

統合したグラフのプライバシー保護リンク解析

森井 正 覚^{†1} 佐久間 淳^{†2}
佐藤 一 誠^{†1} 中川 裕 志^{†1}

リンク解析はエンティティとそれらの関係を表すリンクによって表現されたグラフ構造から有用な情報を抽出する手法として用いられている。複数のパーティが秘密に保持するグラフを集めて統合することにより1つのグラフを構成し、リンク解析をすることを考える。既存のリンク解析をそのまま用いると、各パーティが保持するデータのプライバシーが守られない。本稿では、このプライバシー保護に関する問題点を解決する方法として、複数のパーティが秘密に保持するグラフを統合したモデルを考え、それらに対するプライバシーを保護したリンク解析手法を提案する。我々の手法は、暗号的ツールを組み合わせて作られており、全パーティが semi-honest に振る舞う限り理論的に安全である。

Secure Link Analysis for Integrated Graphs

SHOGAKU MORII,^{†1} JUN SAKUMA,^{†2} ISSEI SATO^{†1}
and HIROSHI NAKAGAWA^{†1}

Link analysis is a method to obtain knowledge from graph structure that is represented by entities and their relationships. We consider collecting graphs that two or more parties hold in secret and composing a graph by integrating these secret graphs. When we analyze the integrated graph using current link analysis methods, the privacy of data that each party holds does not be preserved. In order to solve this privacy issue, we propose a secure method to analyze links for graphs that are integrated from the graphs that each party holds. Our method consists of cryptographic tools and provides theoretical security as long as all parties are semi-honest.

^{†1} 東京大学
The University of Tokyo
^{†2} 筑波大学
University of Tsukuba

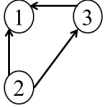
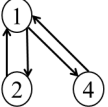
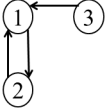
1. はじめに

情報技術の普及にともない、web 訪問履歴や購買履歴といった個人情報を扱うサービスがさかんに提供されている。そのような個人情報が漏えいした場合の社会的影響は深刻であり、サービス提供者には慎重な扱いが求められる。一方、複数のサービス提供者の保持する詳細な個人情報を統合して用いるデータマイニングは、実社会における情報活用に大きく貢献すると期待される。そのような状況下で、データのプライバシーを保護しながらデータマイニングを実現する研究（プライバシー保護データマイニング）がさかんに行われている。多くのデータマイニング手法に対して、データのプライバシーを保護する手法が提案されているが、本稿では、リンク解析におけるプライバシー保護について扱う。

リンク解析とは、エンティティとそれらの関係を表すリンクによって表現されたグラフ構造から有用な情報を抽出する手法である。既存のリンク解析は、解析者にエンティティのリンク構造全体が見えているということ为前提としている。しかしながら、人間関係や企業取引などの実世界でのリンク情報が公であることは稀である。Sakuma らは、そのような秘密のリンク関係を持つエンティティのグラフにおけるプライバシーモデルを定義し、それらのモデルに基づくプライバシーを保護したリンク解析を提案した⁵⁾。彼らの提案したリンク解析は自身に関係するリンク情報しか知りえないモデルにおけるリンク解析である。それゆえ、彼らの研究ではグラフ上のエンティティ単体を1つのパーティと見なしている。我々の研究では、グラフ上の複数のエンティティ（例：通信事業者の顧客またはシンクタンクによって調査された企業）に関するデータベースをパーティ（例：通信事業者またはシンクタンク）が秘密に保持することを想定する。各パーティがエンティティ間のリンク情報の一部を保持しており、そのリンク情報を他パーティに対しては知らせたくない状況を考える。我々の手法は、Sakuma らの手法と以下の3つの点で異なっている。第1に、各パーティは複数のエンティティについてのリンク情報を保持していることである。第2に、各パーティは同じエンティティについて異なるリンク情報を保持しているかもしれないことである。最後に、パーティは単体のエンティティである必要はないということである。我々は、複数のパーティが秘密に保持するグラフを統合したグラフに対してのリンク解析を提案する。

我々の問題設定を通話記録のマイニングを例に説明する。通信事業者とその顧客をそれぞれパーティとエンティティと見なす。簡単のため、通信事業者が Alice 社と Bob 社の2社の場合を考える。Alice 社の顧客は、Alice 社の他の顧客または Bob 社の顧客と通話とすることができ、逆も同様である。この例では、各通信事業者は通話記録をリンクとリン

表 1 グラフの統合の例
Table 1 Example of integration of graphs.

パーティ	グラフ	重み行列	統合したグラフの重み行列
Alice		$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 4 & 0 & 4 & 0 \\ 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	加法的統合 $\begin{bmatrix} 0 & 4 & 0 & 1 \\ 9 & 0 & 4 & 0 \\ 8 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{bmatrix}$
Bob		$\begin{bmatrix} 0 & 3 & 0 & 1 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{bmatrix}$	
Carol		$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	
			平均的統合 $\begin{bmatrix} 0 & 2 & 0 & 1 \\ 3 & 0 & 4 & 0 \\ 4 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{bmatrix}$

クの重みとして保持しているとし、エンティティ間のリンクは通話履歴に対応し、エンティティ間のリンクの重みは通話時間に対応しているとする。通話記録は2種類に分けることができる。1つは、通信事業者が異なる顧客同士の通話記録で、これは両通信会社が共有している情報である。あと1つは、通信事業者が同じ顧客同士の通話記録で、これはその通信事業者のみが秘密に保持しており、他の通信事業者とは共有したくない情報である。目的は、互いに共有されていない秘密の情報を用いて、仮想的に統合した通話記録に対してリンク解析をすることである。たとえば、エンティティのランキングをし、重要な顧客を知ることは、通信事業者にとっては有益なことだろう。さらに、1社が保持しているグラフを用いてリンク解析をするよりも、各通信事業者が保持しているグラフの重み行列を足したグラフに対してリンク解析を行ったほうが、正確な結果が得られることが期待できる。また、一般的な問題設定としては、共有しているリンク情報はまったく存在しなくてもよい。

