1C1-1

# プライバシアウェアなPeer-to-Peer k-meansクラスタリング

## Privacy-aware k-means Clustering in Peer-to-Peer Network

佐久間 淳*1　　小林 重信*1
Jun Sakuma　　Shigenobu Kobayashi

*1東京工業大学 大学院総合理工学研究科
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology

A new privacy-preserving k-means clustering is presented and described. Conventionally, organizations that collect a large amount of personal information are assumed as participants of Privacy-Preserving Data Mining (PPDM). As a contrasting privacy-preserving concept, we propose User-Centric Privacy Preservation where users store their personal information in their local storage and manage their data by themselves. In this framework, because the database is partitioned into a small fraction, the number of parties tends to be large. As a data-mining algorithm with this new privacy preservation manner, we propose a novel and scalable protocol for $k$-means clustering. In large-scale networks, we must resolve two difficulties: (1) the global synchronization over the entire network is not guaranteed; and (2) communication between participants might be terminated suddenly. Considering them, we propose two novel protocols: private asynchronous average computation and private Euclidean distance comparison. Based on them, a k-means clustering with user-centric privacy preservation is designed. Experimental results show that our protocol is scalable and works asynchronously, even with 1000 parties.

## 1. はじめに

センサーネットワークやプローブカー, ユーザーのローカルストレージなど, 自律的に動作する情報源により構成されるPeer-to-Peer(P2P) ネットワークにおけるデータマイニングが注目されている. P2P ネットワークにおけるプロトコルは, 非中央化・非同期化が容易で, 大規模なネットワークにおいてスケールしやすいという利点があるが, P2P では各ノードがデータの流通を中継するという性質上, 機密情報や個人情報の扱いが困難である点が問題となる.

Privacy Preserving Data Mining (PPDM) は, データが複数サイトに分散している状況において, プライバシーを保護したままデータマイニングの出力のみを共有するための技術である. PPDM では, 必要な計算が可能なようにデータを暗号化や乱数による摂動によりマスクする. プロトコルはそのマスクを解除せずにデータマイニングを実行し, 終了後マスクを解除し必要なデータマイニング結果のみを得る. PPDM に拡張されたデータマイニングの例として, 相関ルール抽出 [1], $k$-means クラスタリング [2], 決定木学習 [3] などが知られる.

上記に示した PPDM は 2 パーティー間での実行を想定した設計がなされ, 参加パーティー数が多い場合には計算量爆発を起こす. 本稿では, 構成ノード数が $10^3$ 程度の Peer-to-Peer(P2P) ネットワークを対象とし, P2P ネットワーク上のプロトコルのスケーラビリティーを損なうことなく, PPDM の技術を活用し, プライバシを保護した $k$-means クラスタリングのためのプロトコルを提案する.

プライバシを保護したクラスタリングの応用例の一つとして, 疫病発生源の特定が挙げられる. 病院が患者の医療履歴を, 各患者が車による移動履歴を保持しているとしよう. 疫病発生時において, 患者の医療履歴および移動履歴をデータとしたクラスタリング結果は, 疫病発生源の予測に有効であるが, 両データともプライバシーの観点から扱いがセンシティブであり, 安易な統合はプライバシーの漏洩を招く. またこの場合, 参加パーティー数は患者数のオーダーであり, 既存の 2 パーティープロトコルにおいては実現が困難である. 一方, 提案プロトコルによれば, 病院および患者は, それぞれプライベートなデータを開示することなく, クラスタリング結果のみを共有することが可能である.

提案プロトコルでは, プライバシ保護と計算効率性を両立させるために, プライベートな非同期平均計算プロトコル (private AAC) [4] とプライベートな距離比較 (private EDC)[5] を用いる. private AAC はプライベートかつ非同期的に, P2P 上の各ノードが持つ値の暗号化された平均値を求めるプロトコルである. private EDC はノードがベクター $x$ を所持し, 二つの暗号化された参照点が与えられたときに, $x$ からどちらの参照点への距離が近いかをプライベートに判定するプロトコルである. 提案法はノードの停止や通信途絶が発生してもプロセス全体への影響がないという意味で非常にロバストであり, また非同期性も高い. プロトコルの計算効率性は数値例に基づき評価される.

