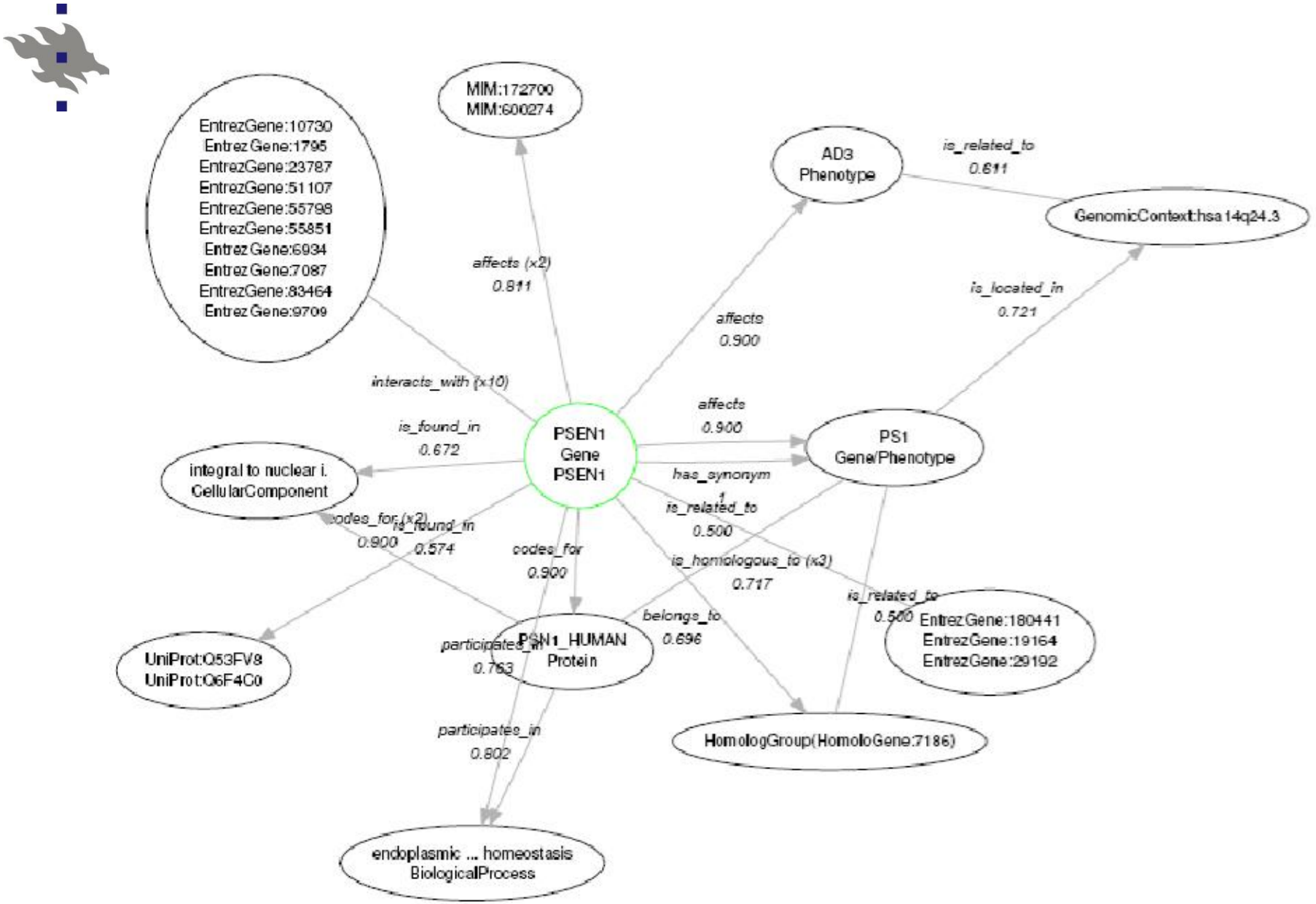
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Biomine search engine for probabilistic graphs

Hannu Toivonen
University of Helsinki

JSI, Slovenia, 2010

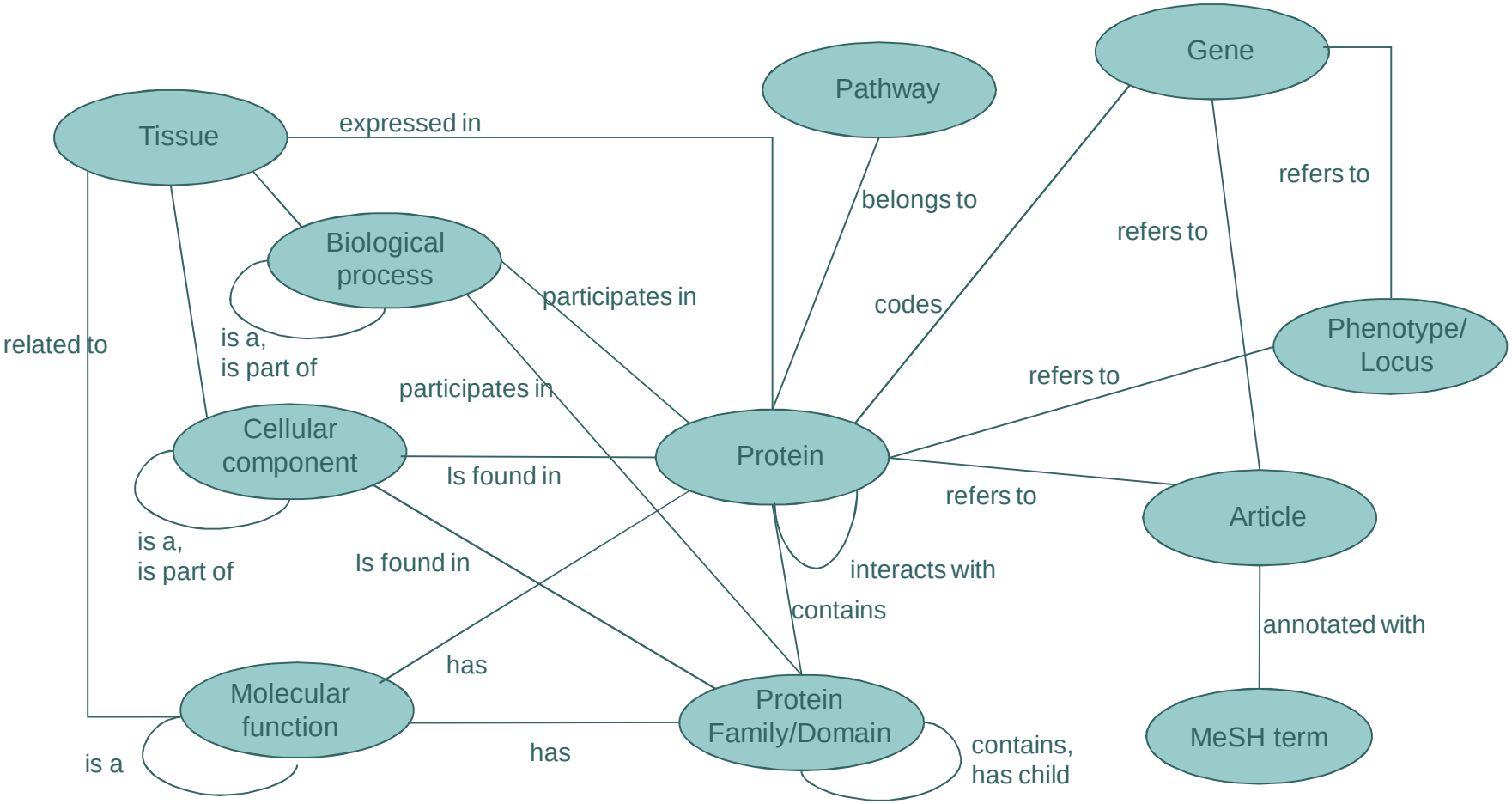
What is known about PSEN1 (presenilin1) gene?


