Optimal Spatial Dominance: An Effective Search of Nearest Neighbor Candidates

XIAOYANG WANG[1], YING ZHANG[2], WENJIE ZHANG[1], XUEMIN LIN[1], MUHAMMAD AAMIR CHEEMA[3]

1. THE UNIVERSITY OF NEW SOUTH WALES, AUSTRALIA
2. UNIVERSITY OF TECHNOLOGY, SYDNEY, AUSTRALIA
3. MONASH UNIVERSITY, AUSTRALIA

UNSW
THE UNIVERSITY OF NEW SOUTH WALES

UNIVERSITY OF TECHNOLOGY SYDNEY

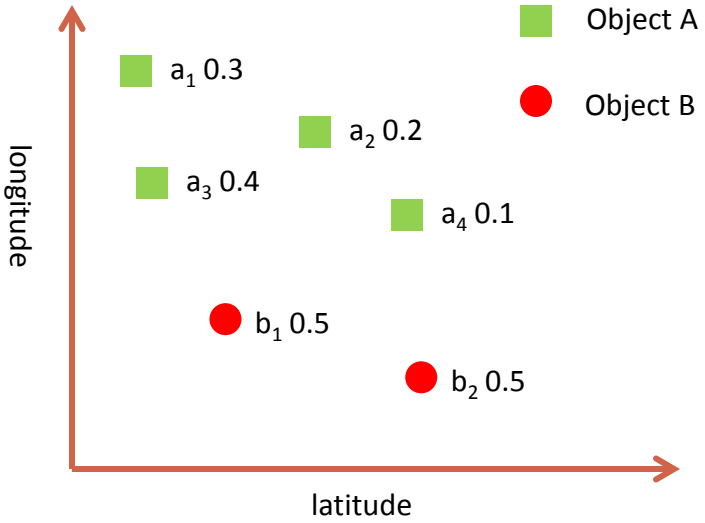
MONASH University

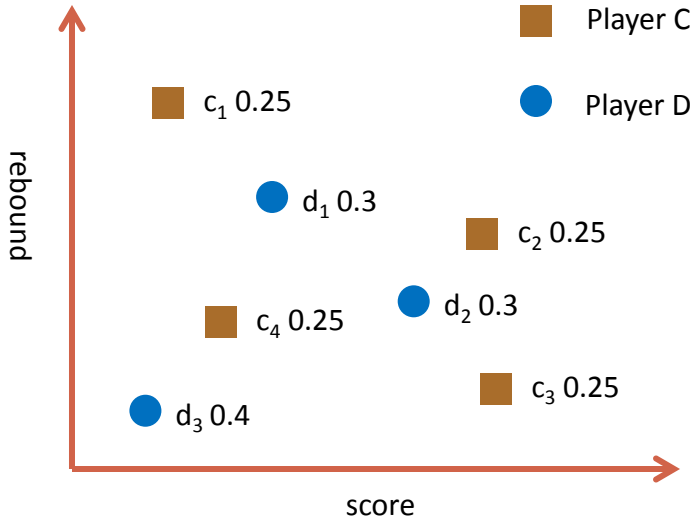
Outline

- Introduction
- Related Work
- Spatial Dominance Operators
- Experiments
- Conclusion

Introduction

- Objects with multiple instances are widely used.
  - Uncertain object (instances are exclusive), e.g., uncertain spatial objects.
  - Multi-valued object (instances are co-occurrence), e.g., NBA player records.

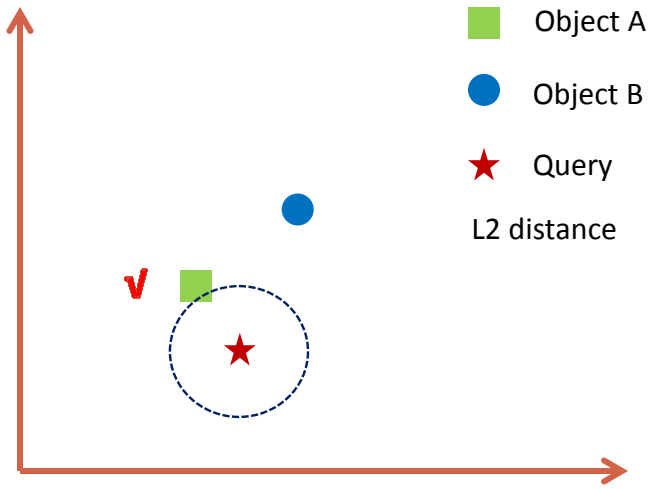




Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.

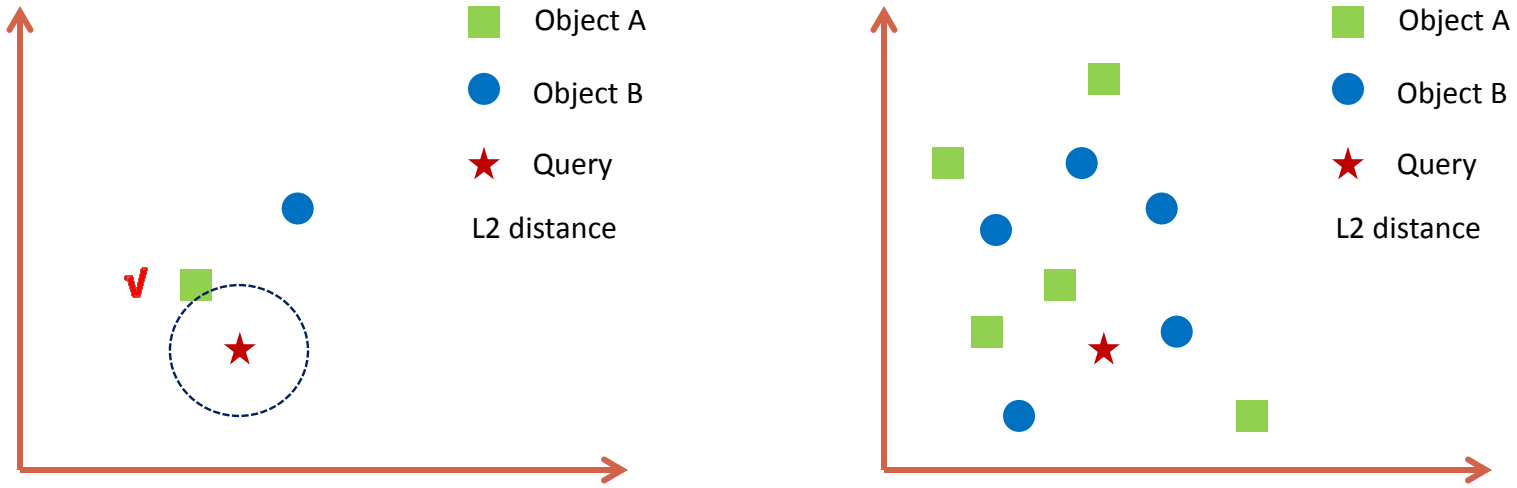
- Easy for objects with single instance when distance metric is given.



Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.

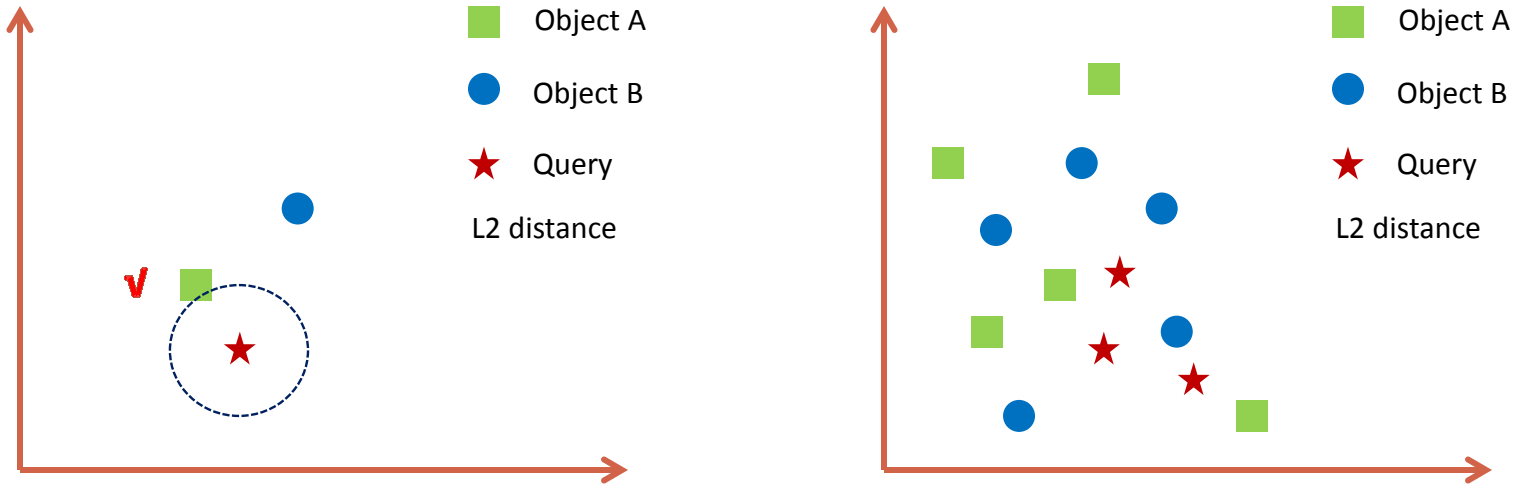
- Easy for objects with single instance when distance metric is given.



Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.

- Easy for objects with single instance when distance metric is given.



Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.
- There are a lot of NN ranking functions for objects with multiple instances.
  - Min/Max, Expected distance, Quantile.
  - NN probability, Expected rank.
  - Earth Mover's distance, Netflow distance.

Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.

  N_1: all pairs based NN for multi-valued or uncertain object
  - E.g., Min/Max, Expected distance, Quantile.

  N_2: possible world based NN for uncertain object
  - E.g., NN probability, Expected rank.

  N_3: selected pairs based NN for multi-valued or uncertain object
  - E.g., Earth Mover's distance, Netflow distance.

Introduction

- Nearest Neighbour (NN) search:
  - Given a query object *Q*, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.

  N_1: all pairs based NN for multi-valued or uncertain object
  - E.g., Min/Max, Expected distance, Quantile.

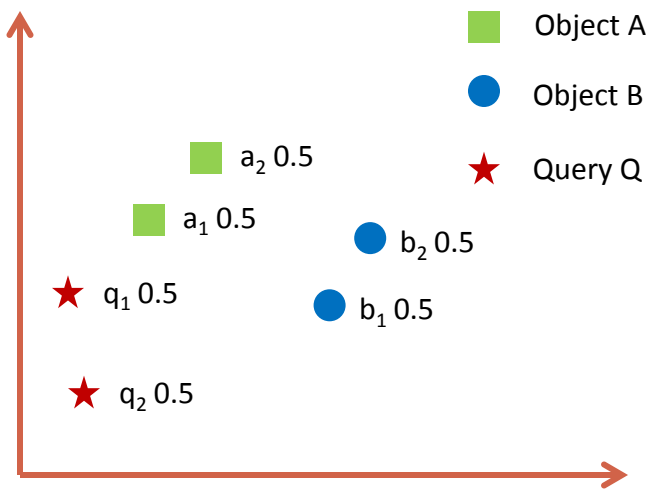
  N_2: possible world based NN for uncertain object
  - E.g., NN probability, Expected rank.

  N_3: selected pairs based NN for multi-valued or uncertain object
  - E.g., Earth Mover's distance, Netflow distance.

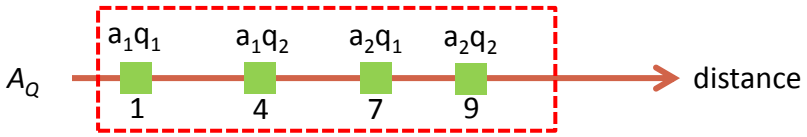
Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

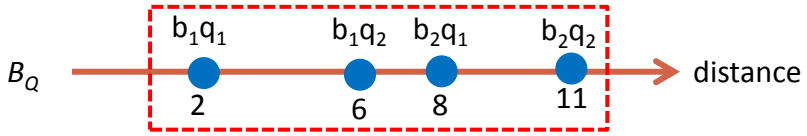
- There are a lot of NN ranking functions for objects with multiple instances.



1. Expected distance (all pairs based NN)



$$E_{dis}(A,Q) = 1 * 0.25 + 4 * 0.25 + 7 * 0.25 + 9 * 0.25 = 5.75$$

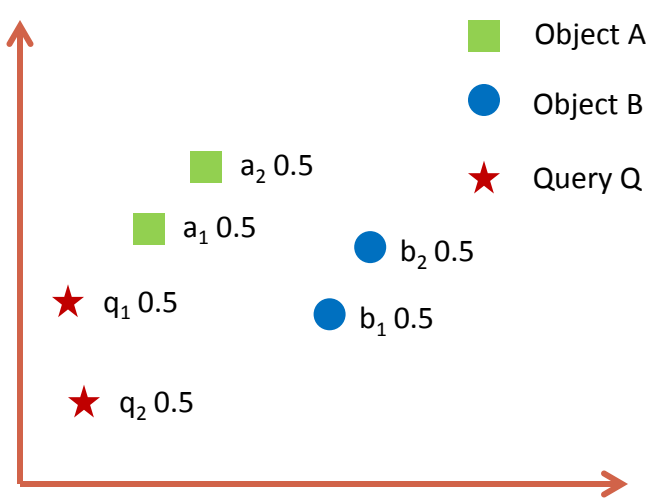


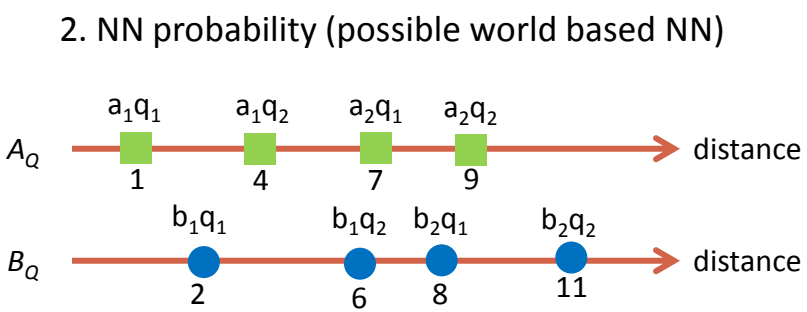
$$E_{dis}(B,Q) = 2 * 0.25 + 6 * 0.25 + 8 * 0.25 + 11 * 0.25 = 6.75$$

Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.

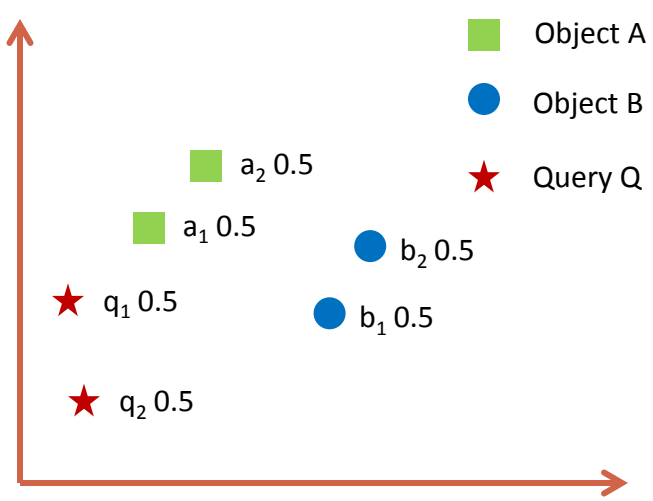


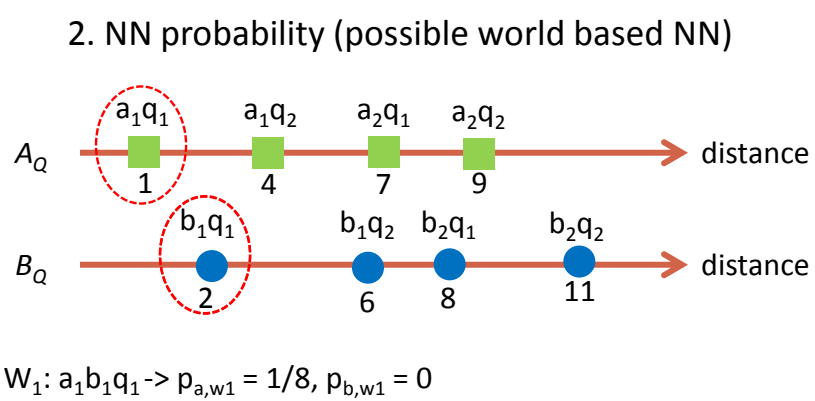


Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.



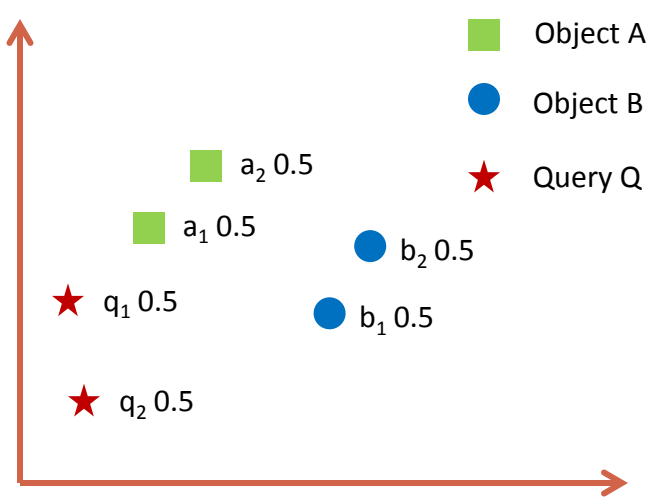


W_1: $a_1b_1q_1$ -> $p_{a,w1} = 1/8$, $p_{b,w1} = 0$

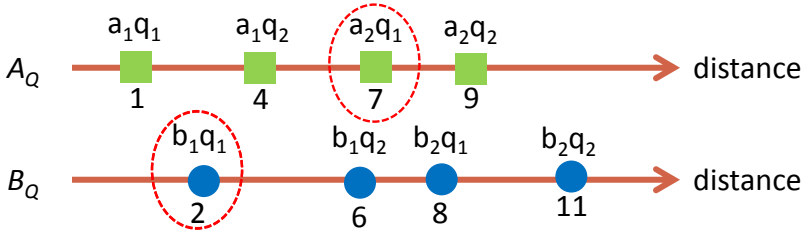
Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.



