Relation Web

- Let D be the set of descriptors.
- Let R be the set of possible relations.
- Let $G = (V, E)$ be a directed graph where $V = D$ and E is a set of edges such that each $v \in V$ has $|R| \times |D|$ edges, one per relation per descriptor (exhaustively listing all possible relations between this descriptor and all other descriptors).
- Goal: assign each edge $e \in E$ a weight $P(e) \in [0, 1]$ that represents the probability of that being a valid relation.
- Idea: as triples are processed, adjust the weight of not only the "explicit" relation between the relevant descriptors that is directly stated with that triple, but also the "implicit" relation(s).

¡++¿

Probability Representation

- Let the set R contain all the relations between all possible relations between all descriptors.
- Sample Space: Assign each $r \in R$ a value $0, 1$ indicating whether or not that relation is valid. Let S be the space of all such outcomes.
- Let T be the set of all training triples.
- Want to find: $s \in S$ such that $P(s|T)$ is maximized.
- Assume perfect relation extraction from triples.
- Let R_E be the set of all explicit relations extracted from T.
- Let R_I be the set of all explicit relations extracted from R_E.

Rudimentary Algorithm

- Explicit relation detection:
  - For each triple, label the detected explicit relations with their probabilities.
- Drawing implications:
  - For every pair of descriptors A, B find all non-zero paths $A \to B$. Store these paths and their vertex labels and edge probabilites in a data structure indexed by descriptor pair.

  ¡++¿

¡++¿