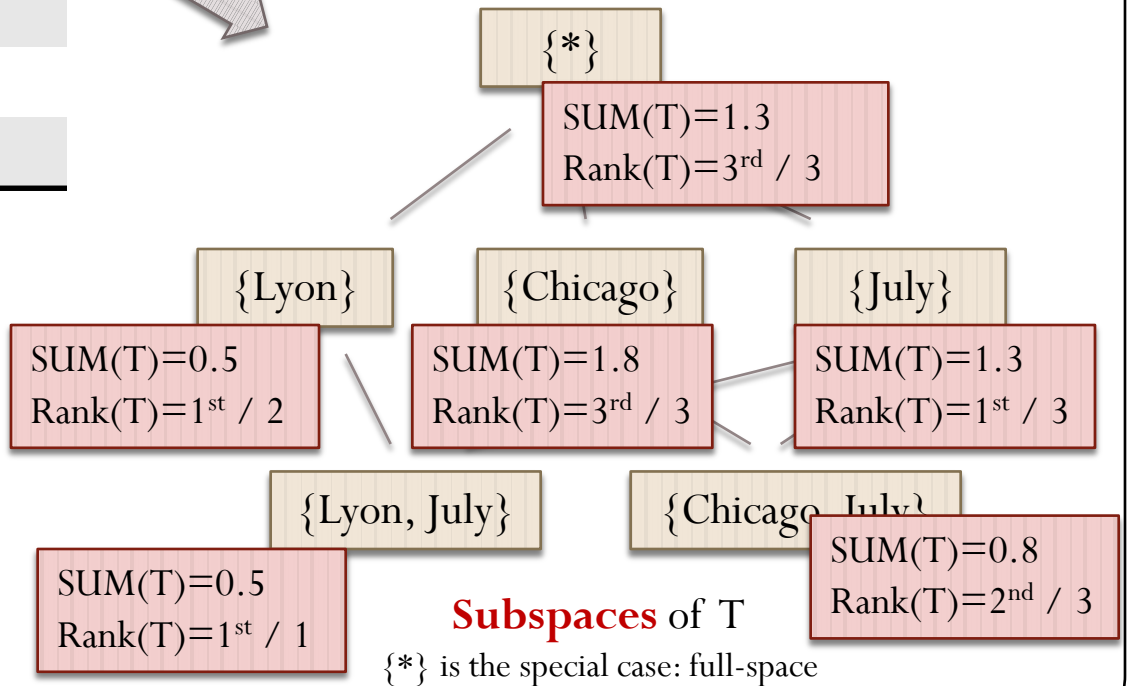


Subspaces

Location	Time	Object	Score
Lyon	July	T	0.5
Chicago	July	T	0.8
Chicago	August	S	1.0
Chicago	July	S	1.0
Lyon	August	V	0.3
Chicago	August	V	0.6
Chicago	July	V	0.7

Given a target object T

Aggregate and compute the target object's rank in each subspace.



Subspaces of T

{*} is the special case: full-space

Query model

- Given a target object T, find the top subspaces which are promotive
- "Promotiveness" : a class of measures to quantify how well a subspace S can promote T
  - $P(S,T) = f(Rank(S,T)) * g(Sig(S))$
    - Higher rank ~ more promotive
    - More significant subspace (e.g., more objects) ~ more promotive
  - Example instantiations
    - Simple ranking: $P(S,T) = Rank^{-1}(S,T)$
    - Iceberg condition: $P(S,T) = Rank^{-1}(S,T) * I(ObjCount(S) > MinSig)$
    - Percentile ranking: $P(S,T) = ObjCount(S) \ / \ Rank(S,T)$
    - …

Query model

- Given a target object T, find the top subspaces which are promotive
- "Promotiveness" : a class of measures to quantify how well a

THE PROMOTION QUERY PROBLEM

Input: a target object T

Output: top-R subspaces with the largest $P(S,T)$ scores

/* assume simple ranking */

  - More significant subspace (e.g., more objects) → more promotive
  - Example instantiations
    - Simple ranking: $P(S,T) = Rank^{-1}(S,T)$
    - Iceberg condition: $P(S,T) = Rank^{-1}(S,T) * I(ObjCount(S) > MinSig)$
    - Percentile ranking: $P(S,T) = ObjCount(S) \ / \ Rank(S,T)$
    - …