本稿では、各パーティ P_k ($k = 1, \dots, l$) は重みつき有向グラフを保持しており、計算能力が多項式オーダに制限された計算機を保持していることを想定する。各パーティが保持するリンクとリンクの重み情報は他のパーティには公開したくない秘密情報とする。また、任意の2パーティ間の通信は、つねに可能であるとする。我々は、そのような秘密情報を含むいくつかのグラフを統合したグラフに対する統合モデルを2つ考える。例として、表1

にあるように3パーティの場合を考える。エンティティ間のリンクはグラフにおける矢印で表されており、それに対応する重みが重み行列として表されている。ただし、各パーティが保持するグラフでリンクが存在しないエンティティに関しては表示されていない。各パーティが保持するグラフを統合するにあたって、重み行列の総和をとる加法的統合と、リンクが存在する重みのみ、存在するパーティ間で平均をとる平均的統合が考えられる。

機密性やプライバシーの問題のために、各エンティティ間のリンク情報が他パーティに対して公にできないことも少なくない。もし秘密のリンク情報を統合したネットワークを対象として、その秘匿性を損なうことなく安全にリンク解析を適用できれば、現実の多様なネットワークからの情報抽出を可能にする。本稿では、その統合したグラフに対する安全なリンク解析アルゴリズムを提案し、これを周知の PageRank アルゴリズムに適用する。

2. リンク解析

本章では、spectral ranking と呼ばれる基本的なリンク解析問題を導入し、有名な手法である PageRank⁴⁾ を導入する。まずは、言葉の定義と記法を定める。

ノード集合 $V = \{1, \dots, n\}$ 、リンク集合 $E = \{e_{ij}\}$ 、および重み行列 $W = (w_{ij})$ からなる非負の重み付き有向グラフ $G = (V, E, W)$ を考える。エンティティはノードとして抽象化される。ノード i とノード j の間にリンクが存在しなければ、 $w_{ij} = 0$ とする。ノード i の度数は $d_i = \sum_{j \in V} w_{ij}$ と定義される。 $D = \text{diag}(d_1, \dots, d_n)$ を度数行列と呼ぶ。隣接行列 $A = (a_{ij})$ は下式によって定義される：

$$a_{ij} = \begin{cases} 1 & (\text{if } e_{ij} \in E) \\ 0 & (\text{o.w.}) \end{cases} \quad (1)$$

2.1 Spectral Ranking

リンク解析は、与えられたグラフのリンク構造をもとに、有用な情報を抽出する手法である。グラフ上のマルコフランダムウォークにおける定常分布密度によりノードのスコアを計算する方法を spectral ranking と呼び、本稿ではリンク解析として spectral ranking に着目する。

ノード i からノード j に、確率 p_{ij} で遷移するマルコフ連鎖を考える。ただし状態遷移確率行列 $P = (p_{ij})$ を $P = D^{-1}W$ として定義する。定常分布 $x = (x_1, \dots, x_n)^T$ は遷移後もその分布を変えないことから以下を満たす：

$$x^T = x^T P. \quad (2)$$

ただし $\sum_i x_i = 1$ である．この定常分布は， P^T の最大の固有値 ($= 1$) と対になる固有ベクトル (主固有ベクトル) に対応することが知られている．

2.2 PageRank

PageRank⁴⁾ は状態遷移確率に下式を用いた spectral ranking の一種である．

$$P = (1 - \epsilon)D^{-1}A + \frac{\epsilon}{n}\mathbf{1}\mathbf{1}^T \quad (3)$$

ここで， $\mathbf{1}$ は全要素が 1 であるようなベクトルである．この状態遷移確率行列は，web 文書において，確率 $1 - \epsilon$ で現在の文書に含まれるハイパーリンクからランダムに選択して遷移し，確率 ϵ で全文書集合からランダムに選択された文書に遷移するようなユーザの行動をモデル化したものである．

主固有ベクトルの計算にはべき乗法がしばしば用いられる．初期値として $\sum_i x_i^{(0)} = 1$ なるベクトルを与え，以下の更新式を繰り返す：

$$(\mathbf{x}^{(t)})^T \leftarrow (\bar{\mathbf{x}}^{(t-1)})^T P, \quad \bar{\mathbf{x}}^{(t)} \leftarrow \frac{\mathbf{x}^{(t)}}{\|\mathbf{x}^{(t)}\|} \quad (4)$$

P が確率行列の場合は，式 (4) の右側の正規化ステップは省略可能である．べき乗法の収束性は以下に示される (証明略)：

補題 1. \mathbf{x} を P の主固有ベクトルとする． $\mathbf{x}^{(t)}$ をべき乗法によって t 回更新した後に得られたベクトルとする．このとき

$$\|\mathbf{x}^{(t)} - \mathbf{x}\| = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^t\right) \quad (5)$$

が成立する．ただし， λ_1, λ_2 はそれぞれ，絶対値が最大，および 2 番目に大きい P の固有値である．

3. 問題の定式化

本章では，各パーティが複数のエンティティのリンク情報を秘密に保持している状況で，それらのグラフを統合したグラフにおけるデータ分割モデルを定義する．データ分割モデルとは，データマイニングの対象になる入力をパーティ間でどのように秘密に共有するかを定めるものである．プライバシ保護データマイニングにおけるデータ分割モデルとしては，垂直分割モデルと水平分割モデルがよく用いられている．垂直分割モデルでは，全データエントリにおける，属性の部分集合が分割されたデータセットを構成する．このモデルでは，全パーティが，各データエントリに関するユニークな ID を共有していることが前提である．

水平分割モデルでは，全属性における，データエントリの部分集合がデータセットを構成する．しかしながら，我々の分割モデルはそれらの典型的なモデルとは異なる．

そして，それらのモデルに基づいたリンク解析を定義する．各パーティ P_k ($k = 1, \dots, l$) が重みつき有向グラフ G^k ($k = 1, \dots, l$) をそれぞれ秘密に保持しているとする状況を考える．本稿では，上付き文字 k は主にパーティ P_k だけが秘密に保持している情報を表す．

