LSDIS
Large Scale Distributed Information Systems

University of Georgia
Computer Science Department

Discovering Informative Subgraphs in RDF Graphs

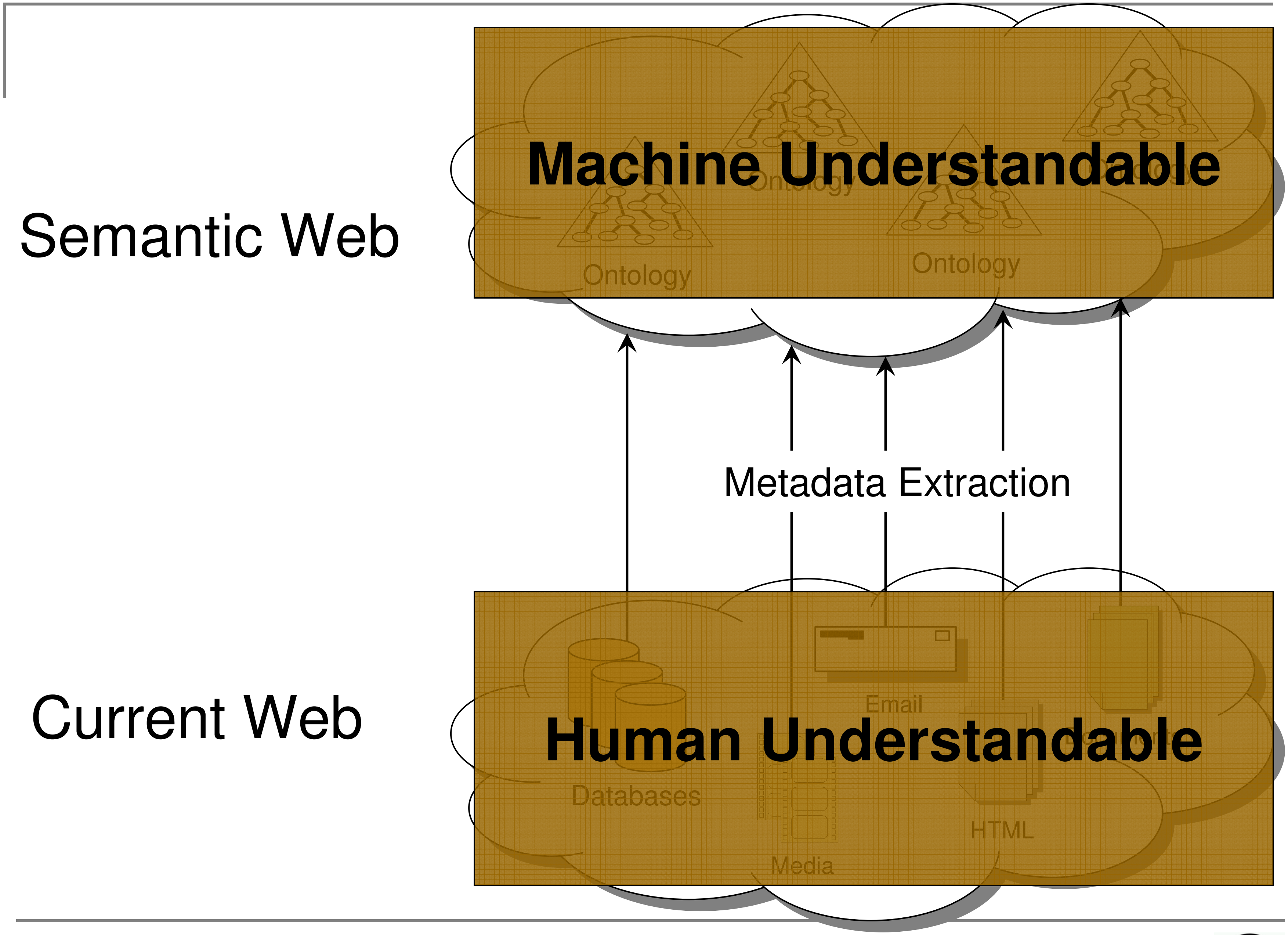
By: Willie Milnor

Advisors: Dr. John A Miller
Dr. Amit P. Sheth

Committee: Dr. Hamid R. Arabnia
Dr. Krysztof J. Kochut

Outline

- Background and Motivation
- Objective
- Algorithms
- Heuristics
- Experimentation
  - Dataset and Scenario
  - Results and Evaluation
- Conclusions and Future Work


Semantic Web
Machine Understandable
Ontology
Ontology
Ontology
Metadata Extraction
Current Web
Human Understandable
Databases
Email
HTML
Media

Semantic Web

- *A framework that allows* ***data*** *to be shared and reused across application, enterprise, and community boundaries* – W3C[1]
  - Integration of heterogeneous data
- Semantic Web Technologies [7]
  - ontologies
  - KR (RDF/S, OWL)
  - entity identification and disambiguation
  - reasoning over relationships

http://www.w3.org/2001/sw/

Ontology

- Agreement over concepts and relationships
  - Specification of conceptualization [5]
- Represent meaning through relationships
  - semantics
- Semantic annotation of distributed information
- Populated through extraction
  - Identify entity objects and relationships
  - Disambiguate multiple mentions of same object

RDF/S

- W3C Recommendation
- Machine understandable representation
- Graph Model:
  - Nodes are entities
  - Edges are relationships
- Triple model: subject, predicate, object
- Schema definition language
- QL's and data storages

RDF Query Languages

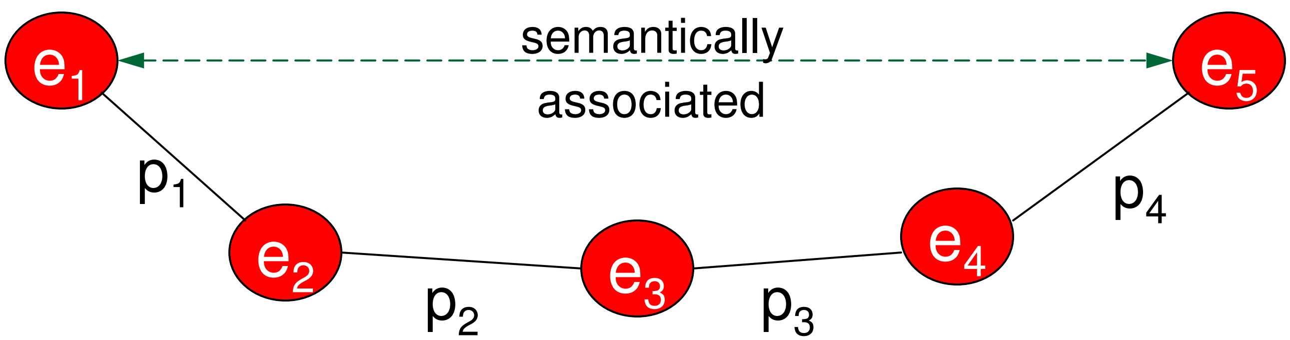
RQL	select RESEARCHER, PUBLICATION from {RESEARCHER} lsdis:authors {PUBLICATION} using namespace lsdis = http://lsdis.cs.uga.edu/sample.rdf#
RDQL	SELECT ?researcher, ?publication WHERE (?researcher lsdis:authors ?publication)USING info FOR <http://lsdis.cs.uga.edu/sample.rdf#>
SPARQL	PREFIX lsdis: http://lsdis.cs.uga.edu/sample.rdf# SELECT ?researcher, ?publication WHERE { ?researcher lsdis:authors ?publication }

Semantic Analytics

- Automatic analysis of semantic metadata
- Mining and searching heterogeneous data sources
  - Millions of entities and explicit relationships
  - i.e. SWETO [2]
- Uncover meaningful complex relationships
- Application areas [8]
  - Terrorist threat assessment
  - Anti-money laundering
  - Financial compliance

Semantic Associations [3]

- Complex relationships between entities
  - Sequence of properties connecting intermediate entities

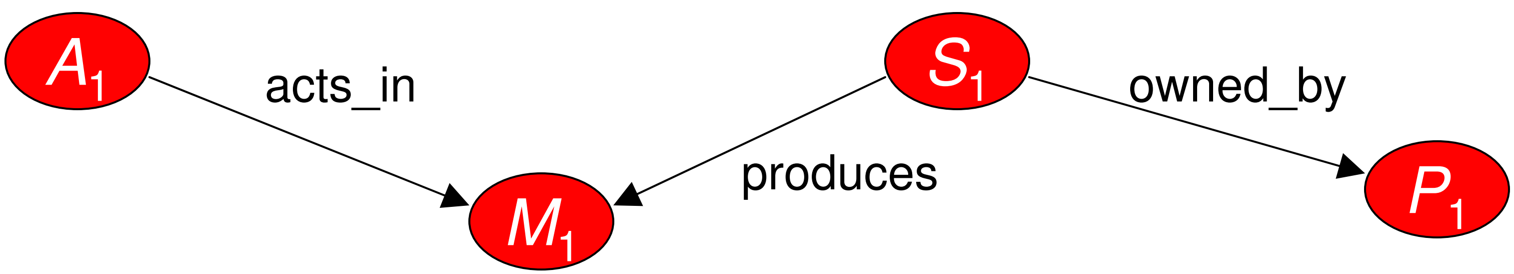


Semantic Associations Defined

- *Semantic Connectivity*
  - An alternating sequence of properties and entities *(semantic path)* exists between two entities
- *Semantic Similarity*
  - An existing pair of matching property sequences where entities in question are respective origins or respective terminuses
- *Semantic Association*
  - Two entities are semantically associated if they are either *semantically connected* or *semantically similar*

Why Undirected Edges?

- Consider 3 statements:
  1) *Actor → acts_in → Movie*
  2) *Studio → produces → Movie*
  3) *Studio → owned_by → Person*
- Instances:



Association Identification

- Association matching
  - Patterns of schema properties/relationships
  - Inference rules
- Require explicit knowledge of ontology
  - Impractical for complex schemas

Association Discovery

- Discovering anomalous patterns, rules, complex relationships
- No predefined patterns or rules
- Limitations
  - Information overload—extremely large result sets
  - Cannot determine significance/relevance

Ranking

- User specified criteria
  - User specifies what is considered significant
  - Criteria can be statistical or semantic [1]
  - Relevance model
- Predefined criteria
  - Rank based on *novelty* or *rarity* [6]
  - May not be of interest

Semantic in Ranking

- Schematic context:
  - Specify classes and properties of interest
  - Create multiple contexts for a single search
- Schematic structure
  - Rank based on property and/or class subsumption
- Trust
  - How well trusted is an explicit relationships
  - How well can a complex relationship be trusted
- Refraction [3]
  - How well does a path conform to a given schema