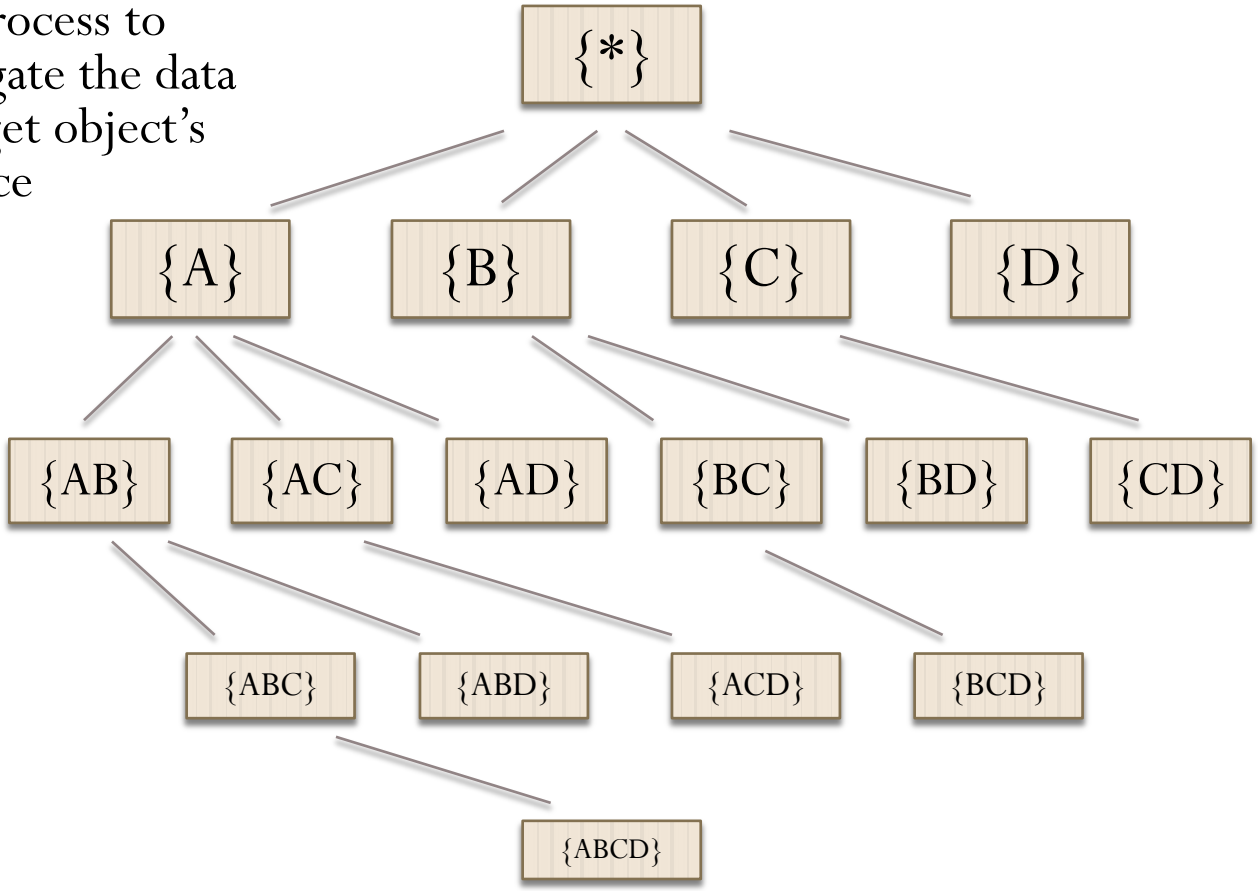
Outline

- Introduction
- Query execution algorithms
  - (1) PromoRank framework
    - (a) Subspace pruning
    - (b) Object pruning
  - (2) Promotion cubes
- Spurious promotion
- Experiment
- Conclusion

The PromoRank framework

Idea: use a recursive process to partition and aggregate the data to compute the target object's rank in each subspace

[Beyer99] The bottom-up method



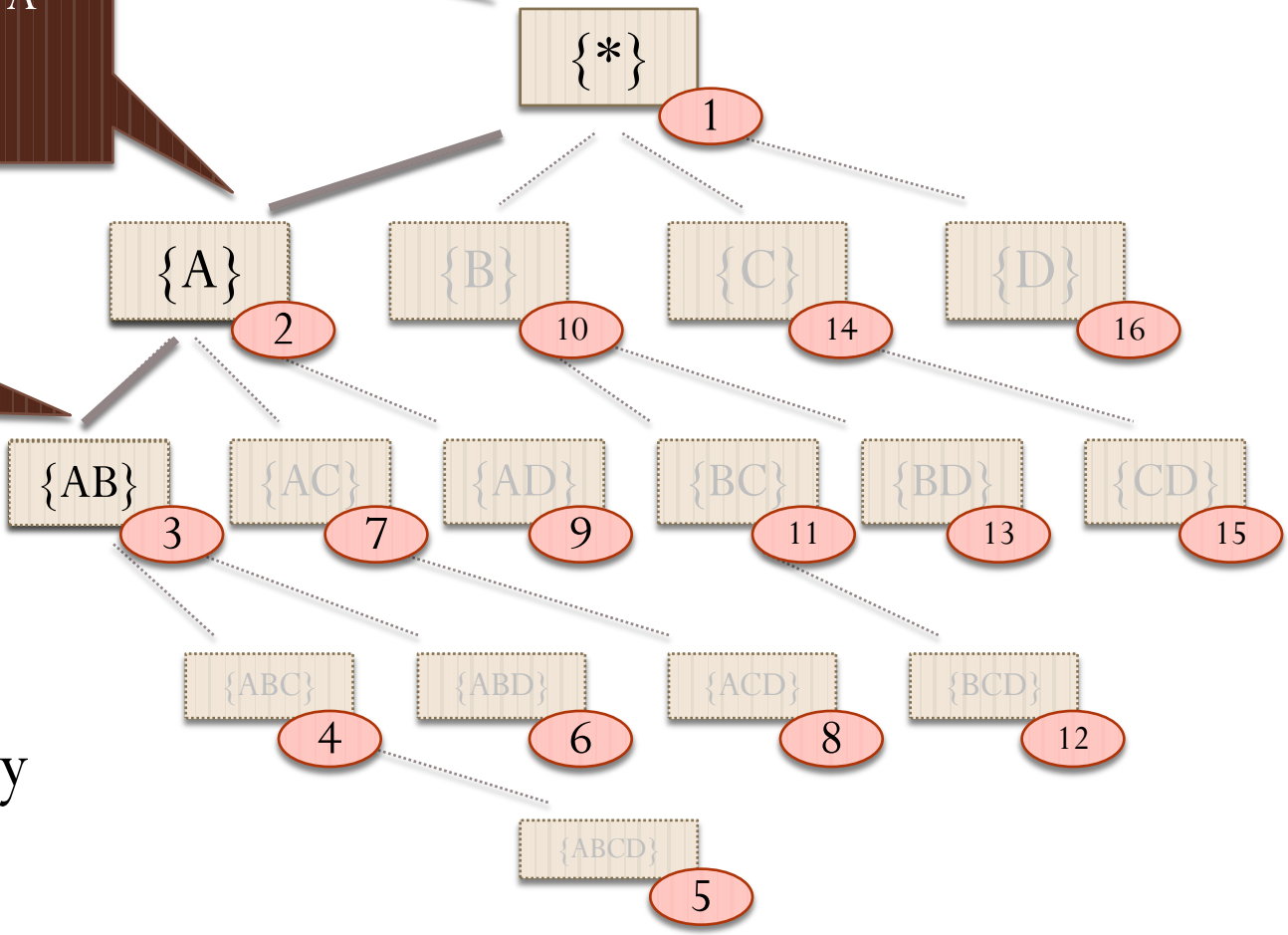
Target object's subspace lattice

ursive process

Compute T's rank in {*}
Method: create a hash table:
HashTable[object] = AggregateScore