連絡先: 佐久間 淳, 東京工業大学 大学院総合理工学研究科, 横浜市緑区長津田町 4259, 045-924-5677, jun@fe.dis.titech.ac.jp

## 2. 研究の背景

### 2.1 k-means クラスタリング

データ集合 $X = \{x_1, ..., x_n\}$ があるとしよう.$k$-means クラスタリングは類似度の高いデータ同士を $k$ 個のクラスタに分類するクラスタラベルを決定するアルゴリズムである. クラスタラベル $z_{i,j}$ は, データ $x_j$ がクラスタ $C_i$ に属するならば $z_{i,j} = 1$, そうでないなら $z_{i,j} = 0$ と定義される. $k$-means は以下の二つのステップからの繰り返し構成される.

1. データ $x_j$ の重みつき平均 $\mu_i = \frac{\sum_j z_{ij} x_j}{\sum_j z_{ij}} (i = 1, ..., k)$ の計算
2. データ $x_j$ に最も近い $\mu_i (i = 1, ..., k)$(クラスタラベル $z_{ij}$) の決定

step 1 はクラスタ中心の計算であり, step 2 はクラスタラベルの割り当ての決定である.

### 2.2 プライバシを保護した k-means クラスタリング

データ集合 $X$ が複数のパーティーに分割されて保持され, それぞれが明かすことのできないプライベートなデータであるときに, それを開示することなく, $k$-mean クラスタリング

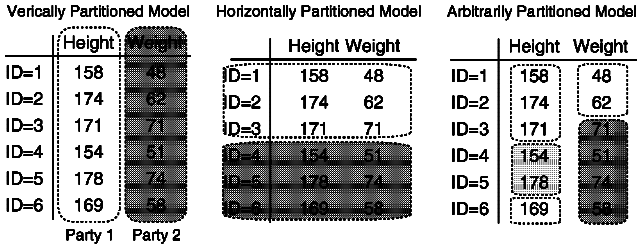


図 1: データ分割モデル

の結果のみを共有するための Privacy-Preserving $k$-means クラスタリング手法が提案されている．データプライバシーはデータ分割モデルに基づき定義される．Vertically Partitioned Model (VPM) は, 各パーティーが全データにおけるある属性のサブセットを保持するモデルである (図 1 左)．Horizontal Partitioned Model (HPM) は, 各パーティーが全ての属性におけるあるデータのサブセットを保持するモデルである (図 1 中)．Arbitrarily Partitioned Model (APM) は両モデルの混合であり, パーティー間のデータの共有は単純なパターンで説明されない (図 1 右)．定義の詳細は [6] や [7] を参照されたい．

Vaidya らは VPM におけるプライバシを保護した $k$-means を提案した [2]．VPM では各ノードは自分が保持する属性についてそのクラスタ中心をローカルに計算することができる．最近接のクラスタ中心は, クラスタ中心とデータ点の間の距離のランダムシェアを順列プロトコル [8] により求め, ランダムシェアを保持する 2 パーティー間の Yao's secure circuit により比較演算を実行することによって決定される．

Jagannathan らは APM において動作する 2 パーティー $k$-means を提案した [7]．APM の場合クラスタ中心はローカルに計算できないため, 二つのパーティーはクラスタ中心への距離のランダムシェアを内積プロトコル [9] より計算し, Yao's secure circuit により最近接のクラスタ中心が決定される．

両アルゴリズムとも分散環境においてセキュアに動作するように設計されているが, Vaidya らの方法は VPM に特化して設計され, 拡張はできない．また Jagannathan らの方法は任意のデータ分割モデルに対応するが, パーティー数 $m$ について通信計算量が $O(m^2)$ になり, P2P 上での利用は非現実的である．またどちらのプロトコルも同期的動作を要求するステップを多数含み, 大規模な P2P ネットワークにおける運用は困難である．

### 2.3 準同型性暗号

プライバシを保護したデータマイニングでは, 他パーティーに自分のデータを明かさずにデータマイニングを実行するために, 必要な計算 (加算と積算) が, 暗文同士において, それらを解読することなく実行可能な性質をもつ準同型性公開鍵暗号系 (Homomprphic Cryptosystem) が一般に用いられる．例えば Paillier 暗号系 [10] は準同型性を有し，強秘匿である．公開鍵暗号系は，それぞれ鍵生成，暗号化，復号化のための確率的多項式時間アルゴリズムの組 $(Gen, Enc, Dec)$ からなる．鍵生成アルゴリズムは有効な秘密鍵と公開鍵の組 $(s_k, p_k)$ を生成する．暗号系が加法的準同型性であるならば，公開鍵と平文 $m_1$ , $m_2 \in Z_m$ について，以下が成立する．

$$Enc(m_1; r_1) \cdot Enc(m_2; r_2) = Enc(m_1 + m_2; r_1 + r_2),$$
$$Enc(m_1; r_1)^{m_2} = Enc(m_1 m_2; r_1).$$