EntrezGene:10730
EntrezGene:1795
EntrezGene:23787
EntrezGene:51107
EntrezGene:55798
EntrezGene:55851
EntrezGene:6934
EntrezGene:7087
EntrezGene:83464
EntrezGene:9709
MIM:172700
MIM:600274
AD3 Phenotype
is_related_to
0.611
GenomicContext:hsa 14q24.3
affects (x2)
0.811
affects
0.900
is_located_in
0.721
interacts_with (x10)
is_found_in
0.672
affects
0.900
PSEN1 Gene PSEN1
PS1 Gene/Phenotype
integral to nuclear i. CellularComponent
has_synonym
1
is_related_to
0.500
codes_for (x2)
0.900
is_found_in
0.574
codes_for
0.900
is_homologous_to (x3)
0.717
is_related_to
0.500
belongs_to
0.696
EntrezGene:180441
EntrezGene:19164
EntrezGene:29192
UniProt:Q53FV8
UniProt:Q6F4C0
participates_in
0.763
PSN1_HUMAN Protein
participates_in
0.802
HomologGroup(HomoloGene:7186)
endoplasmic ... homeostasis BiologicalProcess

Biomine: Search in biological graphs

- A graphical representation of biological data
  - *Nodes*: genes, proteins, tissues, processes, pathways, homology groups, phenotypes, …
  - *Edges*: known, reported or predicted relationships between nodes
  - Edges have weights to describe their certainty (and relevance and informativeness)
- A data mining goal: discovery of novel, non-trivial (indirect) relationships
  - E.g. possible explanations for a biological hypothesis, or discovery of new hypotheses

Biomine graph schema



- Edge weight = probability

Databases and nodes indexed by Biomine

Source databases:

Entrez Gene
Entrez Protein
GO
HGNC
HomoloGene
InterPro
KEGG
MIM
MeSH
PubMed
STRING
UniProt

Node types and counts:

Article	409219
Protein	355188
Gene	175230
HomologGroup	39493
GO	25875
Ligand	24149
Compound	15003
BiologicalProcess	14919
GenomicContext	14730
OrthologGroup	11345
MolecularFunction	8789
Drug	6637
Phenotype	6331

…

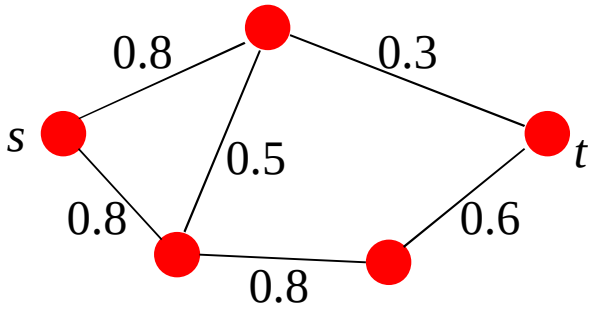
#Nodes: 1 083 891

#Edges: 6 653 464

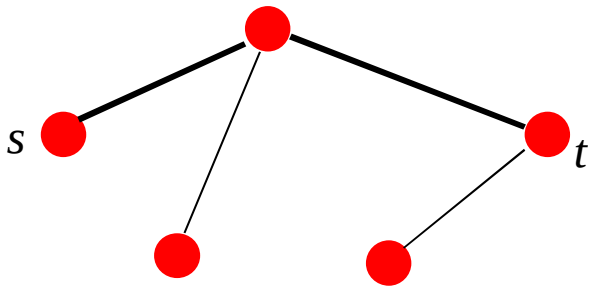
Probabilistic (Bernoulli random) graphs

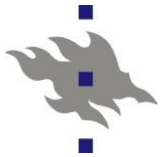
- A weighted graph G=(V, E, P)
- V, E as in standard graphs
- P(e) is the probability of e in E
  - Edge e is true (or exists) with probability P(e)
  - Edges are mutually independent

Probabilistic graph G



A random realization of G





Connectivity between nodes

- An elementary question: how strongly are two nodes s and t connected?
  - Given a node s, search for nodes t that are strongly connected to s
  - Given nodes s, t1, t2, ..., rank t1, t2, ... by their connectivity to s

Measures of connectivity

- *Reliability*: the probability that nodes s, t are connected in the probabilistic graph (i.e., that there exists a path of true edges connecting s and t)
  - Known as two-terminal network reliability from 60's
- Simple alternative: *probability of the best path* connecting s and t

Properties of reliability

- Penalizes long paths (long chains of uncertain inference)
- Rewards parallelism (alternative explanations)
- A natural probabilistic interpretation

- Related models and measures
  - Maximum network flow
    - does not penalize path length
  - Current in resistor networks (Faloutsos et al., 2004)
    - no easy intuitive interpretation
  - Expected time to meeting/arrival in random walks (SimRank; Jeh&Widom, 2002)
    - does not reward parallelism

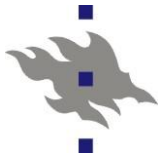
Notes on computation

- Computing the probability of the best path: trivial
- Finding the best path
  - Can be solved with shortest path algorithms

- Computing (two-terminal network) reliability
  - Investigated since the 1960's
  - NP-hard (Valiant 1970)
  - Approximation methods
    - Monte Carlo simulation
    - Exact computation (with BDDs) for a subgraph
    - Lower (and upper) bounds by exact computation
    - Series-parallel reductions
    - ...

Origin of probabilities in Biomine

- Probabilities are computed from three factors
  - Reliability of the link source
    - Method or database specific, e.g., based on sequence similarity or strength of association
  - Relevance to the user
    - Subjective view of what is interesting
  - Rarity of the link
    - Informativeness of an edge, low for nodes with a high degree
- Reliability, relevance, rarity are in [0,1]
- Edge probability = reliabity x relevance x rarity