Partition the data based on A
Method: sorting
Compute T's rank in {A}

Recursively repeat…



Top-R promotive subspaces: priority queue

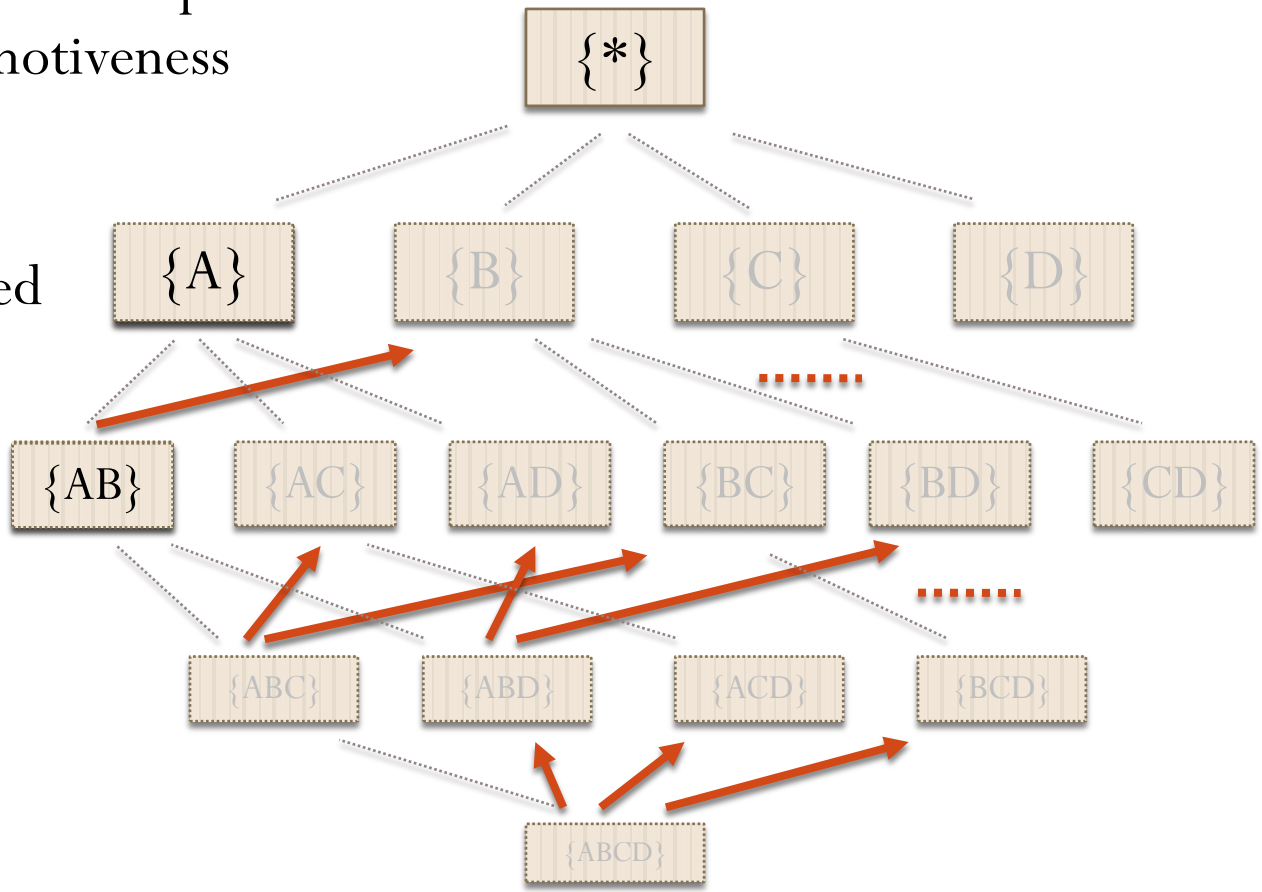
(1.1) Subspace pruning

- Idea: reuse previous results
- Goal: prune out unseen subspaces

by bounding their promotiveness scores

Sig(S): bounded

Rank(S, T) : bounded



Subspace pruning

- Keys:
  - Compute T's highest possible Rank: LBRank
  - Use the monotonicity of the aggregate measure (e.g. SUM, MAX)

Any unseen subspace with low LBRank(T) can be pruned

Thus, LBRank(T) = | {V, S} | + 1 = 3rd

SUM(V) > SUM(T)
SUM(S) > SUM(T)

How to prune an unseen one? {B} 10

SUM(T) = 1.9

SUM(V) = 5.5
SUM(S) = 2.2

Given a seen (aggregated) subspace {AB} 3

SUM(V) = 5.5
SUM(S) = 2.2
SUM(T) = 1.1
Rank(T) = 3rd / 3

(1.2) Object pruning

Idea: avoid computing objects which do not affect rank

Goal: reduce the partitioning and aggregation cost

Power-law distribution: objects at the long-tail can be pruned

W and Z can be pruned!

