

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).

i++j

Rudimentary Algorithm

- Explicit relation detection:
 - For each triple, label the detected explicit relations with their probabilities (e.g., logistic regression confidence measure).
- Drawing implications:
 - For every pair of descriptors A, B find all non-zero paths $A \rightarrow B$. Store these paths and their vertex labels and edge probabilities in a data structure indexed by descriptor pair.
 - “Collate” the entries for every pair of descriptors of the same respective labels, finding clusters of paths that tend to be correlated with one another by finding correlation scores for each pair of paths.
 - When processing a new triple:
 - * Update explicitly identified edge(s) with confidence values.
 - * For each path this edge is a part of, update all correlated paths by an amount proportional to this path’s edge weights and the correlation score.