2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
July 6-11, 2014, Beijing, China

Fuzzy Co-clustering of Vertically Partitioned Cooccurrence Data With Privacy Consideration

Katsuhiro Honda, Toshiya Oda and Akira Notsu

Abstract— **This paper considers fuzzy co-clustering of distributed cooccurrence data, where vertically partitioned cooccurrence information among objects and items are stored in several sites. In order to utilize such distributed data sets without fear of information leaks, a privacy preserving procedure is introduced to fuzzy clustering for categorical multivariate data (FCCM). Withholding each element of cooccurrence matrices, only object memberships are shared by multiple sites and their (implicit) joint co-cluster structures are revealed through an iterative clustering process. Several experimental results demonstrate the ability of improving the individual co-clustering results of each site by combining the distributed data sets.**

I. Introduction

In these days, innumerable databases are constructed with various purposes and big data analysis is regarded as a promising technique for providing business benefits. Privacy preserving data mining (PPDM) [1] is a fundamental approach for utilizing multiple databases including personal information without fear of information leaks. In this paper, the problem of extracting cluster structures from distributed databases, which are independently stored in multiple sites, is considered. For applying k-means-type clustering algorithms to distributed databases, several procedures have been proposed. Assume that we have n objects in conjunction with their m-dimensional observations and the goal is to partition the objects into C clusters, which are represented by prototypical centroids. In horizontally partitioned databases, each site stores m-dimensional observations only on a part of n objects and C cluster centroids are estimated cooperating with each other [2], [3], [4]. Weighted sum of each attributes can be shared for estimating centroids. On the other hand, in vertically partitioned databases, each site stores all n record with different attributes, which are a part of m-dimensional observations. Because each site can have prototypical information of clusters only on their observable attributes, only clustering criterion on each object can be shared by multiple sites [5], [6].

The goal of co-clustering is to simultaneously partition objects and items considering their cooccurrence information. In co-clustering tasks, an $n \times m$ cooccurrence matrix is given, whose elements are cooccurrence degree among n objects and m items, e.g., the number of appearance of keywords in documents and the frequencies of purchase of items by customers. Fuzzy clustering for categorical multivariate data (FCCM) [7] is an FCM-type fuzzy co-clustering algorithm, which simultaneously estimates two types of fuzzy memberships so that the degree of co-cluster aggregations of objects and items is maximized. In this paper, an extended algorithm for FCCM is considered with the goal of applying to vertically distributed cooccurrence matrices. Assume that several companies share their common customers but have the purchase history only on their own products. In the conventional business model, each company can utilize their own purchase history data only and supply such services as collaborative filtering [8], [9], [10] only on their products. If they can jointly utilize such distributed databases, it is expected that they will supply much greater services with high reliability.

In order to extract intrinsic co-cluster structures from vertically distributed cooccurrence matrices, an alternately iterative procedure is proposed, in which personal privacy is preserved by withholding individual cooccurrence information each other. Fuzzy memberships are shared by multiple sites while item memberships are estimated and kept within each site. Several experimental results demonstrate the ability of improving the individual co-clustering results of each site by combining the distributed data sets.

The remaining parts of this paper are organized as follows: Section II gives a brief review on fuzzy clustering and fuzzy co-clustering. An extended procedure for applying FCCM clustering to vertically distributed databases is proposed in Section III. The applicability of the proposed procedures is demonstrated in Section IV and the summary conclusions are given in Section V.

II. FCM-type Clustering Models

A. k-Means and Fuzzy c-Means

Let $\boldsymbol{x}_i, i = 1, \ldots, n$ be a set of n data objects. k-Means [11] is the most well-known non-hierarchical clustering algorithm that assigns data object $\boldsymbol{x}_i$ composed of m-dimensional observation to the nearest cluster center $\boldsymbol{b}_c$, which is the mean vector in the cth cluster. The objective function to be minimized is the sum of the squared errors in clusters $\boldsymbol{G}_c$, $c = 1, \ldots, C$:

$$\min \quad L_{km} = \sum_{c=1}^{C} u_{ci} ||\boldsymbol{x}_i - \boldsymbol{b}_c||^2. \tag{1}$$

u_{ci} represents its assignment such that $u_{ci} = 1$ for $i \in G_c$ and $u_{ci} = 0$ for otherwise. The k-Means algorithm is an iterative procedure composed of two steps: nearest prototype

Katsuhiro Honda, Toshiya Oda and Akira Notsu are with the Graduate School of Engineering, Osaka Prefecture University, Osaka, Japan (email: {honda, notsu}@cs.osakafu-u.ac.jp).