Heuristic Based Discovery

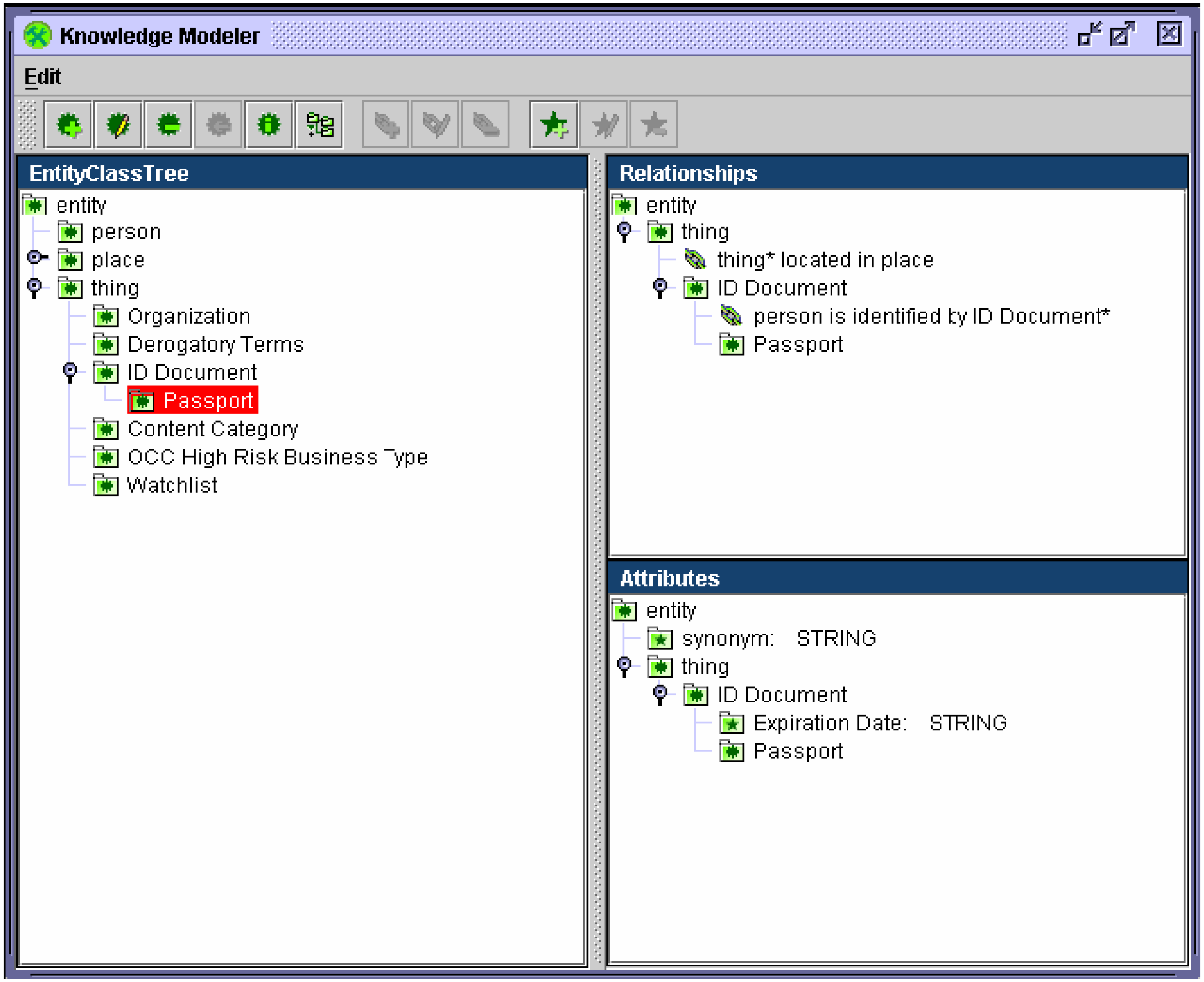
- High complexity in uninformed search
- Informed *(a priori* knowledge*):*
  - Pruning of large search space
  - Certain associations ignored during processing
- Disadvantage: incomplete results
- Could utilize user configurable criteria

Semantic Visualization

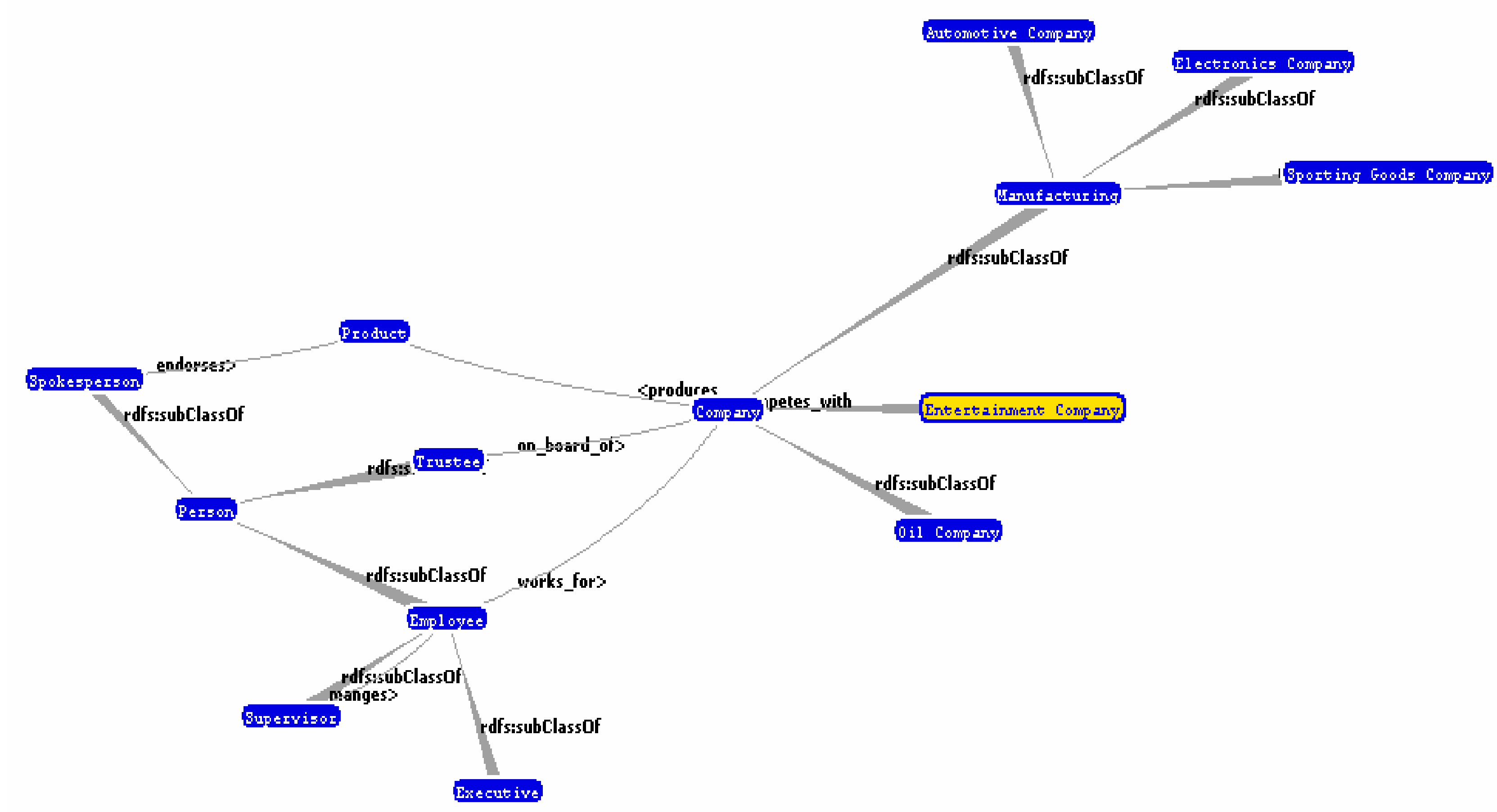
- Ability to browse/visualize ontology is crucial to Semantic Analytics [8]
  - Ontological navigation
- Graphical interfaces for schema development
  - Protégé[1]
  - Semagix Freedom[2]
  - Aid user in gaining cognitive understanding of schema
- Graphical representation of results

1. Protégé. http://protege.stanford.edu/
2. Semagix, Inc. http://www.semagix.com/

Development Interface



Graphical Visualization



Objective

- Heuristic based approach for computing Semantic Associations in Undirected edge-weighted graphs
- Adapt $O(n^3)$ time algorithm for *connection subgraph problem* [4].
  - Originally for single-typed edges in a social network
- Compute edge weights based on semantics
- Obtain relevant, visualizable subgraph

Algorithms

- Input is a weighted RDF graph
- Compute a *candidate graph*
  - Candidate to contain the most relevant associations
- Model graph as an electrical network
- Compute a *display graph* with at most *b* nodes
- *ρ-graph:*
  - Subgraph composed of semantic associations between a pair of entities

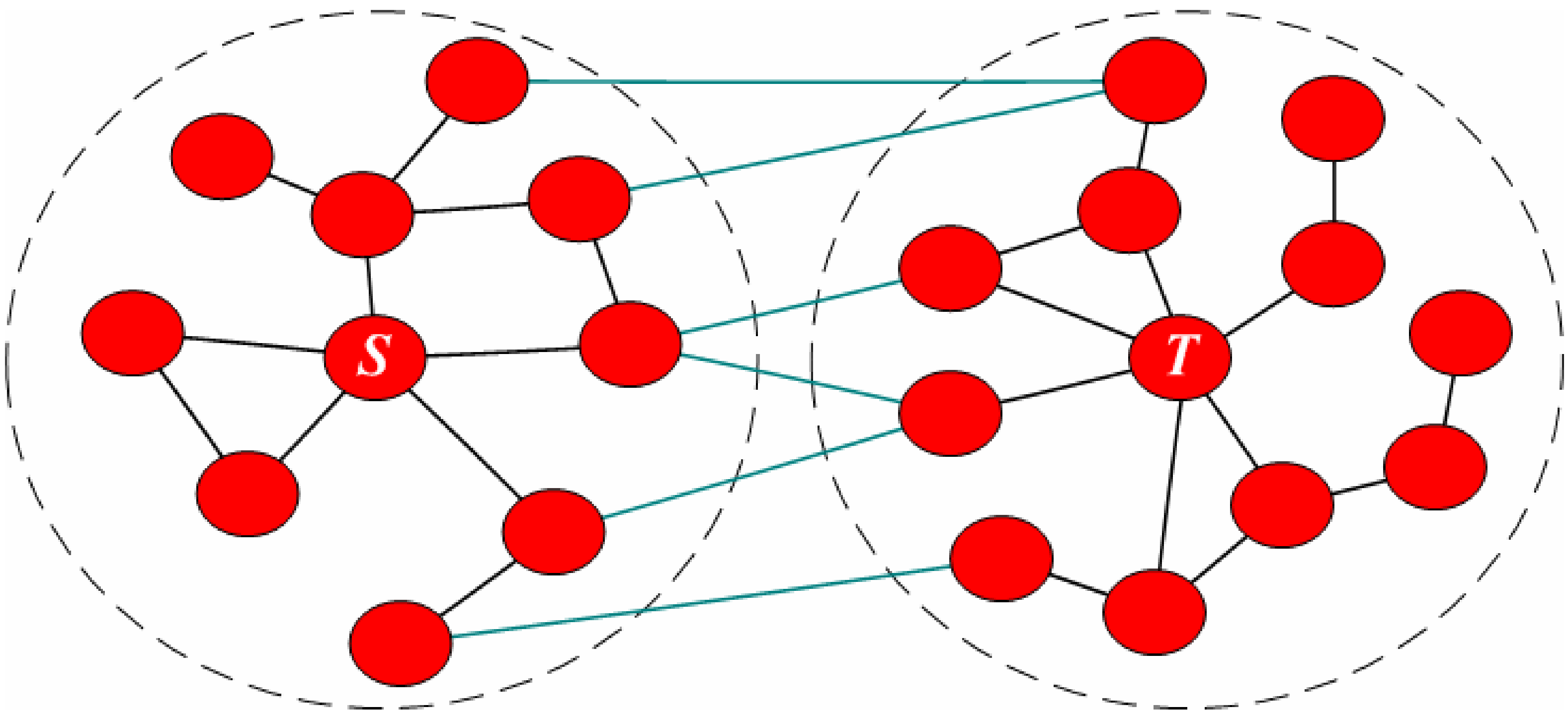
Candidate ρ-Graph

- Given nodes *S* and *T*
- Expand nodes to grow neighborhoods around *S* and *T*
- Use a *pick heuristic* method to select next node for expansion
  - Pick pending node closest to respective root
  - Based on notion of *distance* for an edge *(u,v)*

$$distance(u,v) = log\left(\frac{(degree(u) + degree(v))^2}{w(u,v)^2}\right)$$

Candidate ρ-Graph

- Abstract candidate graph structure



Display ρ-Graph

- Greedy algorithm
- Start with an empty subgraph
- Use dynamic programming to select next path to add to the subgraph
  - At each iteration, add the next path delivering maximum current to sink node proportional to the number of new nodes being added to the subgraph

