ラベル付きグラフに対するプライバシ保護半教師付き学習法

荒井　ひろみ †　　佐久間　淳 †‡

† 筑波大学大学院システム情報工学研究科
305–8573 茨城県つくば市天王台 1–1–1
arai.hiromi.ga@u.tsukuba.ac.jp,
jun@cs.tsukuba.ac.jp

‡ 科学技術振興機構
102–8666 東京都千代田区四番町 5–3

あらまし 本研究では秘密情報を含むグラフ構造データにおいて，秘密を保護してノードのラベルを予測するプロトコルを提案する．実社会におけるグラフ構造データ，例えばノードが個人，リンクが接触，ノードのラベルが感染状態である感染症ネットワークを考える．ラベル予測のための従来の半教師付き学習方法はリンクやラベルの公開を前提としている．本方式は加法準同型性公開鍵暗号を用いたマルチパーティーコンピュテーションによって半教師付き学習の一方式であるラベル伝播法を実行する．ラベルとリンクが各ノードの秘密情報である場合，本方式を用いるとノードの秘密を保護しラベル予測を実現できる．

Privacy-preserving semi-supervised learning for labeled graphs

Hiromi Arai†　　Jun Sakuma†‡

†Department of Computer Science, University of Tsukuba
1–1–1 Tenoudai, Tsukuba, Ibaraki, 305–8573, JAPAN
arai.hiromi.ga@u.tsukuba.ac.jp,jun@cs.tsukuba.ac.jp

‡Japan Science and Technology Agency
5–3, Yonban-cho, Chiyoda-ku, Tokyo, 102–8666, JAPAN

Abstract We propose a privacy-preserving protocol that predicts node labels for graphs including sensitive information. In the real world graph structured data, such as disease infection (node labels) over individuals (nodes) through human contact (links). Traditional semi-supervised learning methods for label prediction assume that all links and labels are disclosed. Our method execute label propagation, one of the major semi-supervised learning methods, by multi party computation using additive homomorphic public-key encryption scheme Our method predicts node labels without disclosing links and labels of nodes.

1　はじめに

グラフ構造データにおいて一部のノードがラベル付けされている場合に，そのラベルを教師として用い学習を行うことにより残りのノードラベルを予測する問題を考える．半教師付き学習はこのような場合にノードラベルの情報に加え，全ノードのなすグラフ構造の情報を用いて学習を行う方法である．半教師付き学習は，データのグラフ構造は比較的手に入りやすいがノードのラベル付けのコストが高い場合に用いられる．例えばタンパク質のクラス分類 [1] におけるタンパク質をノード，機能をラベル，類似度をリンクとした予測がある．本研究では，実社

会における個人や企業の活動のなすグラフ構造データへの応用を見込み，ラベルやリンクの情報がデータ保有者のプライバシである場合にラベル予測を安全に実現する方法を提案する．

秘密情報を含むラベル付きグラフの例として人間間の接触ネットワークにおける性感染症の感染状態を考える．これは人間をノード，接触をリンクの有無，接触回数をリンクの重みとして記述できる．一部の人間の感染状態しかわからない場合，半教師付き学習を行えば全員の感染状態を予測できると期待される．しかし，感染状態及び接触相手はどちらも重要な個人情報であり，他人への開示は現実的でない．

本研究ではラベルの付いた無向グラフにおける典型的なプライバシモデルを定義し，モデル上で安全に実行可能なラベル予測法を提案する．さらに，提案法の有効性を人工データと性的接触ネットワークの実データを用いて検証した．

2 問題定義

本章では行列を用いたグラフ構造データの表現を与え，それらを用いてグラフのプライバシの定義及び問題の定式化を行う．

$G = (V, E)$ をノード集合 $V = \{1, \ldots, n\}$，リンク集合 $E = \{e_{ij}\}$ からなるグラフとする．なお，本研究では無向グラフを取り扱う．リンク e_{ij} には重み w_{ij} が付き，$e_{ij} \notin E$ のときは $w_{ij} = 0$ とする．ノード i のリンク先のノード集合を $N(i) = \{j | j \in V, e_{ij} \in E\}$ とする．