定義 1. (Integrated private-weighted graph; IPWG) パーティ P_k ($k = 1, \dots, l$) はグラフ $G^k = (V, E^k, W^k)$ だけを知っているとす．ただし， $|V| = n$ とす．有限集合を S とし，リンク統合関数 $g: (S, \dots, S) \rightarrow S$ と重み統合関数 $h: (S, \mathbb{R}^{n \times n}) \times \dots \times (S, \mathbb{R}^{n \times n}) \rightarrow \mathbb{R}^{n \times n}$ を考える．それらの統合関数を用いて，integrated private-weighted graph G を以下で定義する．

$$G = (V, g(E^1, \dots, E^l), h((E^1, W^1), \dots, (E^l, W^l))) \quad (6)$$

ノード集合 V は全パーティ間で共有していると仮定する．携帯電話会社のシナリオでは，グラフのエンティティに対応する各電話番号は，ある規則に従った数字列として定まる．それらを昇順に並べることにより，全パーティは各エンティティに一意のノード番号を割り当てることができる．それゆえ，この例のような状況下でこの仮定は妥当である．複数のグラフを統合して 1 つの IPWG を作る時，そのエッジ集合は複数のグラフのエッジ集合の和集合とするのが自然である．しかし，重み行列の統合の仕方は複数考えられる．実世界に現れる 2 種類のプライバシモデルを定義する．

定義 2. (Additive IPWG) 定義 1 において，リンク統合関数と重み統合関数をそれぞれ $g(E^1, \dots, E^l) = \cup_{k=1}^l E^k$ と $h((E^1, W^1), \dots, (E^l, W^l)) = \sum_{k=1}^l W^k$ とす．IPWG $G = (V, \cup_{k=1}^l E^k, \sum_{k=1}^l W^k)$ を additive integrated private-weighted graph と呼ぶ．

定義 3. (Average IPWG) 定義 1 において，リンク統合関数と重み統合関数をそれぞれ $g(E^1, \dots, E^l) = \cup_{k=1}^l E^k$ と $h((E^1, W^1), \dots, (E^l, W^l)) = W = (w_{ij})$ とす．ここで，

$$w_{ij} = \begin{cases} 0 & (\text{if for all } k, e_{ij}^k \notin E^k) \\ \frac{1}{\#\{k | e_{ij}^k \in E^k\}} \sum_{k | e_{ij}^k \in E^k} w_{ij}^k & (\text{o.w.}) \end{cases} \quad (7)$$

とする．IPWG $G = (V, \cup_{k=1}^l E^k, W)$ を average integrated private-weighted graph と呼ぶ．

続いて，IPWG 上でのリンク解析を定義する． $f: \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^n$ をリンク解析のためのス

コアリング関数とする． f は重み行列 $W \in \mathbb{R}^{n \times n}$ を入力として，スコアベクトル $x \in \mathbb{R}^n$ を出力する．このとき secure integrated link analysis は以下のように定義される．

定義 4. (*Secure integrated link analysis*) $G = (V, E, W)$ を IPWG とする．*Secure integrated link analysis* の実行後， $f(W) \rightarrow x$ は正しく評価され，各パーティは x を知るが，それ以外の知識は得ない．

4. 暗号学的ツール

本章では，定義 4 によるリンク解析を実現するために必要ないくつかの暗号学的ツールを導入する．各パーティは自身が保持する情報を他パーティに対して公開したくないが，べき乗法を用いて秘密の行列の固有ベクトルを求めることができれば，プライバシーを保護したリンク解析が可能になる．べき乗法の計算は加算と乗算から構成されるので，暗号化したまま加算と乗算が可能な準同型性公開鍵暗号を導入する．また，統合したグラフから各パーティの情報が推測されないようにするためランダムシェアという概念を導入し，サブプロトコルである weighted average random share protocol を提案する．

4.1 準同型性公開鍵暗号

公開鍵暗号系において，暗号化は公にされた公開鍵 pk を，解読には受信者のみが保持する公開鍵に対応した秘密鍵 sk を用いる．平文 m について， $c = \text{Enc}_{pk}(m; \rho)$ は m の確率暗号による暗号化を， $m = \text{Dec}_{sk}(c)$ はその解読を表す． ρ が $\mathbb{Z}_N (= \{0, 1, \dots, N-1\})$ 上で一様ランダムに選ばれたならば，暗文 c も同様に \mathbb{Z}_N で一様ランダムに分布する．加法的準同型公開鍵暗号は，秘密鍵の知識なしに，暗文どうしの加算

$$\text{Enc}_{pk}(m_1 + m_2; \rho) = \text{Enc}_{pk}(m_1; \rho_1) * \text{Enc}_{pk}(m_2; \rho_2) \quad (8)$$

が可能である．ここで， ρ_1 が ρ_2 の少なくともどちらか 1 つが \mathbb{Z}_N 上で一様ランダムならば，同様に ρ は一様ランダムである．この性質に基づき，定数 κ と暗文 $\text{Enc}_{pk}(m; \rho)$ の乗算が， $*$ の繰返しにより以下のように実現される．

$$\text{Enc}_{pk}(\kappa m; \rho) = \prod_{i=1}^{\kappa} \text{Enc}_{pk}(m; \rho_i) = \text{Enc}_{pk}(m)^{\kappa}. \quad (9)$$

ρ は $\rho_1, \dots, \rho_{\kappa}$ の少なくともどちらか 1 つが \mathbb{Z}_N 上で一様ランダムならば，同様に一様ランダムである．以降は，簡単のために乱数 ρ は表示しない．

(u, y) -閾値暗号系では， y パーティが共通の公開鍵 pk を保持し，各パーティはそれぞれ u 個に分割された異なる秘密鍵 sk^1, \dots, sk^j を保持している．各パーティは共通の公開鍵により任

意のメッセージを暗号化可能である．一方，解読には，少なくとも y 以上のパーティのグループが協力し，公開鍵とそれぞれのノードが持つ decryption shares $\text{Dec}_{sk^1}(c), \dots, \text{Dec}_{sk^j}(c)$ を指数にとる recovery アルゴリズムを実行する必要がある．本稿で示すプロトコルは，加法的準同型性を持つ閾値暗号系を用いる．

