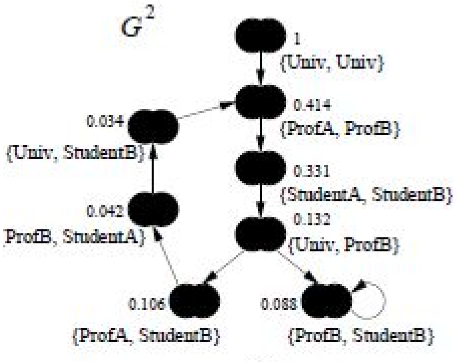
SimRank arbejdsblad

Two objects are similar if they are referenced by similar objects and a object is maximally similar to itself.



(b)

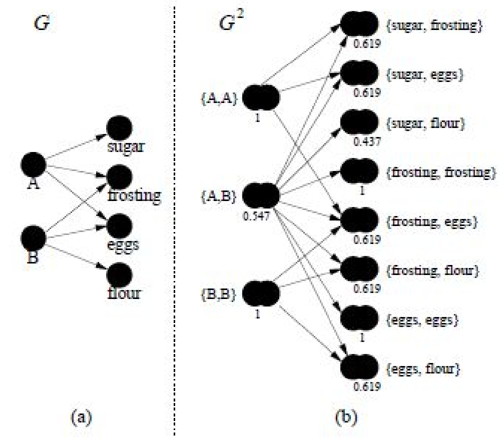
As seen on the figure {Univ, Univ} is 1 as it is the same object and therefore the similarity is 1 (The highest value). The rest of the values are calculated using the formula.
similarity between objects a and b is denoted as s(a,b) E [0, 1]
If a = b then s(a, b) is defined to be 1.

$$s(a,b) = \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b))$$

Where C is a constant between 0 and 1 and is the rate of decay on each iteration, I(a) and I(b) is all neighbours of a and b respectively. We divide by the amount of neighbours to normalize the value.

As an example of how to use SimRank on Recommender Systems, the article actually made this idea: Suppose person A and B purchase a set of items {eggs, frosting, sugar} and {eggs, frosting, flour} respectively. Clearly these 2 are similar as the items are pretty similar (66% similar) we can therefore for example recommend that person A might purchase flour. But there is also another way these two are similar. Not only did the buy similar items, but the two people bought Flour and Sugar which is usually bought by similar people, cake-bakers.

- So people are similar because they buy similar items
- Items are similar because they are bought by similar people



One peculiar thing is however, 2 different item sets like {eggs, frosting} and {sugar, frosting} has the same value, even though the first one is mentioned twice, this is because we normalize the value to only look at the percentage of times the items are purchased.

This can however be done using absolute values instead if necessary.

Problems with SimRank:
Requires a large amount of memory, (Space requirement is O(n^2)) a way to reduce this we can prune. The standard approach is computing every node-pair, however if we reduce the amount of recursion we can reduce the size greatly. So for example in a typical graph, the nodes near a node will be a small piece of a bigger domain, however have a higher similarity with each other than the rest of the nodes.

Limited-information Problem - New things have a little to no information.

So to summarize:
SimRank is a mathematical formula which can be used to find similarity between nodes in a graph. Nodes are maximally similar to themselves. Similarity decay after each step and is represented by constant C.