This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology, Japan, through a Grant-in-Aid for Scientific Research (#23500283, #26330281).

assignment and prototype estimation, and converges to a local optimal solution.

Fuzzy c-Means [12] is a fuzzified version of k-Means, in which membership indicators are drawn from the interval of [0,1] instead of the alternative assignment of $\{0,1\}$ used in k-Means:

$$\min \quad L_{fcm} = \sum_{c=1}^{C}\sum_{i=1}^{n} u_{ci}^{\theta}||\boldsymbol{x}_i - \boldsymbol{b}_c||^2 \tag{2}$$

$$\text{s.t.} \quad u_{ci} \in [0,1], \quad i = 1,\cdots,n : c = 1,\cdots,C \tag{3}$$

$$\sum_{c=1}^{C} u_{ci} = 1, \quad i = 1,\cdots,n \tag{4}$$

where u_{ci} denotes the membership degree of ith object to the cth cluster. In order to fuzzify memberships u_{ci}, the weighting exponent θ ($\theta > 1$) was introduced into the k-Means objective function, which is a linear function with respect to u_{ci}. The larger the θ is, the fuzzier the partition is.

The k-Means objective function can also be non-linearized by other approaches for fuzzifying memberships u_{ci}. In FCM by regularization with entropy (eFCM) [13], [14], an entropy-based non-linear term is introduced into the k-Means objective function as:

$$\min \quad L_{efcm} = \sum_{c=1}^{C}\sum_{i=1}^{n} u_{ci}||\boldsymbol{x}_i - \boldsymbol{b}_c||^2 + \lambda \sum_{c=1}^{C}\sum_{i=1}^{n} u_{ci}\log u_{ci} \tag{5}$$

$$\text{s.t.} \quad u_{ci} \in [0,1], \quad i = 1,\cdots,n : c = 1,\cdots,C \tag{6}$$

$$\sum_{c=1}^{C} u_{ci} = 1, \quad i = 1,\cdots,n \tag{7}$$

where the entropy term works like the weighting exponent in the standard FCM algorithm. The larger the λ is, the fuzzier the partition is.

B. Fuzzy Co-clustering

Assume that we have a cooccurrence matrix $R = \{r_{ij}\}$ on objects $i = 1,\dots,n$ and items $j = 1,\dots,m$, in which r_{ij} represent the degree of cooccurrence of item j with object i. The goal of co-clustering is to simultaneously partition objects and items by estimating two types of fuzzy memberships. The fuzzy memberships of objects u_{ci} are designed in a similar concept to FCM, in which they represent exclusive assignment of objects to clusters such that $\sum_{c=1}^{C} u_{ci} = 1$. On the other hand, in order to avoid trivial solutions, the fuzzy memberships of items w_{cj} are mostly responsible for representing the mutual typicalities in each cluster such that $\sum_{j=1}^{m} w_{cj} = 1$.

Oh *et al.* [7] proposed the FCM-type co-clustering model, which is called FCCM, by modifying the FCM algorithm for handling cooccurrence information, where the cluster aggregation degree of each cluster is maximized:

$$\max \quad L_{fccm} = \sum_{c=1}^{C}\sum_{i=1}^{n}\sum_{j=1}^{m} u_{ci}w_{cj}r_{ij} - \lambda_u \sum_{c=1}^{C}\sum_{i=1}^{n} u_{ci}\log u_{ci} - \lambda_w \sum_{c=1}^{C}\sum_{j=1}^{m} w_{cj}\log w_{cj} \tag{8}$$

In FCCM, the entropy-based fuzzification method [13], [14] was adopted instead of the standard approach in FCM because the exponential weight θ in FCM can work only in the minimization framework of positive objective functions. λ_u and λ_w play a similar role to θ, where larger λ brings fuzzier partitions while small λ brings crisp partitions.