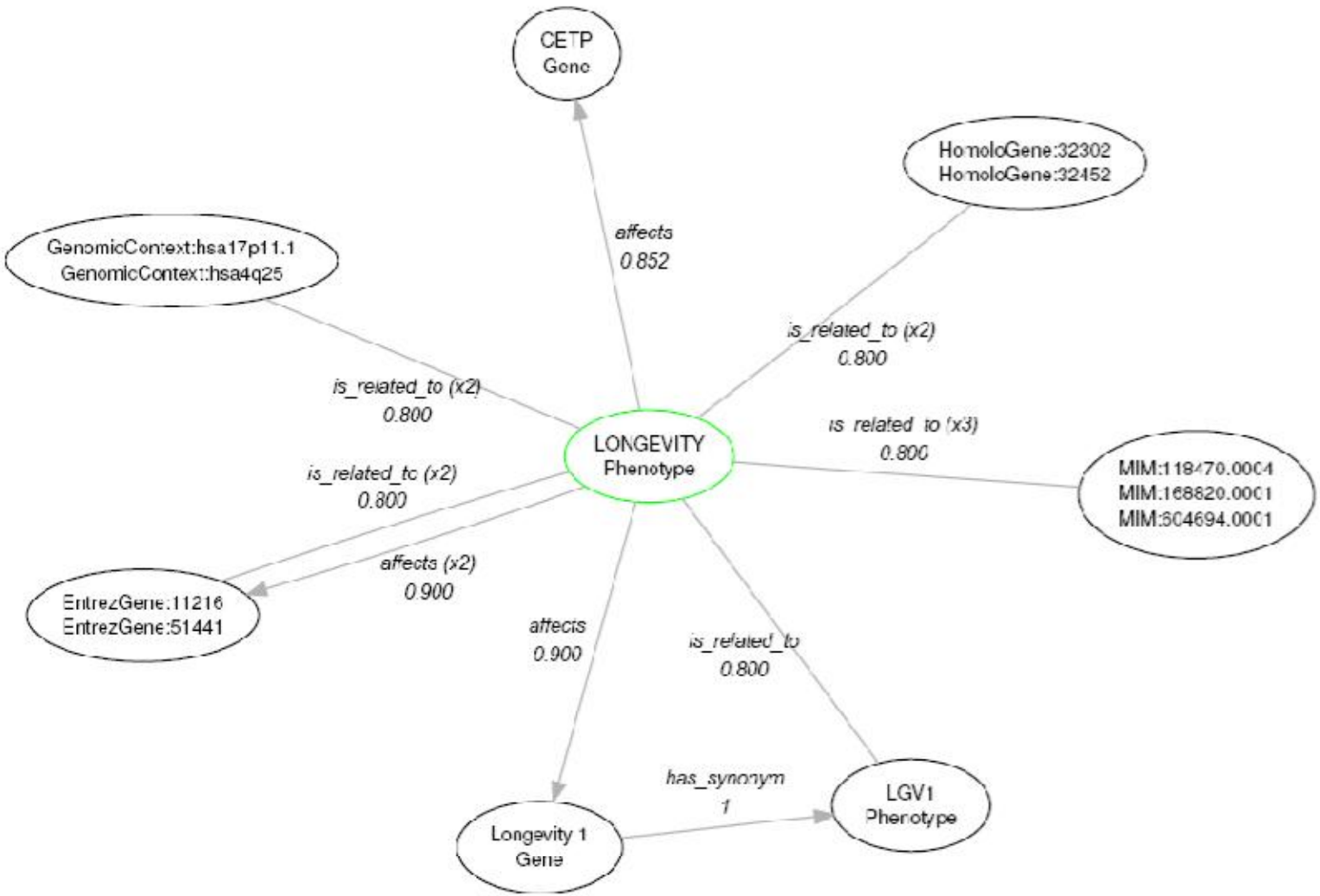
Two search problems

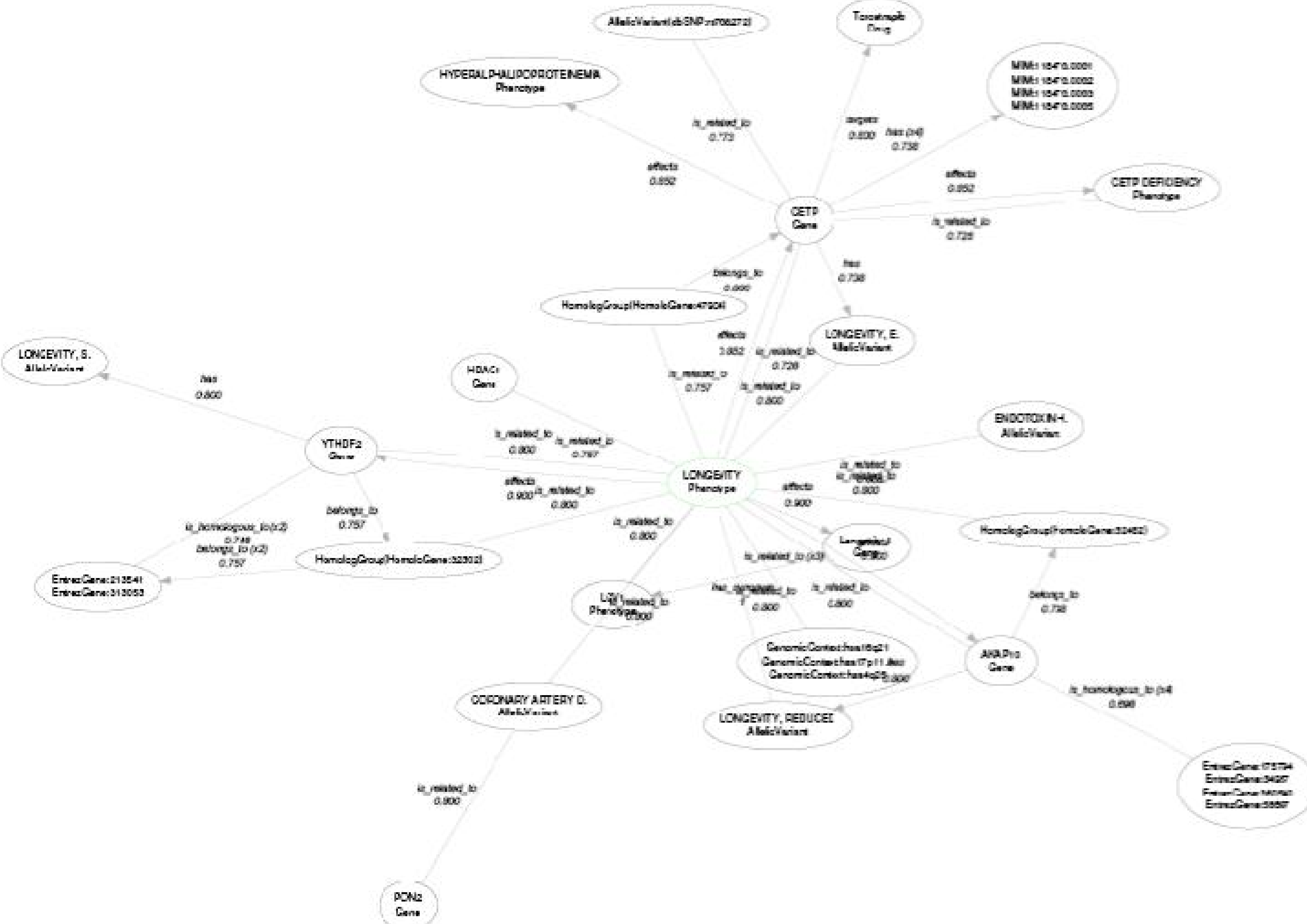
- Consider search types where
  - input consists of a node or a set of nodes
  - output is a subgraph (or a set of nodes)
- and where the general goal is to
  - maximise the probability that nodes in the output are connected to nodes in the input (i.e., the reliability of the output graph wrt the input nodes)

1. Neighborhood query

- Given a query node s, retrieve its neighbors
- (Or, given a set of query nodes, return the union of their neighborhoods)

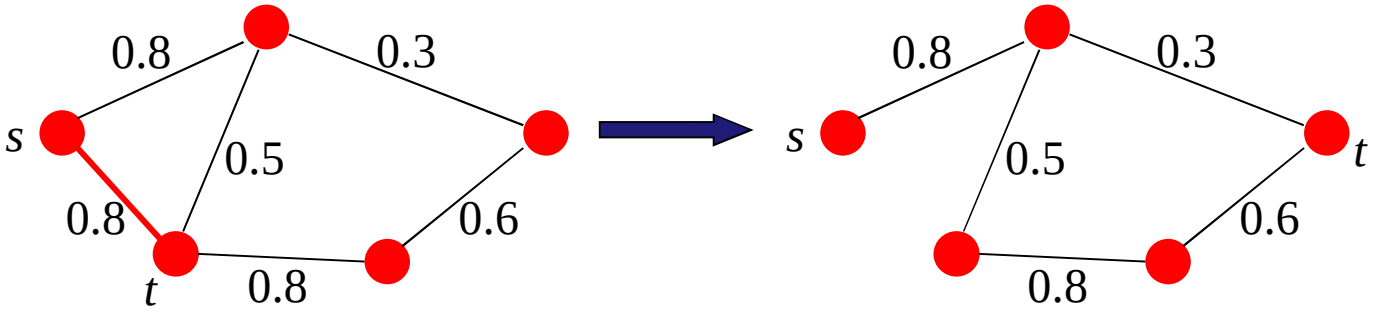
- Find those k nodes that have the highest reliability of being connected with node s


CETP
Gene
HomoloGene:32302
HomoloGene:32452
GenomicContext:hsa17p11.1
GenomicContext::hsa4q25
affects
0.852
is_related_to (x2)
0.800
is_related_to (x2)
0.800
LONGEVITY
Phenotype
is_related_to (x3)
0.800
MIM:118470.0004
MIM:168820.0001
MIM:604694.0001
is_related_to (x2)
0.800
affects (x2)
0.900
EntrezGene:11216
EntrezGene:51441
affects
0.900
is_related_to
0.800
has_synonym
1
Longevity 1
Gene
LGV1
Phenotype


HYPERALPHALIPOPROTEINEMIA
Phenotype
CETP
Gene
CETP DEFICIENCY
Phenotype
LONGEVITY, E.
AllelicVariant
LONGEVITY, S.
AllelicVariant
YTHDF2
Gene
LONGEVITY
Phenotype
ENDOTOXIN-H.
AllelicVariant
CORONARY ARTERY D.
AllelicVariant
LONGEVITY, REDUCED
AllelicVariant
affects
has
belongs_to
is_related_to

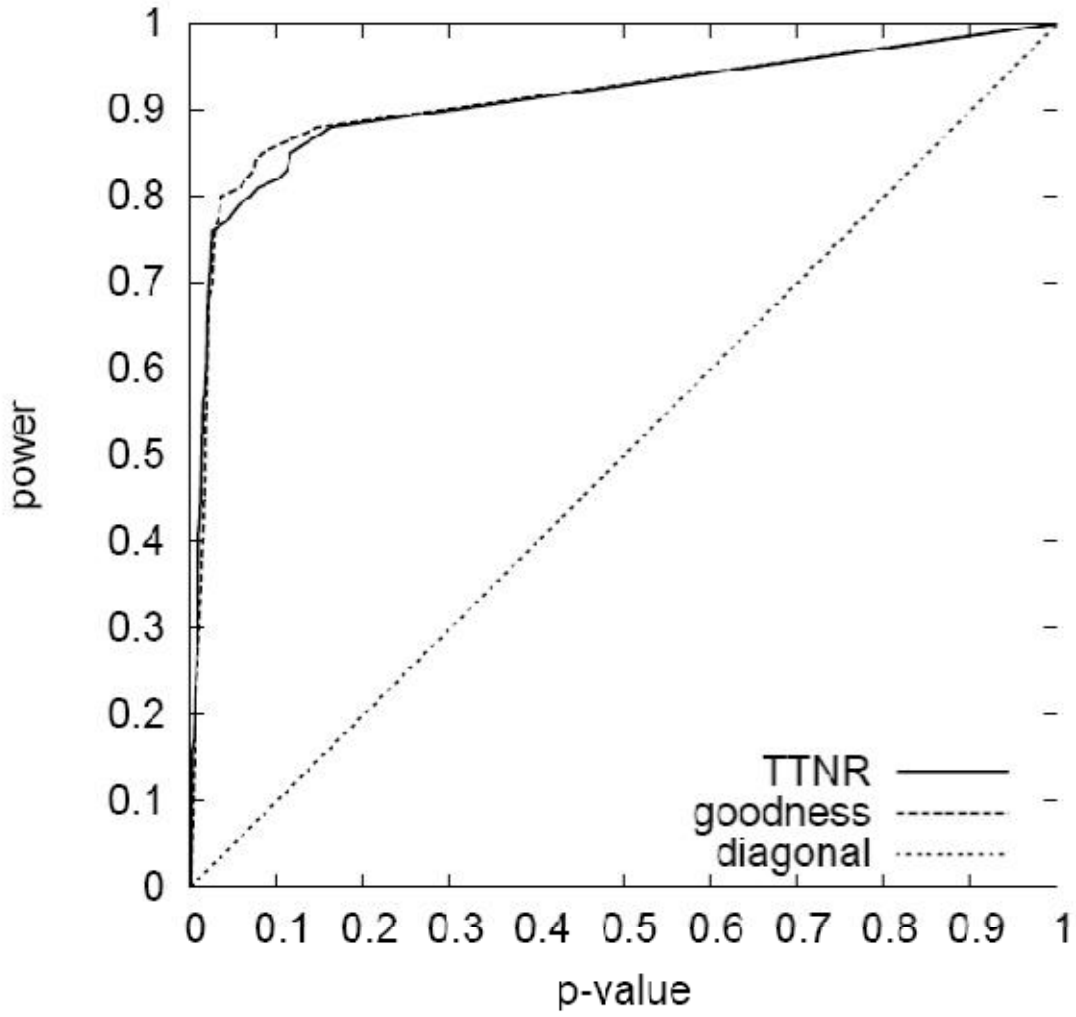
Are neighborhood queries useful?