4.2 ランダムシェア

行列 $X \in \mathbb{Z}_N^{n_1 \times n_2}$ の各要素 $x_{ij} \in \mathbb{Z}_N$ (for all i, j) が， $\sum_{k=1}^l r_{ij}^k \bmod N = x_{ij}$ ，を満たすように \mathbb{Z}_N から一様ランダムに選択された $r_{ij}^1, \dots, r_{ij}^l$ に分割されているとする．パーティ P_1, \dots, P_l が X を知らずに $r_{ij}^1, \dots, r_{ij}^l$ をそれぞれ保持しているとき，これを X のランダムシェアによる秘密共有，と呼ぶ．本稿では，すべての演算が有限体 \mathbb{Z}_N 上で行われるように， N は十分大きい自然数とする．

4.3 Secure Scalar Product Protocol

パーティ P_1 と P_2 がそれぞれベクトル $x^1 = (x_1^1, \dots, x_{n_1}^1)^T$ と $x^2 = (x_1^2, \dots, x_{n_1}^2)^T$ を保持しているとする．それらの内積のランダムシェアを計算するために Goethals らによって提案された secure scalar product protocol²⁾ を用いる．以下では，secure scalar product protocol を \mathcal{P}_{SSP} と略記し，アルゴリズムの説明に用いる．

4.4 Weighted Average Random Share Protocol

パーティ P_k ($k = 1, \dots, l$) が自然数 x^k, α^k のペア (x^k, α^k) を保持しているとする．各パーティは互いに協力し， $\sum_{k=1}^l x^k / \sum_{k=1}^l \alpha^k$ をランダムシェアによって秘密共有を試みる．つまり，以下の機能を持つプライバシー保護プロトコルが必要である．

$$((x^1, \alpha^1), (x^2, \alpha^2), \dots, (x^l, \alpha^l)) \mapsto (r^1, r^2, \dots, r^l), \quad (10)$$

ここで， r^k ($k = 1, \dots, l$) は $\sum_{k=1}^l r^k = \sum_{k=1}^l x^k / \sum_{k=1}^l \alpha^k$ を満たすような一様ランダムな数である．ただし， $\alpha^1 = \dots = \alpha^l = 0$ のときは， $\sum_{k=1}^l r^k = 0$ とする．式 (10) は，パーティ P_k がプロトコルに入力として (x^k, α^k) を与え，出力として r^k を受け取ることを意味する．

以下では，weighted average random share protocol を \mathcal{P}_{WARS} とおく．閾値準同型性公開鍵暗号系の鍵集合を $\mathcal{K} = \{pk, sk^1, \dots, sk^j\}$ とする．秘密鍵は， sk^1, \dots, sk^j と m 個に分割され， sk^i はパーティ P_i のみが保持する．公開鍵 pk は全ノードが共有する．また， $\Delta \geq \sum_{k=1}^l \alpha^k$ であり， Δ はすべてのパーティが知っているとする． \mathcal{P}_{WARS} の手続きをアルゴリズム 1 に示す．

すべてのパーティが semi-honest に振る舞うことを想定する．つまり，各パーティは定め

られたプロトコルを逸脱しないが、実行途中で受け取ったすべての情報から他パーティの情報を推測しようとする。このとき、以下の補題が示される。

補題 2. アルゴリズム 1 に示されたプロトコルの出力の総和 $\sum_{k=1}^l r^k$ は、 $\alpha^1, \dots, \alpha^l$ を重みとする x^1, \dots, x^l の加重平均に等しい。

証明. ステップ 1 の c を $(\omega_1, \dots, \omega_{\Delta+1})$ とおく。ステップ 2(a) で、パーティ P_k は、 c の要素をそれぞれ α^k だけ移動させる。 $\sum_{k=1}^l \alpha^k$ を β とする。ステップ 3 で、パーティ P_1 によって全体に送信された $c[1]$ を $\omega_{\beta+1}$ とおく。ここで、もし、 $\beta = 0$ ならば $\text{Dec}_{\text{sk}}(\omega_{\beta+1}) = 1$ であり、そうでないならば、 $\text{Dec}_{\text{sk}}(\omega_{\beta+1}) = \Delta!/\beta$ である。準同型性公開鍵暗号は、加算や乗算はできるが除算はできない。また、平文は整数である必要があるため、 $1/\beta$ に $\Delta!$ をかけて整数にしている。暗号系の準同型性により、ステップ 4(b) でパーティ P_k が受け取った暗文は以下ようになる。

$$c'_k = \begin{cases} \text{Enc}_{\text{pk}}(x^k - s^k + s^{k+1 \bmod l}) & (\text{if } \beta = 0) \\ \text{Enc}_{\text{pk}}\left(\frac{\Delta!}{\beta}(x^k - s^k + s^{k+1 \bmod l})\right) & (\text{o.w.}) \end{cases} \quad (11)$$

ステップ 4(c) で、パーティ P_k は、 $r^k = \text{Dec}_{\text{sk}}(c'_k)/\Delta!$ を得るが、それらの値の総和は、

$$\sum_{k=1}^l r^k = \begin{cases} \sum_{k=1}^l x^k & (\text{if } \beta = 0) \\ \frac{1}{\beta} \sum_{k=1}^l x^k & (\text{o.w.}). \end{cases} \quad (12)$$

ここで、 $\beta = 0$ のとき、 $\sum_{k=1}^l x^k = 0$ であり、 $\beta = \sum_{k=1}^l \alpha^k$ だったので、補題は示せた。□

定理 1. すべてのパーティが *semi-honest* に振る舞い、かつ u パーティ以上の結託がないならば、アルゴリズム 1 に示されたプロトコルは正しくかつ安全に加重平均のランダムシェアを計算する。

証明. 証明の概略を示す。アルゴリズム 1 に示されたプロトコルが正しく加重平均のランダムシェアを計算することは、補題 2 よりただちに導かれる。全パーティが *semi-honest* に振る舞うことを想定する。プロトコルを通して交わされるメッセージは、それぞれステップ 2(c)、3 と 4(b) における c 、 $c[1]$ と c_k, c'_k である。それらのメッセージは、閾値準同型性公開鍵暗号系で暗号化されており、もし、あるパーティ P_k がそれらすべてのメッセージを知ったとしても、閾値準同型性公開鍵暗号系が安全で、 u パーティ以上の結託がない限り

は、パーティ P_k は他パーティの入力 $(x^{k'}, \alpha^{k'})$ ($k' \neq k$) を知ることはできない。それゆえ、プロトコルの安全性が証明できる。□

アルゴリズム 1 Weighted Average Random Share Protocol

Require: 公的な入力: $\Delta \geq \sum_{k=1}^l \alpha^k$ なる Δ .