The clustering algorithm is an iterative process of updating u_{ci} and w_{cj} using the following rules:

$$u_{ci} = \frac{\exp\left(\lambda_u^{-1}\sum_{j=1}^{m} w_{cj}r_{ij}\right)}{\sum_{\ell=1}^{C}\exp\left(\lambda_u^{-1}\sum_{j=1}^{m} w_{\ell j}r_{ij}\right)}, \tag{9}$$

and

$$w_{cj} = \frac{\exp\left(\lambda_w^{-1}\sum_{i=1}^{n} u_{ci}r_{ij}\right)}{\sum_{\ell=1}^{m}\exp\left(\lambda_w^{-1}\sum_{i=1}^{n} u_{ci}r_{i\ell}\right)}. \tag{10}$$

III. Application to Vertically Distributed Databases With Privacy Consideration

A. Privacy Preserving k-Means Clustering

Assume that T sites ($t = 1,\dots,T$) has m_t-dimensional observations $\boldsymbol{x}_i^t = (x_{i1}^t,\dots,x_{im_t}^t)^\top$ on common n objects ($i = 1,\dots,n$) and $\sum_{t=1}^{T} m_t = m$. If we can gather the data pieces into a whole data set $\boldsymbol{x}_i$, the nearest centroid in k-Means process is searched for by calculating the distances between objects $\boldsymbol{x}_i$ and centroids $\boldsymbol{b}_c$ as:

$$||\boldsymbol{x}_i - \boldsymbol{b}_c||^2 = \sum_{t=1}^{T} ||\boldsymbol{x}_i^t - \boldsymbol{b}_c^t||^2, \tag{11}$$

where $d_{ci}^t = ||\boldsymbol{x}_i^t - \boldsymbol{b}_c^t||^2$ is the within-cluster square distance in site t. This implies that we can find the nearest centroid only with d_{ci}^t without broadcasting each element x_{ij}^t. Then, in privacy preserving k-Means clustering, d_{ci}^t instead of x_{ij}^t are utilized for performing k-Means clustering in a secure manner.

By the way, in fuzzy co-clustering context, we do not have such representative prototypes as centroids. In this paper, it is considered to directly utilize object memberships u_{ci} for item membership estimation in each site.

B. Fuzzy Co-clustering of Vertically Distributed Cooccurrence Matrices

In the following, the task of extracting co-cluster structures from vertically distributed cooccurrence matrices is considered. Assume that T sites ($t = 1,\ldots,T$) share common n objects ($i = 1,\ldots,n$) and have different cooccurrence information on different items, which are summarized into $n \times m_t$ matrices $R_t = \{r_{ij}^t\}$, where m_t is the number of items in site t and $\sum_{t=1}^{T} m_t = m$.

If we do not care the privacy issues, the distributed matrices should be gathered into a full $n \times m$ matrix to be analyzed in a single process without information losses. Taking the privacy preservation into account, however, each matrix should be processed in each site without broadcasting personal information although the reliability of each co-cluster structure may not be enough satisfied because of information losses.

The goal of the task in distributed database analysis is to estimate object and item memberships as similar to the full-data case as possible by sharing object partition information without broadcasting cooccurrence information r_{ij}^t. Object memberships u_{ci} to be shared by sites are common and are defined in the same manner with the conventional FCCM. On the other hand, item memberships w_{cj} are somewhat different because they follow the within-cluster sum constraint. In this subsection, it is assumed that item memberships are independently estimated in each site following the site-wise constraint $\sum_{j=1}^{m_t} w_{cj}^t = 1$, where w_{cj}^t is the item membership on item j in site t.

1) Two-site case ($T = 2$): First, a two-site case ($T = 2$) is considered. In order to jointly perform co-clustering in two sites t_1 and t_2, a strategy of sharing *object memberships* is considered.

