Link Analysis for Private Weighted Graphs

Introduction to Information Retrieval
CS 221
Donald J. Patterson

Content based on the SIGIR2009 paper located here:
http://doi.acm.org/10.1145/1571941.1571983

Link Analysis for Private Weighted Graphs

Link Analysis for Private Weighted Graphs

Link Analysis for Private Weighted Graphs

Jun Sakuma
University of Tsukuba
1-1-1 Tennodai,
Tsukuba, Japan
jun@cs.tsukuba.ac.jp

Shigenobu Kobayashi
Tokyo Institute of Technology
4259 Nagatsuta-cho
Yokohama, Japan
kobayasi@dis.titech.ac.jp

ABSTRACT

Link analysis methods have been used successfully for knowledge discovery from the link structure of mutually linking entities. Existing link analysis methods have been inherently designed based on the fact that the entire link structure of the target graph is observable such as public web documents; however, link information in graphs in the real world, such as human relationship or economic activities, is rarely open to public. If link analysis can be performed using graphs with private links in a privacy-preserving way, it enables us to rank entities connected with private ties, such as people, organizations, or business transactions. In this paper, we present a secure link analysis for graphs with private links by means of cryptographic protocols. Our solutions are designed as privacy-preserving expansions of well-known link analysis methods, PageRank and HITS. The outcomes of our protocols are completely equivalent to those of PageRank and HITS. Furthermore, our protocols theoretically guarantee that the private link information possessed by each node is not revealed to other nodes.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms

Keywords

link analysis, privacy, ranking, HITS, PageRank

1. INTRODUCTION

Link-based analysis has been developed in the form of algorithms that discover useful information from the link structure of mutually linking entities. In particular, HITS [7] and PageRank [9] have been successfully used for the ranking of hyperlinked web documents. These link analysis methods were originally designed for the analysis of web documents; however, these can be readily applied to mutually linking entities, such as referenced academic papers, protein-protein interactions, and so on.

In general, link analysis methods take the entire link structure as its input. Indeed, for the computation of Google's PageRank, the linking structures of web documents are collected by crawling agents which actually wander around public web documents. The same holds for citation graphs of academic papers or interaction graphs of protein networks. As shown, existing link analysis methods have inherently been designed based on the fact that the entire link structure of the target graph is observable; however, link information in the real world, such as human relationships or economic activities, is rarely open to public.

In this paper, we present link analysis solutions for graphs of privately connected entities. Let there be a directed weighted graph $G = (V, E, W)$ where V is a set of vertices, E is a set of edges, and W is a weight matrix. Throughout this paper, we assume that the set of vertices corresponds to a collection of distributed nodes where the computational power of each node is polynomial. Edges correspond to links between nodes; weights of edges correspond to weights of these links. Let there be a link of node i pointing to node j. In our setting, we assume that link e_{ij} and weight of the link w_{ij} are not desired to be known by nodes other than node i and node j. Furthermore, we design our link analysis solutions based on the three privacy models of graphs described as below:

Weight-aware model. If both the head node i and the tail node j know the existence of the link and the weight value, this is designated as *weight-aware link-aware* model (or weight-aware model for short). For example, consider commercial relationships among enterprises. Each enterprise may conduct business transactions with the other enterprises. Let the ith enterprise purchase some products from the jth enterprise. This transaction corresponds to link e_{ij} and the transaction value corresponds to weight w_{ij}. In this case, both the ith and jth enterprise are aware of the existence of this link and know the weight value, but enterprises other than i and j do not know the existence of this transaction and the transaction value.

Link-aware model. If the head node i and the tail node j know the existence of the link, but the weight value is only known by the head node i, this is designated as *link-aware weight-unaware* model (or link-aware model for short). For example, consider call logs of cell-phones. Let caller i make a phone call to receiver j. This call corresponds to link e_{ij} and the probability that i makes a phone call to j corresponds to the weight w_{ij} of e_{ij}. In this case, both caller i and receiver j are aware of the existence of the link, but the call probability w_{ij} are known only by caller i.

Link-unaware model. If only the head node i knows the existence of the link and the weight value, but the tail node j knows nothing, this is designated as *link-unaware weight-unaware* model (or link-unaware model for short). For example, consider a peer evaluation scheme among members of personnel. Each member can choose a limited number of other members and provide eval-

Link Analysis for Private Weighted Graphs

- It would be nice if we could compute PageRank without exposing the connection network.