- Test setting
  - Use a hold-out set of edges
  - For each hold-out edge (s, t), compute the reliability of the graph wrt. s and t



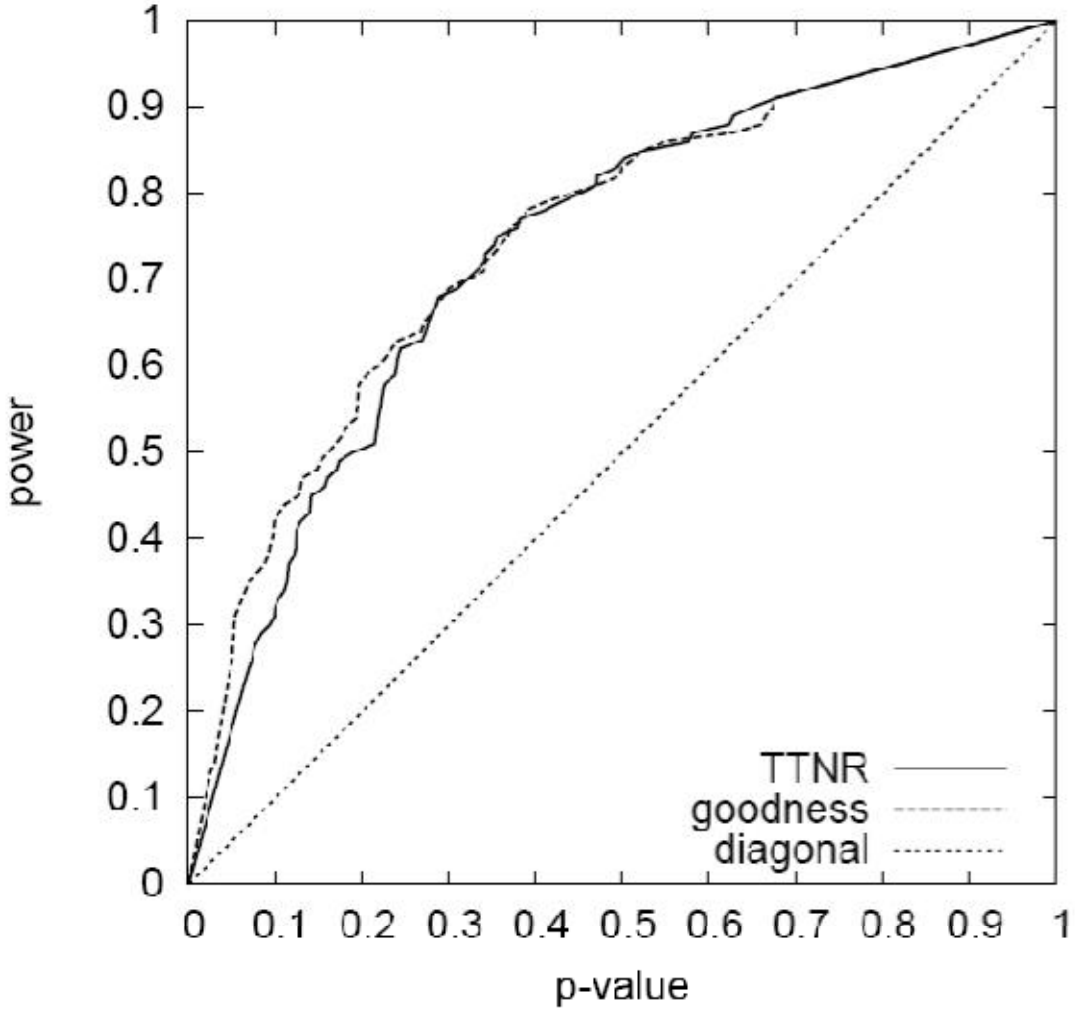
  - Compute the reliability for random node pairs (chosen to be similar to s and t) (null distribution, ~negative examples)
  - Obtain a p-value for edge (s, t)

Prediction of missing protein interactions

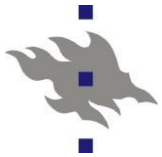


- (Gene interactions and gene-phonotype relations were also removed)

Prediction of future gene interactions

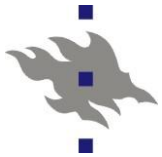


- (Note: comparison is against interactions discovered within the next six months, not true ones.)



Are neighborhood queries useful?

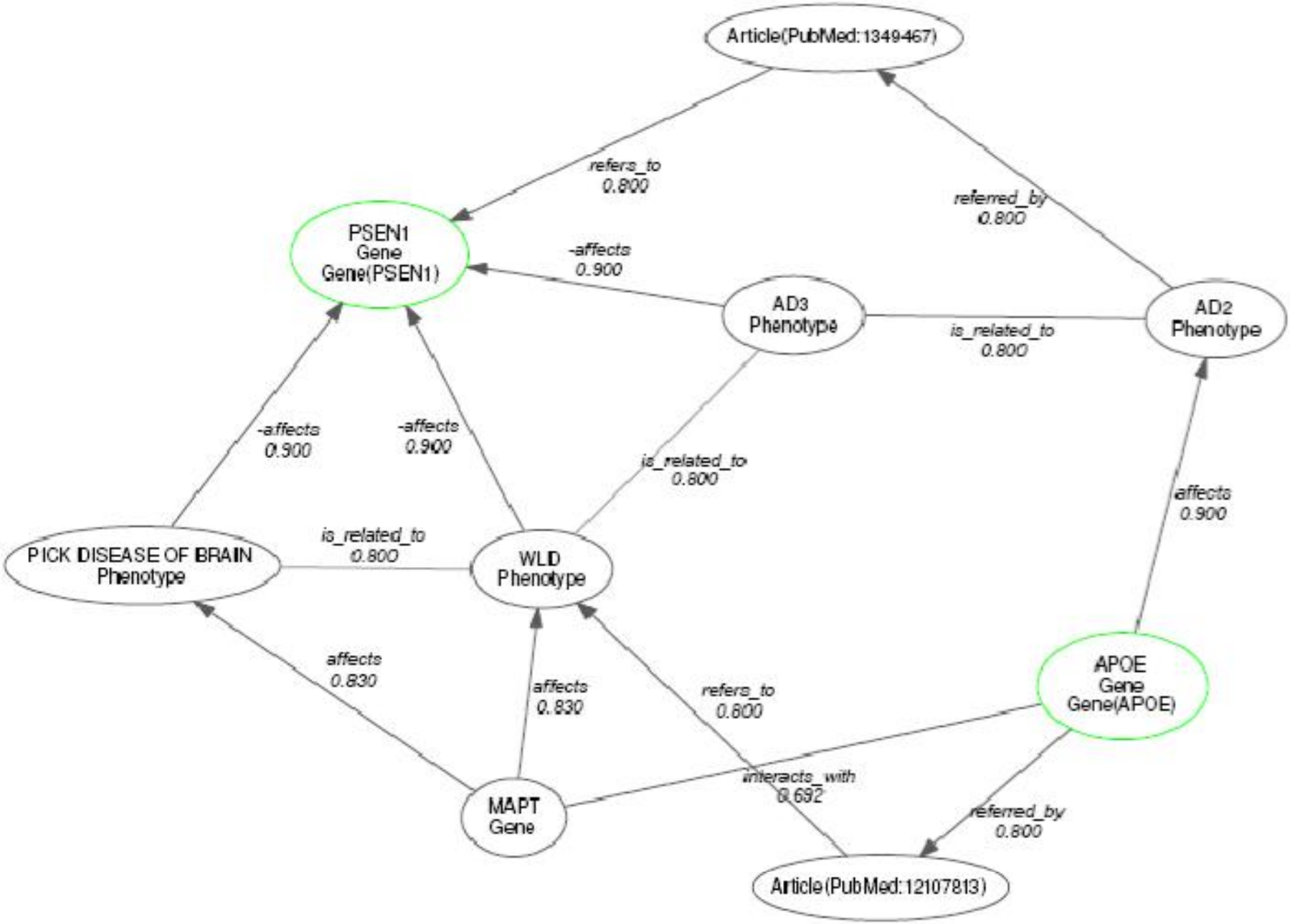
- Apparently yes: there is potential to predict links
  - Reliability and probability of the best path seem to perform equally well
- BTW, no machine learning so far
  - Given a training set, we could fit our model (the probabilistic graph) better to the data
  - E.g., learn data source specific reliabilities or edge type relevances, even individual edge probabilities