SUM(S) = 6.5
SUM(T) = 2.2
SUM(U) = 1.5
SUM(W) = 1.0
SUM(Z) = 0.8

SUM(W)<MinScore(T)
SUM(Z)<MinScore(T)

Seen (aggregated) subspace {A}

Unseen subtree of subspaces

{AB} {AC}

SUM(T) = 1.2

SUM(T) = 1.9

MinScore(T) = 1.1

{ABC}

SUM(T) = 1.1

(2) Promotion cubes

Observation:
(1) T: tends to be highly ranked in a top subspace;
(2) A top subspace is likely to contain many objects

- Method: promotion cube
  - Offline materialization
  - Structure
    - For each subspace with *Sig(S)>MinSig*
      - parameter: *MinSig*
    - Materialize a selected sample of top-*k* aggregate scores in each subspace
      - Parameter(s): *k* and *k'*

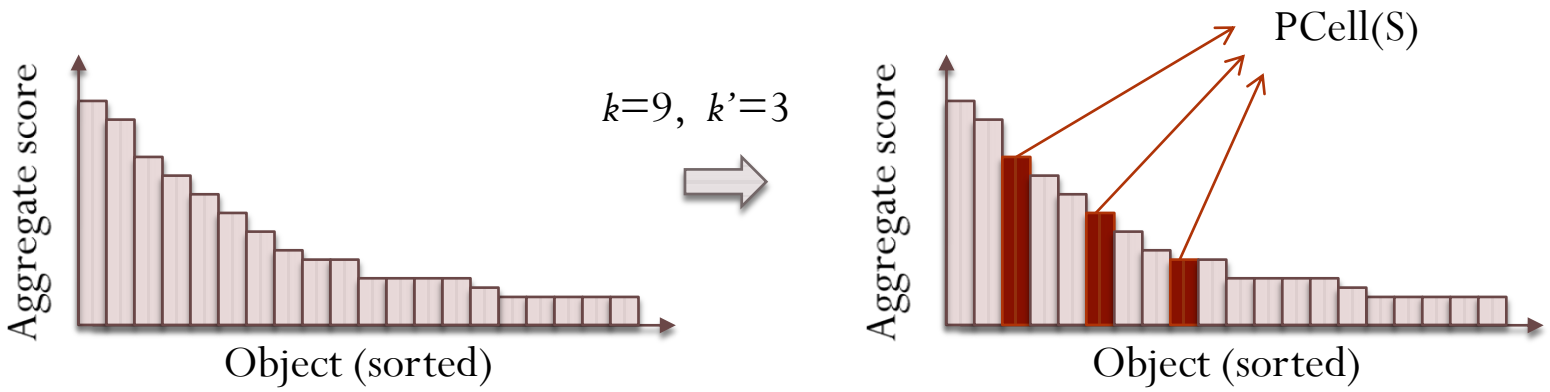
Promotion cell

- For each "significant" subspace S, create a "promotion cell"

Subspace S

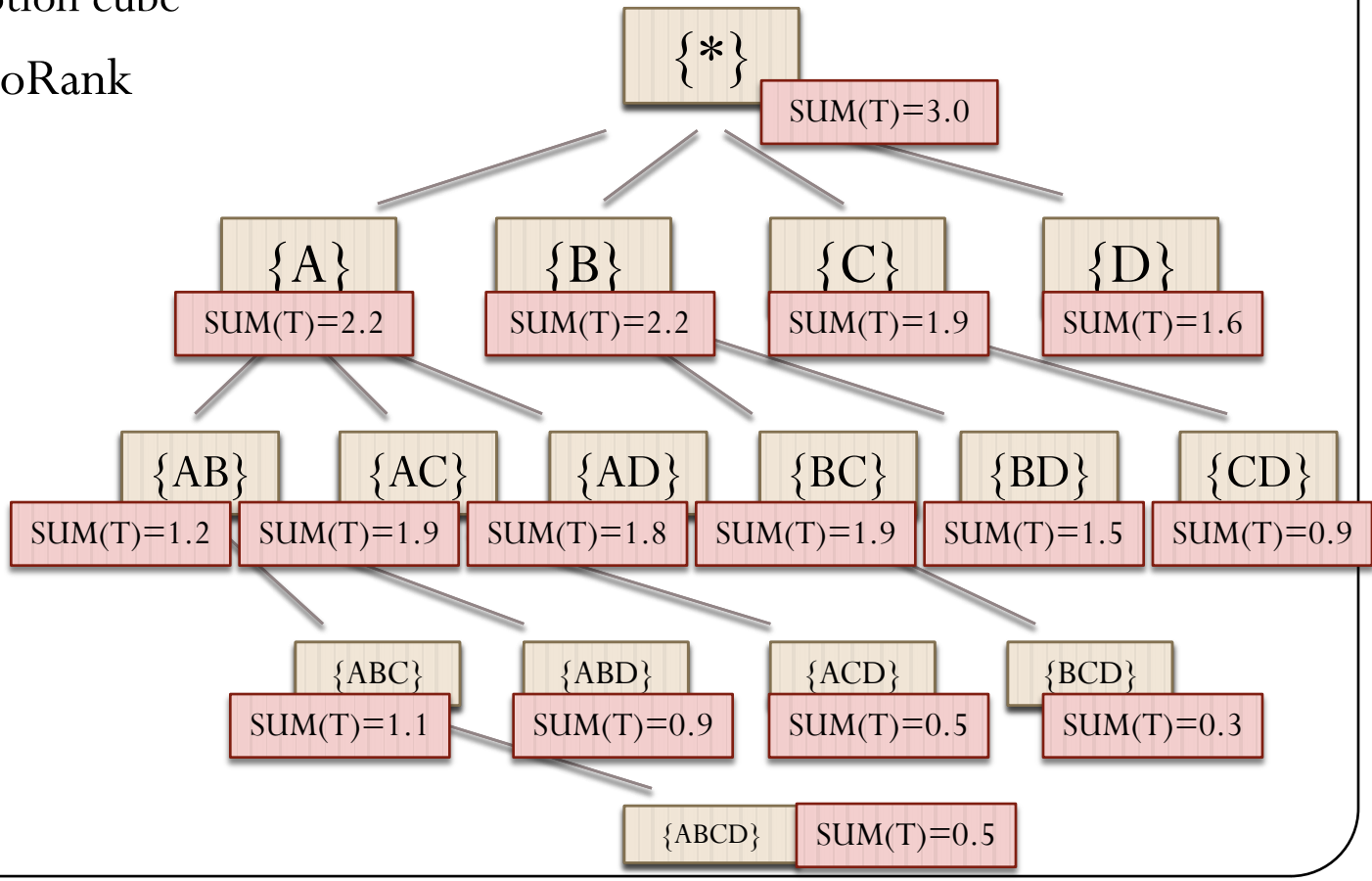
Passing the *MinSig* threshold

➢Promotion cell:
- ➢Store aggregate scores; no object IDs
- ➢Parameters *MinSig*, *k*, and *k'*: chosen to yield a space-time tradeoff; application dependent
- ➢Does not restrict query processing