Link Analysis for Private Weighted Graphs

- Commercial Relationships
  - Business i and Business j do w amount of business
  - Only i and j know this
  - "Weight-aware model"

Link Analysis for Private Weighted Graphs

- Personal Relationships
  - Person i calls person j do w amount per month
  - Persons i and j know about the call, but only i knows w (j doesn't know who else i calls, or how much)
  - No one else knows about the call
  - "Link-aware model"

Link Analysis for Private Weighted Graphs

- Professional Relationships
  - Person i ranks person j as being w compared to peers
  - Person i knows the ranking and the value
  - Neither j, nor anyone else knows about the call
  - "Link-unaware model"

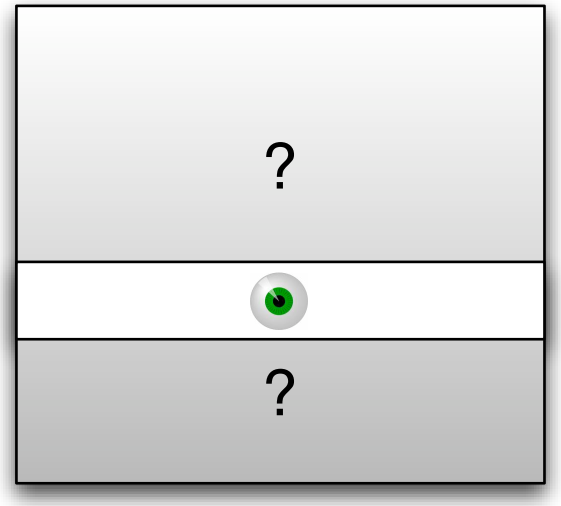
Link Analysis for Private Weighted Graphs

- The goal of this paper is:
    - To present secure protocols for link analysis with private weighted graphs

Link Analysis for Private Weighted Graphs

- M is an n by n matrix
  - Row Privacy

Definition 1. (Row private) Let there be a $n \times n$ matrix M and n parties. For all i, if the ith party knows a row vector $\boldsymbol{m}_{i*}$, but does not know other row vectors $\boldsymbol{m}_{p*} (p \neq i)$ of M, then M is *row private*.



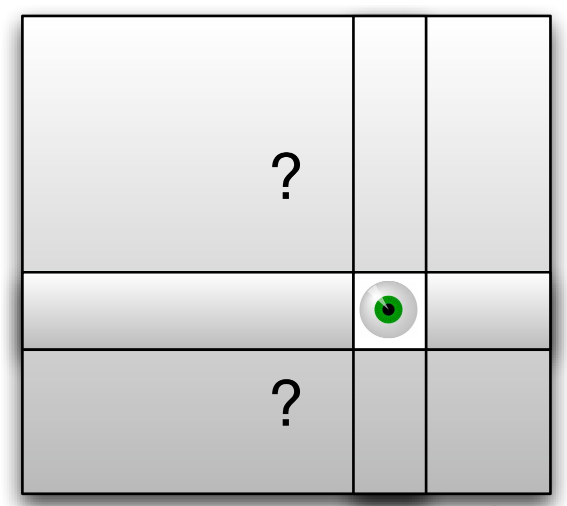
Link Analysis for Private Weighted Graphs

- M is an n by n matrix
  - Row Privacy

    Definition 1. (Row private) Let there be a $n \times n$ matrix M and n parties. For all i, if the ith party knows a row vector $\boldsymbol{m}_{i*}$, but does not know other row vectors $\boldsymbol{m}_{p*} (p \neq i)$ of M, then M is *row private*.

  - Symmetrically Private

    Definition 2. (Symmetrically private) Let there be a $n \times n$ matrix M and n parties. For all i, if the ith party knows $\boldsymbol{m}_{i*}$ and $\boldsymbol{m}_{*i}$, but does not know other elements m_{pq} where $p, q \neq i$, then M is *symmetrically private*.



Link Analysis for Private Weighted Graphs

- Graph is $G = \{V, A, W\}$
  - V are vertices
  - A is adjacency matrix (0,1)
  - W is weight matrix
- Weight-Aware Private Graph
  - A and W are symmetrically private
- Link-Aware Private Graph
  - A is symmetrically private, W is row private
- Link-Unaware Private Graph
  - A and W are row private

Link Analysis for Private Weighted Graphs

- Assumption
  - Global Dual-Key Cryptographic System
    - One public key p_k
    - One private key for each node k_i

Link Analysis for Private Weighted Graphs

- We are going to have each node in the graph compute PageRank for itself

$$p_i = \sum_j p_j A_{ji}$$

- This is very similar to PageRank on MapReduce
- Where each node does the Reduce Work for itself

Link Analysis for Private Weighted Graphs

- Keys to achieving this:
  - Additive Homomorphic Cryptosystem

    $$Enc_{pk}(x+y) = Enc_{pk}(x) \cdot Enc_{pk}(y)$$

    - Enable calculating a sum without knowing the individual elements
  - Threshold Decryption which allows $Enc_{pk}(x)$ to be decrypted when at least t nodes agree to decrypt
  - Onion Routing to prevent knowledge of source of summands