2. NN probability (possible world based NN)



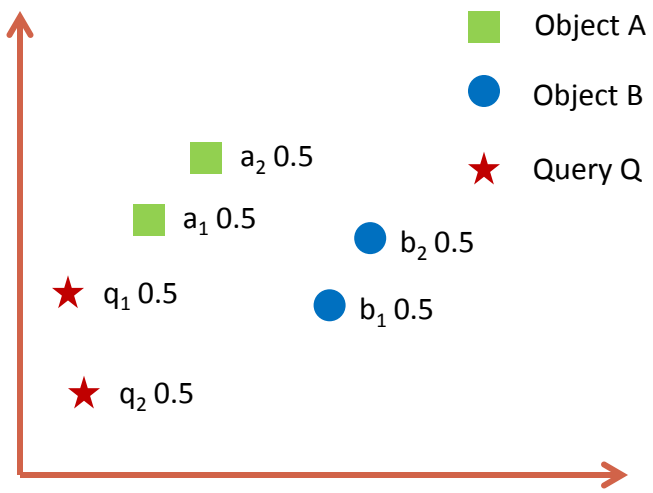
W_1: $a_1b_1q_1$ -> $p_{a,w1} = 1/8$, $p_{b,w1} = 0$

W_2: $a_2b_1q_1$ -> $p_{a,w2} = 0$, $p_{b,w2} = 1/8$

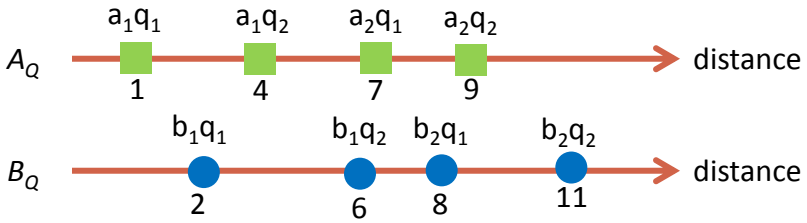
Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.



2. NN probability (possible world based NN)



W_1: $a_1b_1q_1$ -> $p_{a,w1} = 1/8$, $p_{b,w1} = 0$

W_2: $a_2b_1q_1$ -> $p_{a,w2} = 0$, $p_{b,w2} = 1/8$

...

W_8: $a_2b_2q_2$ -> $p_{a,w8} = 1/8$, $p_{b,w8} = 0$

=>

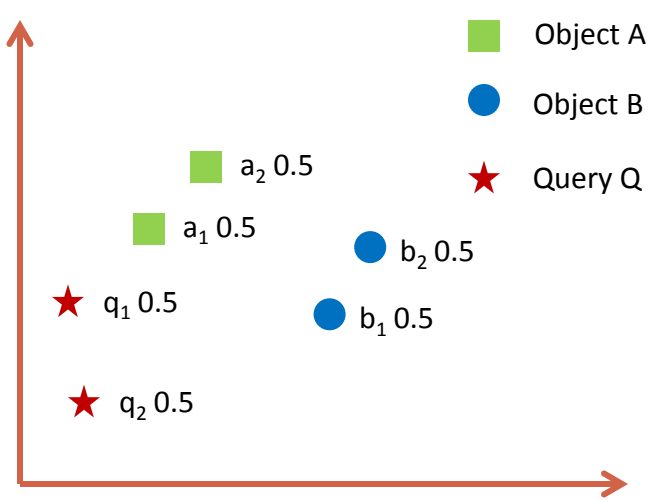
$NN_p(A,Q) = \sum p_{a,wi} = 6/8$

$NN_p(B,Q) = \sum p_{b,wi} = 2/8$

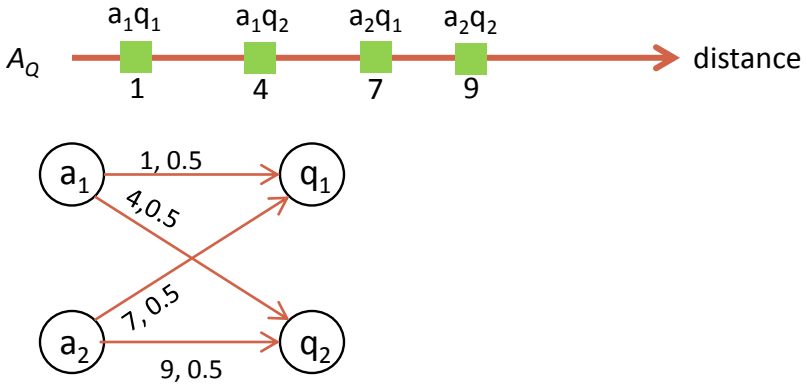
Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.



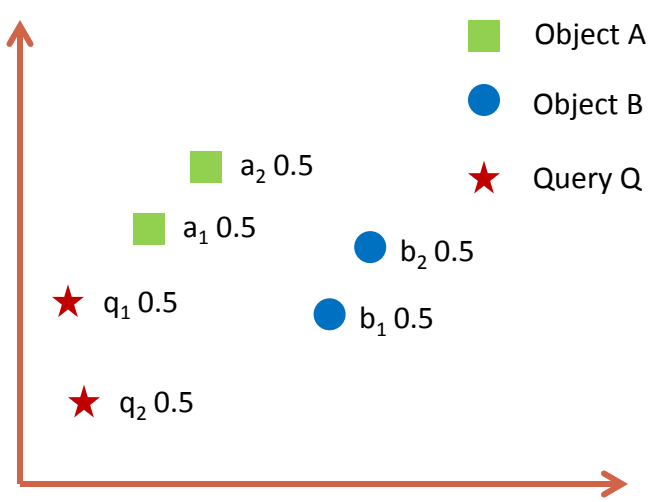
3. Earth Mover's distance (selected pairs based NN)

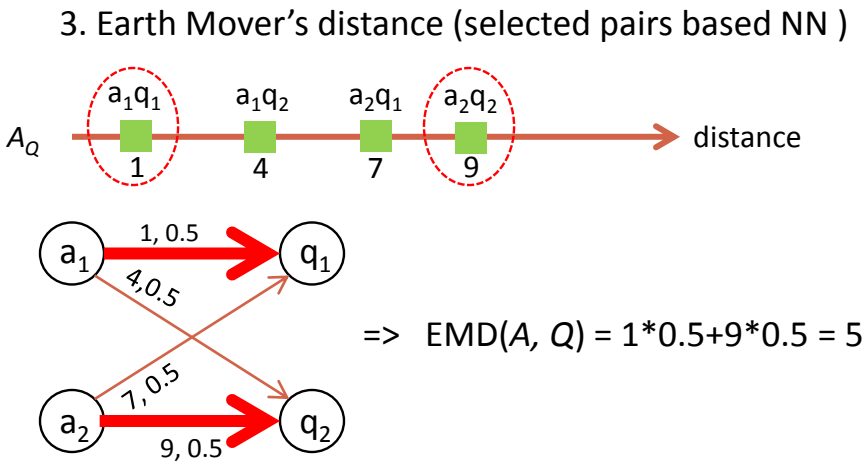


Introduction

- Nearest Neighbour (NN) search:
  - Given a query object Q, return the nearest object to the query.

- There are a lot of NN ranking functions for objects with multiple instances.





Introduction

- There are a lot of NN ranking functions for objects with multiple instances.
- The parameters in NN function can be set to infinite value, e.g., quantile distance.

Introduction

- There are a lot of NN ranking functions for objects with multiple instances.
- The parameters in NN function can be set to infinite value, e.g., quantile distance.

- **Motivation**
  - **A user may not have a specific NN function in mind, it is desirable to provide her with a set of NN candidates.**

Introduction

- A spatial dominate (SD) operator is used to define the partial order between objects regarding a query *Q*, i.e., SD(*A, B, Q*) means *A* dominates *B*, otherwise $\neg$SD(*A, B, Q*).

- Given a SD operator, the NN candidate set consists of the objects that are not dominated by any other objects.

- Optimal SD operator w.r.t a family *F* of NN functions
  - Correctness: SD(*A, B, Q*) $\Rightarrow \forall f \in F$, that $f(A) \leq f(B)$
  - Completeness: $\neg$SD(*A, B, Q*) $\Rightarrow \exists f \in F$, that $f(A) > f(B)$

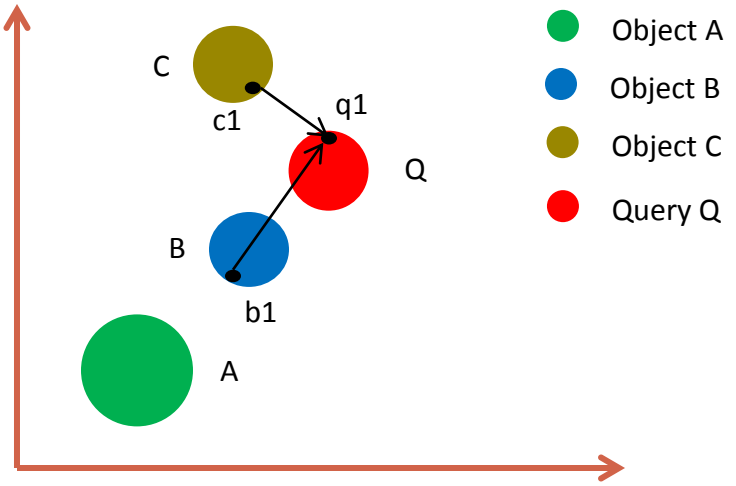
Related Work

- Full dominance operator (F-SD)
  - We have FSD(*A,B,Q*), if and only if $\forall q \in Q, \max(A,q) \leq \min(B,q)$ (SIGMOD10, SIGMOD14)

Cheng Long, et. al, The Hong Kong University of Science and Technology

Tobias Emrich, et. al, Institute for Informatics, Ludwig-Maximilians-Universität München