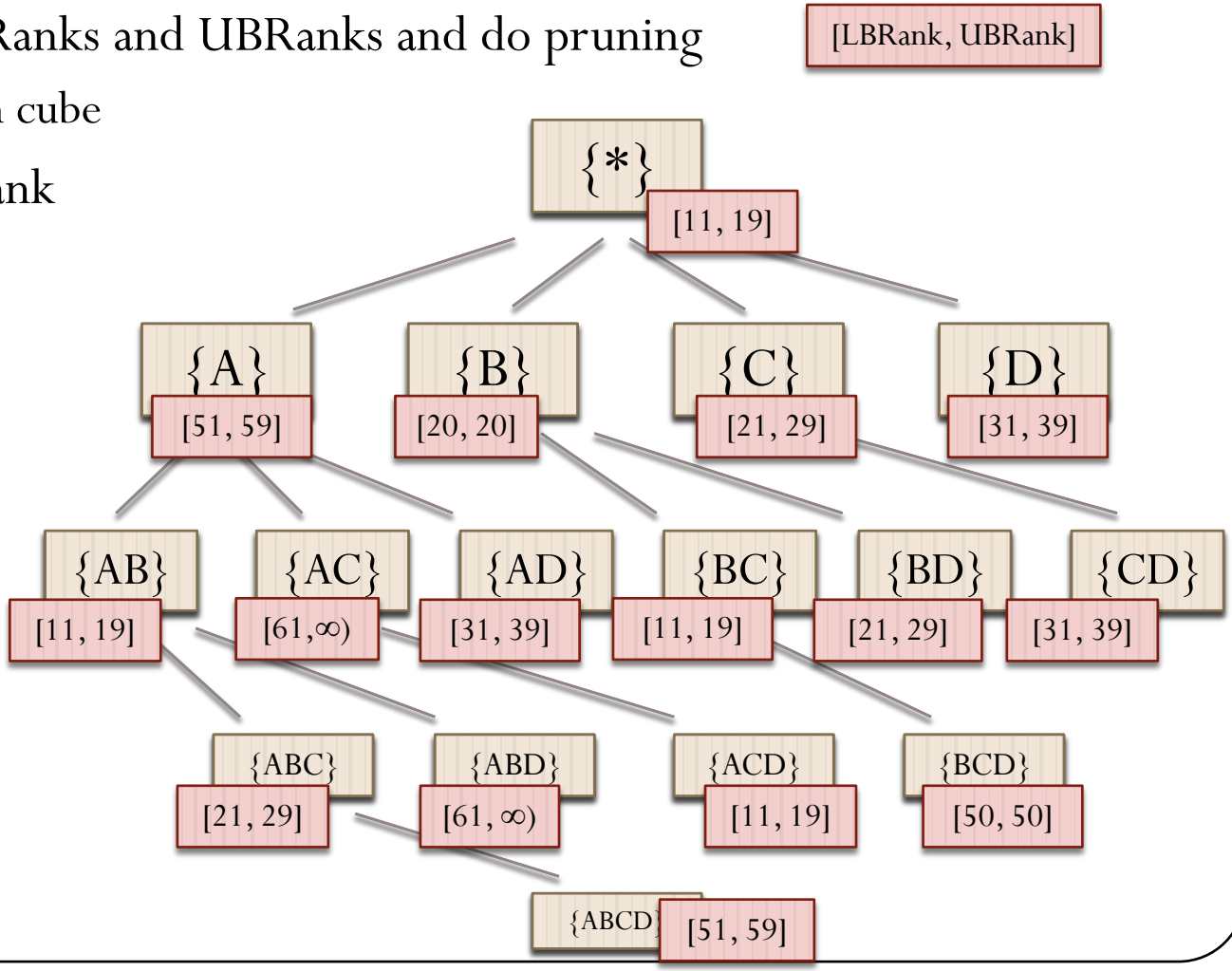
Query execution using promotion cube

- Step 1: Compute T's aggregate scores
- Step 2: Compute LBRanks and UBRanks and do pruning
  - Using the promotion cube
- Step 3: Call PromoRank



Query execution using promotion cube

- Step 1: Compute T's aggregate scores
- Step 2: Compute LBRanks and UBRanks and do pruning
  - Using the promotion cube
- Step 3: Call PromoRank



Outline

- Introduction
- Query execution algorithms
- Spurious promotion
- Experiment
- Conclusion

The spurious promotion problem

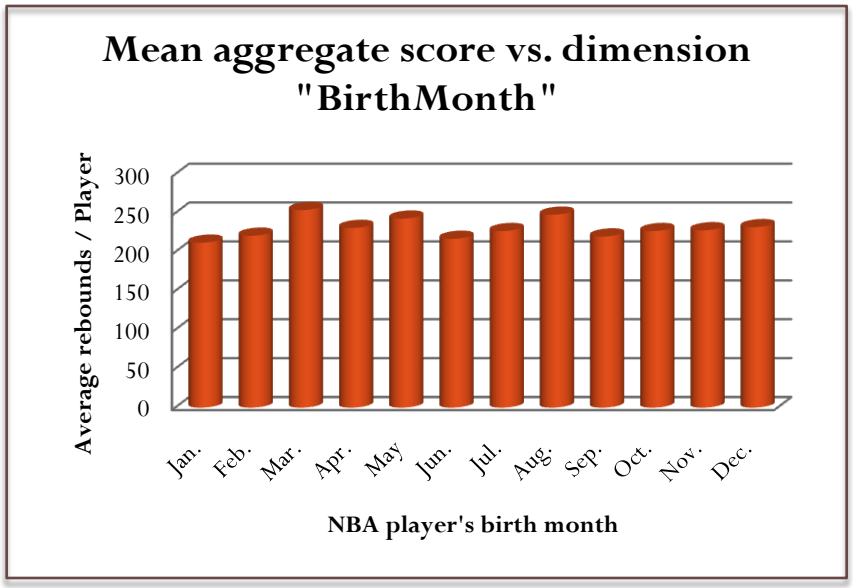
- Spurious promotion
  - The target object is highly ranked in a subspace due to random perturbation: not meaningful
- Example: Michael Jordan (NBA player)

