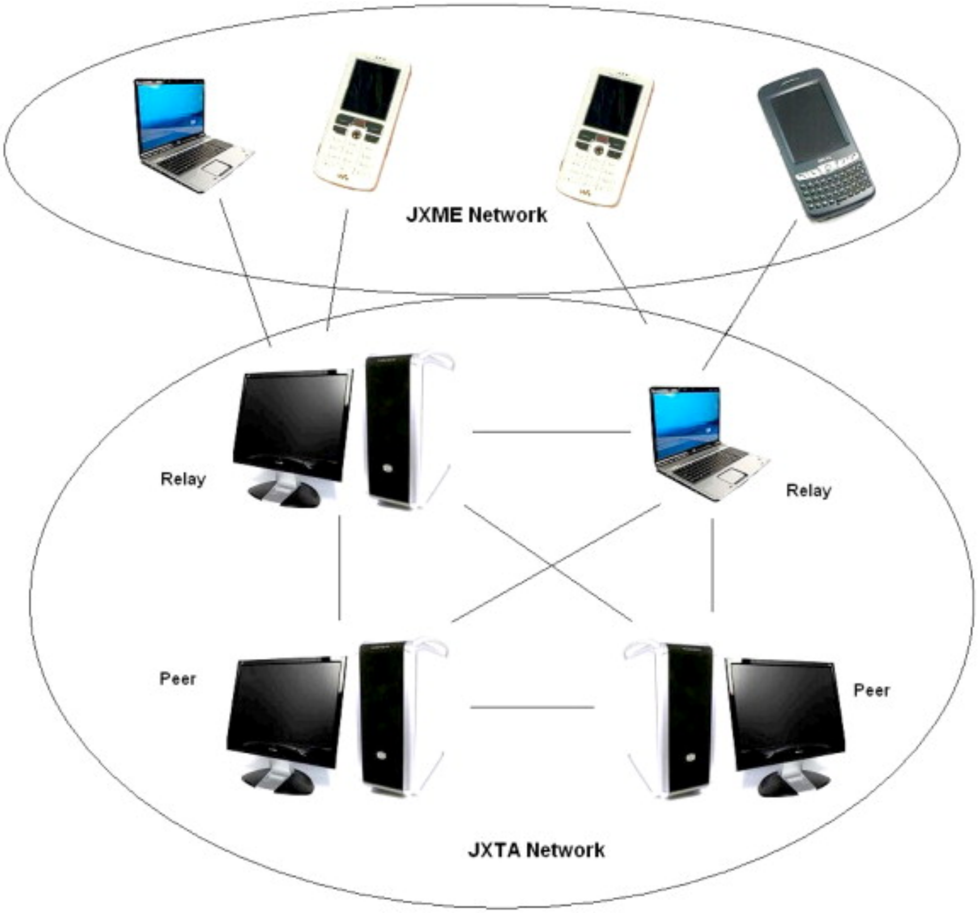
INTERNATIONAL UNIVERSITY

Pre-Thesis

P2P Social Network in Link Recommendation

Research Report



Advisor: MSc. Vo Duy Khoi

Student: Le Quoc Thanh
ID: ITIU09028

I) Introduction:

Social networks have been mostly based on a centralized infrastructure where the owner hosts all the data and services. This model of "fat server & thin clients" results in many systems and practical problems such as privacy, censorship, scalability, and fault-tolerance. P2P systems can be an alternative to the client-server model for social networks, where data is maintained in all peers and the users define their level of privacy. The P2P link prediction problem differs due to the distributed nature, where each node has only local information about the network. We propose an approach by adapting a top-k search algorithm to P2P link recommendation.

II) Background:

2.1) Link Prediction:

Methods for link prediction rely on content shared among nodes and topology of the network. Topological methods are based on paths between nodes and neighborhoods. Paths between nodes approach uses shortest path, ensemble of paths or their variants to handle the link prediction problem. For example, Adamic and Adar use weighted neighborhood information to find relationship between individuals [6]. Intuition behind Common Neighbor is that a node is more likely to interact with another node if the number of overlaps between their neighbors is high.