ここで $r_1, r_2$ は乱数である．

## 3. P2P ネットワークにおけるプライバシを保護した k-means クラスタリング

### 3.1 問題の定義

本稿で対象とする P2P ネットワークは, 任意のノード間およびノード・サーバ間の通信およびブロードキャストが可能である．ネットワークは $n$ 個のノード $P_j (j = 1, ..., n)$ と一つのサーバ $S$ から構成される．データは $d$ 個の整数値で表現される属性からなる $x \in Z_m^d$ とする[*1]．またノード $P_j$ はある一つのデータ $x_j$ において全ての属性を保持する．つまりデータ分割モデルとして HPM を想定する．サーバはデータを保持せず, 参加ノードのアドレス管理や公開鍵の発行などを行う．

データマイニングにおけるプライバシーは, 各パーティーが, 他のパーティーのデータについて, 出力 (データマイニング結果) とその出力より推測される事実を除き, 何の知識も得ることができないことを意味する．$k$-means の場合, 各ノードの入力はデータ $x_j$, 出力はそのノードが保持するデータにおけるクラスタラベル $z_{ij}$ である．

**Definition 1.** プライベートな $k$-means クラスタリング: データ $x_j$ を保持するノード $P_j (j = 1, ..., n)$ とサーバ $S$ からなる P2P ネットワークにおいて, プロトコル実行後, 各ノードは, 各データに対応するクラスタラベル $z_j$ を得るがそれ以外は何も得ない．サーバは何も得ない．

前節に示したように, $k$-means は, クラスタ中心の計算と, 最近接クラスタ中心の決定の二つのステップから構成されている．そこで本論文では, "プライベートな重みつき平均計算" と "プライベートな再近接点の決定" を実現する二つのプロトコルの交互の実行により $k$-means を構成する．サーバは秘密鍵と公開鍵を作成し, 公開鍵をノードにブロードキャストする．両プロトコルにおいて, ノードでの多くの計算は暗文とノード自身が持つデータを用いた演算である．以下, この二つのプロトコルを説明する．

### 3.2 プライベートな平均計算

P2P ネットワークにおける, 効率的な統計計算のためのプロトコルとして, Gosship-based プロトコル (疫学的 (epidemic) プロトコルとも呼ばれる) が知られている．Gosship-based プロトコルは単純性とスケーラビリティーを維持しつつ, 非同期かつ非停止に P2P ネットワークから情報を収集するアプローチである [11]．Kowalczyk らは, P2P 上の各ノードに分散した値について, それらを中央サーバに集約することなく, その平均 $\mu = \frac{\sum_j x_j}{n}$ を計算する "newscast" と呼ばれるプロトコルを提案した [12]．ノード $P_j$ は自ノードのデータで初期化された局所的推定値 $\mu_j$ を保持しているとしよう．newscast では, ランダムに選択された 2 ノード $P_j, P_{j'}$ において, 以下の更新式を非同期的に実行する．

$$\mu_j \leftarrow \frac{\mu_j + \mu_{j'}}{2}, \mu_{j'} \leftarrow \frac{\mu_j + \mu_{j'}}{2}. \quad (1)$$

この更新式において, 各ノードのメッセージ交換回数 $t$ を**サイクル**と呼ぶ．newscast プロトコルの非同期的実行の後, 各ノードの局所推定 $\mu_j$ は, サイクル $t$ の極限 $t \to \infty$ において $\mu$ に収束することが示されている．またノード数にかかわらず, $\mu_j$ の分散は, 平均ではサイクル $t$ 毎に $1/\lambda$ ずつ減少することが示されている．

*1 k-means では一般に実数値データを扱うが, 本稿では適当な定数を乗じ, 整数としてデータを扱うものとする．

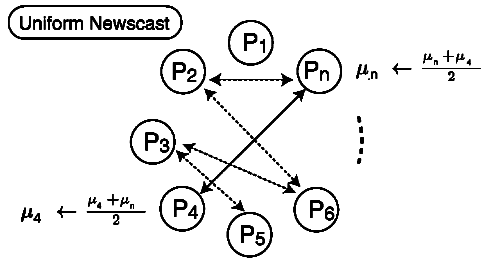


図 2: Newscast プロトコル

newscast プロトコルでは ノード同士は自分が保持するデータを直接交換するわけではないが, ノードが持つデータのプライバシは考慮されていなく, 取得したメッセージからノードが保持するデータの推測は容易である. そのため準同型性暗号のための拡張された newscast プロトコル [4] が提案されている. このプロトコルでは, 各ノードの局所推定値を準同型性暗号によって $c_j = Enc_{p_k}(x_j)$, $c_{j'} = Enc_{p_k}(x_{j'})$ のように暗号化する. このとき, 準同型性によって, 更新式 1 と等価な更新は, 以下のように実行することができる.