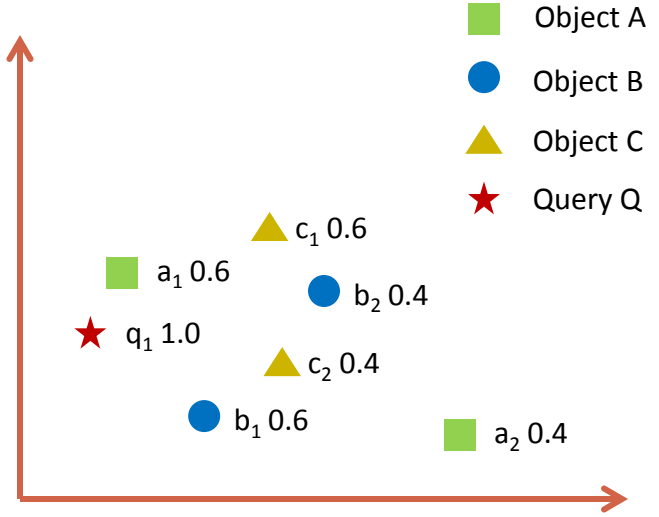
$FSD(B,A,Q)$
$\neg FSD(B,C,Q)$

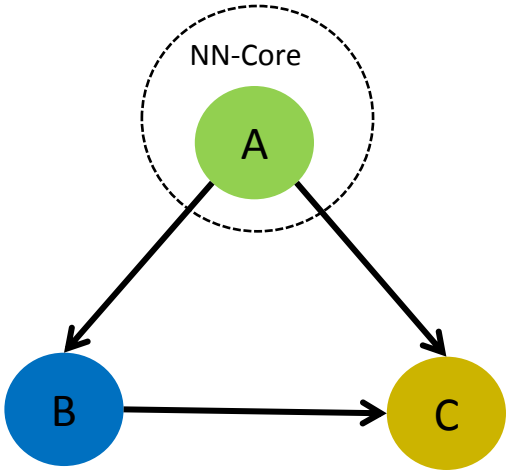


Over pessimistic: contain many redundant objects (no completeness property)

Related Work

- NN Core (TKDE10) ← Sze Man Yuen, et. al, Chinese University of Hong Kong
  - A → B: if A has higher chance to be closer to the query than B;
  - NN candidates : Minimal set of objects, each of which beats any object not in the NN candidates.



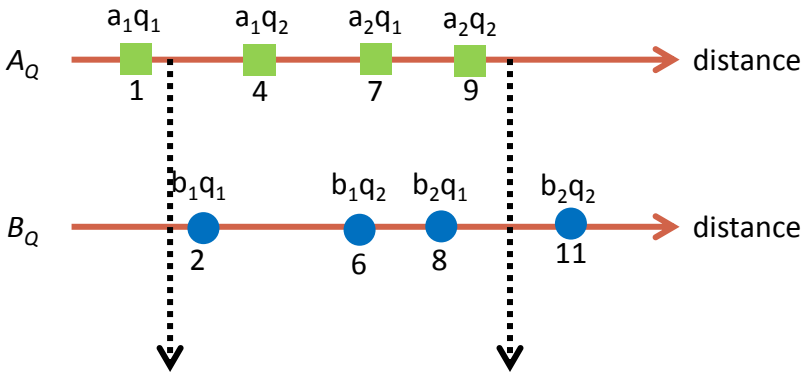


Too aggressive: violate correctness property

Related Work

- Stochastic order has been widely used in various domains to compare the "goodness" of two random variables distributions.

Stochastic Order. Given two independent random variables *X* and *Y* , we say *X* is smaller than *Y* in usual stochastic order, denoted by $X \preccurlyeq_{st} Y$, if $Pr(X \leq \lambda) \geq Pr(Y \leq \lambda)$ for every $\lambda \in R$.



$\Pr(A_Q \leq 1.5) = 0.25; \Pr(B_Q \leq 1.5) = 0$

$\Pr(A_Q \leq 10) = 1; \Pr(B_Q \leq 10) = 0.75$

Problem Definition

- Objects with multiple instances
  - An object *U* consists of a set $\{u_i\}$ of instances, and a discrete probability mass function assigns each instance u_i a probability value, denoted by $p(u_i)$, where $\sum p(u_i) = 1$.
  - The multi-valued objects are treated as discrete random variable if their weight can be normalized.

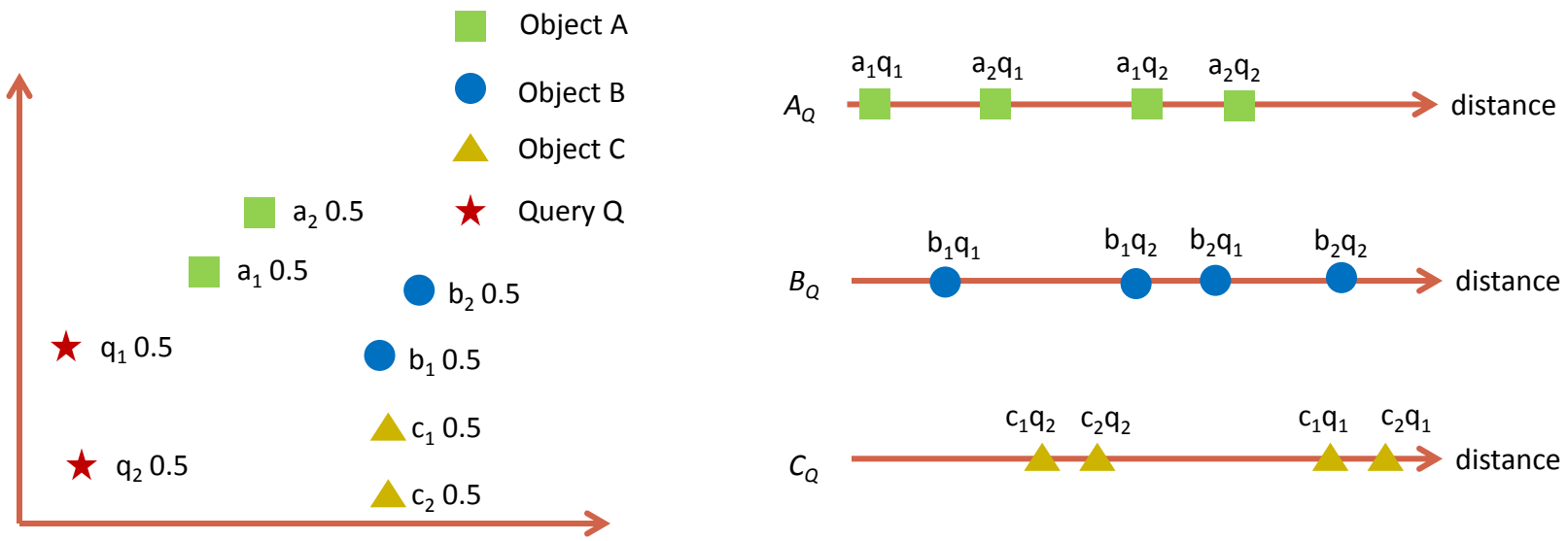
- Problem statement
  - Devise spatial dominance (SD) operators by carefully considering various NN function families.

Classify NN Functions

- N_1: all pairs based NN
  - Aggregation over pair-wise distances g(A_Q), e.g., Min/Max, Expected distance, Quantile.
- N_2: possible world based NN
  - Aggregation over score on each possible world g(A_W), e.g., NN probability, Expected rank.
- N_3: selected pairs based NN
  - Aggregation over selected pairs g(S(A_Q)), e.g., Earth Mover's distance, Netflow distance.

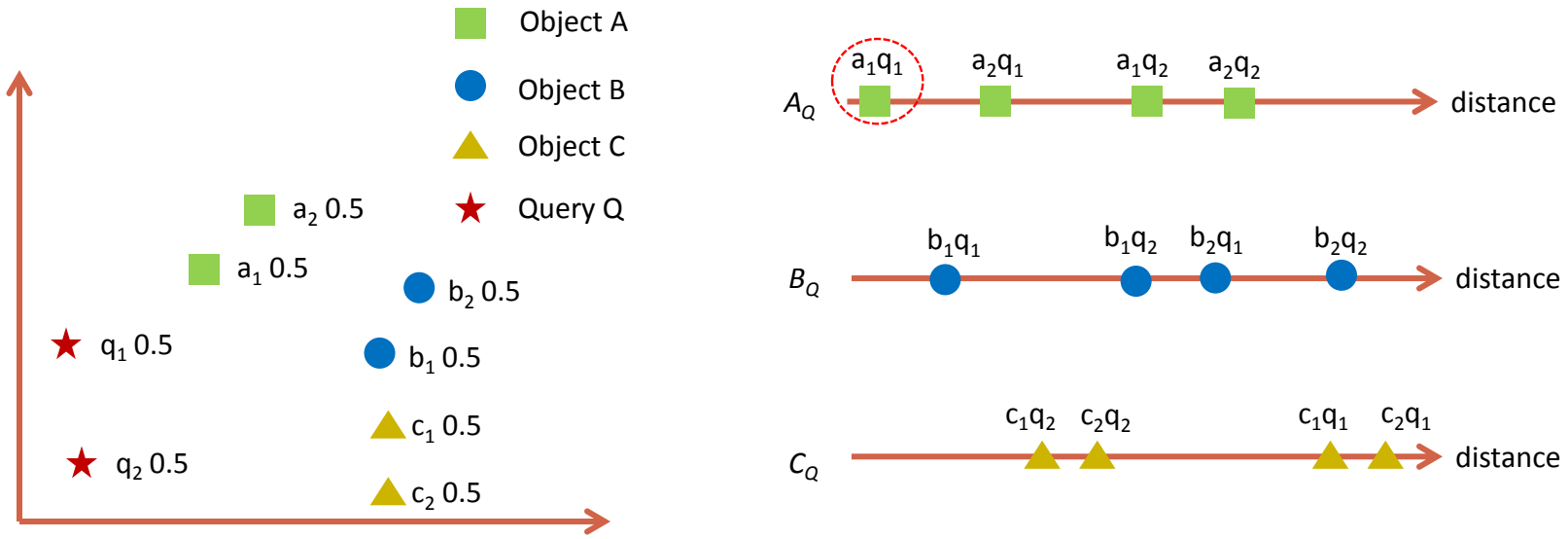
SD Operator

- Stochastic-SD (*S-SD*, opt. w.r.t N_1)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSD*(*U,V,Q*) if and only if $U_Q \preccurlyeq_{st} V_Q$ and $U_Q \neq V_Q$.