2.2) P2P Infrastructures

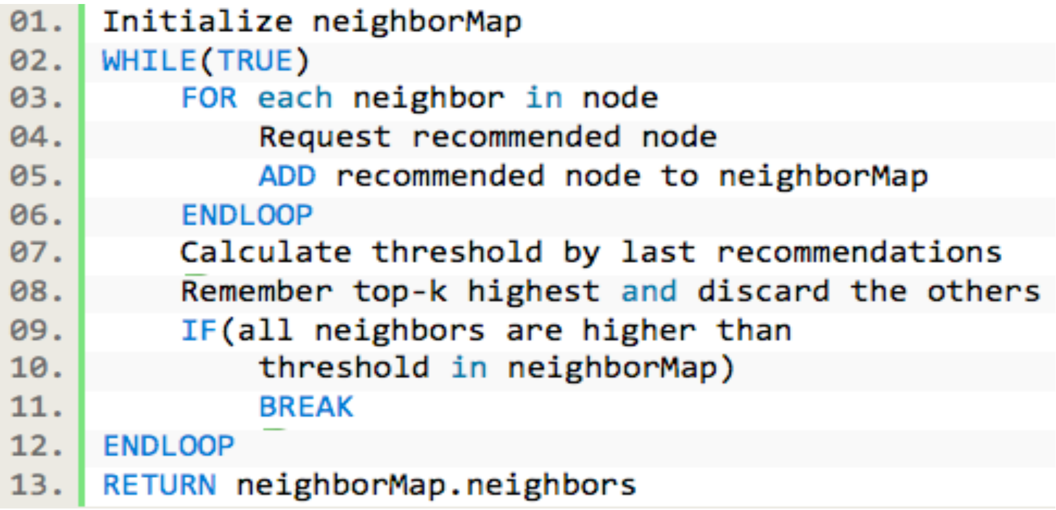
P2P systems enable sharing data and resources between the peers. P2P systems can be categorized according to the degree of centralization: purely decentralized, partially centralized and hybrid decentralized. Hybrid systems offer an effective tradeoff for implementing P2P social networks, since fully decentralized systems have disadvantages like recalling peer addresses. In a case where all peers are offline, all information can be lost in a decentralized system while we at least expect one super peer can take care of peer addresses. DNS like protocols can be easily implemented in hybrid system using the power of super peers. In a P2P environment, the link prediction problem becomes different since we do not have control on the full network. Each node has partial information about the network. Thus, there is a need for distributed algorithms to find predicted links in P2P infrastructure. One can utilize a Common Neighbor approach to locally gather and merge link strengths from neighbors. We approach this merging problem as a top-k query processing [9] and implement a set of distributed top-k link recommendation algorithm.

The closest work to our study is the CNP (Common Neighbor Predictor) method. It uses a distributed algorithm to predict future links in a P2P infrastructure. They express their ground truth through the following formula: $\alpha = \frac{|E_{pre} \cap E_2|}{|E_{pre}|}$

where *E* two time instants t1 and t2. Furthermore, *E2* is the set of links that are newly created by the network between times t1 and t2. First, NCNP (Neighbor's Common Neighbor Predictor) is proposed that considers neighbors' Common Neighbor when predicting a new link, when at least two neighbors of a node share the same node in common as a neighbor.

III) Common Neighbor Recommendation

To solve the P2P link recommendation problem, we start by adapting a Common Neighbor based approach which produces a recommendation to a node *x* based on highly weighted common neighbors of *x*'s neighbors. We utilize Fagin's top-k algorithm (FA) and an approximate threshold algorithm (TA) that were originally proposed for top-k Common Neighbor processing. In FA, each node initializes its neighbor map and enters a loop until finding *k* neighbor recommendations retrieved from all its neighbors. For each iteration *i*, neighbor recommendations are retrieved from each neighbor, which is the i^{th} top neighbor of the requested neighbor. Then the neighbor map is updated and the node checks if *k* recommendations have been retrieved from all its neighbors. If so, it stops and calculates top-k neighbors by assessing weights; otherwise it starts another iteration. The threshold algorithm is the modified version of FA differing for its stopping condition [13]. Even though FA is optimal, it can result in the worst-case scenario with high probability. To avoid this problem TA was proposed where it stops at least as early as FA. Thus, TA calculates a threshold value at each iteration for the last retrieved recommendations. If there are *k* neighbors that have higher rate than the threshold value, the algorithm terminates.



```
Initialize neighborMap
WHILE(TRUE)
    FOR each neighbor in node
        Request recommended node
        ADD recommended node to neighborMap
    ENDLOOP
    Calculate threshold by last recommendations
    Remember top-k highest and discard the others
    IF(all neighbors are higher than
        threshold in neighborMap)
        BREAK
ENDLOOP
RETURN neighborMap.neighbors
```

Figure 2 – Top-K TA Common Neighbor algorithm

IV) Conclusion: We presented problem of link recommendation in P2P social networks. We studied two algorithms, Top-K FA and TA Common Neighbors to find recommended links for a node. To implement social network, we propose using hybrid P2P infrastructure, where there are simple peers and super peers. Each peer in the system has a super-peer to provide other peer addresses such as neighbor peers. In order to provide peer addresses, super peers are designed to have a DNS like protocol in which each super-peer delegates address inquiry message to parent super-peer if peer-address is not found in local repository. As a result, there must be at least one super-peer, which has permanent address for system start-up. This super-peer would keep track of super-addresses and if there is no other super- peer, it would also keep track of simple-peers.