パーティ P_k ($k = 1, \dots, l$) の秘密の入力: x^k, α^k .

鍵の設定: すべてのパーティが共同で鍵集合 $\mathcal{K} = \{\text{pk}, \text{sk}^1, \dots, \text{sk}^l\}$ を生成し、 pk はすべてのパーティが所持し、 sk^i はパーティ P_i だけが所持するように配布する。

1. パーティ P_1 は、暗文のベクトル $c = (\text{Enc}_{\text{pk}}(1), \text{Enc}_{\text{pk}}(\frac{\Delta!}{1}), \dots, \text{Enc}_{\text{pk}}(\frac{\Delta!}{\Delta}))$ を計算する。 $c[i]$ を c の i 番目の要素とする。

2. パーティ P_k ($k = 1, \dots, l$) は以下を行う。

(a) パーティ P_k は $c \leftarrow (c[\alpha^k + 1], c[\alpha^k + 2], \dots, c[\Delta + 1], c[1], c[2], \dots, c[\alpha^k])$ を計算する。

(b) パーティ P_k は $c \leftarrow (c[1] \cdot \text{Enc}_{\text{pk}}(0), c[2] \cdot \text{Enc}_{\text{pk}}(0), \dots, c[\Delta + 1] \cdot \text{Enc}_{\text{pk}}(0))$ を計算する。

(c) パーティ P_k は c をパーティ P_{k+1} へ送る。

3. パーティ P_l は $c[1]$ をすべてのパーティに知らせる。

4. パーティ P_k ($k = 1, \dots, l$) は以下を行う。

(a) パーティ P_k は乱数 $s^k \in_r \mathbb{Z}_N$ を生成する。

(b) パーティ P_k は $c_k = x^k - s^k$ をパーティ $P_{k+1 \bmod l}$ に送る。パーティ $P_{k+1 \bmod l}$ は $c'_k = c[1]^{c_k + s^{k+1 \bmod l}}$ をパーティ P_k に送る。

(c) パーティ P_k は recovery アルゴリズムを実行し、 $r^k = \text{Dec}_{\text{sk}}(c'_k)/\Delta!$ を得る。

5. Secure Integrated Link Analysis

本章では、定義 4 に基づく additive/average IPWG に対する secure integrated spectral ranking (SISR) を提案し、その PageRank への拡張を示す。SISR の全体的な手順をアルゴリズム 2 に示す。

5.1 Secure Integrated Spectral Ranking

SISR のステップは、確率遷移行列のランダムシェア、初期設定、べき乗法の更新、正規

アルゴリズム 2 Secure Integrated Spectral Ranking

Require: 公的な入力: $K \in \mathbb{Z}_N, L \in \mathbb{Z}_N$ s.t. $Lp_{ij}^k \in \mathbb{Z}_N$ for all i, j, k , IPWG の種類 .

パーティ P_k ($k = 1, \dots, l$) の秘密の入力: W^k, A^k .

鍵の設定: すべてのパーティが共同で鍵集合 $\mathcal{K} = \{pk, sk^i, \dots, sk^j\}$ を生成し, pk はすべてのパーティが所持し, sk^i はパーティ P_i だけが所持するように配布する .

1. (B のランダムシェア) 各パーティは $\mathcal{P}_{\text{WARS}}$ を用いて $B = LP$ のランダムシェアを得る . パーティ P_k はランダムシェア B^k を保持している . ここで, $B = \sum_{k=1}^l B^k$ である .

2. (初期化) パーティ P_k はすべての i について, 以下のように設定する .

$$q_i^{(0),k} \leftarrow K_i^k \text{ s.t. } \sum_{k=1}^l \sum_{i=1}^n K_i^k = K, t \leftarrow 1$$

3. (べき乗法) 各パーティ P_k は以下の計算を収束するまで繰り返す .

(a) パーティ P_k は $B^k q^{(t-1),k}$ を計算する .

(b) すべての i と $k' \neq k$ なるすべての k' について, パーティ P_k は \mathcal{P}_{SSP} を用いてランダムシェア $r_{i,k'}^{(t-1),k}, s_{i,k}^{(t-1),k'}$ を得る . ここで, $r_{i,k'}^{(t-1),k} + s_{i,k}^{(t-1),k'}$ は, B^k の i 番目の行と $q^{(t-1),k'}$ の内積である .

(c) すべての i について, パーティ P_k は $q_i^{(t),k} \leftarrow B^k q^{(t-1),k} + \sum_{k' \neq k} (r_{i,k'}^{(t-1),k} + s_{i,k}^{(t-1),k'})$ を計算する .

(d) パーティ P_i とランダムに選ばれたパーティ P_j ($j \in \{1, \dots, l\} \setminus \{i\}$) は収束を判定するプロトコルを実行する . 収束していなければ, “未収束” とすべてのパーティに知らせる . もしそのようなメッセージがなければ, ステップ 4 へ進み, そうでなければ, ステップ 3(a) へ戻る .

4. (復号) パーティ P_k は, 復号を実行し $q^{(t),k}$ を得, それをすべてのパーティに知らせる . それゆえ, 出力は $\pi^{(t)} = \sum_{k=1}^l q^{(t),k} / KL^{t-1} = q^{(t)} / KL^{t-1}$ となる .

化と収束検出からなる . 以下の項では, それらのステップについての詳細な説明を行う .

5.1.1 確率遷移行列のランダムシェア

問題の定式化で述べたように, リンクとリンク間の重みは公にすべき情報ではない . Spectral ranking には IPWG から計算される確率遷移行列 P が必要であるが, 各パーティは P について何も情報を得ない状況で, SISR を実現したい . そのために, P をランダムシェアで秘密共有する . ランダムシェアを計算するために, $\mathcal{P}_{\text{WARS}}$ を用いる . Additive/average

IPWG の重み行列に対するランダムシェアアルゴリズムを以下に示す .