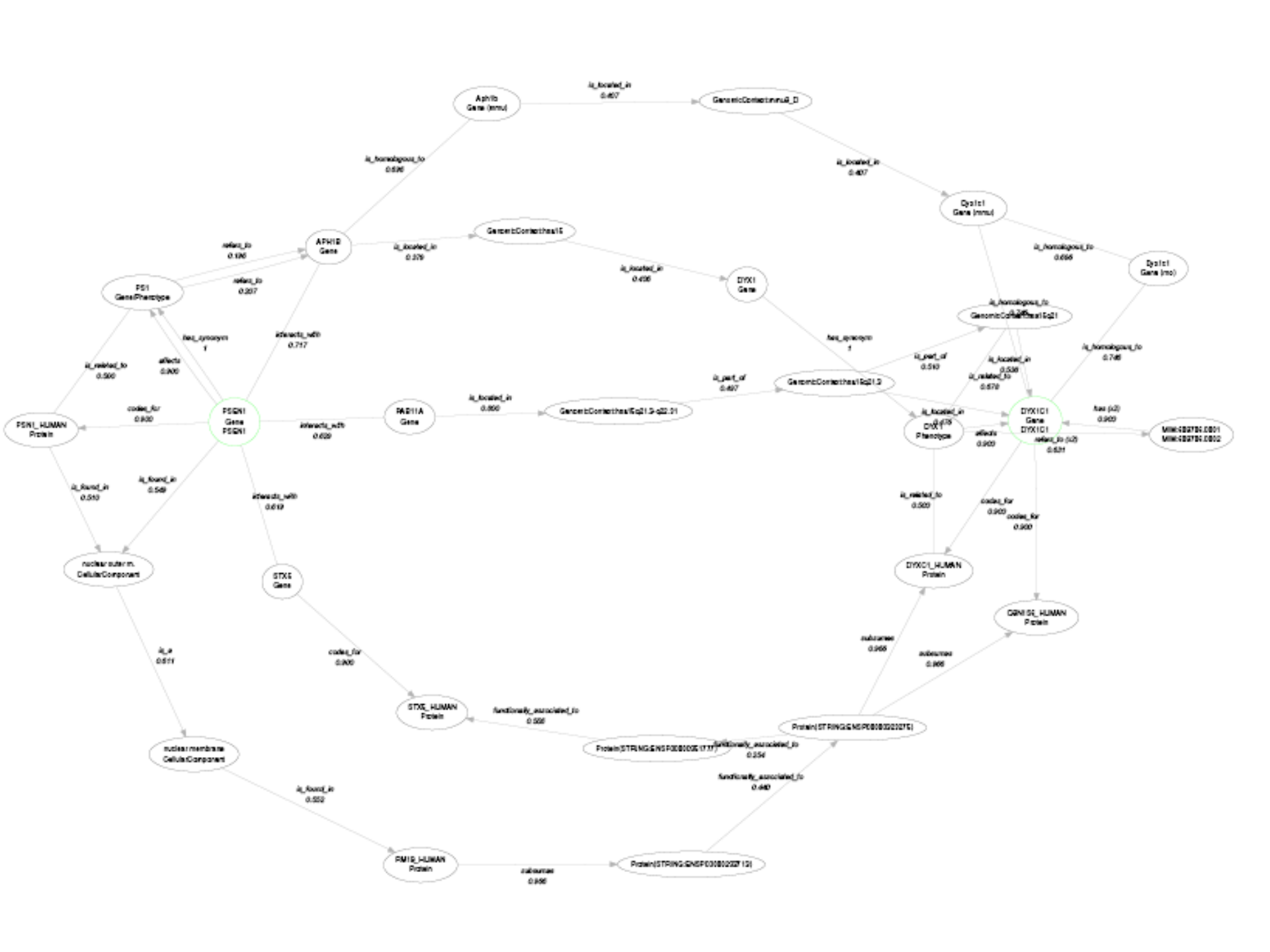
2. The most reliable subgraph problem

- Given two query nodes s and t, find a subgraph (of a limited maximum size) that connects the query nodes as strongly as possible
- Motivation
  - Visualization
  - Preprocessing for computationally intensive methods
- For a probabilistic graph: extract the most reliable subgraph (of size at most k) wrt. to s and t
  - Ensures relevance wrt to both s and t
  - Favors results with little redundancy

How are genes PSEN1 (presenilin1) and APOE (apolipoprotein E) related?

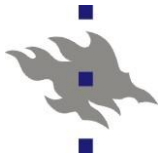

Article(PubMed:1349467)
refers_to
0.800
referred_by
0.800
PSEN1
Gene
Gene(PSEN1)
-affects
0.900
AD3
Phenotype
AD2
Phenotype
is_related_to
0.800
-affects
0.900
-affects
0.900
is_related_to
0.800
affects
0.900
PICK DISEASE OF BRAIN
Phenotype
is_related_to
0.800
WLD
Phenotype
affects
0.830
affects
0.830
refers_to
0.800
APOE
Gene
Gene(APOE)
interacts_with
0.692
MAPT
Gene
referred_by
0.800
Article(PubMed:12107813)

How are genes PSEN1 (presenilin1) and DYX1C1 (apolipoprotein E) possibly related?


Aphlb
Gene (mmu)
is_located_in
0.407
GenomicContextmu8_0
is_homologous_to
0.696
is_located_in
0.407
Qsi1c1
Gene (mmu)
is_homologous_to
0.696
Qsi1c1
Gene (mo)
APH1B
Gene
is_located_in
0.370
GenomicContext8q15
is_located_in
0.436
DYX1
Gene
PS1
GenePhenotype
is_homologous_to
0.748
GenomicContext8q21
has_synonym
1
has_synonym
1
is_part_of
0.510
is_located_in
0.538
is_related_to
0.572
is_related_to
0.500
affects
0.900
interacts_with
0.717
is_part_of
0.497
GenomicContext6q21.3
is_homologous_to
0.748
codes_for
0.900
PSEN1
Gene
PSEN1
interacts_with
0.629
RAD11A
Gene
is_located_in
0.950
GenomicContext6q21.3-q22.31
is_located_in
0.478
DYX1C1
Phenotype
DYX1C1
Gene
DYX1C1
has (x2)
0.900
MIM:608780.0001
MIM:608780.0002
affects
0.900
relates_to (x2)
0.521
PSN1_HUMAN
Protein
is_found_in
0.510
is_found_in
0.549
interacts_with
0.619
is_related_to
0.500
codes_for
0.900
codes_for
0.900
nuclear outer m.
CellularComponent
STX5
Gene
DYXC1_HUMAN
Protein
DNH5I_HUMAN
Protein
is_a
0.511
codes_for
0.900
subunitwo
0.966
subunitwo
0.966
STX5_HUMAN
Protein
functionally_associated_to
0.500
Protein(STRING:ENSP00000320275)
functionally_associated_to
0.254
functionally_associated_to
0.440
nuclear membrane
CellularComponent
is_found_in
0.552
PM19_HUMAN
Protein
subunitwo
0.956
Protein(STRING:ENSP00000327131)

Subgraph extraction

- Related work
  - Faloutsos et al (2004): connection subgraphs
    - model: current in resistor networks
  - De Raedt, Kimmig, Toivonen (2008): ProbLog theory compression
    - Similar to Biomine, but in first-order logic
- Two opposite heuristic approaches
  - Prune the original graph until the required size is reached
    - Complexity depends on the size of the original graph
  - Construct a subgraph incrementally
    - Depends (more) on the size of the result