Link Analysis for Private Weighted Graphs

- Keys to achieving this:
  - Ability to detect convergence

$$(Enc_{pk}(p_i) - Enc_{pk}(p_{i-1})) \stackrel{?}{<} \theta$$

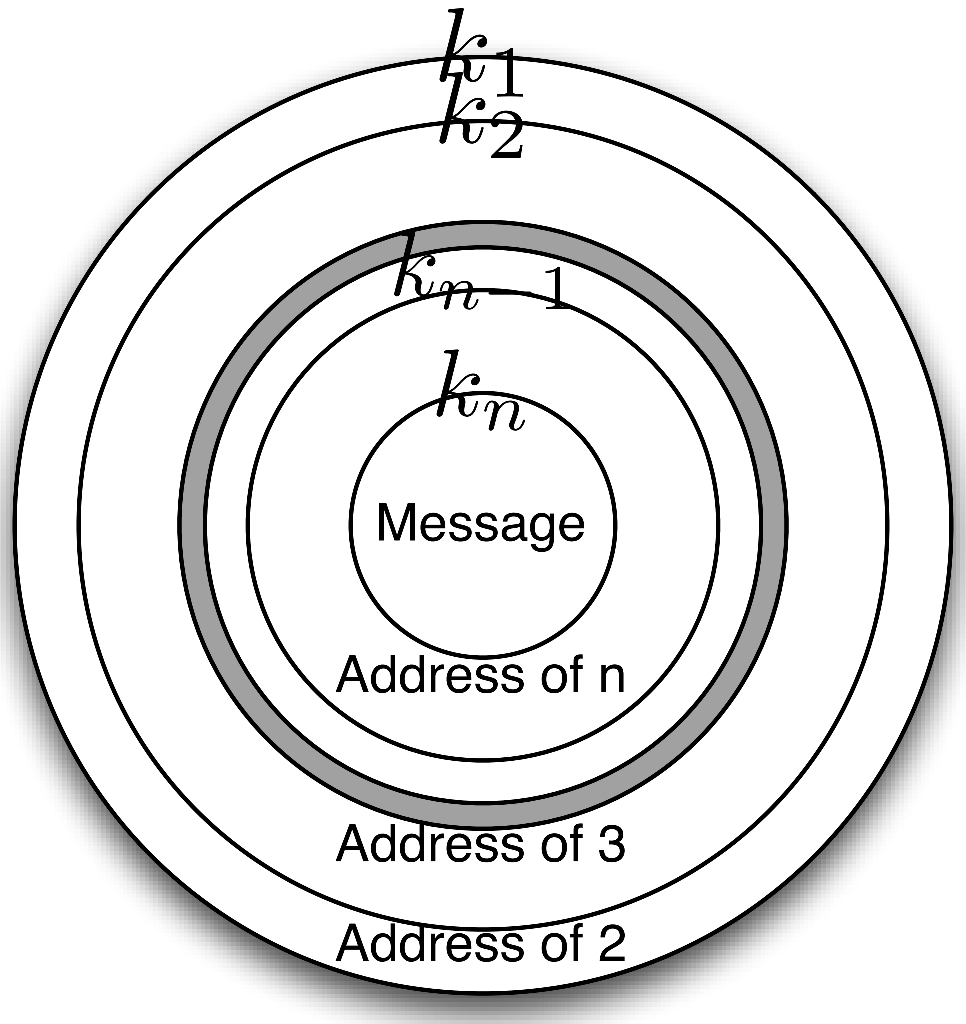
  - Ability to normalize

$$Enc_{pk}(p_{i+1}) = \frac{\sum Enc_{pk}(p_i)}{n}$$

  - Solution:
    - Secure Function Evaluation

Link Analysis for Private Weighted Graphs

- Onion Routing
  - Sender picks a route
    - Encodes the message and the route in an "onion"
    - Each node can decrypt the next hop only
    - Destination can decrypt the message



Link Analysis for Private Weighted Graphs

- Basic Flow
    - Keys are distributed
    - Probabilities are initialized
    - Each node encrypts $p_i a_{ij}$ and passes it to j
    - j securely sums incoming results
    - j normalizes
    - repeat until convergence between step i and i+1
        - globally coordinate stepping
        - globally coordinate covergence checking
    - globally decrypt

Link Analysis for Private Weighted Graphs

- Summary
  - Using the same technique as PageRank on MapReduce private PageRank can be calculated
  - It requires
    - A global public key
    - A distribution of private keys
    - Nodes to calculate their own PageRank
    - Additive Homomorphic CryptoSystem
    - Onion routing
    - Secure Function Evaluation