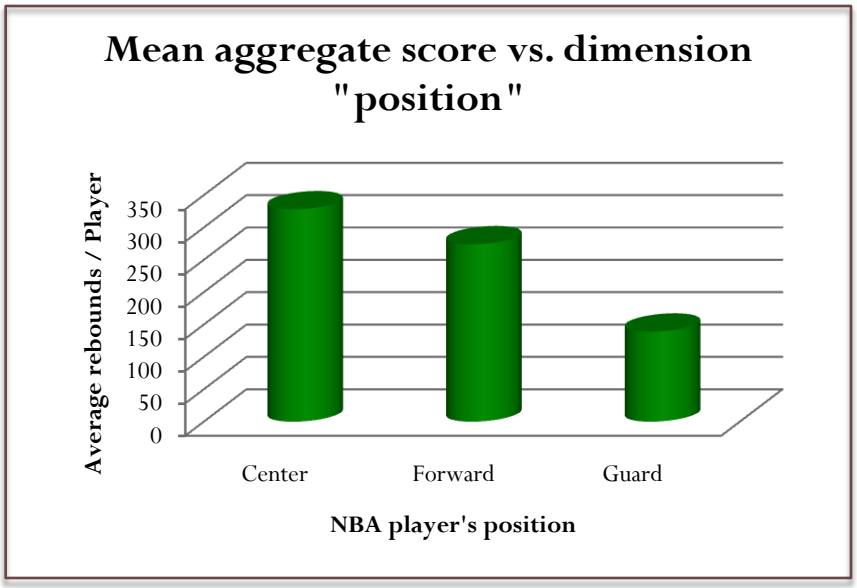
Rank	Subspace	
# 1	{Year = 1995}	OK
# 1	{MonthOfBirth = February}	Spurious
# 1	{Weather = Sunny}	Spurious

Due to random perturbation

Avoid spurious promotion

- How to avoid such meaningless subspaces?
- Observation
  - Spuriously promotive dimension: mean aggregate scores tend to be similar across different dimension values

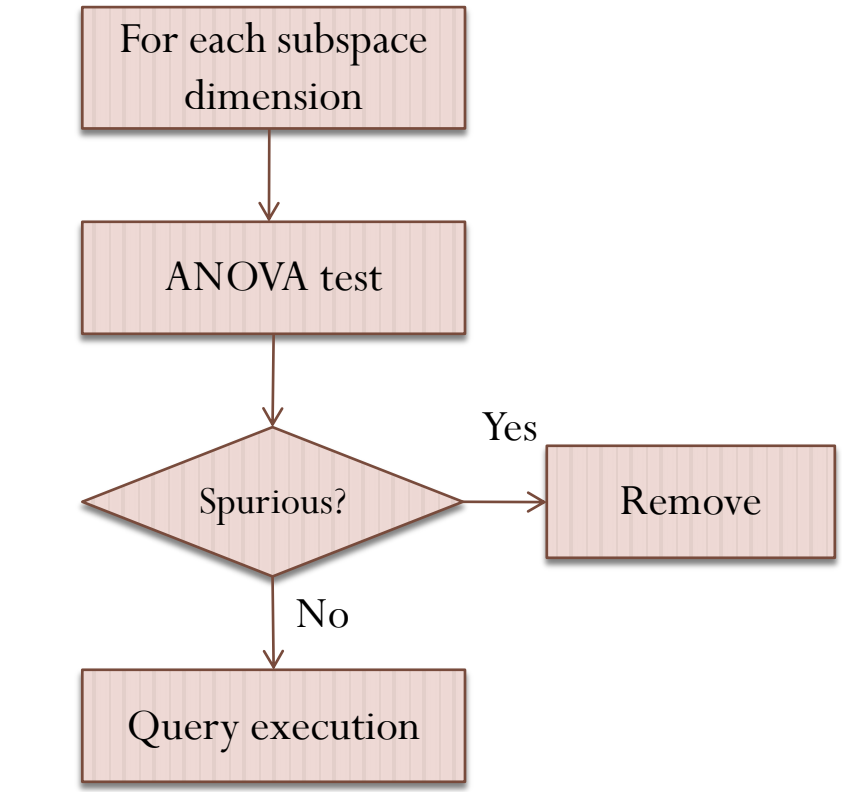




Preprocessing to filter out spurious dimensions

Method:

- ANOVA (analysis of variance) test
- Given a subspace dimension A
  - $|A|$ groups of scores
  - Between-group sum of squared deviation $SS_B = \sum_i \frac{\sigma_i}{size_i} - \frac{(\sum_i \sigma_i)^2}{n}$
  - Within-group sum of squared deviation $SS_W = \sum_i \sum_j (s_j^i - \mu_i)^2$
  - **F-ratio**$(A) = SS_B / SS_W$
  - **F-ratio** too small: H_0 rejected; no correlation with score.