Eq.(9) implies that each object membership function is dependent on $\sum_{j=1}^{m} w_{cj} r_{ij}$, which is the sum of independent information $\sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$. Therefore, by sharing $\sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ instead of u_{ci}, each site can update u_{ci} by itself. Once u_{ci} are updated in each site, the item memberships are updated in each site and $\sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ are also updated.

Following the above consideration, a secure process for two-site case is described as follows:

[FCCM for Vertically Distributed Cooccurrence Matrices: FCCM-VD (two-site case)]

1) Given $n \times m_1$ matrix R_1 and $n \times m_2$ matrix R_2, let C be the number of clusters. Choose the fuzzification weights λ_u and λ_w. (We can also use different λ_w in different sites because item memberships are available only in each site.)
2) **[Initialization]** Perform the conventional FCCM in site t_1 for estimating u_{ci} and w_{cj}^1. Using u_{ci}, estimate w_{cj}^2 in site t_2.
3) **[Iterative process]** Iterate the following process until convergence of all u_{ci}.
   a) In site t_1, calculate $\sum_{j=1}^{m_1} w_{cj}^1 r_{ij}^1$ and send them to site t_2.
   b) In site t_2, update u_{ci} and w_{cj}^2. Calculate $\sum_{j=1}^{m_2} w_{cj}^2 r_{ij}^2$ and send them to site t_1.
   c) In site t_1, update u_{ci} and w_{cj}^1.

Here, we should note that, in this two-site model, the two sites can mutually know the co-cluster structures of the other site from shared information although the cooccurrence information components r_{ij} are concealed.

2) Multi-site case ($T > 2$): In the multi-site case of $T > 2$, we can consider more secure process following the encryption approach in [5]. In the same manner with the two-site case, the object partition is shared referring to $\sum_{j=1}^{m} w_{cj} r_{ij}$ while their actual values are concealed each other. The secure process is implemented by at least three sites. Here, two sites of t_1 and t_T are selected as representative sites. Site t_1 generates a length C random vector $\boldsymbol{v}_t = (v_{t1},\ldots,v_{tC})^\top$ for each site t, such that $\sum_{t=1}^{T} \boldsymbol{v}_t = \mathbf{0}$. Then, sites $t_1 \ldots t_{T-1}$ send $v_{tc} + \sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ to site t_T and their total amount $\sum_{t=1}^{T}(v_{tc} + \sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t)$ is calculated for estimating u_{ci} in site t_T. $\sum_{t=1}^{T} \boldsymbol{v}_t = \mathbf{0}$ implies that the total amount is equivalent to $\sum_{t=1}^{T}\sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ although the individual value of each site is concealed by v_{tc}. In this scheme, no site can reveal the actual value of $\sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ on other sites.

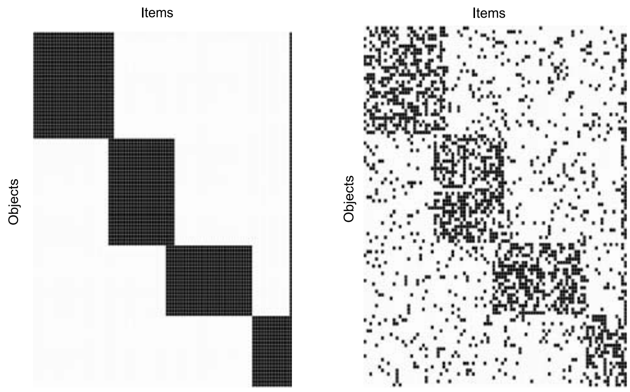
Following the above consideration, a secure process for multi-site case is described as follows:

[FCCM for Vertically Distributed Cooccurrence Matrices: FCCM-VD (multi-site case)]