ノードラベルは集合 $\{1, \ldots, h\}$ によって記述する．ラベル行列 $\mathbf{F} = (f_{ik}) \in \mathbb{R}^{n \times h}$ を導入し各行はノードを，各列はラベルのクラスに与えられるスコアを示すとする．ノード i のラベルは $\mathbf{F}$ の i 行目の最大スコアを持つ列 $s = \arg\max_{1 \le k \le h} f_{ik}$ として示す．

行列表記したグラフのプライバシの議論のため，n ノードの集合 V とある行列 $\mathbf{M} \in \mathbb{R}^{n \times r}$ に対して，情報分割モデル row private，symmetrically private $N(i)$-row private を定義する．

定義 1. *(row private)* $n \times n$ 行列 $\mathbf{M}$ において，すべての i について i 番目のノードが行列 $\mathbf{M}$ の第 i 行 $\mathbf{m}_{i*}$ のみを知っているが，他の行 $\mathbf{m}_{p*}(p \neq i)$ を知ることができない場合，行列 $\mathbf{M}$ は *row private* である．

定義 2. *(symmetrically private)* $n \times n$ 行列 $\mathbf{M}$ において，すべての i について i 番目のノードが行列 $\mathbf{M}$ の第 i 行 $\mathbf{m}_{i*}$ 及びの第 i 列 $\mathbf{m}_{*i}$ のみを知っており，他の要素 $\mathbf{m}_{pq}(p \neq i$ かつ $q \neq i)$ を知ることができない場合，行列 $\mathbf{M}$ は *symmetrically private* である．

上述の行列の情報分割モデルを用い，秘密情報である重み行列 $\mathbf{W}$ とラベル行列 $\mathbf{F}$ に関するグラフプライバシモデルを定義する．

定義 3. *(label-unaware PWG)* グラフ $G = (V, E)$ において，重み行列 $\mathbf{W}$ が *symmetrically private* 及びラベル行列 $\mathbf{F}$ が *row private* ならば，G は *label-unaware private weighted graph* である．

$\mathbf{W}$ が row private とは，ノード i がリンク先 $N(i)$ 及びリンク重みのみを知る場合である．$\mathbf{F}$ が $N(i)$-row private となるのは，ノード i が i 及びリンク先 $j \in N(i)$ のノードラベルを示す $\mathbf{F}$ の行 $\mathbf{f}_{j*}$ のみを知る場合である．$\mathbf{F}$ が row private のとき，ノード i が自分のラベルを示す $\mathbf{F}$ の行 $\mathbf{f}_{i*}$ のみを知る．

label-unaware PWG は各ノードが自分の持つリンクやラベルの情報をいかなる相手に対しても秘匿する状況を表している．これはリンク及びラベルが非常に秘匿性の高い個人情報である場合に対応する．例えば人間間の性的接触ネットワークにおける性感染症の感染状態の情報が当てはまる．このような状況では人間 (ノード) はその接触相手 (リンク先) 以外にはその接触関係 (リンク) を知られたくない，また人間はその性感染症の感染状態 (ラベル) は自分の接触相手を含む他のいかなる人間に対しても開示しないと考えられる．

さらに，label-unaware PWG においてノードがプライバシを犠牲にしてラベルの情報をそのリンク先ノードに開示した場合を label-aware PWG として定義する label-unaware PWG は各ノードが自分の持つリンクをいかなる相手対しても秘匿し，ラベルはリンク先にのみ開示する状況を表している．これはリンクが非常に秘匿性が高く，ラベルの秘匿性も高い場合に対応