Electrical Circuit Network

- Model the *Candidate ρ-graph* as a network of electrical circuits
  - *S* is source, *T* is sink
  - Edge weights are analogous to conductance
  - Need node voltages and edge currents

Electrical Circuit Network

- Let:
  - *C(u,v)* be the conductance along edge *(u,v)*
  - *C(u)* be the total conductance of edges incident on *u*
  - *V(u)* be the voltage of node *u*
  - *I(u,v)* be the current flow from *u* to *v*

Electrical Circuit Network

- Ohm's Law:

$$\forall u,v : I(u,v) = (V(u) - V(v))C(u,v)$$

- Kirchoff's Law:

$$\forall v \neq s,t : \sum_{u} I(u,v) = 0$$

Electrical Circuit Network

- Given:

$$V(s) = 1$$

$$V(t) = 0$$

- System of linear equations based on laws

$$V(u) = \sum_{v} \frac{V(v)C(u,v)}{C(u)} \qquad \forall u \neq s,t$$

Display ρ-Graph

- Successively add next path which maximizes ratio of delivered current to number of new nodes
- Delivered current $\hat{I}(u,v)$

$$\hat{I}(s,u) = I(s,u)$$

$$\hat{I}(s = u_1,\ldots,u_i) = \hat{I}(s = u_1,\ldots,u_{i-1})\frac{I(u_{i-1},u_i)}{I_{out}(u_{i-1})}$$

$$I_{out}(u) = \sum_v I(u,v), \qquad \forall v : V(u) > V(v)$$

Heuristics

- Loosely based on semantics
- Define schemas *S* as union of class and property sets
- Define an RDF store as union of schemas and corresponding instance triples
- Edge weight is the sum of the heuristic values

Class and Property Specificity (CS, PS)

- More specific classes and properties convey more information

- Specificity of property p_i:

$$\mu(p_i) = \frac{d(p_i)}{d(p_{iH})}$$

  - $d(p_i)$ is the depth of p_i
  - $d(p_i)$ is the depth of the branch containing p_i

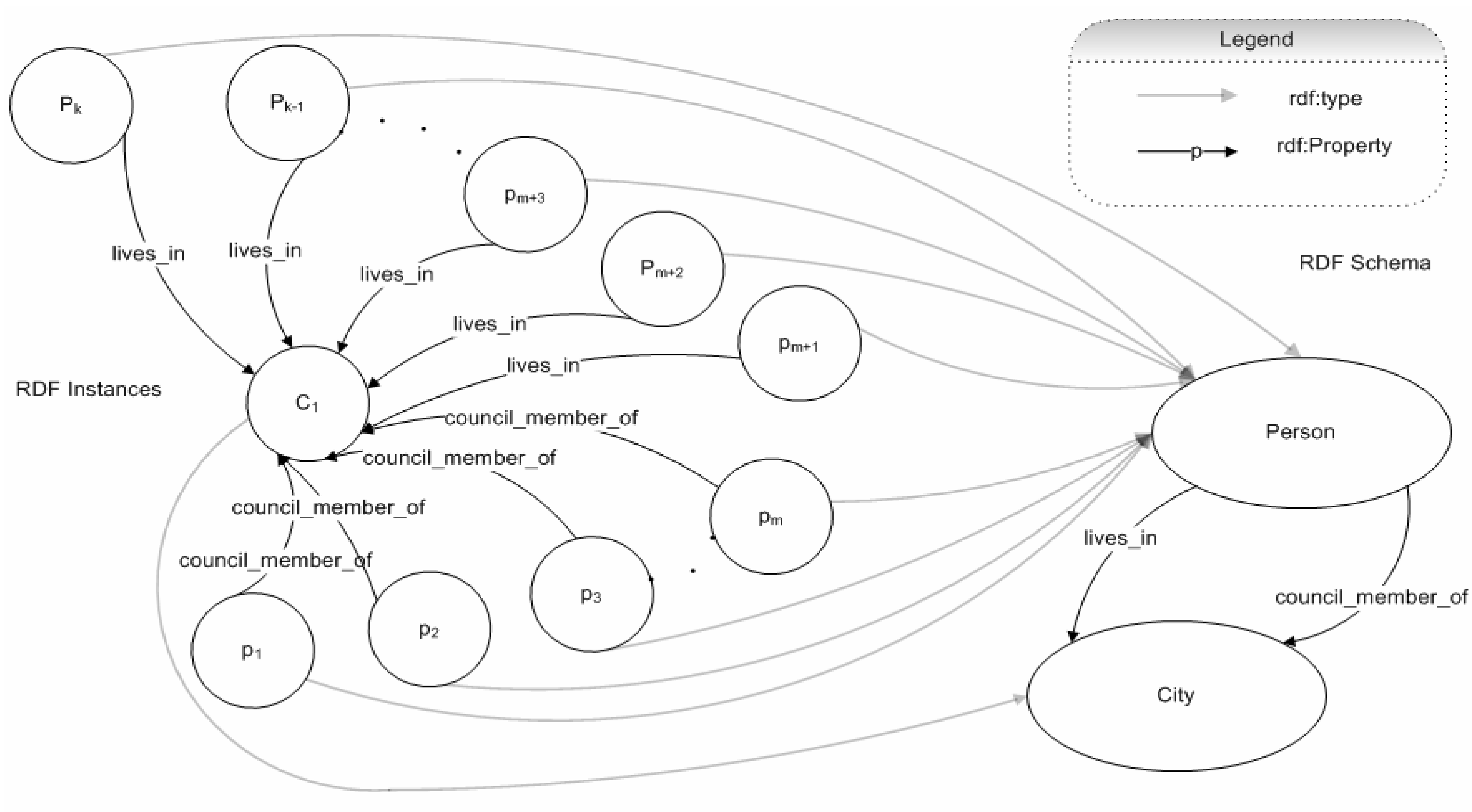
- Specificity of class c_j:

$$\mu(c_j) = \frac{d(c_j)}{d(c_{jH'})}$$

  - $d(p_{iH})$ is the depth of c_j
  - $d(p_{iH'})$ is the depth of the branch containing c_j

Instance Participation Selectivity (ISP)

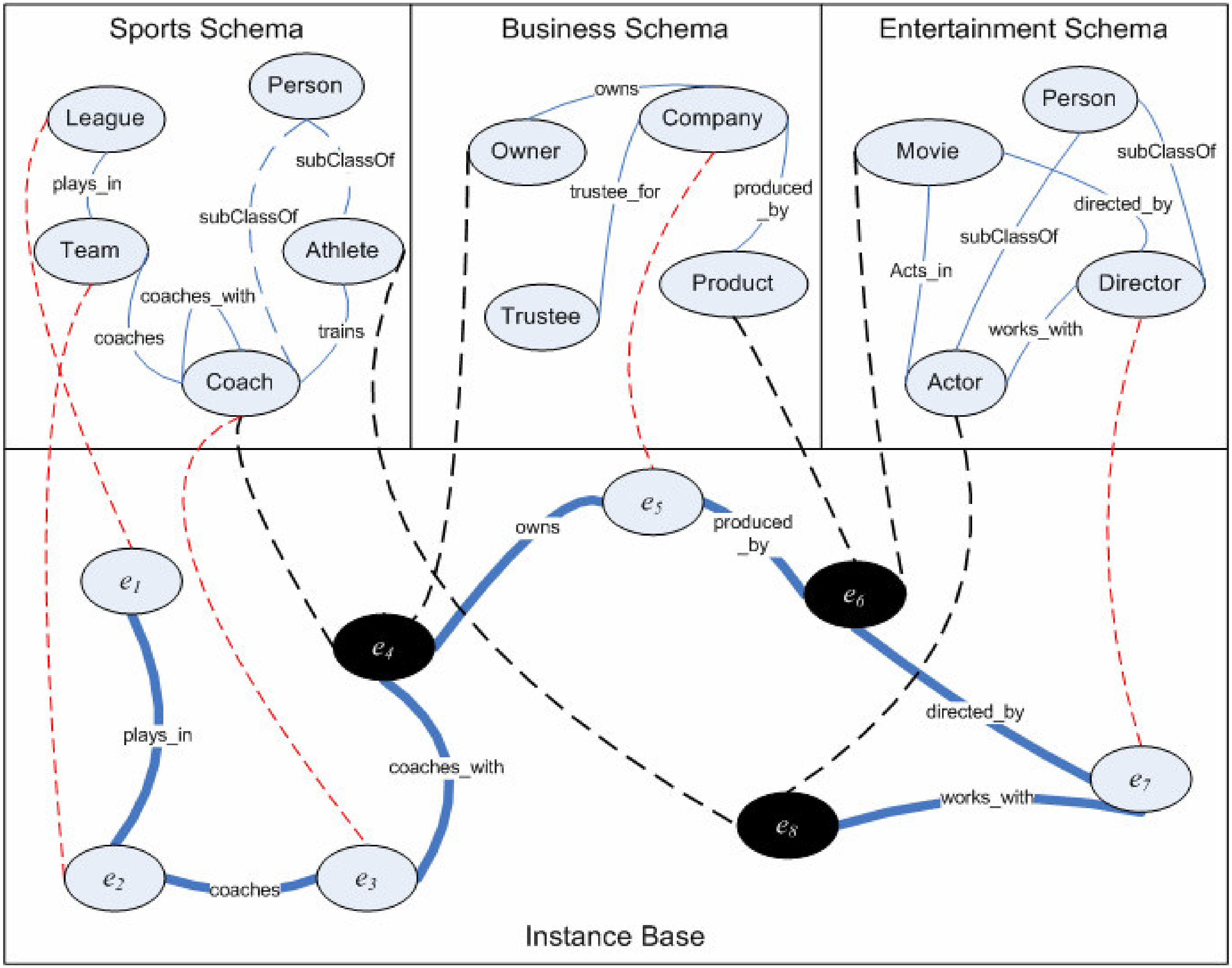
- Rare facts are more informative than frequent facts
- Define a *type* of an statement RDF $<s,p,o>$
  - Triple $\pi = <C_i,p_j,C_k>$
    - $typeOf(s) = C_i$
    - $typeOf(t) = C_k$
- $/ \pi /$ = number of statements of type π in an RDF instance base
- *ISP* for a statement: $\sigma_\pi = 1/|\pi|$