1) Given $n \times m_1$ matrix $R_1 \ldots n \times m_T$ matrix R_T, let C be the number of clusters. Choose the fuzzification weights λ_u and λ_w. (We can also use different λ_w in different sites because item memberships are available only in each site.)
2) **[Initialization]** Randomly initialize u_{ci} such that $\sum_{c=1}^{C} u_{ci} = 1$ and broadcast them to all sites.
3) **[Iterative process]** Iterate the following process until convergence of all u_{ci}.
   a) For $i = 1,\ldots,n$
      i) In site t_1, generate random vectors $\boldsymbol{v}_t = (v_{t1},\ldots,v_{tC})^\top$, $t = 1,\ldots,T$ such that $\sum_{t=1}^{T} \boldsymbol{v}_t = 0$, and send $\boldsymbol{v}_t$ to site t.
      ii) In sites $t_1,\ldots,t_T$, update w_{cj}^t using the current values of u_{ci}. Calculate $v_{tc} + \sum_{j=1}^{m_t} w_{cj}^t r_{ij}^t$ and send them to site t_T.
      iii) In site t_T, update u_{ci} and broadcast them to all sites.
   b) Check the termination condition.

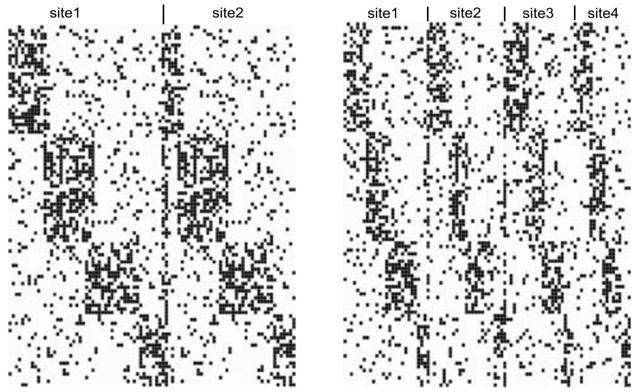
IV. Experimental Results

Several results of comparative experiments are presented for demonstrating the characteristic features of the proposed method. The experiments were performed with artificially



(a)noiseless (b)noise

Fig. 1. Artificial coocurrance matrices

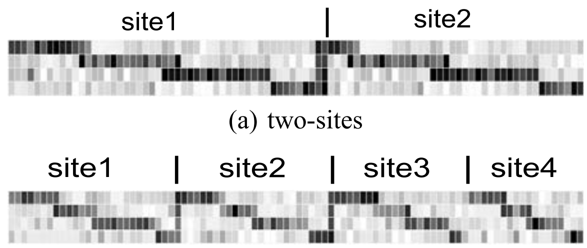


(a) two-sites (b) four-sites

Fig. 2. Arranged matrices from Fig. 1

TABLE I
COMPARISON OF PARTITION QUALITY MEASURED BY CORRELATION COEFFICIENTS AMONG ITEM MEMBERSHIPS (TWO-SITES CASE)

		site 1	site 2
FCCM-VD	Best (Max.)	0.990	0.989
	Mean	0.929	0.918
Site-wise FCCM	Best (Max.)	0.986	0.966
	Mean	0.868	0.756



(b) four-sites

Fig. 3. Item memberships of each cluster in full matrices cases

generated cooccurrence matrices. A base (noise-less) cooccurrence matrix composed of 100 objects and 90 items includes 4 co-cluster structures. Figure 1-(a) shows the 100×90 cooccurrence matrix $R = \{r_{ij}\}$, in which black and white cells imply $r_{ij} = 1$ and $r_{ij} = 0$, respectively. The 100 objects belong to a single co-cluster while some items are shared by multiple clusters. A noisy cooccurrence matrix shown in Fig. 1-(b) was generated from the base matrix by replacing '1' elements with '0' at a rate of 50% and '0' elements with '1' at a rate of 10%. Disturbed by noise, the co-cluster structures are only weakly recognized.

A. Matrices Arrangement and Full Matrix Case Results

Two types of vertically distributed cooccurrence submatrices were generated by arranging the 100×90 noisy matrix into two and four sites. Figure 2 shows the arranged cooccurrence matrices, in which the order of items were arranged from Fig. 1-(b). Figures 2-(a) and 2-(b) are for two-sites case and four-sites case, respectively.