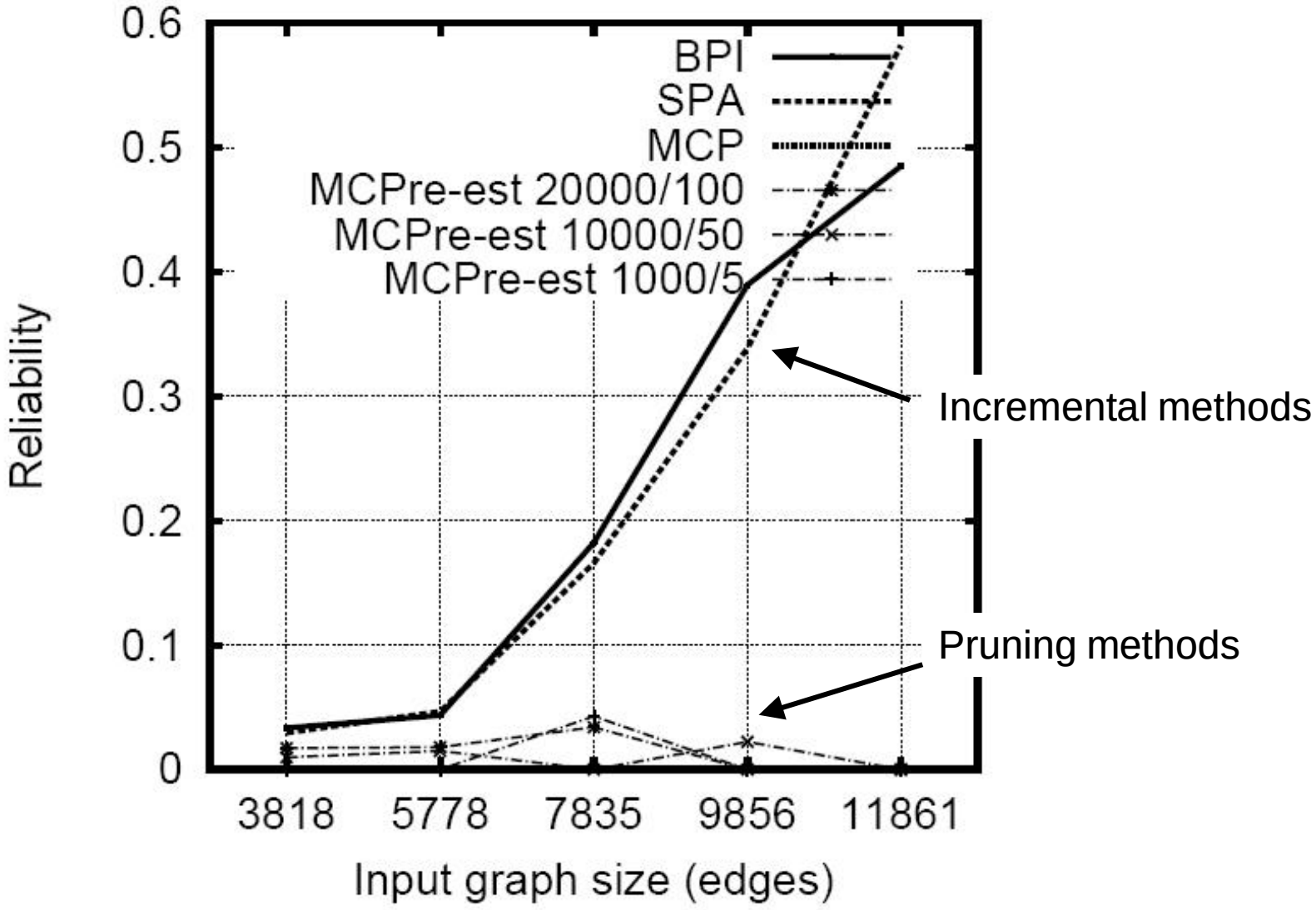


Incremental constructive methods

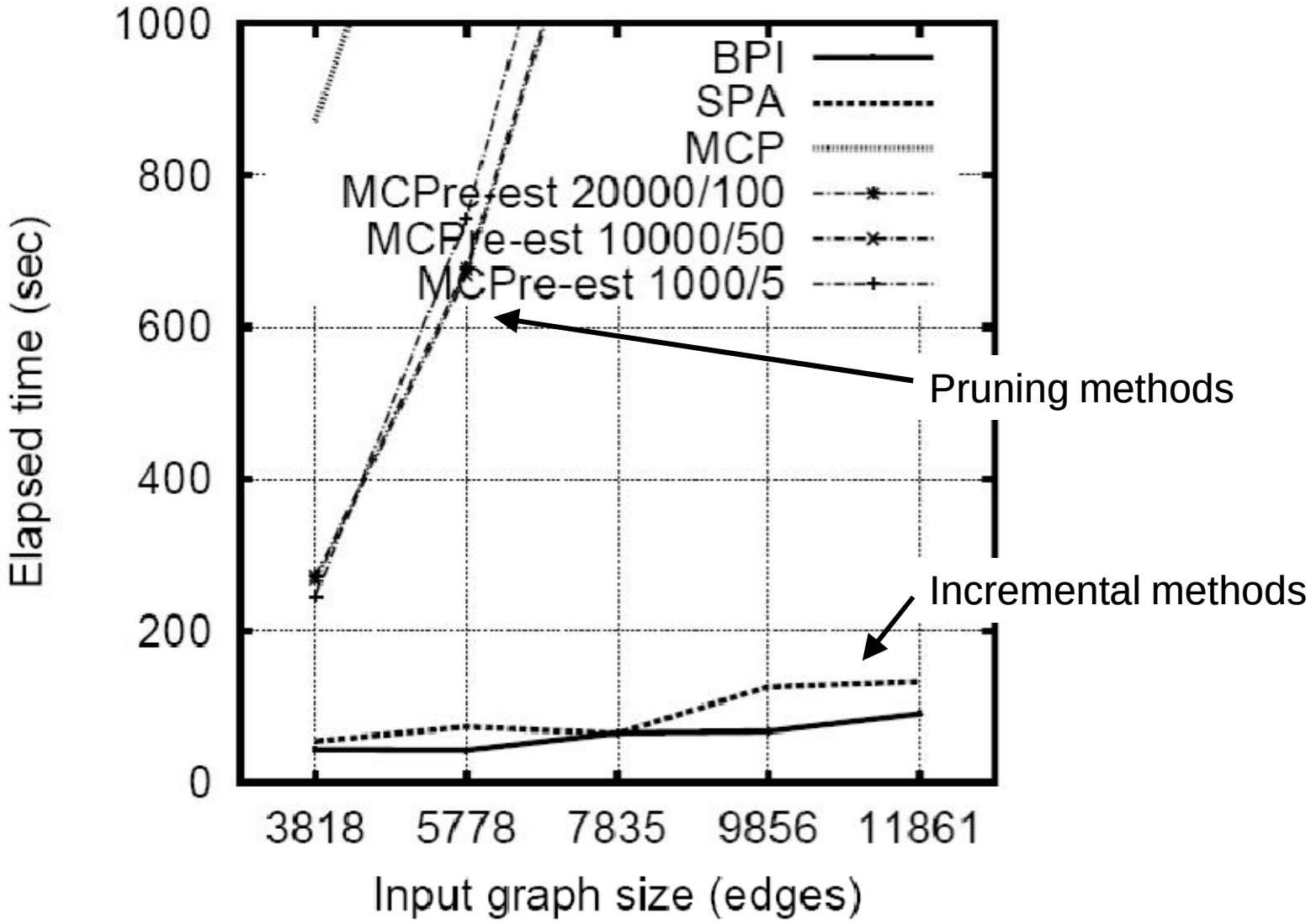
- Parameter k: upper limit for the size of the result

- BPI, Best Paths Incremental
  - Take K best paths, such that they span a graph of size k
- SPA, Series-Parallel Augmentation
  - Greedily builds a series-parallel graph of size at most k
  - Hintsanen and Toivonen (PKDD/DAMI 2008)
- PathCover
  - Stochastic selection of candidate paths (or trees) + set cover of Monte Carlo simulations by candidates
  - Hintsanen and Toivonen; Kasari et al. (2010)