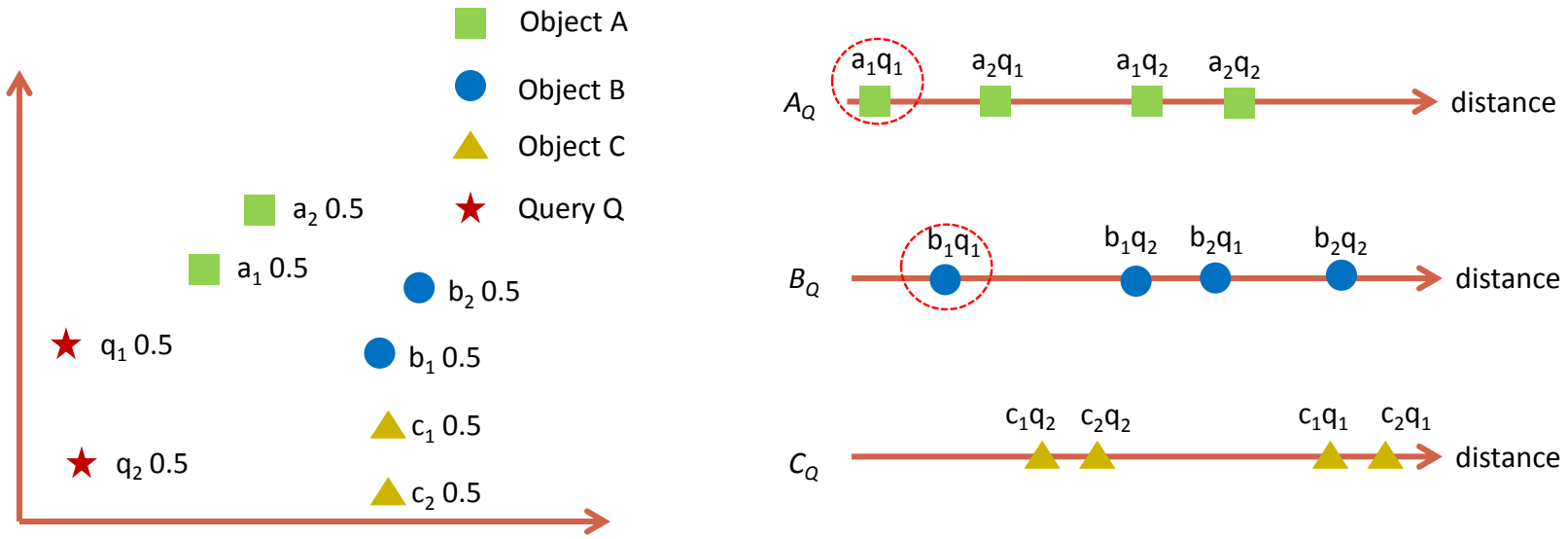
SD Operator

- Stochastic-SD (*S-SD*, opt. w.r.t N_1)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSD*(*U,V,Q*) if and only if $U_Q \preccurlyeq_{st} V_Q$ and $U_Q \neq V_Q$.



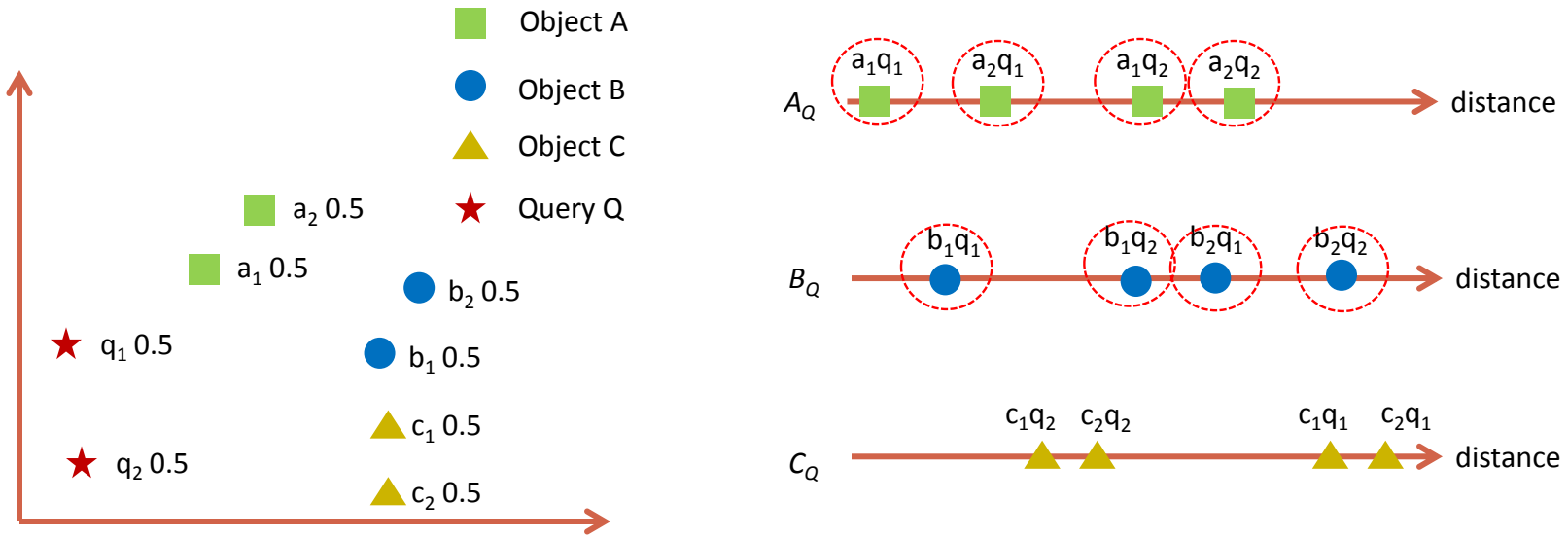
SD Operator

- Stochastic-SD (*S-SD*, opt. w.r.t N_1)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSD*(*U,V,Q*) if and only if $U_Q \preccurlyeq_{st} V_Q$ and $U_Q \neq V_Q$.



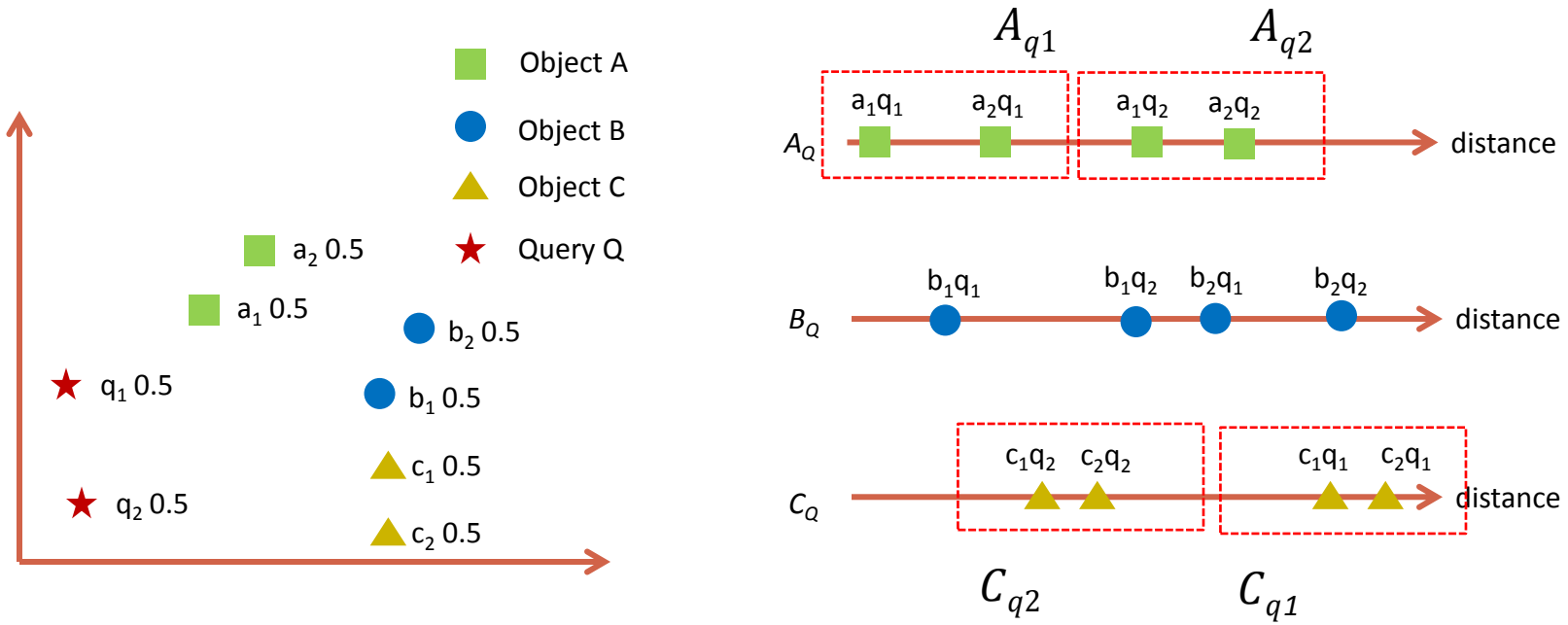
SD Operator

- Stochastic-SD (*S-SD*, opt. w.r.t N_1)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSD*(*U,V,Q*) if and only if $U_Q \preccurlyeq_{st} V_Q$ and $U_Q \neq V_Q$.



