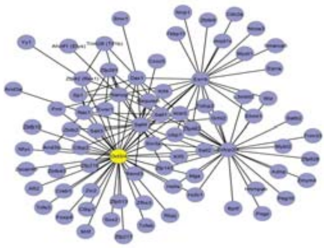
Streaming Link Prediction on Dynamic Graph Streams

Introduction

Real-world **information networks** are **dynamic** and **of massive volume**. Noteworthy examples include social networks, PPI networks, communication networks, and the Web.

Link Prediction

Given a snapshot of an information network at time *t*, we seek to accurately predict the edges that will be added to the network during the interval from time *t* to a given future time t_0

Graph Stream Model

An information network can be formalized as a graph stream that receives a sequence of edges of the form $< EId; i, j >$. At any given moment in time *t*, the edges in the stream imply a graph $G(t) = (N(t), E(t))$, where $N(t)$ is the set of nodes and $A(t)$ is the set of *distinct* edges at time *t*

•***Dynamics***: *t* can be infinitively large

•***Massiveness***: *G(t)* is too large to be safely stored even on disk for effective analysis

Can we predict potential links on massive , dynamic graph streams accurately and efficiently?

Main Idea

We consider proposing and re-examining high-quality **approximations** to the state-of-the-art link prediction metrics on the massive graph stream scenario:

1.Common neighbor:

$$|\tau(i,t) \cap \tau(j,t)|$$

2.Jaccard coefficient

$$\frac{|\tau(i,t) \cap \tau(j,t)|}{|\tau(i,t) \cup \tau(j,t)|}$$

3.Adamic-Ada

$$\sum_{k \in \tau(i,t) \cap \tau(j,t)} (1/\log(|\tau(k,t)|))$$

Algorithms

1. Minhashing based approximation

Consider a streaming algorithm maintaining values of each node *i*, its *minimum adjacent hash value*, $v(i)$, and *minimum adjacent hash index*, $I(i)$

Theorem: *The probability that $I(i) = I(j)$ is exactly equal to the Jaccard coefficient between nodes i and j*

2. Node-biased sampling

A ***reservoir*** of budget *L* is associated with each node to dynamically sample *L* incident edges of each node

Theorem: *Let (i) and (j) be the fraction of nodes incident on i and j respectively, which are sampled. Then, the total number of common neighbors C_{ij} , between nodes i and j, can be estimated*

$$C_{ij} = \frac{|S(i) \cap S(j)|}{min\{\eta(i), \eta(j)\}}$$

Experiments

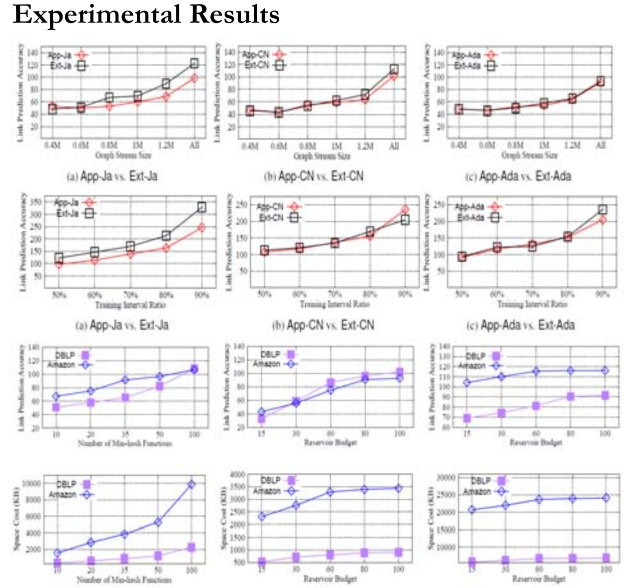
Real-world Datasets

- **DBLP**: a streaming co-authorship graph comprising 1,954,776 author-pairs
- **Amazon Co-purchasing Network**: a product co-purchasing network comprising 410,236 nodes and 3,356,824 edges

Evaluation Metrics

- Link prediction accuracy
- Link prediction cost

Experimental Results



(a) App-Ja vs. Ext-Ja (b) App-CN vs. Ext-CN (c) App-Ada vs. Ext-Ada

(a) App-Ja vs. Ext-Ja (b) App-CN vs. Ext-CN (c) App-Ada vs. Ext-Ada

Peixiang Zhao, Computer Science, FSU, zhao@cs.fsu.edu