- π = <Person, lives_in, City>
- π' = <Person, council_member_of, City>
- $\sigma_\pi = 1/(k-m)$ and $\sigma_\pi' = 1/m$, and if $k-m>m$ then $\sigma_\pi' > \sigma_\pi$

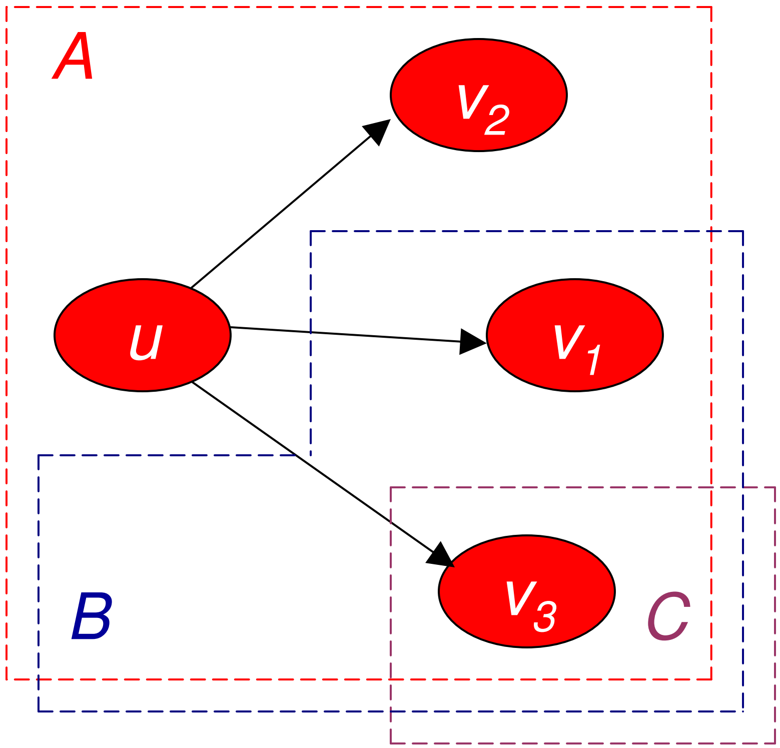
Span Heuristic (SPAN)

- RDF allows Multiple classification of entities
  - Possibly classified in different schemas
  - Tie different schemas together
- *Refraction* [3] measures how well a path conforms to a schema
  - Indicative of anomalous paths
- SPAN favors *refracting* paths


Sports Schema
Business Schema
Entertainment Schema
Person
League
plays_in
subClassOf
subClassOf
Team
Athlete
coaches_with
coaches
trains
Coach
owns
Owner
Company
trustee_for
produced
_by
Product
Trustee
Person
Movie
subClassOf
directed_by
subClassOf
Acts_in
Director
works_with
Actor
owns
produced
_by
directed_by
plays_in
coaches_with
works_with
coaches
Instance Base

Uncharted Schemas

- Schema classifications for *u:*
  - {A}
- Schema classification for v_1
  - *{A,B}*
- Schema classification for v_2
  - *{A}*
- Schema classification for v_3
  - *{A,B,C}*
- Order to favor: v_3, v_1, v_2



Schema Coverage

- *m* schemas
- How many schemas does *v* cover?

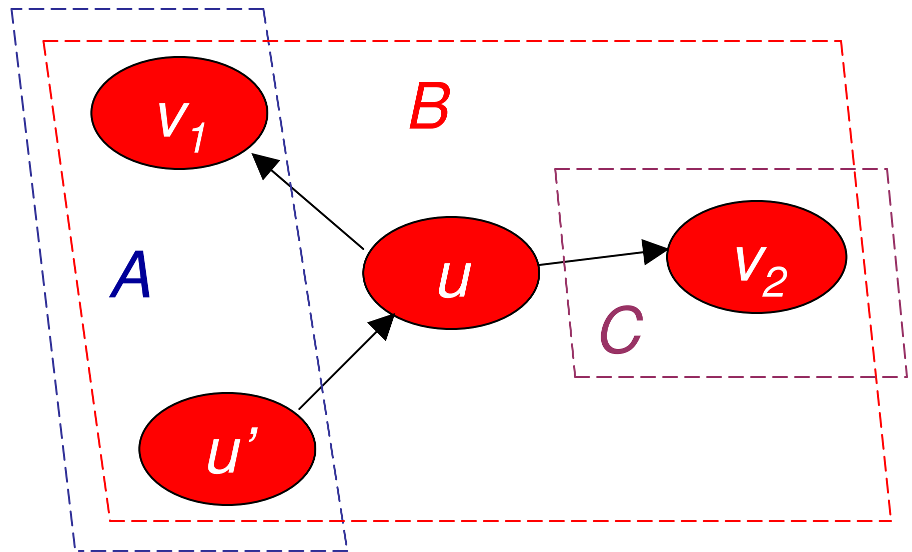
$$SchemaCover(v) = \{S \mid \exists C \in S \wedge typeOf(v) = C\}$$

- How many schemas does *(u,v)* cover?

$$\alpha(u,v) = \frac{1}{2}\left(\frac{|SchemaCover(u)| + |SchemaCover(v)|}{m}\right)$$

Always Moving Forward

SchemaCover(u')={A,B}
SchemaCover(u')={B}
SchemaCover(u')={A,B}
SchemaCover(u')={B,C}



- $\alpha(u,v_1) = \alpha(u,v_2)$
- But, more schemas are covered along (u',u,v_2) than along (u',u,v_1)

Cumulative Schema Coverage

- Schema difference between nodes

$$SDiff(u,v) = |SchemaCover(v) - SchemaCover(u)|$$

- Cumulative schema difference
  - For a two hop path *(u',u,v)*

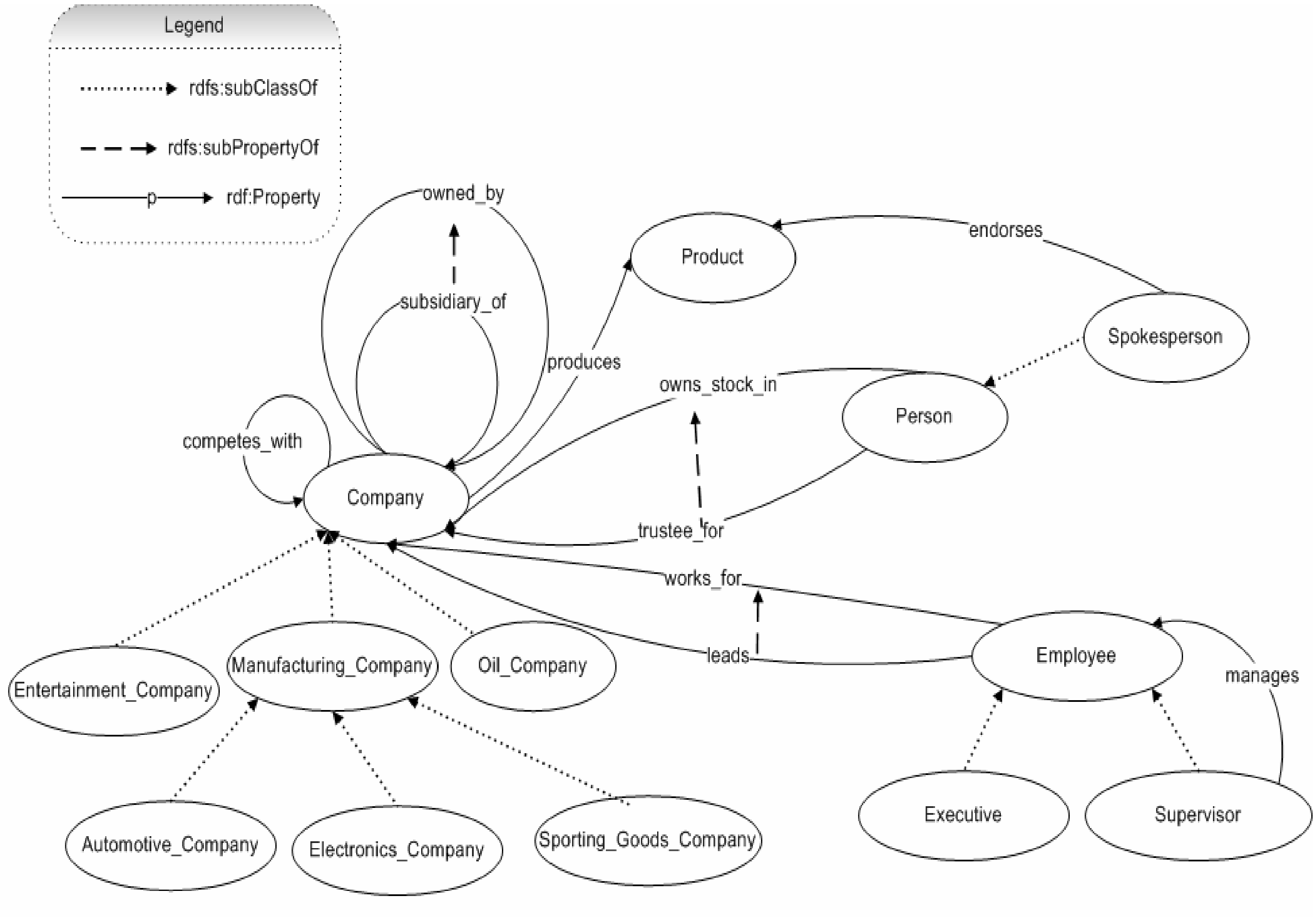
$$CSDiff(u,u',v) = 1 + SDiff(u, v) + SDiff(u', v)$$