Top-R non-spurious subspaces

Outline

- Introduction
- Query execution algorithms
- Spurious promotion
- Experiment
- Conclusion

Experiment

- Evaluation
  - Effectiveness (case study)
  - Efficiency (space-time tradeoff)
- Data sets
  - NBA
  - DBLP
  - TPC-H
- Methods
  - PromoRank
  - PromoRank++ (with the pruning methods)
  - PromoCube
- Implementation
  - Pentium 3GHz CPU / 2G memory
  - WinXP / Microsoft Visual C# 2008 (in-memory)

DBLP data set

- Subspace dimensions
  - *Conference* (2,506)
  - *Year* (50)
  - *Database* (boolean)
  - *Data mining* (boolean)
  - *Information retrieval* (boolean)
  - *Machine learning* (boolean)

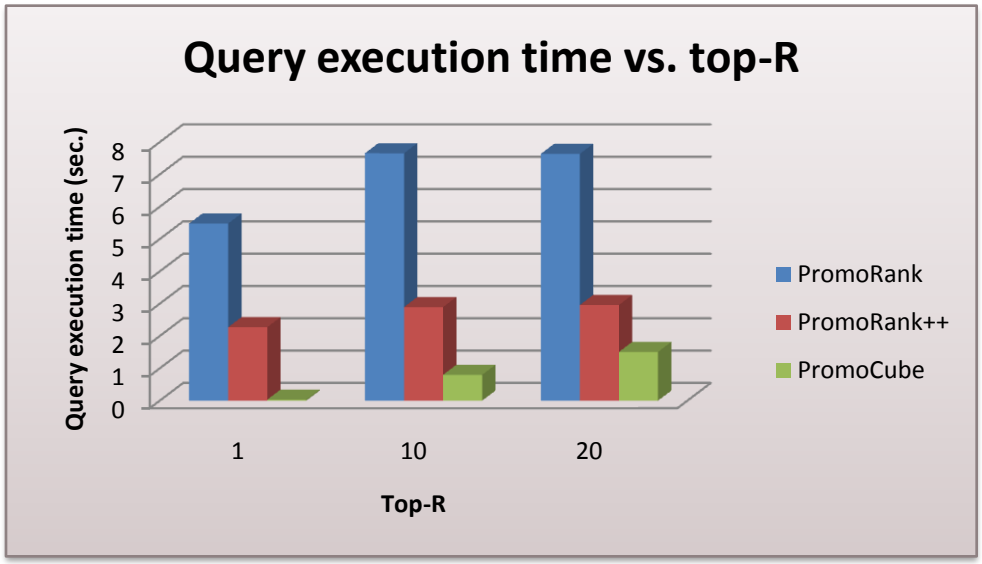
  } From *title*
- Object dimension: *Author* (*450K*)
- Score dimension: *Paper count*
- Base tuples (1.76M)

A case study on DBLP

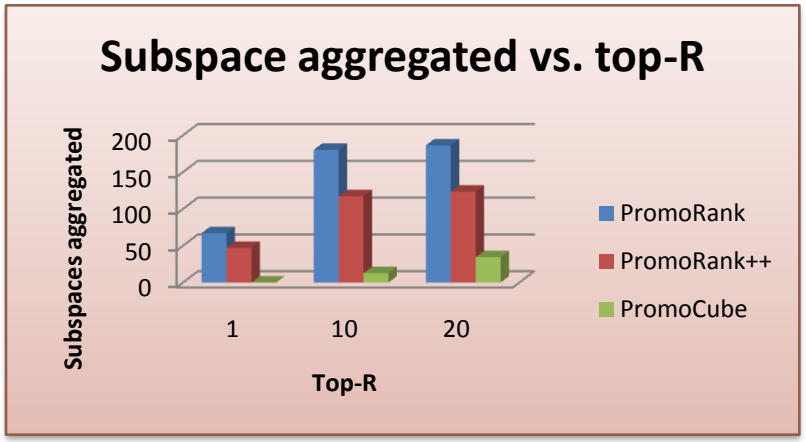
Query object	Top-3 subspaces	Rank	Authors	Top-%
David Dewitt	{*}	376th	451,316	0.08%
	{Database}	16th	65,321	0.02%
	{1990}	2nd	13,170	0.02%
	{SIGMOD}	2nd	3,519	0.06%
Yufei Tao	{*}	3325th	451,316	0.74%
	{Database, 2003}	11th	6,707	0.16%
	{Database, 2004}	18th	8,877	0.20%
	{ICDE}	30th	4,822	0.62%

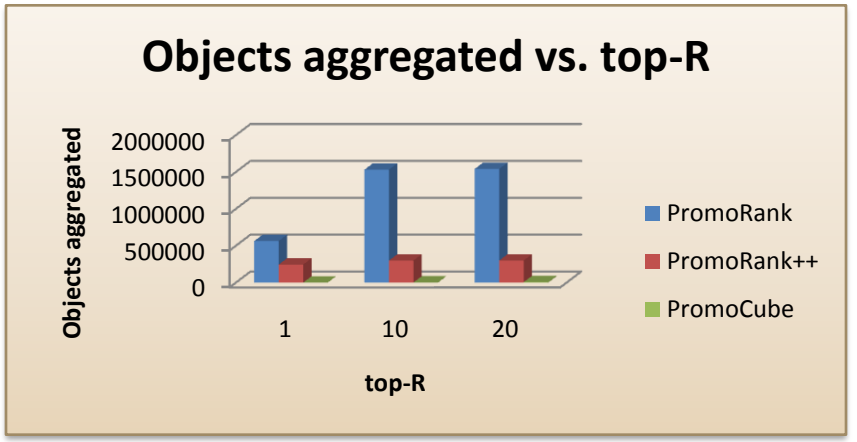
Promotiveness measure decided by rank and a penalty for small subspace

Query execution time (DBLP)