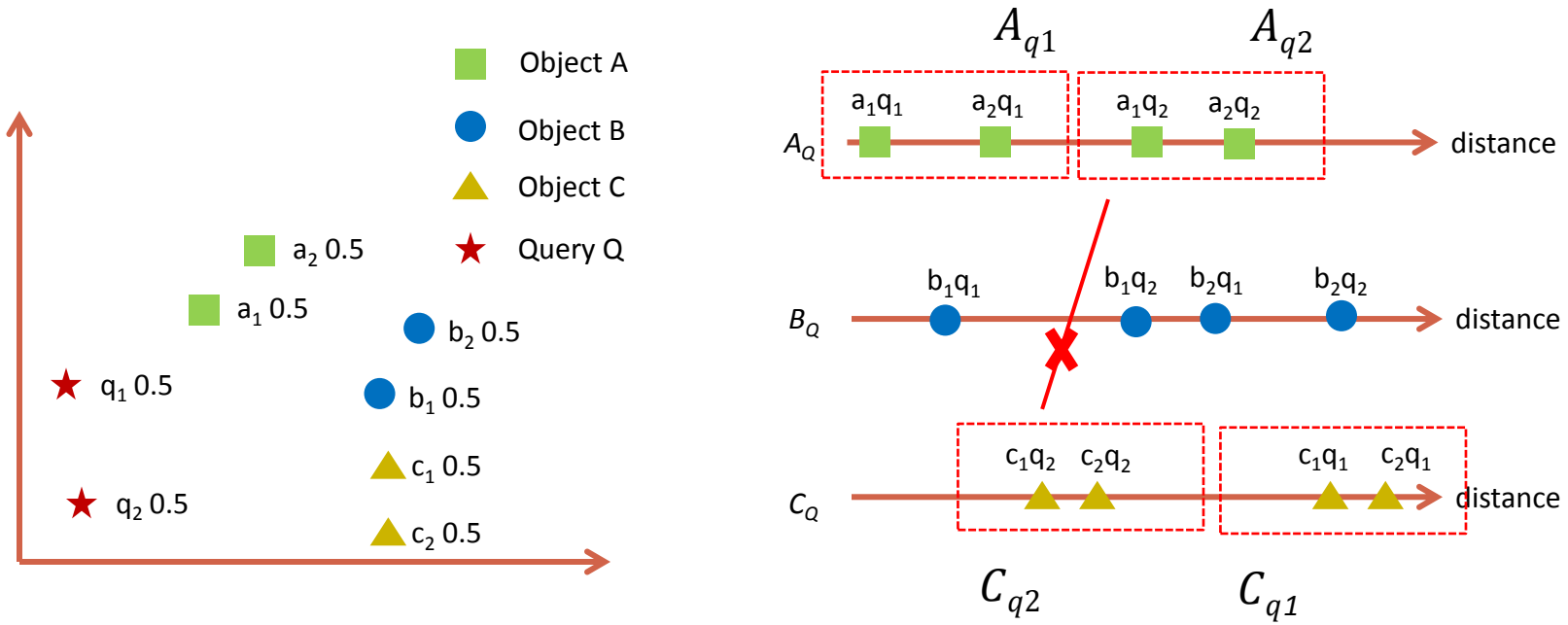
SD Operator

- Strict Stochastic-SD (*SS-SD*, opt. w.r.t $N_{1,2}$)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSSD*(*U*,*V*,*Q*) if and only if $U_q \preccurlyeq_{st} V_q$ for $\forall q \in Q$ and $U_Q \neq V_Q$.



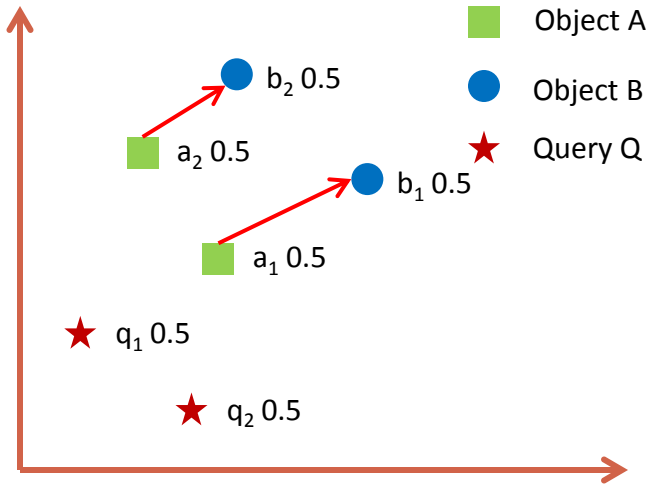
SD Operator

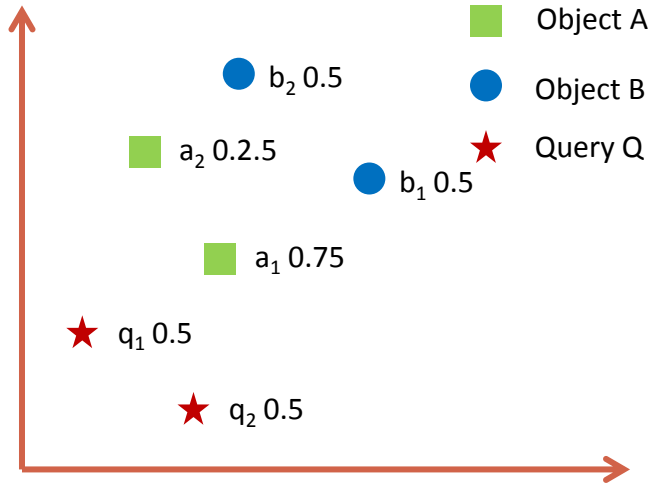
- Strict Stochastic-SD (*SS-SD*, opt. w.r.t $N_{1,2}$)
  - Given two objects *U* and *V*, and the query *Q*, we have *SSSD*(*U*,*V*,*Q*) if and only if $U_q \preccurlyeq_{st} V_q$ for $\forall q \in Q$ and $U_Q \neq V_Q$.



SD Operator

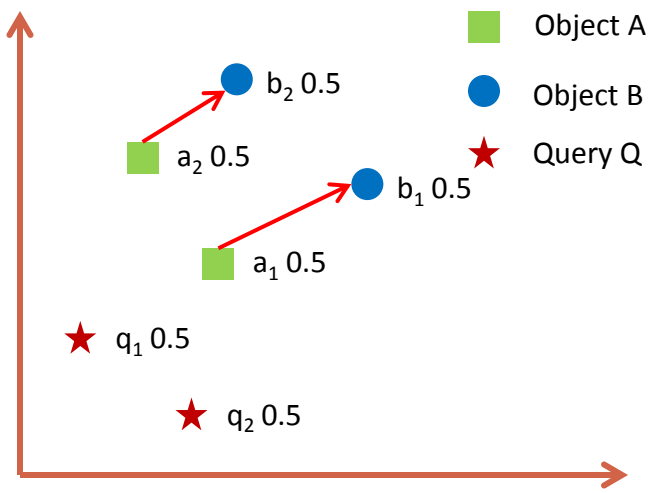
- Peer-SD (*P-SD*, opt. w.r.t $N_{1,2,3}$)
  - Given two objects *U* and *V*, and the query *Q*, we have *PSD*(*U*,*V*,*Q*) if there is a mapping between *U* and *V*, for each pair <*u*,*v*>, *u* is always closer to *Q* than *v* with same weight.

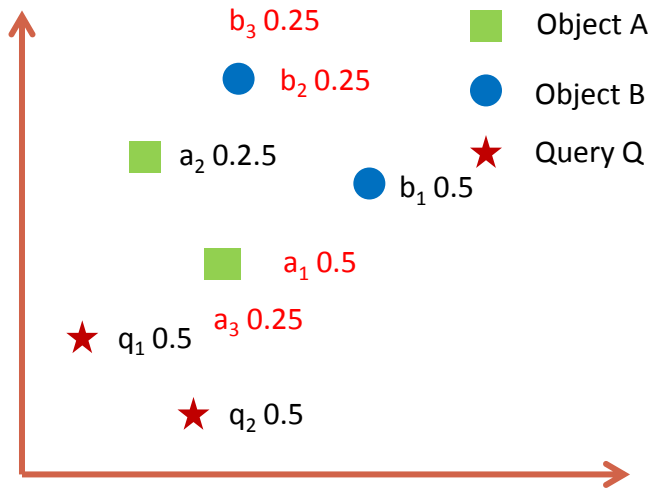




SD Operator

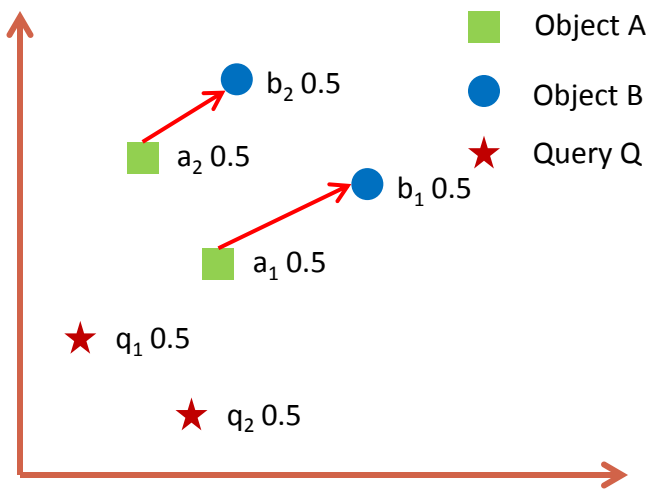
- Peer-SD (*P-SD*, opt. w.r.t $N_{1,2,3}$)
  - Given two objects *U* and *V*, and the query *Q*, we have *PSD*(*U,V,Q*) if there is a mapping between *U* and *V*, for each pair <*u*,*v*>, *u* is always closer to *Q* than *v* with same weight.