$$\beta_{u' \rightarrow u \rightarrow v} = \frac{CSDiff}{1 + 2(m-1)}$$

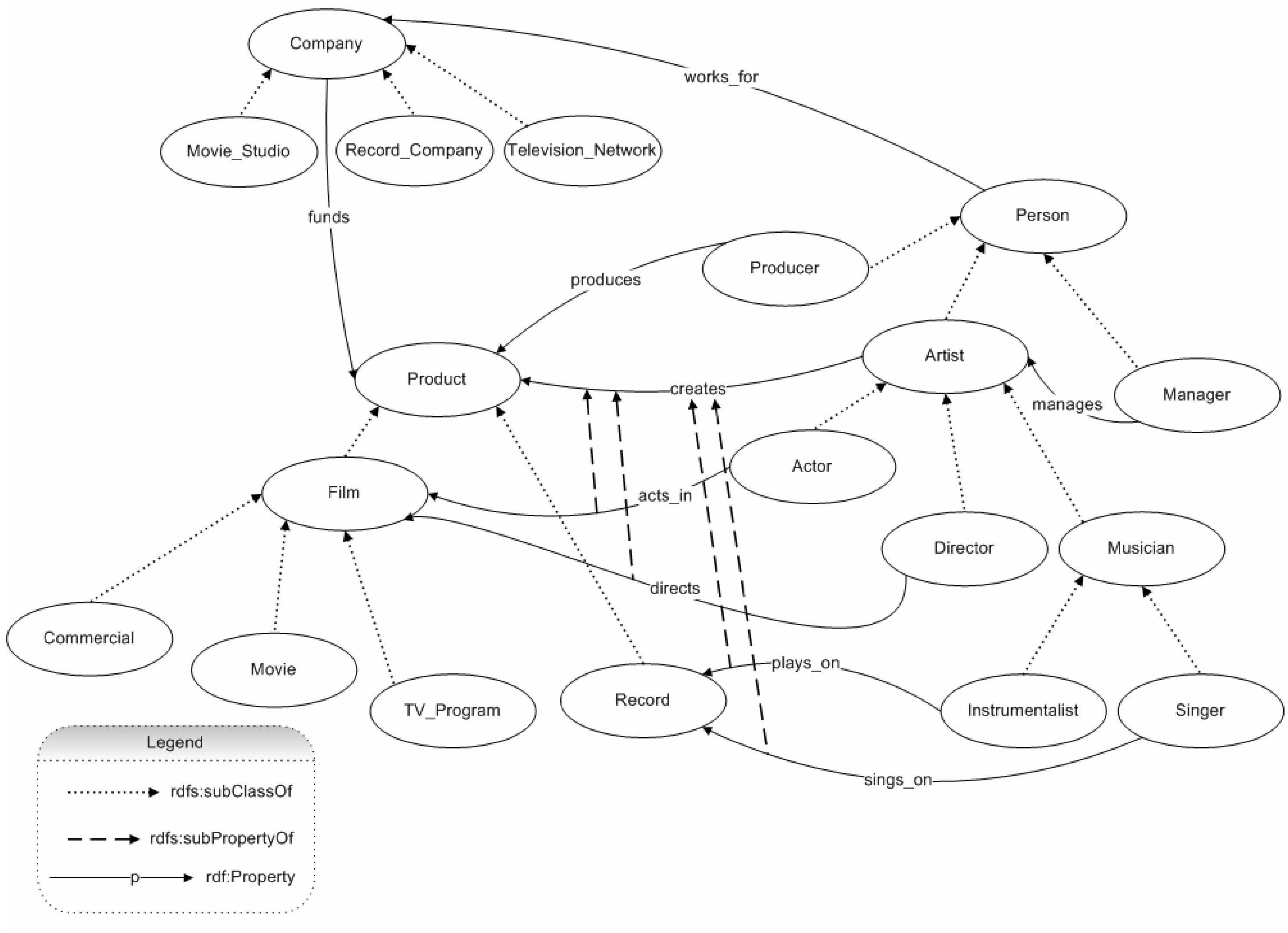
Dataset

- Obstacle:
  - Few publicly available datasets
    - Many contain sensitive information
  - Datasets do not reflect real-world distributions
- Solution:
  - Developed synthetic instance base
  - Ability to control characteristics
  - Entities classified by 3 schemas

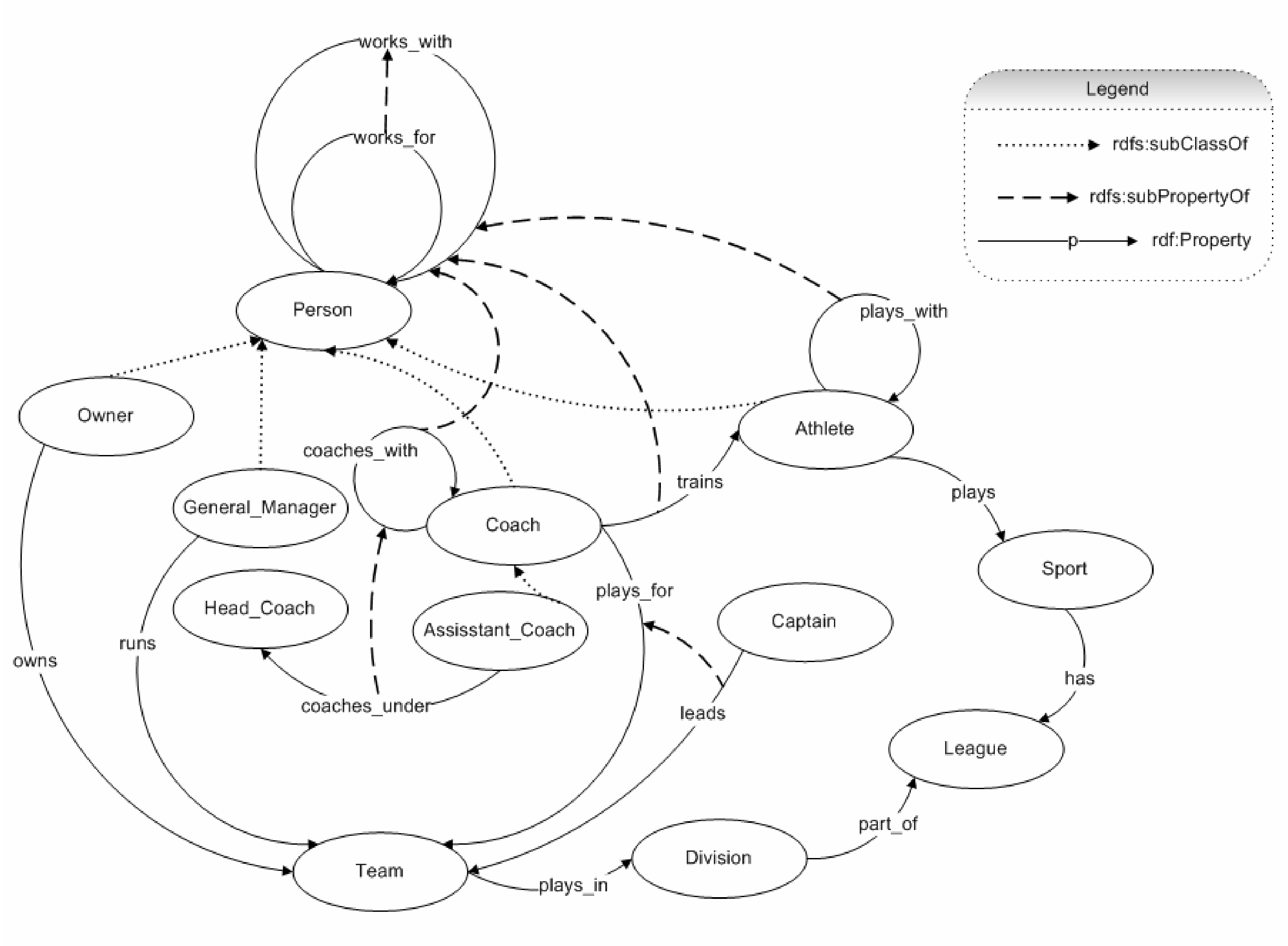
Business Schema



Entertainment Schema

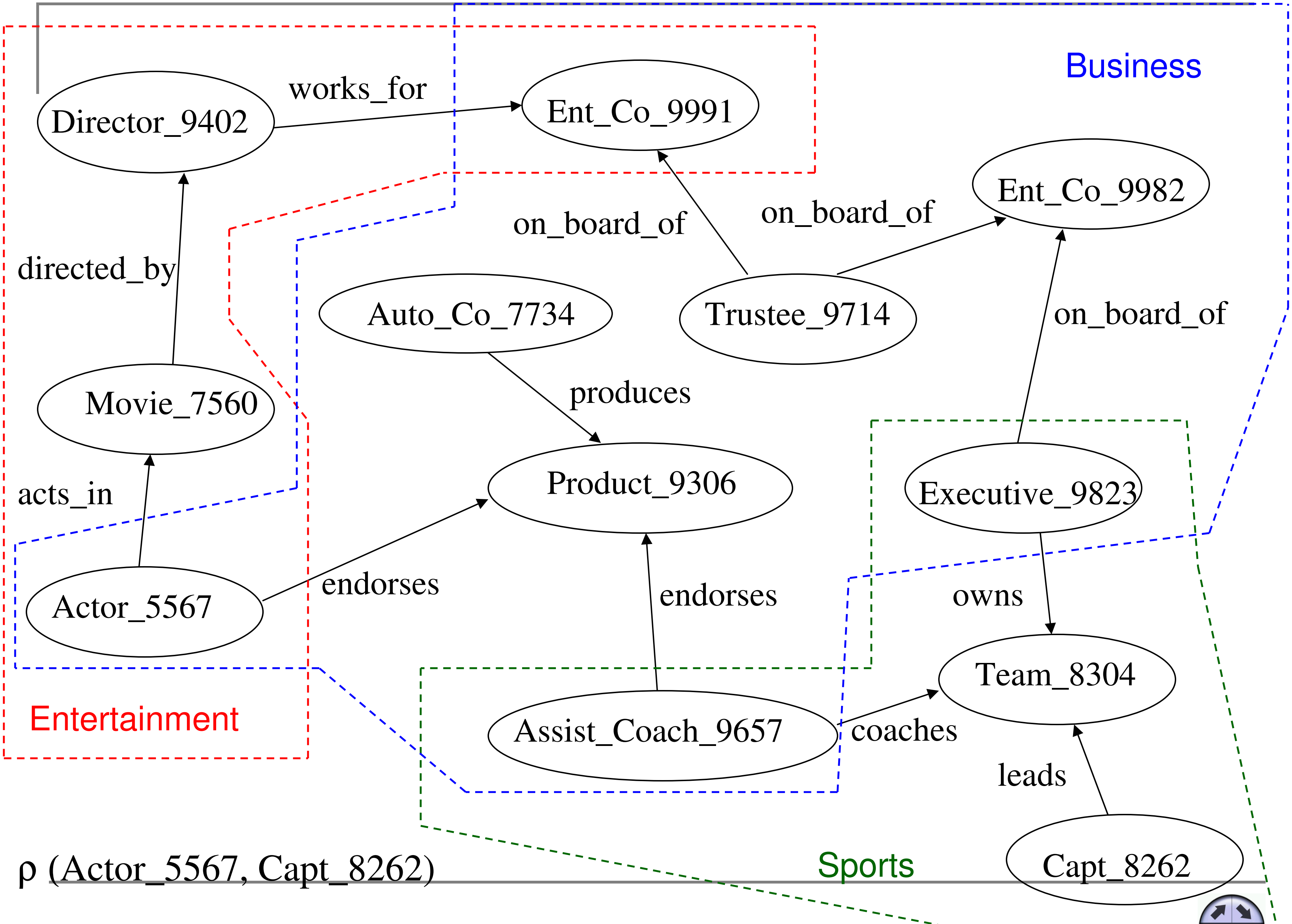


Sports Schema



Scenario

- Insider trading example
- Fraud investigator is given:
  - Stock in *Ent_Co_9991* plummeted
  - Prior to price drop:
    - *Capt_8262* sold all shares
    - *Actor_5567* sold 70% of shares
- Why did they both sell so many shares so quickly?