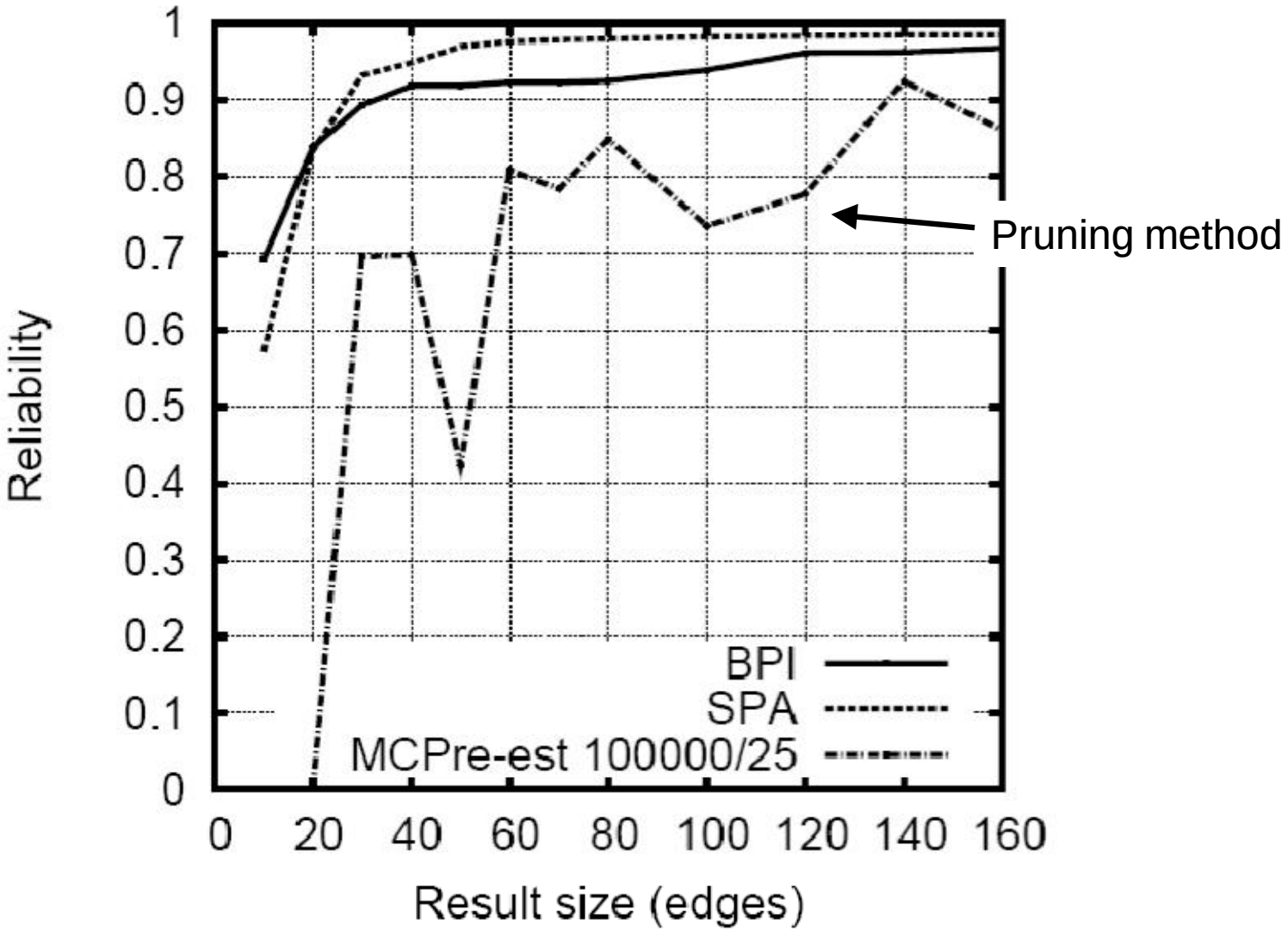
Quality of the extracted subgraph (as a function of the size of the input)



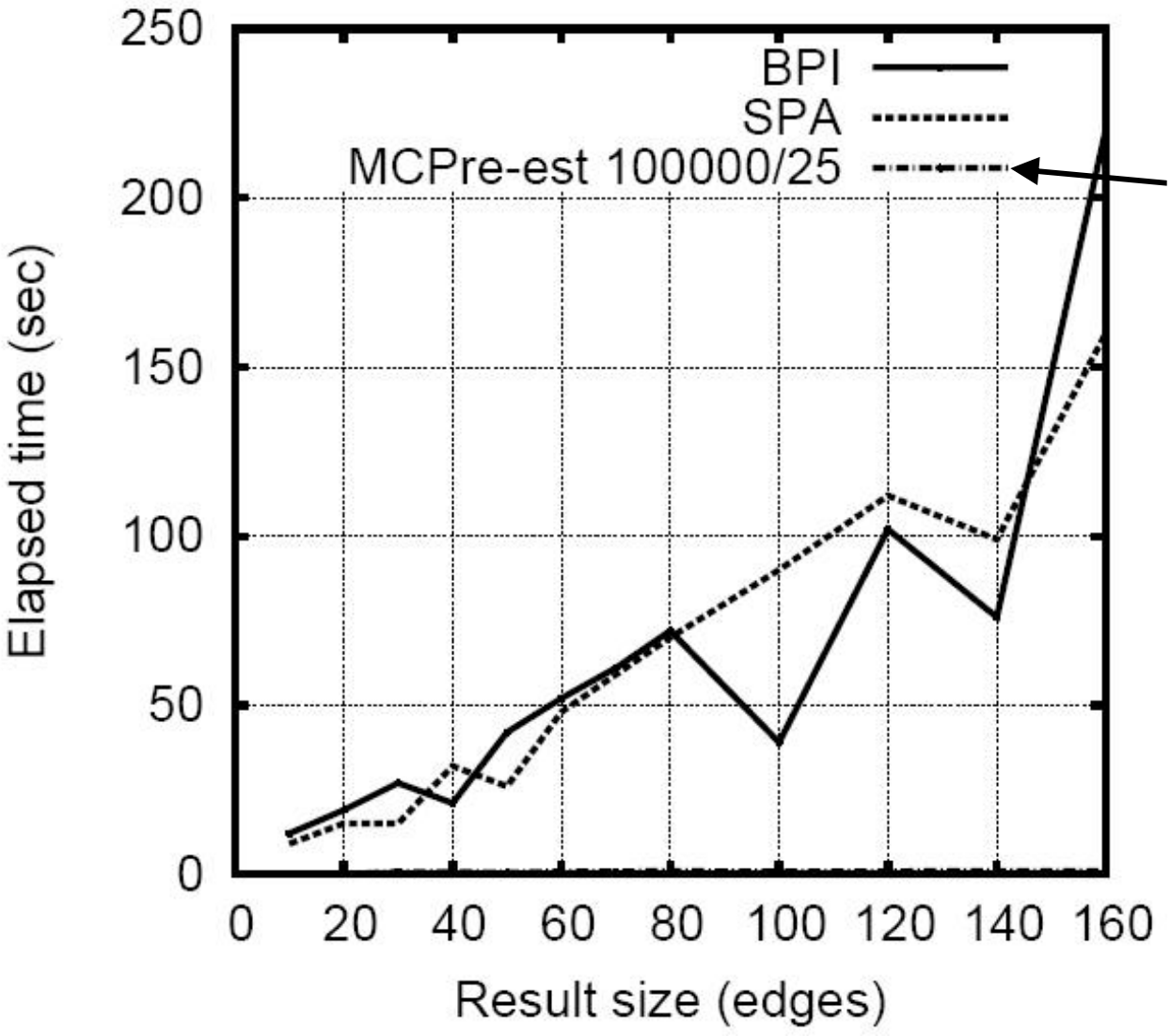
Time to extract a subgraph (as a function of the size of the input)



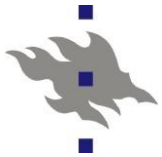
Quality of the extracted subgraph (as a function of the size of the output)



Time to extract a subgraph (as a function of the size of the output)



Pruning method appr. constant 1400 seconds



Slide 33/44

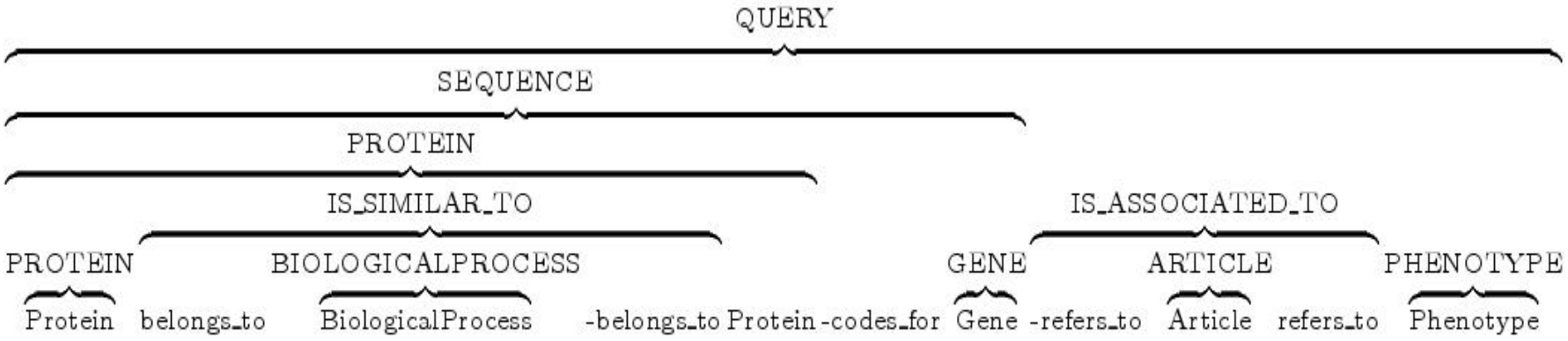
- We have now looked at
  - Probabilistic graphs
  - Reliability and path probability as measures of connectedness
  - The most reliable subgraph extraction problem

