1 PathSim

PathSim is discussed in details in the lecture slides, for more information, please see [2].

2 Personalized PageRank

Personalized PageRank [1] is variant of PageRank\(^1\), where instead of jumping/teleporting to an arbitrary node in the network uniformly with some probability, P-PageRank jumps to a set of preferred nodes. In the similarity search setting, the preferred node set only contains the queried node. In linear algebra terminology, both PageRank and P-PageRank try to find the \( \mathbf{v} \) in

\[
\mathbf{v} = (1 - c)A\mathbf{v} + c\mathbf{u} \quad (2.1)
\]

where \( c \in (0, 1) \) is the teleportation constant, \( A \) is the adjacency matrix raw normalized to make sure every row sums to 1, \( \mathbf{v} \) is the fixed-point probability distribution of nodes, i.e., pagerank scores. The only difference is that in normal pagerank \( \mathbf{u} = \mathbf{1}/|V| \), where \( V \) is the entire node set, whereas in P-PageRank, \( \mathbf{u} \) is the preference vector where \( \mathbf{u}(p) = 1/|P| \) if \( p \in P \), and \( \mathbf{u}(p) = 0 \) if \( p \notin P \), here \( P \) is the preferred node set.

To implement the P-PageRank in our similarity task, we suggest you to use iterative algorithm, where you initialize \( \mathbf{v} = \mathbf{1}/|V| \). In each iteration, we redistribute the probability of each node to its neighbors and the preferred node set (the queried node). After \( t \) iterations, output the nodes with top page rank scores as the result.

In our experiment, we use \( t = 10 \) iterations and \( c = 0.15 \). Please directly use the provided APVPA net and APTPA net for computing P-PageRank.

References


\(^1\)https://en.wikipedia.org/wiki/PageRank