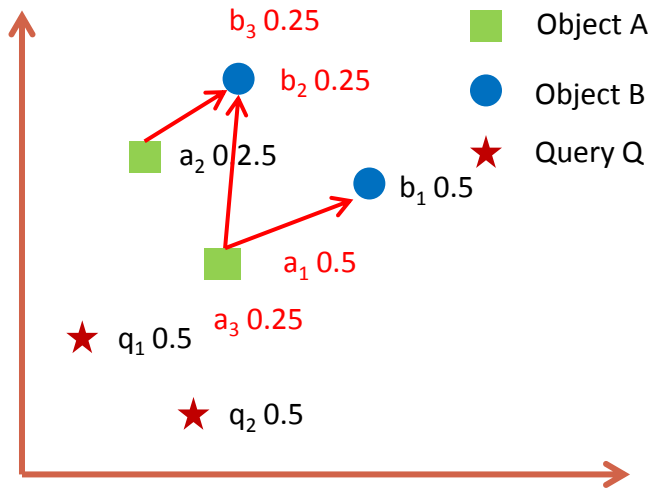




SD Operator

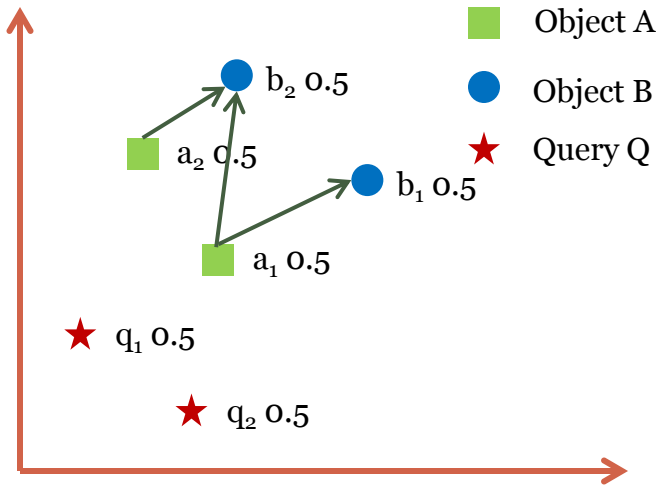
- Peer-SD (*P-SD*, opt. w.r.t $N_{1,2,3}$)
  - Given two objects *U* and *V*, and the query *Q*, we have *PSD*(*U*,*V*,*Q*) if there is a mapping between *U* and *V*, for each pair <*u*,*v*>, *u* is always closer to *Q* than *v* with same weight.

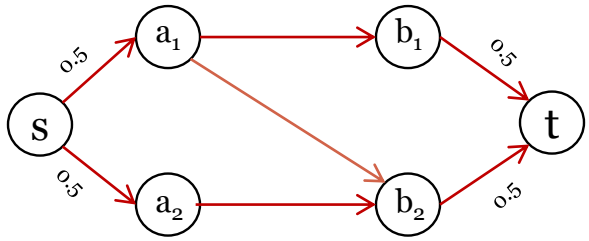




P-SD Check

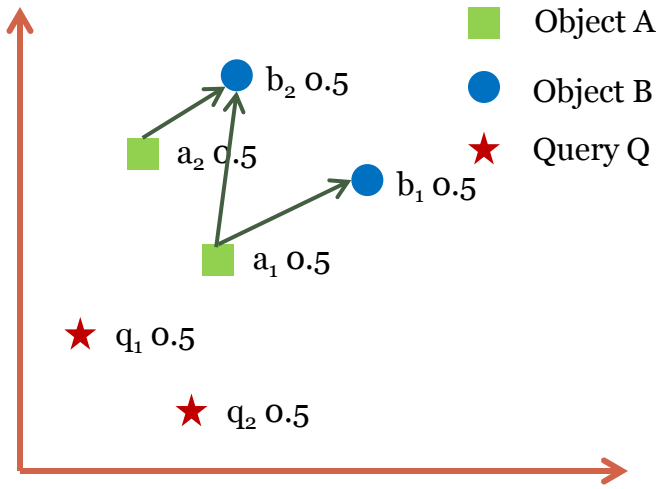
- *P-SD* check
  - Naively have to check all the mapping.
  - The P-SD check between *U* and *V* can be reduced to compute the network flow problem, *PSD*(*U*,*V*,*Q*) iff the network flow is 1

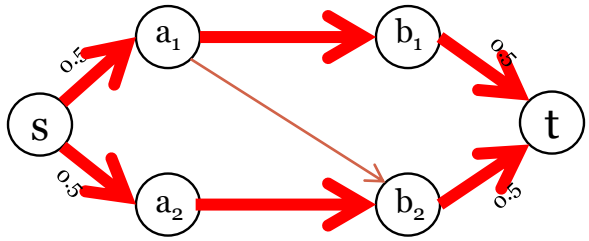




P-SD Check

- *P-SD* check
  - Naively have to check all the mapping.
  - The P-SD check between *U* and *V* can be reduced to compute the network flow problem, *PSD*(*U*,*V*,*Q*) iff the network flow is 1

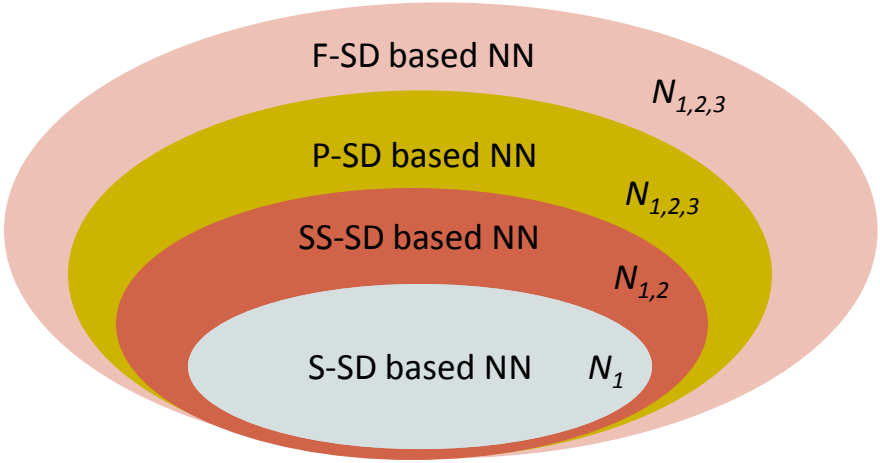




SD Operator Properties

- SD operator containment

$$NNC(O, Q, S\text{-}SD) \subseteq NNC(O, Q, SS\text{-}SD) \subseteq NNC(O, Q, P\text{-}SD) \subseteq NNC(O, Q, F\text{-}SD)$$



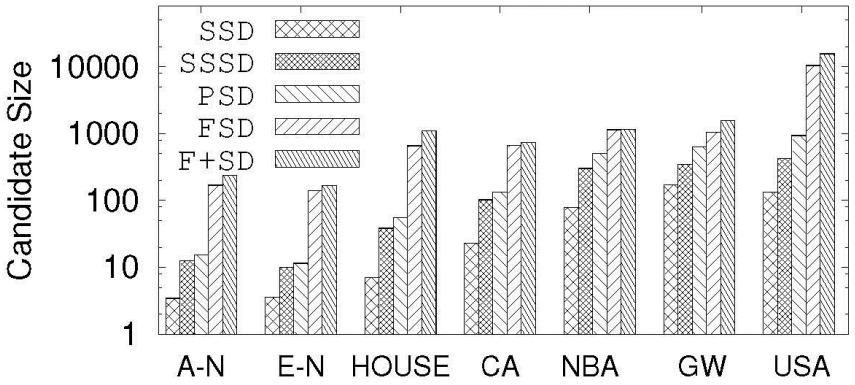
- NN candidate search
  - Based on branch and bound framework like skyline search

Experiments

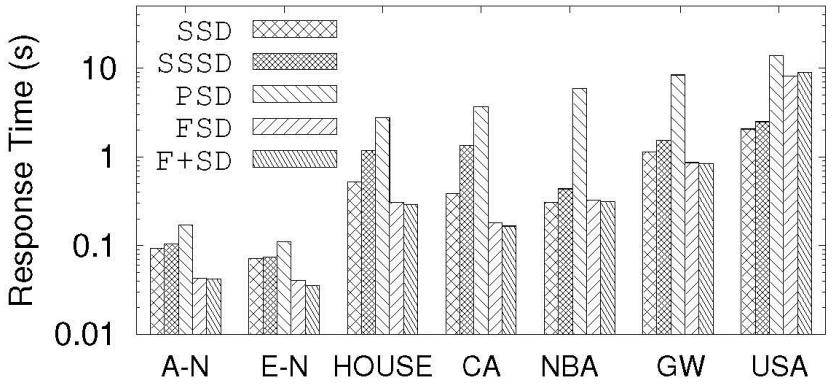
- Compare Algorithm
  - SSD, SSSD, PSD, FSD and F^+SD
- Datasets:
  - real dataset: NBA, Gowalla;
  - semi-real dataset: House, CA, USA;
  - synthetic dataset.