する．例えば人間間の近接ネットワークにおける風邪の感染状態の情報が当てはまる．このような状況では人間 (ノード) はその接触相手 (リンク先) 以外にはその近接関係 (リンク) を知られなくない．また風邪の感染状態 (ラベル) は自分に近接する人には知られてもよいが広く公共には公開したくないと考えられる．

label-unaware PWG 上での安全なラベル予測法を secure label prediction として定義し，これを実現するプロトコルを 4 章で，基礎となるアルゴリズムを 3 章で述べる．また，プライバシの制約を緩め label-aware PWG 上を許容する場合の議論を 5 章で行う．

3 ラベル伝播法

グラフ構造データに用いられる半教師付き学習法の一つであるラベル伝播法 [2] のプライバシモデル上での安全性を議論する．

ラベル伝播法は初期ラベル行列と重み行列 $\mathbf{W}$ を入力とする．G 上の確率行列を $\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}$ とする．度数行列 $\mathbf{D}$ は $\mathbf{D} = \mathrm{diag}(d_1, \ldots, d_n)$, $d_i = \sum_{j \in V} w_{ij}$ として $\mathbf{W}$ から求められる．初期ラベル行列を $\mathbf{Y} = (y_{ij}) \in \mathbb{R}^{n \times c}$，ノード i のラベルが p ならば $y_{ip} = 1$，それ以外では $y_{ip} = 0$ とする．

ラベル伝播法では確率行列に従ってラベル情報を伝播させる以下の更新式を用いる．

$$\mathbf{F}^{(t+1)} = \alpha \mathbf{P}\mathbf{F}^{(t)} + (1-\alpha)\mathbf{Y}. \qquad (1)$$

ここで $0 \leq \alpha < 1$，第一項はラベル伝播，第二項は初期ラベルの影響をそれぞれ示す．この更新式は収束し，収束値を $\mathbf{F}^*$ としてラベル予測に用いる．

式 (1) を分散化して，ラベル伝播法のセキュリティを考察する．$\mathbf{F}$ の要素に着目すると，要素 f_{ik} の更新式は

$$f_{ik}^{(t)} \leftarrow \alpha\Big(\sum_{j \in N(i)} p_{ij} f_{jk}^{(t-1)} \Big) + (1-\alpha) y_{ik} \qquad (2)$$

となる．この更新式が label-unaware PWG 上で安全に実行できるか検討する．ノード i が式 (2) を更新するには，p_{ij}，$f_{jk}(j \in N(i))$，y_{ik} を必要とする．$p_{ij} = w_{ij} / \sum_{j \in N(i)} w_{ij}$ であり，$\mathbf{W}$ は row private であるためノード i は p_{ij} を計算できる．また $\mathbf{Y}$ が row private であるためノード i にとって y_{ik} は既知である．ラベル行列に関しては $f_{jk}^{(t-1)}(j \in N(i))$ はノード i がそのリンク先 $j \in N(i)$ から取得する必要がある．そのため更新式 (2) は label-aware PWG 上では実行できる．しかし label-unaware PWG 上では $\mathbf{F}$ は row private であるため，安全に実行できない．label-unaware PWG 上で安全にラベル予測を実現するには，ノード i は $f_{jk}(\ j \in N(i))$ を観測せずに式 (2) を評価する必要がある．これはのちに示すように，準同型性公開鍵暗号を導入することにより達成される．

4 プライバシ保護ラベル伝播法

分散ラベル伝播法に準同型性公開鍵暗号系を用いることによって，label-unaware undirected PWG 上で安全に実行できるプライバシ保護ラベル伝播法 (PPLP) を提案する．以下にプロトコルに用いた暗号系と提案法を示す．