$$c_j \leftarrow c_j \cdot c_{j'}^{2^{t_j - t_{j'}}} \quad \text{if } t_j \geq t_{j'}, \tag{2}$$

$$c_j \leftarrow c_{j'} \cdot c_j^{2^{t_{j'} - t_j}} \quad \text{otherwise.} \tag{3}$$

ただし $t_j, t_{j'}$ はそれぞれノード $P_j, P_{j'}$ のサイクルである. このプロトコルによって, ある定数 $T$ について, 各ノードは自分のデータ $x_j$ を開示することなく, 暗号化された平均値 $Enc_{p_k}(2^T \mu_j)$ をプライベートに計算可能であることが証明されている [4]. また, 重み付き平均および二乗平均への拡張も同様に可能である.

### 3.3 プライベートな距離比較

プライベートな距離比較によって, 暗号化されたクラスタ中心 $Enc_{p_k}(2^T \mu_j)$ が各ノードにプライベートに共有されることを前節に示した. 本節では, これを用いて, プライベートな距離比較 (Private Euclidian Distance Comparison, Private EDC) を実行するための方法を示す.

ノード $P_j$ はデータ $x_j$ と, 二つの暗号化されたクラスタ中心 $Enc_{p_k}(2^T \mu_1), Enc_{p_k}(2^T \mu_2)$ を保持し, サーバがこの公開鍵暗号系の秘密鍵を保持しているとしよう. このとき, [5] に示されたプロトコルを用いることによって, ノードとサーバはプライベートに以下の結果を得ることが示されている.

- ノードは $d(x_j, \mu_1) > d(x_j, \mu_2), d(x_j, \mu_1) = d(x_j, \mu_2), d(x_j, \mu_1) < d(x_j, \mu_2)$ の三つの不等式のうち, 成立する一つを得るが, それ以外は何も得ない
- サーバは $d(x_j, \mu_1) = d(x_j, \mu_2)$ が成立するか否かを得るが、それ以外は何も得ない

$k$-means では $k$ 個のクラスタ中心から保持データに最も近いクラスタ中心を決定する必要があるが, 上記のプロトコルをを $k$ 回繰り返すことによって, プライベートな平均計算によって得た, 暗号化された $k$ 個のクラスタ中心から, プライベートに, 自分のデータに最も近接するクラスタ中心を決定することができる.

### 3.4 P2P ネットワークにおけるプライバシを保護した k-means のためのプロトコル

private AAC と private EDC を用いて, プライバシー保護を考慮した P2P 上の $k$-means クラスタリングプロトコルを構築する. 図 3 にプロトコルの詳細を示す.

ステップ 1 および 2 は初期化である. サーバは公開鍵・秘密鍵を作成し, 公開鍵をノードにブロードキャストする. これらの鍵は private EDC, private AAC の両方で用いられる. ステップ 3 および 4 は, private $AWAC_{sq}$(重み付き二乗平均プロトコル) のための準備ステップで実行は一度のみである. ステップ 5 は, クラスタ中心を計算する private AWAC(重み付き平均) およびクラスタ中心の二乗を計算する private $AWAC_{sq}$ が実行される. 得られた平均および二乗平均の暗文は, ステップ 6 の private EDC における $e(S)$ の計算に利用される. ステップ 5 および 6 は, それぞれ非同期的に実行可能であり, またあるノードにおいて通信が途絶しても, 全てのステップにおいて初期化や再スタートが不要であることに注意されたい.

続いてプライベートな $k$-means のためのプロトコルのセキュリティに関する定理を示す. 定義 1 では, "各ノードはデータ $x_j$ に対応するクラスタラベル $z_j$ を得るがそれ以外は何も得ない. サーバは何も得ない "とした. 残念ながらこのプロトコルではこれは達成されず, プライバシーは以下の定義に基づく.