ρ (Actor_5567, Capt_8262)

Queries for Evaluation

- 30 queries over synthetic dataset
  - Evaluation averaged over all queries
- Evaluation:
  - All queries
  - Separate query types
- *ρ-graphs* for all combinations of heuristics
  - 4 heuristics → 2^4 → 16 possible settings

Ranking/Scoring a ρ-Graph

- Need objective measure *ρ-graph* quality
- 3 ranking schemes
  - User specified criteria: *[1]*
  - rarity of an association type: *RarityRank*
  - Relevance model: *[3]*
- How well "ranked" is a *ρ-graph*?
  - Compare to each ranking scheme

Ranking a ρ-Graph

- *$FGPaths_k$:*
  - Set of all paths found in *k-hop* limited search
  - *$CGPaths_k$*: paths in *candidate ρ-graph*
  - *$DGPaths_k$*: paths in *display ρ-graph*
- Use $k = 9$ for feasible path enumeration
  - 60 million paths when $k = 13$
- Compare *ρ-graph* to *$FGPaths_9$*

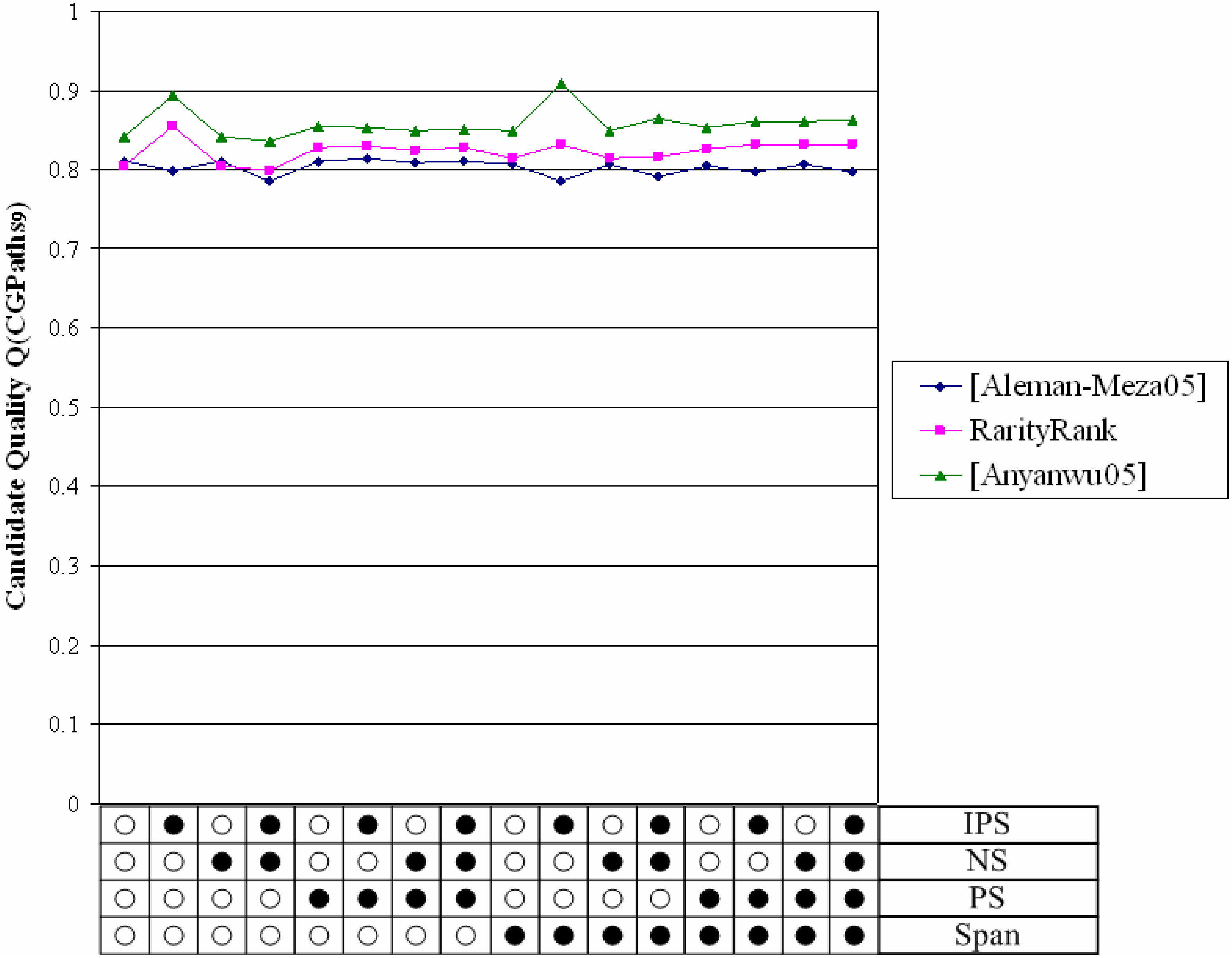
Candidate ρ-Graph Quality

1. Score each path, $p_{candidate} \in CGpath_9$:

$$score(p_{candidate}) = |FGRankedPaths| - rank(p_{candidate})$$

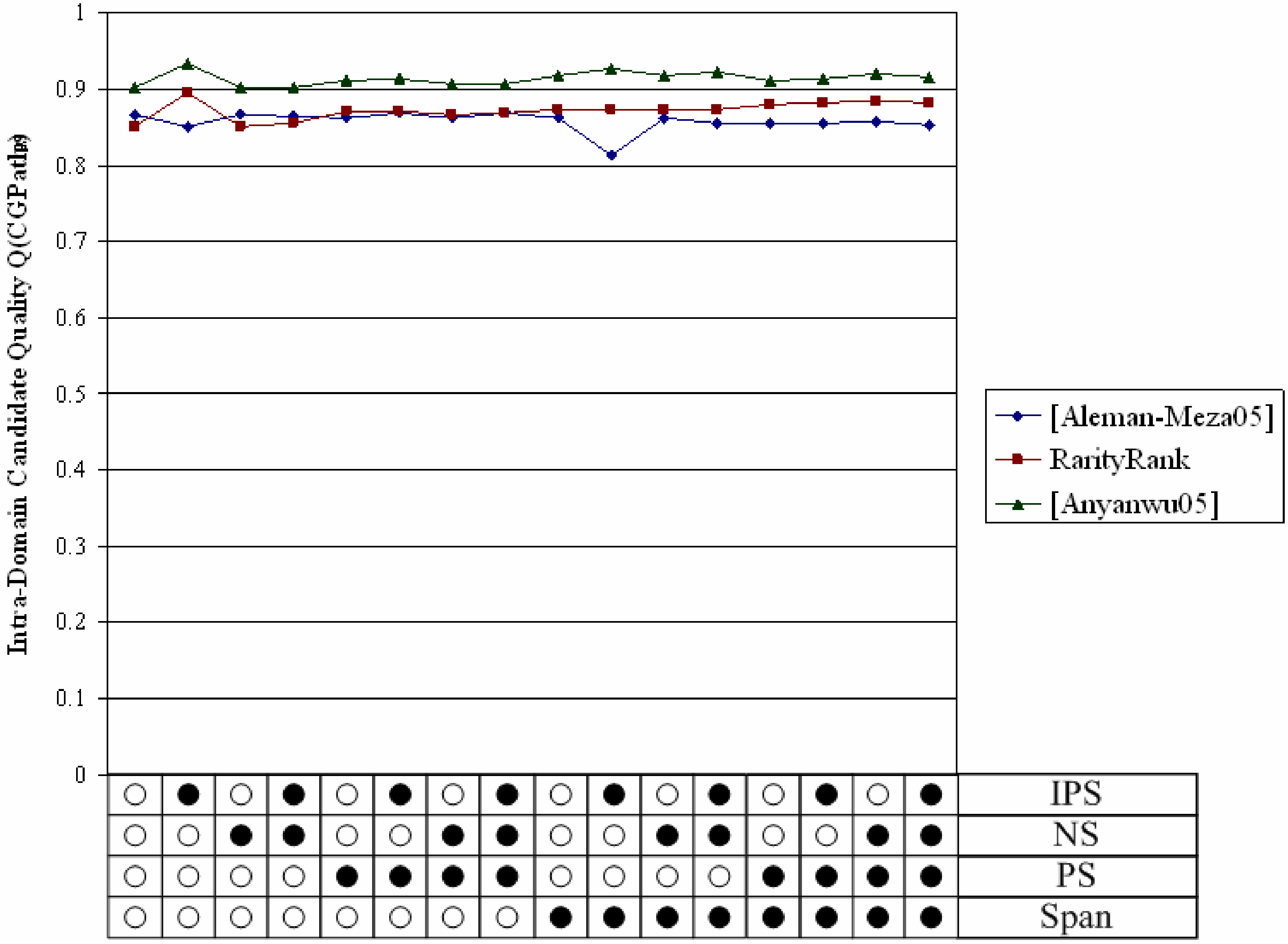
2. Score a *Candidate ρ-graph*, $Q(CGPaths_9)$:

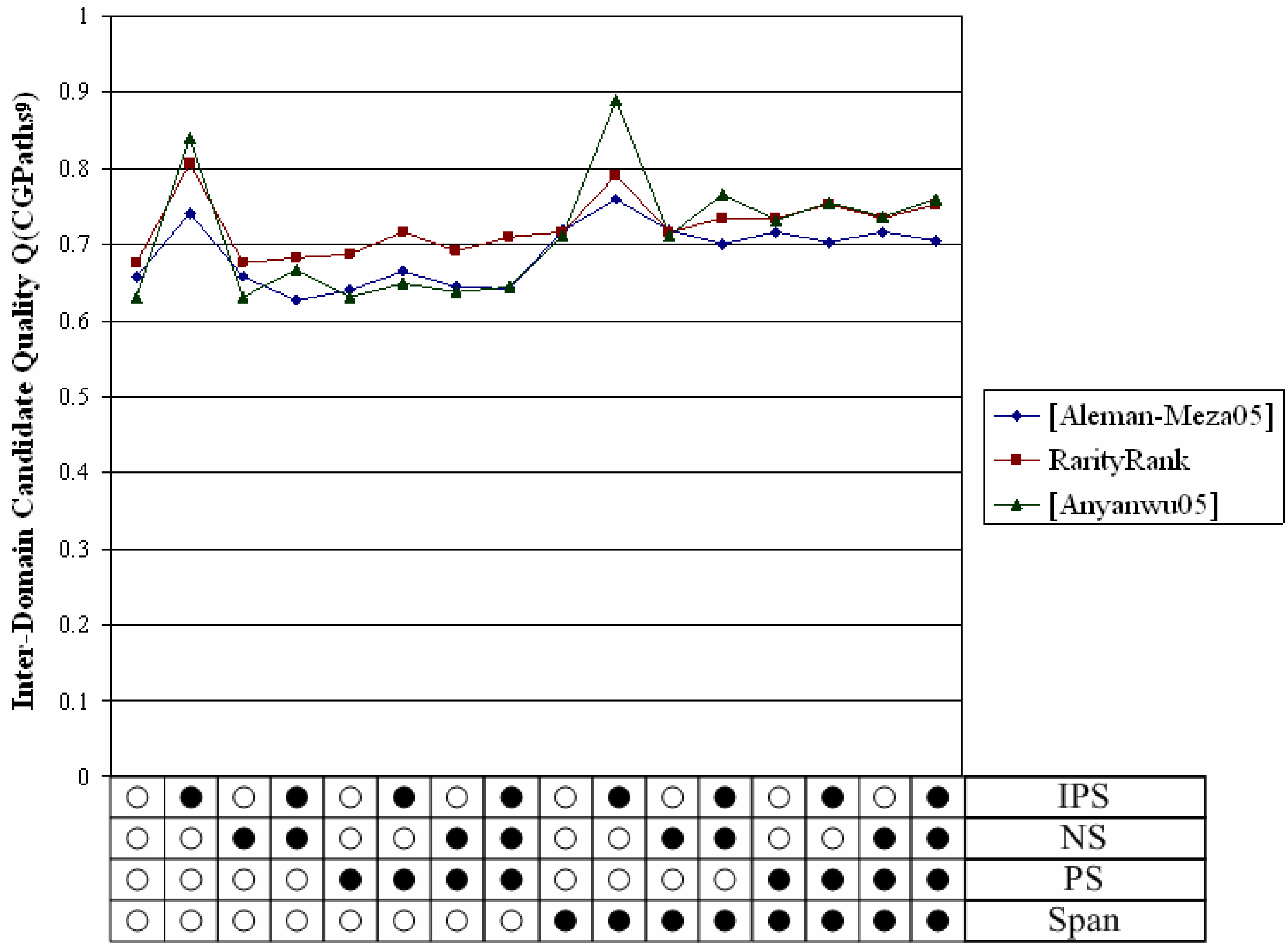
$$Q(CGPaths_9) = \frac{\sum_{p_{candidate} \in CGPaths_9} (score(p_{candidate}))}{\sum_{r=1}^{|CGPaths_9|} (|FGRankedPaths_9| - r)}$$