4.1 加法準同型性公開鍵暗号

公開鍵暗号系において，公開鍵 pk，対応する秘密鍵 sk とする．平文 $m \in \mathbb{Z}_N$(N はセキュリティパラメータ) について，$c = \mathrm{Enc}_{\mathsf{pk}}(m; l)$ は m の確率暗号による暗号化を，$m = \mathrm{Dec}_{\mathsf{sk}}(\mathrm{c})$ はその復号を表す．加法準同型性公開鍵暗号では下式が成立する．

$$\mathrm{Enc}_{\mathsf{pk}}(m_1 + m_2; l) = \mathrm{Enc}_{\mathsf{pk}}(m_1; l_1) \cdot \mathrm{Enc}_{\mathsf{pk}}(m_2; l_2)$$

$$\mathrm{Enc}_{\mathrm{pk}}(km; l) = \prod_{i=1}^{k} \mathrm{Enc}_{\mathsf{pk}}(m; l_i) = \mathrm{Enc}_{\mathsf{pk}}(m)^k$$

なお，準同型性公開鍵暗号の演算は法 N 上で行われる．以降乱数 l は略記する．

(m, θ)-閾値暗号系では n ノードが共通の公開鍵 pk を保持し，各ノードはそれぞれ異なる秘密鍵 $sk^1, \ldots, sk^n$ を保持する．暗号の復号には少なくとも θ 個以上のノードが協力して recovery アルゴリズムを実行する必要がある．

Procedure of PPLP

- Public input: α and $L \in \mathbb{Z}_N$ s.t $\alpha L p_{ij} \in \mathbb{Z}_N$ and $(1-\alpha)L y_{ik} \in \mathbb{Z}_N$ for all i, j, k.
- Private input of node i: link weights $\mathbf{w}_{i*}$ and label vector $\mathbf{y}_{i*}$
- Key setup: All nodes share public key pk; node i holds secret key sk^i for threshold decryption.

1. (Initialization) For all $j \in N(i)$, node i computes:
   (a) $p_{ij} \leftarrow w_{ij} / \sum_{j \in N(i)} w_{ij}$, $\tilde{p}_{ij} \leftarrow \alpha L p_{ij}$,
   (b) For all $k \in \{1, ..., h\}$, $\tilde{f}_{ik}^{(0)} \leftarrow y_{ik}$, $\tilde{y}_{ik} \leftarrow (1-\alpha) L y_{ik}$,
   (c) $c_{ik}^{(0)} \leftarrow \mathrm{Enc}_{\mathsf{pk}}(\tilde{f}_{ik}^{(0)})$ for all $k \in \{1, ..., h\}$ and $t \leftarrow 1$.
2. (Iteration) For all $k \in \{1, ..., h\}$:
   (a) Node i sends $c_{ik}^{(t-1)}$ to all $j \in N(i)$,
   (b) Node i receives $c_{jk}^{(t-1)}$ from all $j \in N(i)$ and updates $c_{ik}^{(t-1)}$ by eq. 3,
   (c) All nodes jointly perform the convergence detection and normalization if needed. If convergence is detected, go to step 3. Else, $t \leftarrow t+1$ and go to step 2 (a).
3. (Decryption) Node i and arbitrary $(\theta - 1)$ nodes jointly perform recovery scheme and output $\mathbf{f}_{i*}^* = (f_{i1}^*, \ldots, f_{ih}^*)$.

図 1: Privacy-preserving label propagation

4.2 提案プロトコル

提案法を図 1 に，詳細を以下に示す．

Setup. 全ノードが協力して pk を公開し，sk^i がそれぞれノード i のみが所有するように鍵を生成する．鍵のセットは分散鍵生成法 [5] によって安全に作成される．また全てのノードが協調して更新式を実行できるように global clock を用意する．ノード i は固有の入力として $\mathbf{w}_{i*}$ and $\mathbf{y}_{i*}$ を持つ．また，$\mathbf{w}_{i*}$ を用いて p_{ij} を計算する (Step 1 (a))．また準同型性公開鍵暗号は平文として整数値のみを取るため，十分大きな定数 L を用いて $\tilde{y}_{ik}, \tilde{p}_{ij} \in \mathbb{Z}_N$ となるように $\tilde{y}_{ik} \leftarrow (1-\alpha) L y_{ik}, \tilde{p}_{ij} \leftarrow \alpha L p_{ij}$ とする (Step 1 (b))．Step 1 (c) でこれらの暗号化を行う．