First, the conventional FCCM was applied to the two full 100×90 cooccurrence matrices with $\lambda_u = 0.001$ and $\lambda_w = 100.0$. The derived item memberships are shown in Fig. 3, where each row represents the 90-dimentional item membership vector of each cluster with gray-scale, i.e., black and white are $w_{cj} = w_{max}$ and $w_{cj} = 0$, respectively.

Second, the arranged cooccurrence matrices of Fig. 2 were vertically distributed. $m = 90$ items were divided into $(m_1, m_2) = (50, 40)$ in two-sites case and $(m_1, m_2, m_3, m_4) = (27, 24, 21, 18)$ in four-sites case, where four co-cluster structures are very weakly implied in each site. The goal of analyzing the vertically distributed matrices is to derive a similar object and item memberships to those of the full matrices cases. Then, in this experiment, the clustering qualities are evaluated by measuring the correlation coefficient between the derived item membership vectors in each cluster and the full matrix case result of Fig. 3.

B. Two-sites Case

In the two-sites case, the FCCM algorithms were applied to the cooccurrence matrices of Fig. 2-(a), which is vertically distributed with 100×50 (site 1) and 100×40 (site 2) matrices. First, by applying the conventional FCCM algorithm several times in each site without collaboration, it was often difficult to reveal the four-cluster structures in both sites because each site has only weak site-wise cooccurrence information.

Second, by applying the proposed FCCM-VD algorithm with $\lambda_u = 0.005$ and $\lambda_w = 100.0$, the item membership vectors (the best one having the maximum correlation coefficient with the base result) shown in Fig. 4 was derived, where 4 site-wise separate item membership vectors are merged into a 90-dimensional one in order to compare with the base results of Fig. 3.

Table I compares the mean (best) correlation coefficients between the base results and the site-wise memberships, which were derived with the proposed FCCM-VD and the conventional site-wise FCCM. The algorithms were implemented in 50 trials with different initialization, and the best

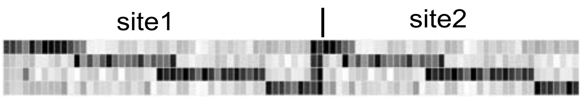


Fig. 4. Item memberships derived by proposed method (two-sites case)

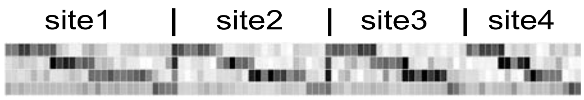


Fig. 5. Item memberships derived by proposed method (four-sites case)

TABLE II

COMPARISON OF PARTITION QUALITY MEASURED BY CORRELATION COEFFICIENTS AMONG ITEM MEMBERSHIPS (FOUR-SITES CASE)

		site 1	site 2	site 3	site 4
FCCM-VD	Best (Max.)	0.998	0.998	0.997	0.999
	Mean	0.945	0.949	0.943	0.947
Site-wise FCCM	Best (Max.)	0.913	0.889	0.935	0.946
	Mean	0.718	0.677	0.851	0.903

TABLE III

COMPARISON OF COMPUTATIONAL TIME AND CONVERGENCE SPEED (FOUR-SITES CASE)

	time (s)	iteration
FCCM-VD	0.01270	35.98
whole data FCCM	0.01318	44.20

and mean values are compared. The table implies that the proposed algorithm is useful for estimating the intrinsic co-cluster structures through collaborative scheme among sites.

C. Four-sites Case

A similar experiment was performed in the four-sites case, where the cooccurrence information of Fig. 2-(b) was vertically distributed in four sites. Because the site-wise information became poorer than the previous experiment, it is much difficult to capture the intrinsic co-cluster structures in all four sites without collaboration.

By applying the proposed FCCM-VD algorithm with $\lambda_u = 0.005$ and $\lambda_w = 100.0$, the item membership vectors (best result) shown in Fig. 5 was estimated. The derived result is quite similar to the base case of Fig. 3.

The partition quality is also compared in Table II, which implies that the proposed FCCM-VD algorithm still work well in multi-site cases.