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Candidate Quality Q(CGPaths9)
[Aleman-Meza05]
RarityRank
[Anyanwu05]
IPS
NS
PS
Span

Types of Candidate ρ-Graph Quality

- 30 queries over synthetic dataset
  - 15 intra-domain queries
  - 15 inter-domain queries
- Quality averaged over all respective queries
- Compute *Candidate ρ-graph* quality for each type


Intra-Domain Candidate Quality Q(CGPaths)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
[Aleman-Meza05]
RarityRank
[Anyanwu05]
IPS
NS
PS
Span


Inter-Domain Candidate Quality Q(CGPaths9)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
[Aleman-Meza05]
RarityRank
[Anyanwu05]
IPS
NS
PS
Span

Display ρ-Graph Quality

- Compute a *Pseudo Display ρ-graph:*
  - Given budget *b*
  - Start with an empty subgrpah
  - Enumerate paths in $FGPaths_g$
  - Add successive paths to subgraph
  - Stop when subgraph contains *b* nodes

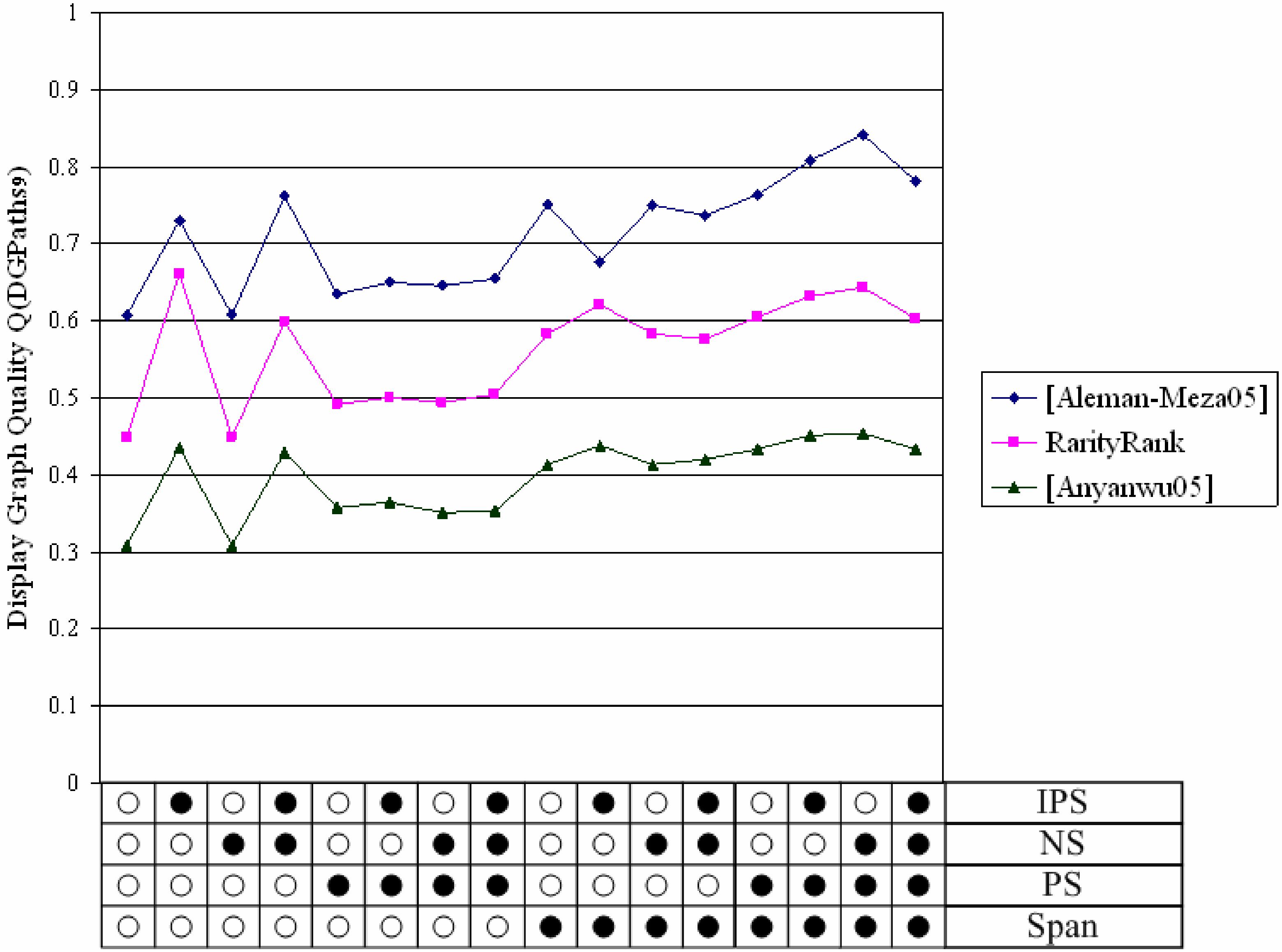
Display ρ-Graph Quality

1. Score each path, $p_{display} \in DGpaths_g$:

$$score(p_{display}) = |FGRankedPaths| - rank(p_{display})$$

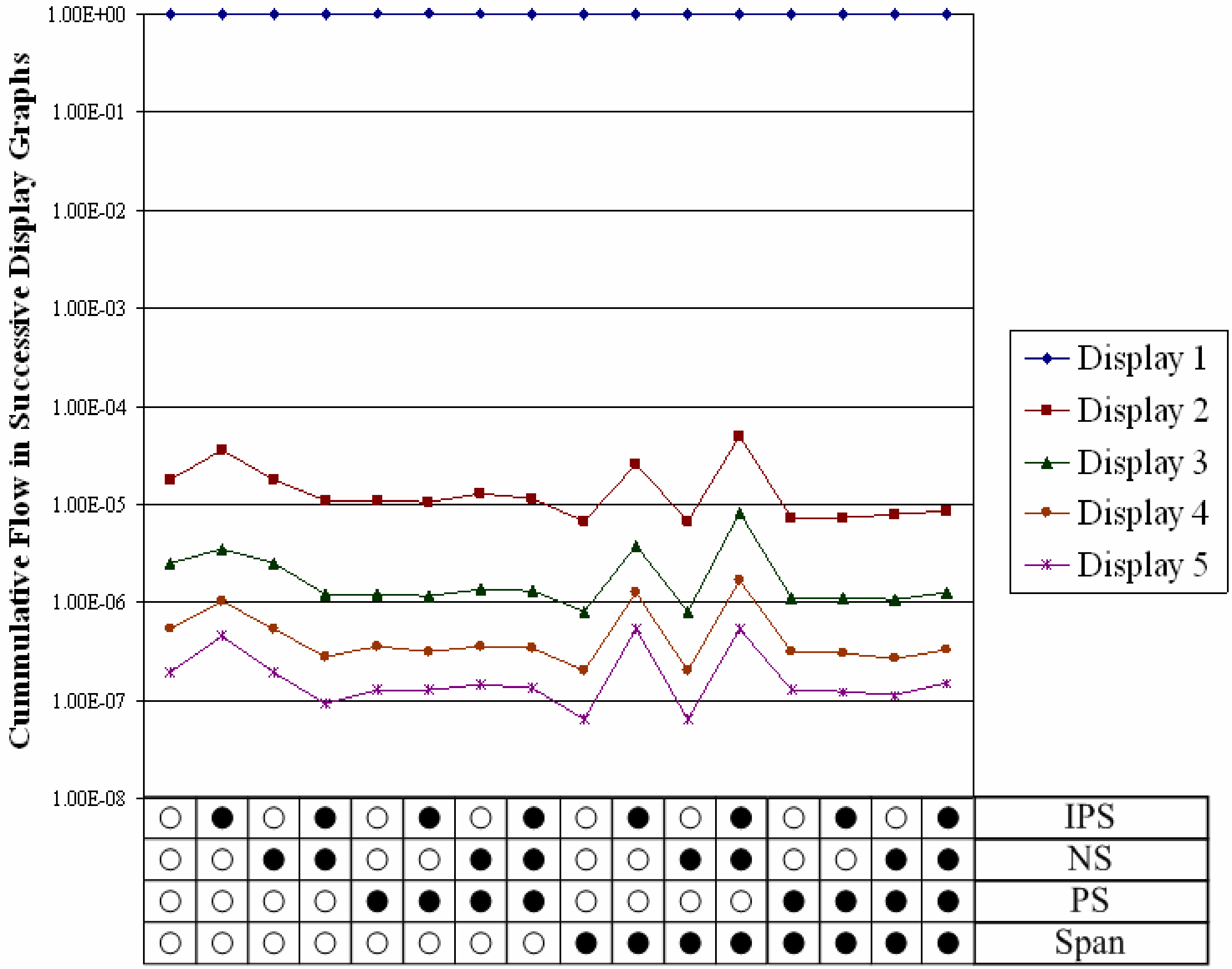
2. Score each path, $p_{display}$ $DGpaths_g$:

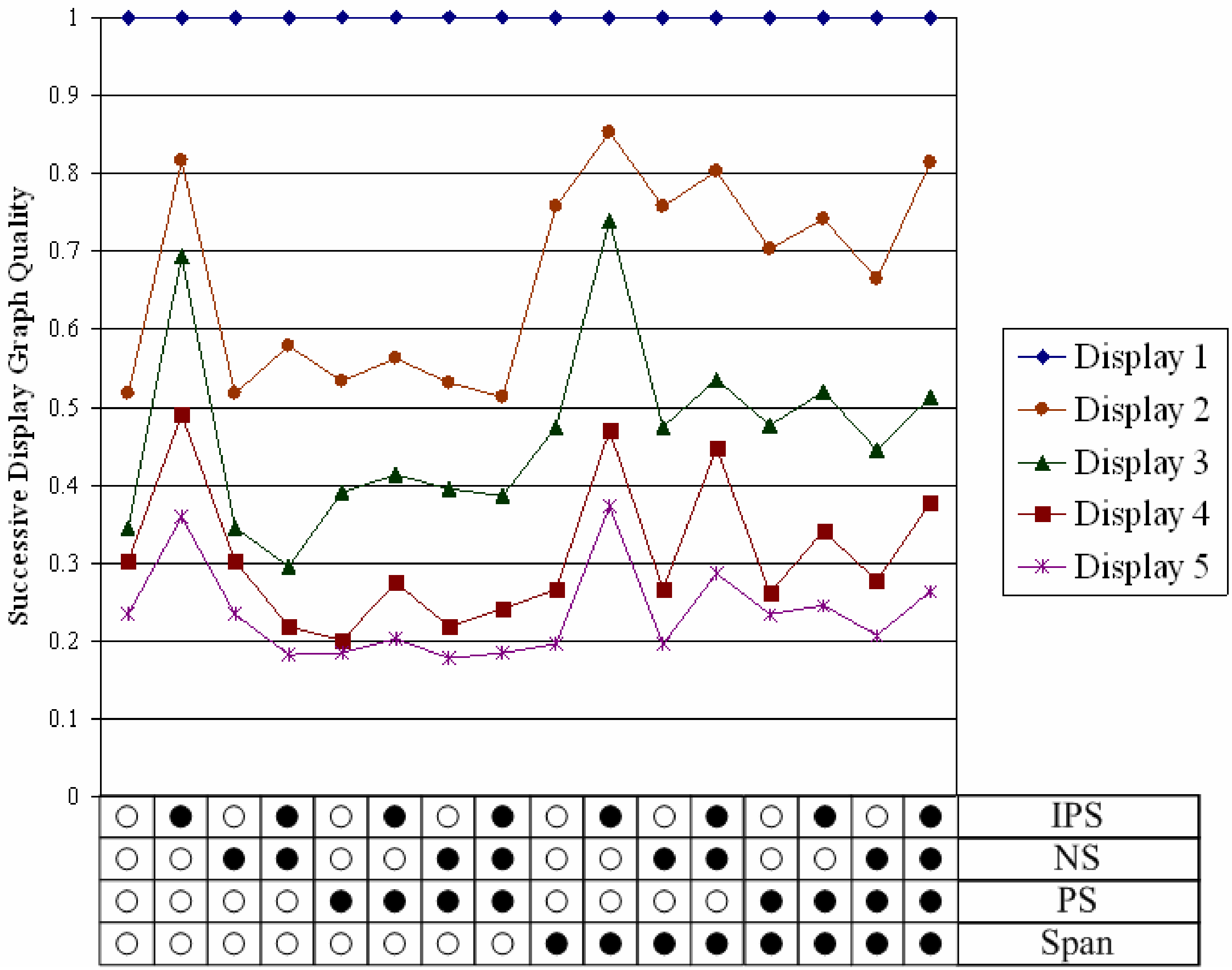
$$Q(DGPaths) = \frac{\sum_{p_{display} \in DGPaths} score(p_{display})}{\sum_{p_{pseudo} \in Pseudo-Display} score(p_{pseudo})}$$


Display Graph Quality Q(DGPaths9)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
[Aleman-Meza05]
RarityRank
[Anyanwu05]
IPS
NS
PS
Span

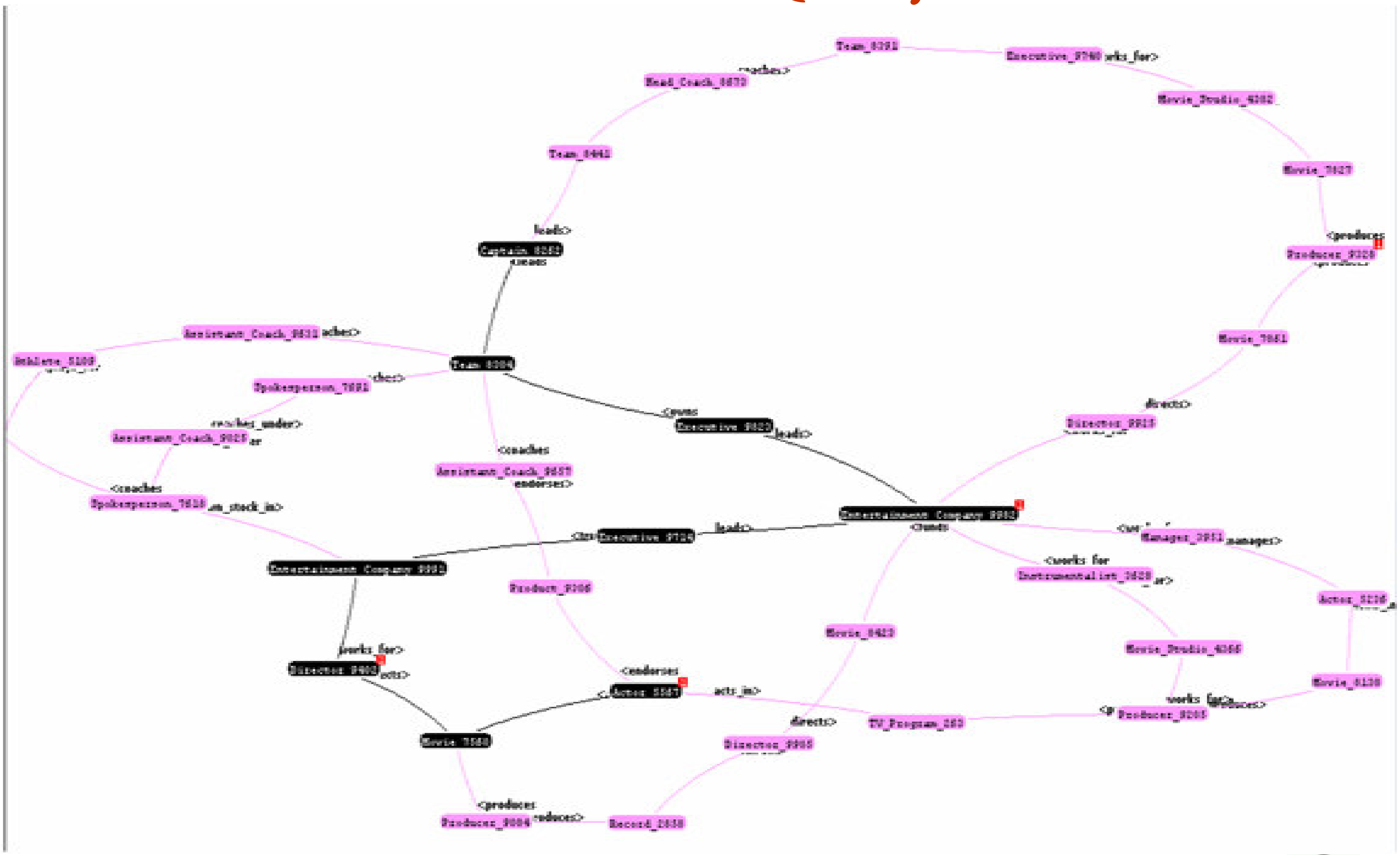
Current Flow Model

- 5 successive *Display ρ-graphs*
  - Compute the first *Display ρ-graph* as usual
  - Compute the second *Display ρ-graph* by starting with the next path of maximum delivered current
  - Continue in this manner
- Intuition:
  - Cumulative flow should decrease successively
  - Quality should decrease successively


Cummulative Flow in Successive Display Graphs
1.00E+00
1.00E-01
1.00E-02
1.00E-03
1.00E-04
1.00E-05
1.00E-06
1.00E-07
1.00E-08
Display 1
Display 2
Display 3
Display 4
Display 5
IPS
NS
PS
Span


Successive Display Graph Quality
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Display 1
Display 2
Display 3
Display 4
Display 5
IPS
NS
PS
Span

Visualizable Scenario Query Result



Timing Evaluation

- Computed time for *Candidate ρ-graph* search
  - *Candidate ρ-graph* generation and subsequent exhaustive search
- Computed time for exhaustive search over full graph
- Bidirectional join algorithm for search
  - Database of triples (and corresponding inverses)
  - Secondary indexes on triple endpoints
  - Joined the table with itself in opposite directions
- Averaged time for all 30 queries and all 16 settings of heuristics

Timing Results

k-hop limit	Full graph search in ms *(λ)*	Candidate ρ-graph search in ms *(φ)*	Ratio: ϕ / λ
5	504	2,389.313	4.740699
6	1,686	2,617.063	1.552232
7	17,354	3,808.938	0.219485
8	1,261,099	7,6063.88	0.060316

Conclusions

- Developed heuristics loosely based on semantics for *semantic association* discovery
- Applied heuristics to compute edge weights
- Presented empirical evaluation of sugraph generation algorithms

Contributions

- Adapted algorithms in [4]:
  - Use *degree(u) + degree(v)* in distance measurement
    - Allowed by main-memory RDF representation
  - Apply algorithms to graphs with multiple edge types
  - Compute edge weights using semantic based heuristics

Future Work

- Use *closeness centrality* for *Candidate ρ-graph* algorithm
  - Expand the next pending node which is closest to the given endpoints
- *n*-point operator
  - Compute a relevant subgraph given *n* endpoints

Future Work

- Formalize the notion of context
  - *Context-aware subgraph discovery*
  - Define context based on query results
- Evaluate based on distance thresholds
  - Given a threshold for maximum distance of a path
  - Compare two sets of paths:
    1. All paths in a *ρ-graph* not exceeding the threshold
    2. All paths in the full graph not exceeding the threshold
  - What is the quality of such paths in the *ρ-graph*?

References

[1] Boanerges Aleman-Meza, Christian Halaschek-Wiener, I. Budak Arpinar, Cartic Ramakrishnan, and Amit Sheth. Ranking Complex Relationships on the Semantic Web. To Appear in *IEEE Internet Computing, Special Issue - Information Discovery: Needles & Haystacks May-June 2005.*

[2] *B. Aleman-Meza, C. Halaschek, A. Sheth, I. B. Arpinar, and G. Sannapareddy, "SWETO: Large-Scale Semantic Web Test-bed", In Proceedings of the 16th International Conference on Software Engineering & Knowledge Engineering (SEKE2004): Workshop on Ontology in Action, Banff, Canada, June 21-24, 2004, pp. 490-493.*

[3] Kemafor Anyanwu, Angela Maduko, Amit Sheth, SemRank: Ranking Complex Relationship Search Results on the Semantic Web. The 14th International World Wide Web Conference, (WWW2005), Chiba, Japan, May 10-14, 2005

References

[4] Christos Faloutsos, Kevin S. McCurley, Andrew Tomkins: Fast discovery of connection subgraphs. KDD 2004: 118-127.

[5] Thomas Gruber. It Is What It Does: The Pragmatics of Ontology. Invited presentation to the meeting of the CIDOC Conceptual Reference Model committee, Smithsonian Museum, Washington, D.C., March 26, 2003.

[6] Shou-de Lin, Hans Chalupsky: Unsupervised Link Discovery in Multi-relational Data via Rarity Analysis. ICDM 2003: 171-178

[7] I. Polikoff and D. Allemang, “Semantic Technology,” TopQuadrant Technology Briefing v1.1, September 2003. http://www.topquadrant.com/documents/TQ04_Semantic_Technology_Briefing.PDF

References

[8] Amit Sheth. Enterprise Applications of Semantic Web: The Sweet Spot of Risk and Compliance. Invited paper: IFIP International Conference on Industrial Applications of Semantic Web (IASW2005), Jyväskylä, Finland, August 25-27, 2005. http://www.cs.jyu.fi/ai/OntoGroup/IASW-2005/

Question & Comments

Thank You!