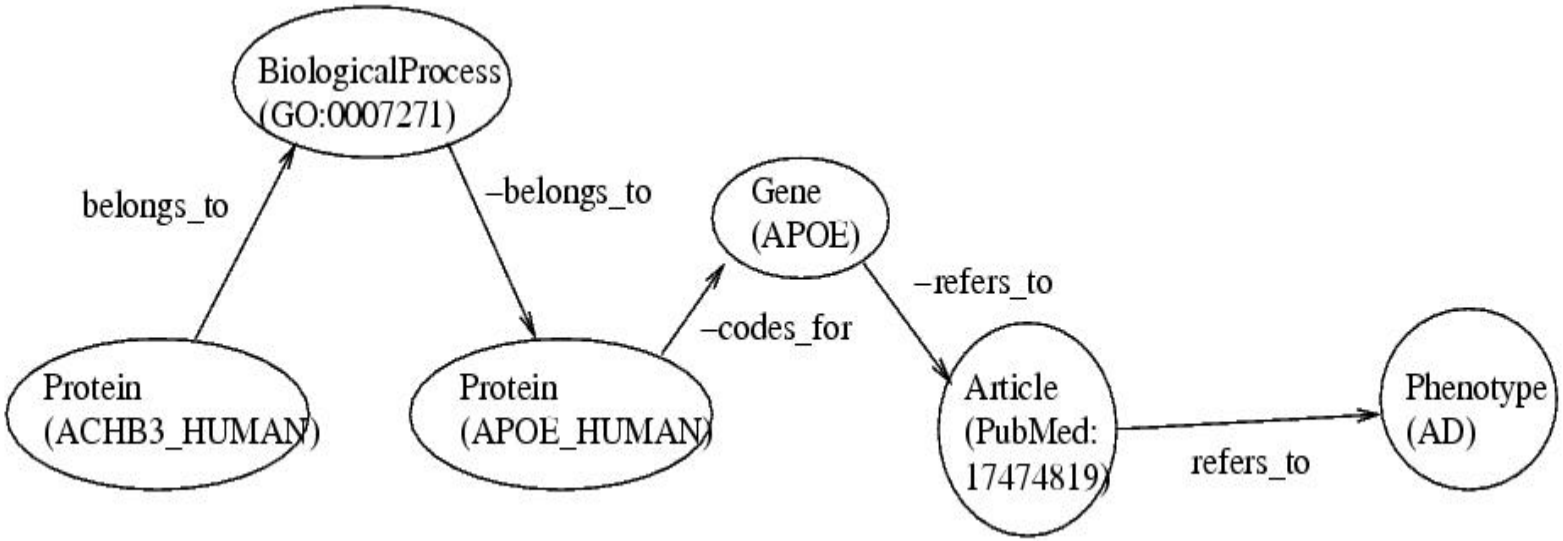
- Coming up next: different views to subgraph extraction
  - Context-free grammars as a qualitative query tool
  - ProbLog: a probabilistic Prolog

Subgraph extraction problem

- The most reliable subgraph problem is quantitative
- Consider a qualitative variant:
  - The user specifies relevant path types
  - The task is to find all paths between s,t, of the given types
  - The method returns the subgraph induced by the set of accepted paths
- Sevon & Eronen (2008)

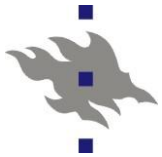
Example: paths from ACHB3_HUMAN to AD





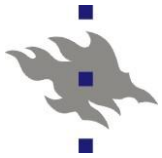
Subgraph queries with context-free grammars

- Path type = the alternating sequence of node and edge types on a path
  - e.g., Gene participates_in Pathway is_related_to Phenotype
- Use CFG to specify the class of acceptable path types
  - terminal symbols = node and edge types
  - nonterminal symbols = path classes
  - starting nonterminal = class of acceptable paths
- Path classes are pre-defined in a background CFG, queries are formulated by specifying the root level production rules



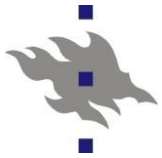
Subgraph queries with context-free grammars

- Key idea: extract a subgraph, spanned by admissible paths
  - The grammar controls what is useful, relevant, or plausible
- Algorithmic issues
  - How to parse all graphs from a graph efficiently
  - Sevon & Eronen (IB 2008)
- Combining qualitative and quantitative approaches: probabilistic grammars
  - Paths (strings) have probabilities, derived from the edges
  - Productions of the grammar can have probabilities, too



ProbLog

- ProbLog = Prolog + probabilities of clauses
  - ```
    0.3779:edge('EntrezProtein_4885045','HGNC_620').
    0.4928:edge('HGNC_620','PubMed_12653567').
    0.6054:edge('EntrezProtein_4885045','HGNC_12850').
    0.9022:edge('PubMed_2322535','HGNC_983').
    0.8750:edge('HomoloGene_20065','HGNC_983').
    ...
    1.0:path(X,Y):-edge(X,Y).
    1.0:path(X,Y):-edge(X,Z),path(Z,Y).
    ```
- Each clause has a probability to be in a Prolog program
- Clauses are mutually independent
- Suitable for representing and querying probabilistic graphs
- [De Raedt, Kimmig, Toivonen, IJCAI 07, PKDD 07]
    ```



ProbLog semantics

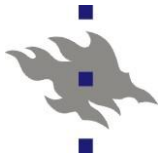
- A ProbLog program $\{p_1 : c_1, ..., p_n : c_n\}$ defines a probability distribution over Prolog programs $L \subseteq \{c_1, ..., c_n\}$

$$P(L|Program) = \prod_{c_i \in L} p_i \prod_{c_j \notin L} (1 - p_j)$$

- The probability of a goal q:

$$P(q|Program) = \sum_{L \subseteq \{c_1,...,c_n\}} P(q|L) \times P(L|Program)$$

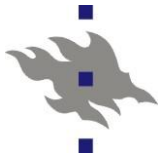
$$P(q|L) = 1 \text{ if } L \models q$$
$$= 0 \text{ otherwise}$$



ProbLog inference

- Given a ProbLog program T and a query q, P(q|T) gives the probability that a proof exists for q in T
- Obvious application to graphs
  - P(path(s,t) | T) is the probability that nodes s and t are connected in graph T
- A generalization of network reliability

- How to compute P(q|T)?
  - [De Raedt, Kimmig, Toivonen, IJCAI 07]



Compression of ProbLog programs

- A generalization of the most reliable subgraph problem to ProbLog
- Given
  - ProbLog program T
  - positive and negative example queries Pos and Neg
  - constant k
- find the program $T' \subseteq T$ of size at most k that maximizes

$$Likelihood(T'|Pos, Neg) = \prod_{p \in Pos} P(p|T') \prod_{n \in Neg} (1 - P(n|T'))$$

- [De Raedt et al., Machine Learning, 2008]

Research problems

Abstraction

- Given a large graph, produce a smaller abstraction, e.g.
  - replace subgraphs by single nodes or edges
  - replace nodes by more general types
  - remove irrelevant details

Discovery query

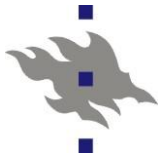
- Goal: discover unknown but plausible indirect relationships
- Balance between a strong connection and novelty
  - (and non-redundancy between proposed discoveries)

Conclusions

- Search in large probabilistic graphs
- Probabilistic definitions of search objectives
  - Input: a set of nodes; output: a subgraph
  - Maximise reliability of output, minimize its size
- Grammars, ProbLog: also qualitative criteria
- Computationally non-trivial solutions (skipped)

- Potential for discoveries/hypothesis generation

- An experimental search engine is available at **http://biomine.cs.helsinki.fi**



Thanks

- Biomine (Helsinki): Petteri Sevon, Lauri Eronen, Petteri Hintsanen, Kimmo Kulovesi, Laura Langohr, Atte Hinkka, Melissa Kasari, Aki Koivisto, Anne Hyvärinen, ...

- Problog (Leuven and Freiburg): Luc De Raedt, Angelika Kimmig, Kristian Kersting, Kate Revorado