Iteration and convergence. Step 2 でラベル行列の各行 f_{i*} の更新を行う．Step 2 (a) でノード i は暗号化したラベル行列の要素 $c_{ik}^{(t-1)}$ をリンク先のノード $j \in N(i)$ に送信，リンク先のノードはこれを受け取る．ラベル要素は次式によって更新される．

$$\mathrm{Enc}_{\mathsf{pk}}(\tilde{f}_{ik}^{(t)}) \leftarrow \prod_{j \in N_{(i)}} \left((c_{jk}^{(t-1)})^{\tilde{p}_{ij}} \right) \cdot \mathrm{Enc}_{\mathsf{pk}} \left(L^{t-1} \tilde{y}_{ik} \right). \tag{3}$$

$\tilde{\mathbf{F}}^{(t)} = (\tilde{f}_{ij}^{(t)})$ とおく．式 (3) の収束を次に示す．

補題 1. *$L\mathbf{F}^* = (1-\alpha)(\mathbf{I} - \alpha\mathbf{P})^{-1}\mathbf{Y}$ とする．$\tilde{\mathbf{F}}^{(0)}$ が式 (3) によって更新されるとき，$\lim_{t\to\infty} \tilde{\mathbf{F}}^{(t)} / L^t = \mathbf{F}^*$ が成り立つ．*

Proof. 暗号の準同型性より式 (3) は以下のように変形される．

$$\mathrm{Enc}_{\mathsf{pk}}(\tilde{f}_{ik}^{(t)}) \leftarrow \mathrm{Enc}_{\mathsf{pk}} \Big(\sum_{j \in N(i)} \tilde{p}_{ij} \tilde{f}_{jk}^{(t-1)} + L^{t-1} \tilde{y}_{ik} \Big).$$

$$= \mathrm{Enc}_{\mathsf{pk}} \Big(L \big(\sum_{j \in N(i)} \alpha p_{ij} \tilde{f}_{jk}^{(t-1)} + L^{t-1}(1-\alpha) y_{ik} \big) \Big).$$

$\tilde{f}_{ik}^{(t)} = L^t f_{ik}^{(t)}$ の成立は帰納的に求められる．$t = 1$ のとき $\tilde{f}_{ik}^{(1)} = L f_{ik}^{(1)}$ である．$\tilde{f}_{ik}^{(u)} = L^u f_{ik}^{(u)}$ がどのような $u \in \mathbb{Z}$ についても成り立つとする．上式より $\tilde{f}_{ik}^{(u+1)} = L^{u+1} f_{ik}^{(u+1)}$ が成り立つ．以上により，$\tilde{\mathbf{F}}^{(t)} / L^t = \mathbf{F}^{(t)}$ が成立し補題 1 が証明される． □

なお，Step 2 (c) において private division [6] を用いて更新式の収束判定及び正規化を行う．これは平文の定数での除算をランダムシェアと Secure function evaluation (SFE) [7] を用いて安全に行うアルゴリズムである．通常 SFE の計算時間は大きいが，どちらの計算も更新式数回に一度実行すればよいため，プロトコル実現の大きな障害にはならない．