**Definition 2.** プライベートな $k$-means クラスタリング: 値 $x_j$ を保持するノード $P_j (j = 1, ..., n)$ とサーバ $S$ からなる P2P ネットワークを考える. プロトコル実行後, 各 iteration において, 各ノードは距離 $d(x_j, \mu_i)$ の $i$ に関するソート結果を得るが, それ以外は何も得ない. サーバは各 iteration において, 各ノードにおいて距離 $d(x_j, \mu_i) = d(x_j, \mu_{i'}) (i \neq i')$ なるケースの発生頻度を知るが, それ以外は何も得ない.

定義 2 のもとで, プライベートな $k$-means クラスタリングについて以下の定理が成立する.

**Theorem 1.** 各ノードが semi-honest に振舞うならば, Privacy-preserving $k$-means in P2P network は, 定義 2 のもとで, $k$-means クラスタリングをプライベートに計算する.

証明は省略するが, 直感的には以下のように説明される. 定義 2 では, ノードは最近接クラスタのみならず, どのクラスタが何番目に近いか, という情報を得る. しかしながら実際のクラスタ中心をノードは知らないため, この情報はノードにとって他ノードのデータを推測する手がかりにはならない. またサーバは $zero$ と判定される頻度を知るが, 実際にはどのクラスタ中心がその判定に対応するかはサーバには知らないため, この情報からノードのデータに関する情報を推測することはできない. 上記により, 定義 1 にに示した理想的な設定と比較しても, 守られるプライバシーはほぼ変わらないことがわかる.

## 4. 計算時間の分析

提案プロトコルにおいては, step 5 および 6 が繰り返し実行されるため, 計算時間および通信時間はほとんどがこの部分で費される. よってこの二つのステップにおける計算時間および計算量を評価する. step 5 において, private AMC は $2kd$ 回実行され, $T$ ラウンドの後打ち切られる. このステップにおいて, 全ての操作は完全に非同期かつ非中央化されている. step 6 では, ノードは private EDC を $k$ 回実行し, サーバは $nk$ のリクエストに応答する必要がある. 全てのリクエストは非同期的に処理してかまわないが, 複号処理はサーバに中央化されて

**[Privacy-preserving k-means clustering in P2P network for node $P_j$]**

- Input of node $P_j$ : data $x_j$
- Output of node $P_j$ : cluster label of $x_j$

1. Server S: Generate a public key and secret key, then broadcast the private key $p_k$ to all nodes
2. Node $P_j$: Initialize cluster label $z_j$ randomly
3. Node $P_j$: Encrypt $Enc_{p_k}(x_{j,\ell}) = c_{j,\ell} (\ell = 1, ..., d)$ and broadcast this to all nodes
4. Node $P_j$ Receive $c_{j',\ell}$ and compute $c'_{j,\ell} = c_{j,\ell} \prod_{j'=1}^{n} c_{j',\ell} (\ell = 1, ..., d)$
5. Execute private AWAC.
   - $AWAC(x_j, z_{i,j}) \rightarrow \bar{c}_{j,\ell} (i = 1, ..., k, \ell = 1, ..., d)$
   - $AWAC_{sq}(x_j, z_{i,j}) \rightarrow \bar{c'}_{j,\ell} (i = 1, ..., k, \ell = 1, ..., d)$
6. Execute private EDC that compares $d(x_j, \mu_i)$ and $d(x_j, \mu_{i'})$ for all combination of i and i′ in randomized order. Then, identify the nearest cluster i*
7. Update cluster label $z_{i^*,j} = 1, z_{i,j} = 0 (i \neq i^*)$. If converges, output cluster label and terminate. Else, go to step 5.

図 3: P2P ネットワーク上のプライバシを保護した $k$-means のためのプロトコル

| | Node | Server |
|---|---|---|
| step 5 | O(kdT) | – |
| step 6 | O(dk) | O(nk) |

| | Node | Server |
|---|---|---|
| case 1 | 14.9 | 26.4 |
| case 2 | 151 | 3880 |
| case 3 | 151 | 194 |

表 1: プロトコルの計算量および計算時間 (sec, cpu time). Case 1: $n = 100, k = 2$, $d = 2$,512-bit key. Case 2: $n = 1000$, $k = 3$, $d = 6$ and 1024-bit key. Case 3 は case2 において並列に動作する 20 server を想定している．全 case について $T = 20k^2$ とした．

いるため, このステップがプロセス全体のボトルネックとなっている．これを緩和するためには, このプロトコルにおいてはサーバの多重化以外に有効な処理はない．表 1(左) に各ステップの計算量を示す．