Additive IPWG に対するランダムシェアアルゴリズム

以下をすべての i, j に対して行う .

(1) 各パーティ P_k ($k = 1, \dots, n$) は, 入力を $(w_{ij}^k, \sum_{j=1}^n w_{ij}^k)$ として $\mathcal{P}_{\text{WARS}}$ を用いてランダムシェア r_{ij}^k を得る .

(2) $\mathcal{P}_{\text{WARS}}$ の出力 r_{ij}^k を P^k の (i, j) 成分とする .

Average IPWG に対するランダムシェアアルゴリズム

以下をすべての i, j に対して行う .

(1) 各パーティ P_k ($k = 1, \dots, n$) は, 入力を $(w_{ij}^k, \sum_{j=1}^n w_{ij}^k)$ として $\mathcal{P}_{\text{WARS}}$ を用いてランダムシェア z_{ij}^k を得る .

(2) 各パーティ P_k ($k = 1, \dots, n$) は, 入力を $(z_{ij}^k, \sum_{j=1}^n z_{ij}^k)$ として $\mathcal{P}_{\text{WARS}}$ を用いてランダムシェア r_{ij}^k を得る .

(3) $\mathcal{P}_{\text{WARS}}$ の出力 r_{ij}^k を P^k の (i, j) 成分とする .

もし, $\mathcal{P}_{\text{WARS}}$ の出力が整数でなければ, 全パーティが十分大きい自然数をかけることによって整数に拡大すればよい .

5.1.2 初期設定

アルゴリズム 2 のステップ 1 で各パーティは確率遷移行列 $P = D^{-1}W$ をランダムシェアにより秘密共有する . 典型的な暗号系は整数のみを指数にとるため, p_{ij}^k は十分に大きい定数 L に基づいて, $b_{ij}^k (= Lp_{ij}^k) \in \mathbb{Z}_N$ となるように拡大される . 同様の理由により, 初期定常分布 q も十分に大きい定数 K を用いて, $\sum_{k=1}^l \sum_{i=1}^n q_i^k = K$ となるように初期化する .

5.1.3 べき乗法

IPWG である $G = (V, E, W)$ に対して, $B = LP$ を l パーティでランダムシェアによる秘密共有をする . ここで, $B = \sum_{k=1}^l B^k$ であり, パーティ P_k は, B^k のみを知っている . また, パーティ P_k のみが知っているベクトル $q^{(t),k}$ に対して, $q^{(t)} = \sum_{k=1}^l q^{(t),k}$ とおく . 正規化ステップを省略したべき乗法の更新式は以下ようになる .

$$\begin{aligned} B^T q^{(t)} &= \left(\sum_{k=1}^l (B^k)^T \right) \left(\sum_{k=1}^l q^{(t),k} \right) \\ &= \sum_{k=1}^l ((B^k)^T (q^{(t),1} + \dots + q^{(t),l})) \end{aligned}$$

$$= (B^1)^T \mathbf{q}^{(t),1} + \dots + (B^l)^T \mathbf{q}^{(t),l} + \dots \\ + (B^1)^T \mathbf{q}^{(t),1} + \dots + (B^l)^T \mathbf{q}^{(t),l}. \quad (13)$$

各パーティ P_k は, $(B^k)^T \mathbf{q}^{(t),k}$ を独自に計算できるので, 式 (13) の計算は, $(B^k)^T \mathbf{q}^{(t),k}$ ($\forall k \neq k'$) の安全な計算ができればよいことになる. これは, \mathcal{P}_{SSP} を用いることにより計算が可能である. 各パーティは, $(B^k)^T$ の行と $\mathbf{q}^{(t),k'}$ の内積をランダムシェアによって秘密共有する. \mathcal{P}_{SSP} を用いた後, パーティ P_k と $P_{k'}$ がランダムシェア $\mathbf{r}_{k'}^{(t),k}$ と $\mathbf{s}_k^{(t),k'}$ をそれぞれ保持しているとする. パーティ P_k は以下のようにして $\mathbf{q}^{(t),k}$ を更新できる.

$$\mathbf{q}^{(t+1),k} \leftarrow (B^k)^T \mathbf{q}^{(t),k} + \sum_{k' \neq k} \left(\mathbf{r}_{k'}^{(t),k} + \mathbf{s}_k^{(t),k'} \right), \quad (14)$$

ここで, $\sum_{k=1}^l \mathbf{q}^{(t+1),k}$ は, $B^T \mathbf{q}^{(t)}$ に等しい. このように, 全パーティは式を秘密に更新できる. 以下の補題にはプロトコルが正しく P の定常分布を求めることを示している.

補題 3. π^* を P によるマルコフランダムウォークの定常分布とし, $SISR$ の更新回数を t とする. このとき $\pi^{(t)}$ は, 以下を満足する確率ベクトルである.

$$\|\pi^* - \pi^{(t)}\| = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^t\right), \quad (15)$$

ここで λ_1 および λ_2 は最大および 2 番目に大きい P の固有値である.

証明. ステップ 3(c) の更新は, 暗号系の準同型性を考慮すると,

$$\mathbf{q}^{(t)} = \sum_{k=1}^l \mathbf{q}^{(t),k} \\ \leftarrow \sum_{k=1}^l \left((B^k)^T \mathbf{q}^{(t-1),k} + \sum_{k' \neq k} \left(\mathbf{r}_{k'}^{(t-1),k} + \mathbf{s}_k^{(t-1),k'} \right) \right) \quad (16)$$

と整理される. ここで, パーティ P_k と $P_{k'}$ ($k' \neq k$) は, それぞれ $\mathbf{r}_{k'}^{(t-1),k}$ と $\mathbf{s}_k^{(t-1),k'}$ を保持しており, $(B^k)^T \mathbf{q}^{(t-1),k}$ をランダムシェアにより秘密共有している. それゆえ, 式 (16) は以下のように変形できる.

$$\mathbf{q}^{(t)} \leftarrow \sum_{k=1}^l \left((B^k)^T \mathbf{q}^{(t-1),k} + \dots + (B^k)^T \mathbf{q}^{(t-1),l} \right)$$

$$= \left(\sum_{k=1}^l (B^k)^T \right) \left(\sum_{k=1}^l \mathbf{q}^{(t-1),k} \right) = B \mathbf{q}^{(t-1)}. \quad (17)$$