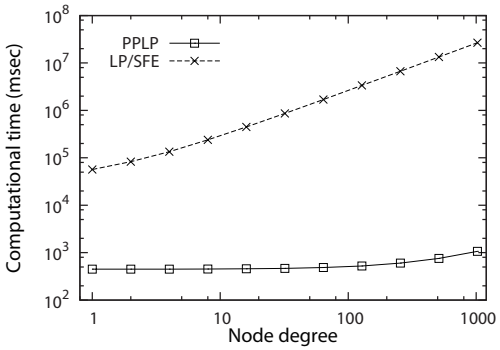

図 2: 人工データに対するノード次数 Δ に対する更新式の計算時間の変化

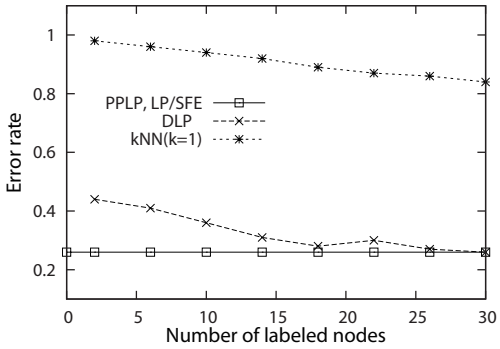

図 3: ROMN に対する開示ラベル数 ℓ に対する予測精度の変化

提案プロトコルのセキュリティは，全てのノードが *semi-honest* であるとすると，以下のように示される (証明略).

補題 2. 全てのノードが *semi-honestly* に振舞う *label-unaware undirected PWG* があるとする．θ 個以上のノードが結託しない限り，*PPLP* の実行後ノード i は $\mathbf{F}^*$ の第 i 行と更新式の回数しか知ることができずグラフは *label-unaware undirected PWG* に保たれる．

上記補題より以下の定理がただちに導かれる．

定理 1. 全てのノードが *semi-honestly* にふるまう場合，*PPLP* は安全にかつ正しく *label-unaware undirected PWG* 上でラベル伝播法を実行する．

5 計算機実験

PPLP の有効性をスケーラビリティ，予測精度と開示情報の関係，及び計算時間の観点から検証した．スケーラビリティには人工データを，その他の 2 つには実データから作成したグラフ構造データを用いた．

実験には PPLP，及び比較のため k 近傍法，式 (3) による分散化したラベル伝播法 (DLP)，SFE を用いたラベル伝播法 (LP/SFE) を用いた．PPLP，k 近傍法，DLP は Java1.6.0 で実装した．暗号系は generalized Paillier 暗号系 [4] を用い，セキュリティパラメータは $N = 2^{1024}$ に設定した．収束判定と正規化はそれぞれ更新式 10 回，50 回ごとに 1 回行った．LP/SFE の実装には Fairplay [9] を用いた．SFE は任意の関数計算を安全に実行できるが，LP/SFE を完全に実装するのは計算量の観点から現実的に困難である．そのため LP/SFE はラベル伝播法の個々の加算，乗算それぞれに対し SFE で計算を行った時間を下限として扱った．

実験には 2.80GHz (CPU), 2GB (RAM) の Linux を用いた．実験は実際のネットワーク上ではなく，単一の計算機上でシミュレーションを行った．また，通信時間は含まれていない．

scalability analysis. PPLP のスケーラビリティは式 (3) に示されるように，ノードの次数に依存する．そのため，グラフ構造データの最大次数を Δ とし，PPLP の更新式 (3) の Δ に対する計算量の変化を計算機実験により調べた．計算時間には収束判定と正規化のための時間も含めた．比較のために同じ条件での LP/SFE の計算時間も調べた．

結果は図 2 に示されるようになり，$\Delta = 2^{10}$ の場合でも PPLP は数秒で処理が行われた．一方 LP/SFE は $\Delta = 2^{10}$ では一更新に数時間を要した．実験を行った範囲では PPLP の方が所用時間が短く，大きなグラフ構造データにも現実的な計算時間で対応できると期待される．

privacy-accuracy trade-off. 実験のためのグラフ構造データ作成には実際の性的接触ネットワークの実データである Romantic Network [8] を用いた．Romantic Network の最大成分 (288 ノード) を用い，感染者情報を人工的に与えた．ランダムに選択した 20 ノード及びそこから 5step 以内の人を感染者として扱い，80 ノードを感染

Method	disclosed info.	comp. time
kNN	y_{jk} $(j \in N(i))$	7.4×10^{-5}(ms)
DLP	$f_{jk}^{(t)}$ $(j \in N(i))$	3.0×10^{-4}(ms)
LP/SFE	none	>88.4 (min.)
PPLP	none	10.7 (sec.)