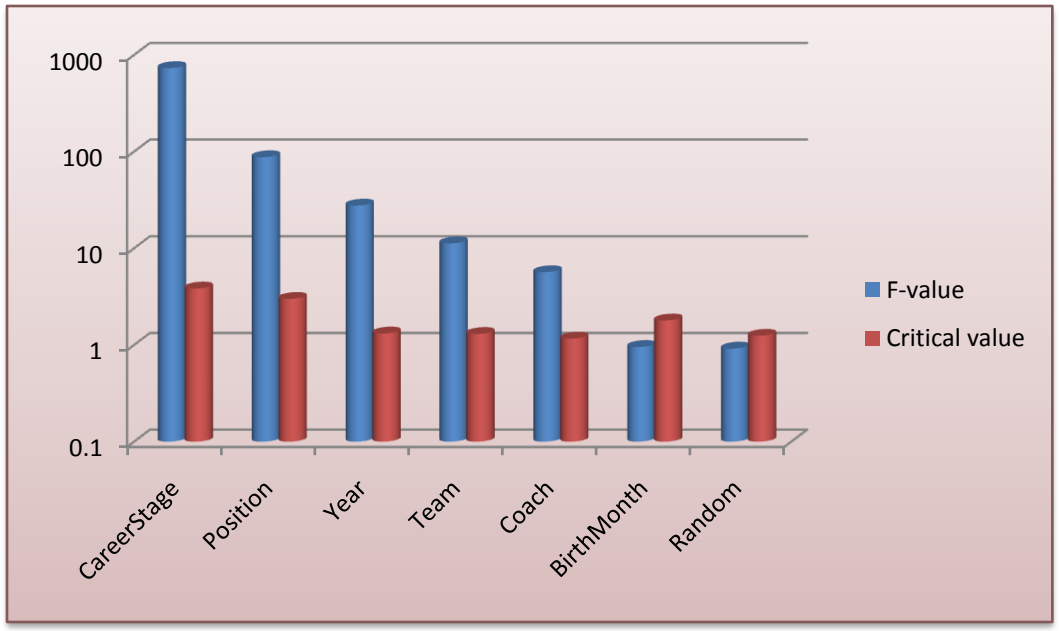
Promotion cube = 310KB
(most aggregate scores are small integers)





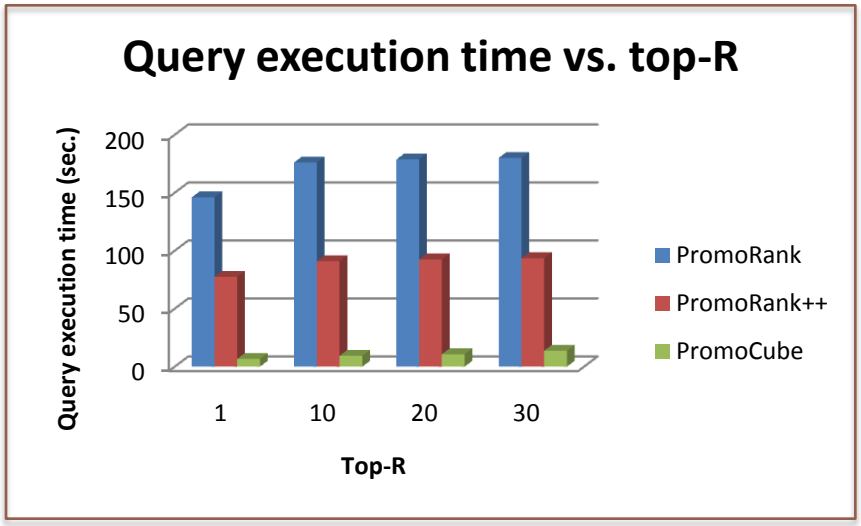
ANOVA test: effectiveness

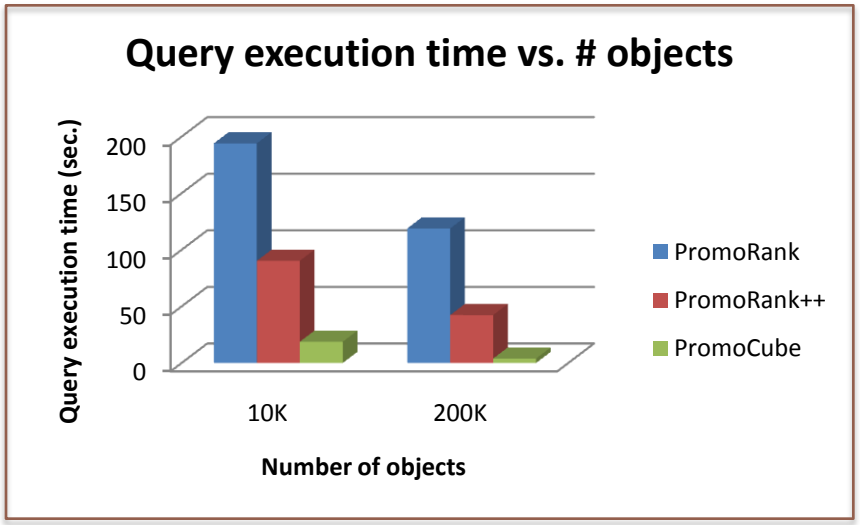
- NBA data
  - 3,460 *players* (objects)
  - *Rebounds* (score)
  - 18,050 base tuples



TPCH benchmark

- 6M tuples
- 6 subspace dimensions
- 10,000 objects
- Promotion cube
  - $k = 1000$, $k' = 8$
  - Size < 1MB





Outline

- Introduction
- Query execution algorithms
- Spurious promotion
- Experiment
- Conclusion

Conclusion

- Promotion analysis: a new direction
  - Search-based advertising
    - *[Borgs WWW 07] Dynamics of bid optimization in online advertisement auctions*
  - Data mining for marketing
    - *[Kleinberg DMKD 98] A microeconomic view of data mining*
  - Finding top-*k* attributes
    - *[Das SIGMOD 06] Ordering the attributes of query results*
    - *[Miah ICDE 08] Standing out in a crowd: Selecting attributes for maximum visibility*
  - Skyline queries
- Future
  - Application: social networks, recommender systems, …
  - Data model: links, textual data, numerical, …

Thank you!

Any questions?

Promotion Analysis in Multi-Dimensional Space

Presenter: **Tianyi Wu**
University of Illinois at Urbana-Champaign