これは, B を用いたべき乗法となる. $(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots)$ を P を用いたべき乗法で得られるベクトル列とする. もし, 初期ベクトル $\mathbf{x}^{(0)}$ が確率ベクトルならば, $\mathbf{x}^{(t)}$ もまた確率ベクトルである. 式 (17) によるベクトル列 $(\mathbf{q}^{(0)}, \mathbf{q}^{(1)}, \dots)$ に対して, $B = LP$ であるので, $\mathbf{q}^{(t)}$ と $\mathbf{x}^{(t)}$ は平行である. しかし, $\mathbf{q}^{(t)}$ は正規化されていないので, 大きさは異なる. $\mathbf{q}^{(t)}$ の各要素の和 $\sum_{i=1}^n q_i^{(t)}$ は, 以下ようになる.

$$\sum_{i=1}^n q_i^{(t)} = \sum_{i=1}^n \sum_{j=1}^n b_{ji} q_j^{(t-1)} = L \sum_{i=1}^n q_i^{(t-1)}. \quad (18)$$

$\sum_{i=1}^n q_i^{(0)} = \sum_{i=1}^n \sum_{k=1}^l q_i^{(0),k} = K$ であることと, 式 (18) から, 帰納法を用いると以下を得る.

$$\sum_{i=1}^n q_i^{(t)} = KL^{t-1} \quad (19)$$

それゆえ, $\mathbf{x}^{(t)} = \mathbf{q}^{(t)}/KL^{t-1}$ である. 補題 1 より, $\mathbf{q}^{(t)}/KL^{t-1}$ は, P の定常分布に収束する. したがって, プロトコルの出力の総和 $\pi^{(t)} = \frac{1}{KL^{t-1}} \mathbf{q}^{(t)}$ は, P の定常分布に収束する. \square

5.1.4 正規化と収束検出

暗号プリミティブの 1 つである secure function evaluation (SFE)^{3),7)} は, 複数のパーティの入力を, 他パーティに対してまったく明らかにすることなくある関数を評価できる. 我々は, 正規化と収束検出に SFE を用いる.

$\mathbf{q}^{(t)}$ の正規化はべき乗法の更新時に毎回必要になるわけではない. しかし, 暗号の演算は, N を法とする合同算術なので, $q_i^{(t)}$ が N より大きいと, 暗号化された整数の加法は正しく行われぬ. $\sum_i q_i^{(t)}$ は幾何級数的に増加するので, 正規化はオーバフローを避けるために必要である. $\mathbf{q}^{(t)}$ を安全に正規化するため, SFE による private division⁶⁾ を用いる. これは, 暗号化された整数をある特別な整数で割ることを可能にする. 典型的な暗号の設定では, $N = 2^{1024}$ であり, 正規化はそれほど多く行われぬ. 収束検出については, 閾値パラメータ θ に対して, $|q_i^{(t)} - q_i^{(t-1)}| > \theta$ が成立しているかどうかを安全に判定する必要がある. これは, 各パーティが SFE を用いて $q_i^{(t)}$ と $q_i^{(t-1)}$ を安全に比較することにより,

容易に実現できる．

5.2 セキュリティ

アルゴリズム 2 に示したプロトコルの安全性の証明の概略を示す．全パーティが semi-honest に振る舞うことを想定する．プロトコルを通して交わされるメッセージは、それぞれステップ 1, 3(b) と 4 におけるランダムシェア B , $r_{k'}^{(t-1),k}$, $s_k^{(t-1),k'}$ と $q^{(t)}$ である．もし、あるパーティ P_k がそれらすべてのメッセージを知ったとしても、 $\mathcal{P}_{\text{WARS}}$ と \mathcal{P}_{SSP} が安全で、 u パーティ以上の結託がない限りは、パーティ P_k は他パーティの保持するグラフ $G^{k'}$ ($k' \neq k$) を知ることはできない．ステップ 4 では、各パーティはランダムシェアを足し合わせるにより、 $q^{(t)}$ を得るが、この最終的な出力はプライバシーを侵害していない．それゆえ、semi-honest モデルにおける composition theorem³⁾ を用いると、SISR は定義 4 において安全であることが証明できる．

5.3 PageRank への拡張

SISR の PageRank への拡張を示す．式 (3) より、まず全パーティは行列 $D^{-1}A$ をランダムシェアにより秘密共有する、隣接行列 A の各要素は 0 か 1 なので、 $\mathcal{P}_{\text{WARS}}$ を用いるには、 $\Delta = n$ とすれば十分である．この状況は、average IPWG の特殊な場合である． $L'D^{-1}A$ のランダムシェアを計算した後、各パーティは R^k を保持している．ここで、 $\sum_{k=1}^l R^k = L'D^{-1}A$ であり、 L' は十分大きな自然数である．そこで、各パーティは $R^k = (r_{ij}^k)$ を、以下のように更新することにより、PageRank への拡張が実現される．

$$b_{ij}^k \leftarrow L \left((1-\epsilon)r_{ij}^k + \epsilon \frac{L'}{nl} \right). \quad (20)$$

6. 評価実験

プロトコルの有効性を検証するための実験を行った．プログラムは Java 1.6.0 で実装し、暗号系としては generalized Paillier 暗号系¹⁾ で 1024-bit の鍵を用いた．実験は、Xeon 5160 3.0 GHz 2 core x 2 (CPU), 32 GB (RAM) の Linux のもとで行った．

まず、我々の提案アルゴリズムが正しい結果を返すことを示すために、エンティティ数 $n = 100$ の人工グラフに対して、真の主固有ベクトル x と、パーティ数 $l = 3$ の SISR アルゴリズムによって得られる固有ベクトル $x^{(t)}$ を計算した．べき乗法の繰返し回数は $t = 40$ とした．それらのベクトルの差のユークリッドノルムは、 $\|x - x^{(t)}\| = 0.918 * 10^{-11}$ となり、実用的な範囲では正しい結果を得られるといえる．

次に、プロトコルの計算効率性を検証した．実験は実際のネットワーク上ではなく単一の

表 2 確率遷移行列のランダムシェアの計算時間 (秒)

Table 2 Computation time of random share of the probability transition matrix (sec).