Evaluation parameter	Values
dimensionality d	2, **3**, 4, 5
# of objects n	**100k**, 200k, 400k, 600k, 1M
# of object instances m_d	20, **40**, 60, 80, 100
edge length of object h_d	100, 200, 300, **400**, 500
object center distribution	**anti (A)**, indep (E)
# of query instances m_q	10, 20, **30**, 40, 50
edge length of query h_q	100, **200**, 300, 400, 500

Experiments

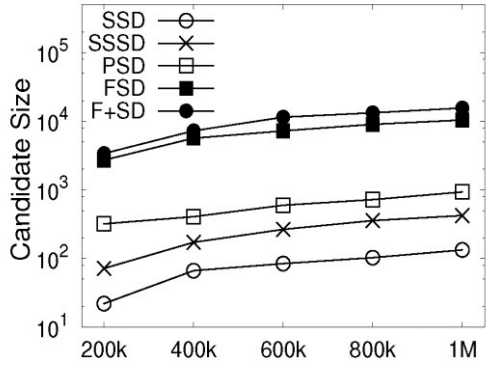


(a) Candidate Size of Different Datasets

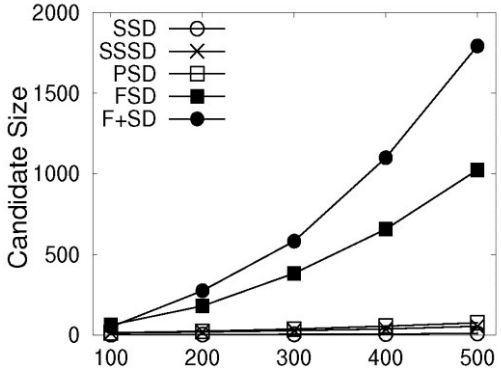


(b) Response Time of Different Datasets

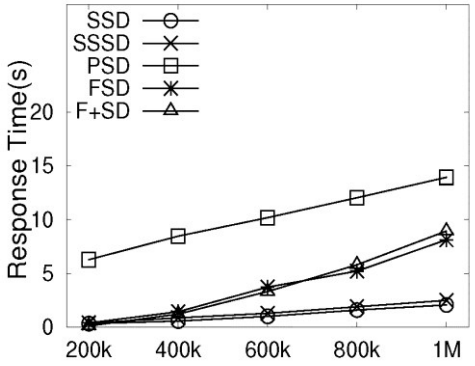
Experiments



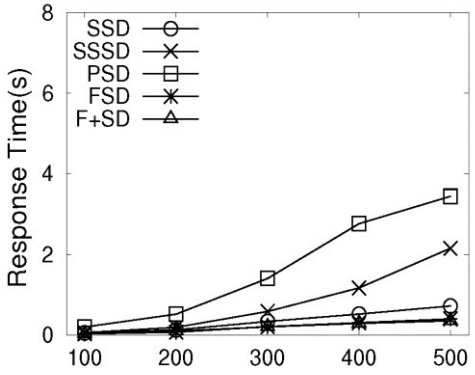
(a) Varying n



(b) Varying h_d



(c) Varying n



(d) Varying h_d

Conclusion

- Formalize three families of NN functions that cover popular NN ranking mechanisms.
- Advocate three SD operator that are optimal to different family of NN functions.
- Propose efficient NN candidate search algorithm for three *SD* operators.

Thanks!
Q&A

Experiments

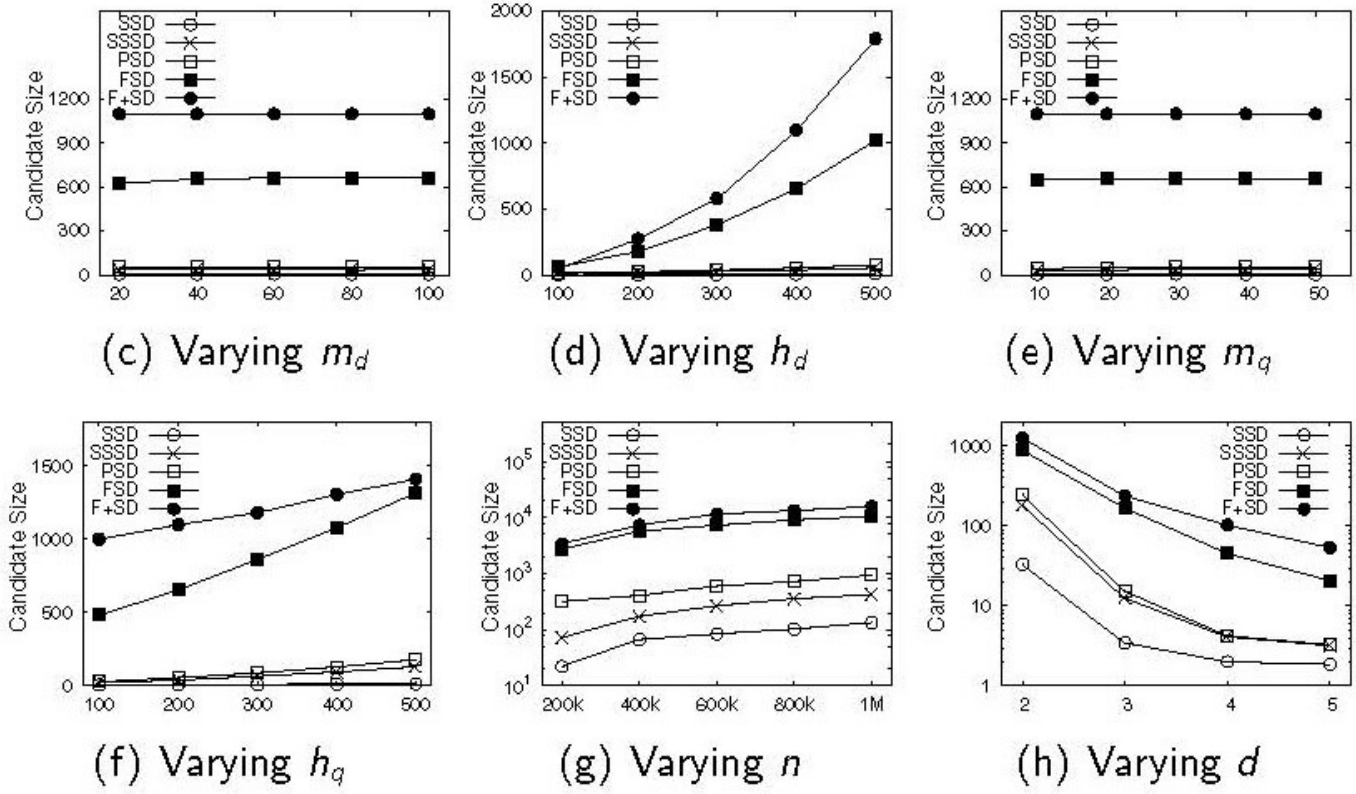


Figure: Impact of Diff. Parameters on Effectiveness

Experiments

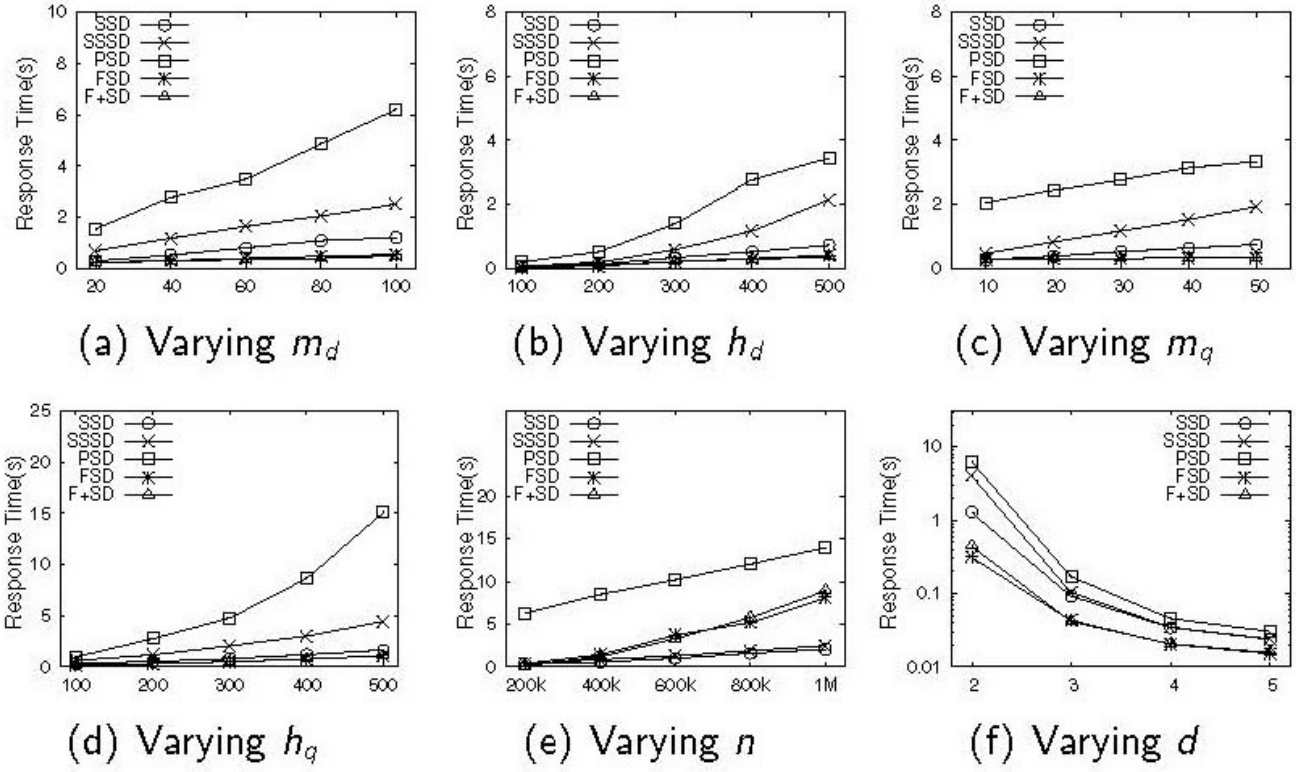


Figure: Impact of Diff. Parameters on Efficiency