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).

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 $A \to B$. Store these paths and their vertex labels and edge probabilites 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.

```
Procedure relation_web(D: descriptors, R:Relations, T:Triples)
    let G be a graph with vertices D and R edges from each vertex to every
        other vertex
    for each t in T:
        let d1 = first descriptor, d2 = second descriptor
        for each r in R:
            assign edge d1 - r -> d2 to p(r | t)

            // begin iterative updates
            until convergence:
                // to hold descriptor path vectors: indexed by descriptor pair
                path_vectors = {};

                for each pair of descriptors d1', d2':
                    get all paths from d1' to d2'
                    compress path into vector d'
                    path_vectors[(d1', d2')] = d'

                for each pair of labels l1, l2:
                    get pairwise entry correlations of vectors with labels
                        l1, l2
```