	$n = 10$	$n = 33$	$n = 100$
$l = 2$	8.3	87.6	737.9
$l = 3$	12.4	129.6	1,193.0
$l = 4$	16.2	173.2	1,589.0

表 3 べき乗法における式更新の計算時間 (秒)

Table 3 Computation time of power method iteration (sec).

	$n = 10$	$n = 33$	$n = 100$
$l = 2$	2.4	7.1	23.5
$l = 3$	6.6	21.0	66.1
$l = 4$	12.8	41.6	131.6

計算機上でシミュレートされたため、通信時間は含まれていないことに注意されたい．我々の SISR アルゴリズムは、ランダムシェアとべき乗法の 2 つのステップに分解できる．ランダムシェアのステップは、主に $\mathcal{P}_{\text{WARS}}$ を使い、べき乗法のステップは、主に \mathcal{P}_{SSP} を用いる．そこで、我々の提案プロトコルである $\mathcal{P}_{\text{WARS}}$ と、 \mathcal{P}_{SSP} を実装し、ランダムに生成した人工グラフを用いて実験した．

$\mathcal{P}_{\text{WARS}}$ を用いた、確率遷移行列のランダムシェアの時間計算量は、パーティ数 l とエンティティ数 n に依存しており、IPWG の種類によって異なる．各パーティは同じ計算を実行しないため、 l パーティの計算時間の総和を計算時間と見なしている．各パーティ数 $l = 2, 3, 4$ に対する、エンティティ数 $n = 10, 33, 100$ と計算時間の関係を表 2 に示す．以上のすべての場合で、 $\Delta = n$ であり、IPWG は、additive IPWG であることを想定している．IPWG 上の PageRank で用いる IPWG は、average IPWG の特殊な場合と考えられるので、それぞれ表 2 に示した計算時間の約 2 倍の時間がかかると考えられる．

\mathcal{P}_{SSP} を用いたべき乗法における 1 回の更新の時間計算量も、パーティ数 l とエンティティ数 n に依存している．ランダムシェアの場合と条件を同じにするため、 l パーティの計算時間の総和を計算時間と見なしている．各パーティ数 $l = 2, 3, 4$ に対する、エンティティ数 $n = 10, 33, 100$ と計算時間の関係を表 3 に示す．確率遷移行列のランダムシェアの計算時間が長いように思えるが、SISR のアルゴリズム中では確率遷移行列のランダムシェアは 1 度だけ行えばよく、一方べき乗法の更新は何度が行わなければならないので、アルゴリズム全体に占める割合でいうとそれほど多いというわけではない．

7. 終わりに

本稿では、integrated private-weighted graph と呼ばれる情報分割モデルを導入し、それに対応した secure integrated link analysis を提案した．我々は、secure integrated spectral ranking の問題が、加重平均のランダムシェアの計算に変形できることを示し、準同型性公

鍵暗号を用いて, weighted average random share protocol を安全に実現した. このプロトコルは, integrated private-weighted graph の確率遷移行列のランダムシェアの計算を可能にする. べき乗法の更新を安全に実行することにより, 統合したグラフの spectral ranking を達成できる. 今後の課題として, ノードクラスタリング, リンク予測, 頻出構造の発見など, グラフマイニングにおける様々な問題を integrated private-weighted graph 上で解決することがあげられる.

参 考 文 献

- 1) Damgård, I. and Jurik, M.: A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System, *Public Key Cryptography*, pp.119–136, Springer (2001).
- 2) Goethals, B., Laur, S., Lipmaa, H. and Mielikäinen, T.: On private scalar product computation for privacy-preserving data mining, *Information Security and Cryptology-ICISC 2004*, pp.104–120 (2005).
- 3) Goldreich, O.: Foundations of Cryptography, volume 2, *Basic Applications* (2004).
- 4) Page, L., Brin, S., Motwani, R. and Winograd, T.: The pagerank citation ranking: Bringing order to the web (1998).
- 5) Sakuma, J. and Kobayashi, S.: Link analysis for private weighted graphs, *Proc. 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp.235–242, ACM (2009).
- 6) Sakuma, J., Kobayashi, S. and Wright, R.N.: Privacy-preserving reinforcement learning, *Proc. 25th international conference on Machine learning*, pp.864–871, ACM (2008).
- 7) Yao, A.C.C.: How to generate and exchange secrets, *Proc. 27th Annual Symposium on Foundations of Computer Science*, pp.162–167, IEEE (1986).

(平成 22 年 12 月 19 日受付)

(平成 23 年 4 月 6 日採録)

(担当編集委員 河野 浩之)



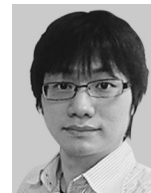
森井 正覚

東京大学大学院情報理工学系研究科修士課程在学中. 2009 年大阪大学基礎工学部情報科学科卒業. プライバシ保護データマイニングの研究に従事.



佐久間 淳

1997 年東京工業大学生命理工学部生物工学科卒業, 2003 年 3 月同大学大学院総合理工学研究科知能システム科学専攻博士後期課程修了. 博士(工学). 同年 4 月日本アイ・ピー・エム株式会社入社, 東京基礎研究所に配属. 2004 年 7 月東京工業大学総合理工学研究科助手, 2007 年 4 月同助教, 2009 年 4 月筑波大学システム情報工学研究科コンピュータサイエンス専攻准教授, 2009 年 10 月科学技術振興事業団さきがけ研究員兼任, 現在に至る. 機械学習と知識発見, セキュリティとプライバシーの研究に従事.



佐藤 一誠 (学生会員)

東京大学大学院情報理工学系研究科博士課程に在籍. 統計的機械学習, 特にベイズ学習の研究に従事.



中川 裕志 (正会員)

1975 年東京大学工学部卒業, 1980 年同大学大学院博士課程修了. 工学博士. 同年より横浜国立大学勤務. 1999 年より東京大学情報基盤センター教授. 統計的機械学習, 情報検索, 自然言語処理の研究に従事.