続いて, プロトコルの計算時間を評価する．このプロトコルでは, 交換されるメッセージ長は極めて短く, 通信計算量より各ノード, サーバにおける時間計算量が支配的であるため, 通信オーバーヘッドは評価しない．ノード数 $n = 100$, クラスタ数 $k = 2$, データの次元数 $d = 2$ の設定において, 鍵長 $512bit$ の Paillier 公開鍵暗号を用い, private AAC を $T = 20$ ラウンドで設定したとしよう．このとき $k$-means の 1-iteration, つまり step 5(平均計算) および step 6(距離比較) では, 42(=14.9+26.4)(sec) を費やす (case 1)．一方, ノード数 $n = 1000$ の Case 2 では step 5 が 151 (sec), step 6 がサーバサイドにおいて 3880 (sec) で, 明らかに step 6 がボトルネックになっている．これを緩和するために, case 2 において 20 台の並列サーバを用いた case 3 では, step 6 の計算時間は 194 (sec) となり, $k$-means の 1-iteration は 245 (sec) となる．

このように, プロトコルの完了時間は, サーバに中央化された処理のために, 大規模ネットワークにおいては必ずしも十分小さいとはいえないが, 提案プロトコルは現実的な時間に完了するプロトコルであることが示された．

## 5. おわりに

本稿では, P2P ネットワークにおけるプライバシーを保護した $k$-means クラスタリングのためのプロトコルを提案した．提案法はプライベートな非同期平均計算 (private AAC) およびプライベートな距離比較 (private EDC) の組み合わせから構成されている．提案法は HPM を想定したが, APM において発生する暗号化されたデータの次元ごとの断片は, 暗号化されたまま一つのベクトルとして結合することによって private EDC, private AAC の両方において用意に利用可能であるため, APM への拡張はごく自然に可能である．

また提案プロトコルは private EDC, private AAC の両方において完全に非同期化されており, ノード毎の通信途絶に対してロバストである．クラスタ中心の更新とクラスタラベルの更新は同期される必要があるが, これの同期はプライバシを考慮しない $k$-means においても必要とされるため, 非同期化・非停止性をほぼ完全に達成したといえる．ただし private EDC においては計算がサーバに中央化されたステップが存在し, Threshold Encryption を利用したその解消が今後の課題である．

## 参考文献


[1] Evfimievski, A., Srikant, A., Agrawal, R,. and Gehrke, J., Privacy Preserving Mining of Association Rules, ACM SIGKDD Int'l conf. on Knowledge discovery in data mining, pp. 217-228 (2002).

[2] Vaidya, J. and Clifton, C., Privacy-preserving k-means clustering over vertically partitioned data, ACM SIGKDD Int'l conf. on Knowledge discovery in data mining, pp. 206 - 215 (2003).

[3] Lindell, Y. and Pinkas B., Privacy Preserving Data Mining, Crypto 2000 pp. 20-24, (2000).

[4] 佐久間, 小林, P2P ネットワークにおける非同期平均計算プロトコル, 暗号と情報セキュリティシンポジウム, CD-ROM 予稿集, (2007).

[5] 佐久間, 小林, プライバシを保護した内積比較プロトコルの提案, 電子情報通信学会 コンピュータセキュリティ研究会 (IPSJ-CSEC) CD-ROM 予稿集, (2006).

[6] Jaideep Vaidya, Chris Clifton and Michael Zhu, Privacy Preserving Data Mining (Advances in Information Security), Springer Verlag (2005).

[7] Geetha Jagannathan and Rebecca N. Wright, Privacy-preserving distributed k-means clustering over arbitrarily partitioned data, ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 593-599 (2005).

[8] Du, W. and Atallah, M., Privacy-preserving statistical analysis, Annual Computer Security Applications Conference, pp. 102-110, (2001).

[9] Bart Goethals, Sven Laur, Helger Lipmaa and Taneli Mielika"inen. On Private Scalar Product Computation for Privacy-Preserving Data Mining, ICISC 2004, vol. 3506 of LNCS, pp. 104-120 (2004).

[10] Pascal Paillier, Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, EUROCRYPT 1999, pp223-238 (1999).

[11] Kempe, D., Dobra, A., and Gehrke, J., Computing aggregate information using gossip. Symposium on Foundations of Computer Science, pp. 482-491, 2003.

[12] Kowalczyk, W. and Vlassis, N., Newscast EM, NIPS 17, MIT Press, (2005).