Finally, computational costs are compared in Table III, which compares the mean computational times (sec.) and iterations needed for convergence in 50 trials. The experiment was performed with an Intel Core i7 CPU (2.80GHz) and 8.0GB memory, and the stopping condition was $\max(|u_{ci}^{NEW} - u_{ci}^{OLD}|) < 1.0 \times 10^{-8}$. The proposed algorithm could achieve faster convergence than the whole data FCCM, in which FCCM was applied to the whole data set at a time. It may be because the site-wise constraint contributed to fast convergent to (site-wise) local optima. This result implies the stable feature of the proposed method.

V. CONCLUSIONS

In this paper, two frameworks for handling vertically distributed cooccurrence information in fuzzy co-clustering were proposed with the goal of estimating stable co-cluster structures through secure collaboration among multiple sites. In the two-sites case, the object memberships are shared by two sites while the each element of cooccurrence information is concealed only in each site. The sharing scheme was also extended to multi-sites cases, where more secure utilization of distributed information is achieved.

In future work, the applicability of the proposed algorithm to such application as co-cluster-based collaborative filtering [9], [10] and document analysis [15] can be studied. Another possible future work is to evaluate the responsibility (utility) degree of each site.

REFERENCES

[1] C. C. Aggarwal and P. S. Yu, *Privacy-Preserving Data Mining: Models and Algorithms*, Springer-Verlag, New York, 2008.

[2] S. Jha, L. Kruger and P. McDaniel, "Privacy preserving clustering," *10th European Symp. on Research in Computer Security,* pp. 397-417, 2005.

[3] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin and M. Y. Zhu, "Tools for privacy preserving distributed data mining," *ACM SIGKDD Explorations Newsletter*. vol. 4, no. 2, pp. 28-34, 2002.

[4] A. Inan, S. V. Kaya, Y. Saygyn, E. Savas, A. A. Hintoglu and A. Levi, "Privacy preserving clustering on horizontally partitioned data," *Data & Knowledge Engineering*, vol. 63, no. 3, pp. 646-666, 2007.

[5] J. Vaidya and C. Clifton, "Privacy-preserving k-means clustering over vertically partitioned data," *Proc. 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 206-215, 2003.

[6] T.-K. Yu, D.T. Lee, S.-M. Chang and J. Zhan, "Multi-party k-means clustering with privacy consideration," *Proc. 2010 International Symposium on Parallel and Distributed Processing with Applications*, pp. 200-207, 2010.

[7] C.-H. Oh, K. Honda and H. Ichihashi, "Fuzzy clustering for categorical multivariate data," *Proc. of Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, pp. 2154-2159, 2001.

[8] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gardon and J. Riedl, "Grouplens: applying collaborative filtering to usenet news," *Communications of the ACM*, vol. 40, no. 3, pp. 77-87, 1997.

[9] K. Honda, A. Notsu and H. Ichihashi, "Collaborative filtering by sequential user-item co-cluster extraction from rectangular relational data," *International Journal of Knowledge Engineering and Soft Data Paradigms*, vol. 2, no. 4, pp. 312-327, 2010.

[10] K. Honda, M. Muranishi, A. Notsu and H. Ichihashi, "FCM-type cluster validation in fuzzy co-clustering and collaborative filtering applicability," *International Journal of Computer Science and Network Security*, vol. 13, no. 1, pp. 24-29, 2013.

[11] J. B. MacQueen, Some methods of classification and analysis of multivariate observations, *Proc. of 5th Berkeley Symposium on Math. Stat. and Prob.*, pp. 281-297, 1967.

[12] J. C. Bezdek: *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, 1981.

[13] S. Miyamoto and M. Mukaidono, "Fuzzy c-means as a regularization and maximum entropy approach," *Proc. of the 7th International Fuzzy Systems Association World Congress*, vol. 2, pp. 86-92, 1997.

[14] S. Miyamoto, H. Ichihashi and K. Honda: *Algorithms for Fuzzy Clustering*, Springer, 2008.

[15] K. Kummamuru, A. Dhawale and R. Krishnapuram, "Fuzzy co-clustering of documents and keywords," *Proc. 2003 IEEE Int'l Conf. Fuzzy Systems*, vol. 2, pp. 772-777, 2003.