表 1: 各手法のノードの開示情報と ROMN に対する計算時間

者，208 ノードを非感染者はとした (ROMN). 実験においては感染者，非感染者からそれぞれ選択した 15 ノード，計 30 ノードのラベルが初期ラベルとして与えられるとし，残りのラベルの予測を行った.

ノードラベルのプライバシ損失と予測精度の関係について検証する．label-unaware PWG において初期ラベルのついたノードのうちいくつかがプライバシの損失を許容してラベルを開示する状況を考える．このような状況では各ノードは自分のはるリンク及びリンク先のうち公開されているラベルの情報を持つ．よって開示されたラベルを用いて k 近傍法の実行が可能である．また，3 章で検討したように，開示されたラベルを用いて DLP を実行することができる．なお DLP では更新式毎に計算の途中経過もノード間で共有されることに注意が必要である.

図 3 に開示ノード数を変化させたときの各手法の予測率の変化を ROMN データセットを用いた計算機実験で検証した結果を示す．k 近傍法と DLP では開示ラベル数が少ないほど誤予測率が高くなった．これよりノードラベルのプライバシ損失と予測精度はトレードオフの関係にあることがわかる．一方，PPLP を用いると,, 開示ラベル数に関係なく，全ての初期ラベルを用いた場合と同等の予測結果を得ることができる.

completion time in ROMN. 全てのノードラベルが開示された場合において ROMN に対するプロトコルの開始から終了までにかかる計算時間を検証した．実験結果をノードの開示情報を合わせて表 1 に示す．この結果より，PPLP のみが情報秘匿と現実的な計算量両方を達成できることがわかる.

6 終わりに

本論文ではラベル付きグラフ構造データのプライバシモデルとして label-unaware private weighted graph を導入し，その上でのでラベル予測の実行を定式化した．さらに label-unaware private weighted graph 上で安全かつ正しくラベル予測を実現する PPLP プロトコルを提案した．今後の課題として，このようなラベル予測法の出力から sensitive な情報を守るために Output Privacy の問題を解決する必要がある.

参考文献

[1] J. Weston, C. Leslie, E. Ie, D. Zhou, A. Elisseeff, and W.S. Noble. Semi-supervised protein classification using cluster kernels. *Bioinformatics*, 21(15):3241, 2005.

[2] X. Zhu, Z. Ghahramani, and J. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions, 2003.

[3] O. Goldreich. *Foundations of cryptography: Basic applications*. Cambridge University Press, 2004.

[4] I. Dåmgard and M. Jurik. A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System. In *Public Key Crypt.* Springer, 2001.

[5] I. Dåmgard and M. Koprowski. Practical threshold RSA signatures without a trusted dealer. *Advances in Cryptology-EUROCRYPT 2001*, pages 152–165, 2001.

[6] J. Sakuma, S. Kobayashi, and R.N. Wright. Privacy-preserving reinforcement learning. In *Proceedings of the 25th international conference on Machine learning*, pages 864–871. ACM, 2008.

[7] A.C.C. Yao. How to generate and exchange secrets. In *Proc. of the 27th IEEE Annual Symposium on Foundations of Computer Science*, pages 162–167, 1986.

[8] P.S. Bearman, J. Moody, and K. Stovel. Chains of affection: The structure of adolescent romantic and sexual networks. *American J. of Sociology*, 110(1):44–91, 2004.

[9] D. Malkhi, N. Nisan, B. Pinkas, and Y. Sella. Fairplay: secure two-party computation system. In *Proc. of the 13th USENIX Security Symposium*, pages 287–302, 2004.