Rings for Privacy: An Architecture for Privacy-Preserving User Profiling

C. Barcellona* I. Tinnirello* M.L. Merani**
*University of Palermo **University of Modena and Reggio Emilia, Italy
cettina.barcellona@unimore.it

In present days, where anyone stays online nearly everywhere and everytime, it is crucial from the viewpoint of service providers to collect consumers data, so that questions such as "what kind of advertisement do they click on?","what do they buy online?","what IP-TV channels do they watch?" and "what do they like to do on social networks?" are answered and some relevant information about the users' habits and likes are extracted, resold for business, or employed to provide services that are better tailored to the customers' interests. Several privacy issues have however to be taken into account, when sensitive data are managed. We believe that approaches which guarantee the users' privacy have to be pursued: accordingly, we put forth a solution that performs users' profiling and keeps sensitive information private; additionally, it lends itself to a decentralized, cost-effective implementation.

We apply our idea to an iterative clustering algorithm for data mining called Fuzzy C-Means (FCM) [**?**] [**?**], that we employ for profiling the users of a generic service in terms of a given number M of metrics. In the original dataset the algorithm recognizes particular structures called clusters and defines the position of the centroids, where the centroid is the representative element of each cluster. Elements within the same cluster are classified as similar, whereas those belonging to different clusters are dissimilar. For clearness' sake, the algorithm steps are briefly reported below.

Given the data of N users are collected and K clusters are sought for, the scheme defines a membership matrix $\mathbf{U} \in [0,1]^{N \times K}$, whose generic element u_{ij} is the membership degree of the i-th user data vector $\mathbf{d}_i^{[1\times M]}$ to the j-th cluster $\mathbf{c}_j^{[1\times M]}$. The FCM goal is to minimize the function

$$\sum_{j=1}^{K}\sum_{i=1}^{N} u_{ij}^{f}\|\mathbf{d}_i - \mathbf{c}_j\|^2 , \tag{1}$$

where $\|\cdot\|$ indicates the Euclidean norm and f is the fuzziness parameter. Starting from a random matrix initialization $\mathbf{U^{(0)}}$, at the t-th step the FCM algorithm works as follows:

1) update the cluster centroids:

$$\mathbf{c}_j^{(t)} = \frac{\sum_{i=1}^{N}\left(u_{ij}^{(t)}\right)^f \cdot \mathbf{d}_i}{\sum_{i=1}^{N}\left(u_{ij}{}^{(t)}\right)^f} ; \tag{2}$$

2) update the membership matrix: the i-th data element is fuzzily assigned to the j-th cluster with membership degree computed as:

$$u_{ij}^{(t)} = \frac{1}{\sum_{l=1}^{K}\left(\frac{\|\mathbf{d}_i - \mathbf{c}_j^{(t)}\|}{\|\mathbf{d}_i - \mathbf{c}_l^{(t)}\|}\right)^{\frac{2}{f-1}}} ; \tag{3}$$

3) convergence verification: if $\|\mathbf{U}^{(t)} - \mathbf{U}^{(t-1)}\| < \epsilon$, then the algorithm halts, otherwise it steps back to 1.

Given the update required at step 1, it is convenient for our proposal to introduce matrix $\mathbf{D}_i^{(t)}$, defined as:

$$\mathbf{D}_i^{(t)} = \begin{pmatrix} (u_{i1}^{(t)})^f d_{i1} & \dots & (u_{iK}^{(t)})^f d_{i1} \\ \vdots & \ddots & \vdots \\ (u_{i1}^{(t)})^f d_{iM} & \cdots & (u_{iK}^{(t)})^f d_{iM} \\ (u_{i1}^{(t)})^f & \cdots & (u_{iK}^{(t)}) f \end{pmatrix}$$

This matrix is autonomously generated and updated by the i-th user at every step of the algorithm.

To run the FCM algorithm in a privacy preserving way, we can resort to some techniques drawn from the subfield of cryptography known as Secure Multi-Party Computation (SMPC). The SMPC main idea is to perform a computation among different parties, where each contributes with an input value that remains private, even if the final output will be public.

In what follows we will assume that all actors, servers and users, are *honest-but-curious*.

As a first solution, we refer to the architecture proposed by some of the authors of the current paper in [**?**] and sketched in Fig.1(a). In this scenario, where S servers are present, $S \geq 2$, we can run the FCM algorithm and still preserve the users' privacy (as long as the number of colluded servers is lower than or equal to $S-1$) employing a secret sharing scheme generating shares as in [**?**], that we detail below:

1) the i-th user makes S shares of its $\mathbf{D}_i^{(t)}$ matrix and sends a share to each server;
2) every server sums all the shares it receives and sends the resulting matrix to the server that is in charge of determining the centroids via equation (2) (it could be any of the S servers);
3) the centroids are sent back to each user, that updates its membership array, hence determines its $\mathbf{D}_i^{(t+1)}$ matrix and then resumes from step 1, unless the convergence condition is reached.

We now introduce an alternative, decentralized architecture, where there is no need to have S servers: as portrayed in Fig.1(b), the N users are grouped in N_{ring} rings, where the number of nodes is $N_{node} = \frac{N}{N_{ring}}\rfloor$ in all rings except one and there is only one server that belongs to all rings. In this scenario, the privacy preserving FCM algorithm is implemented in the following manner:

1) the server sends a random matrix $\mathbf{R}_j^{(t)}$, of the same size as $\mathbf{D}_i$, to the first node of the j-th ring, $j = 1, 2, \dots, N_{ring}$;
2) the first node starts the Secure Sum Computation [**?**], summing in modular arithmetic its $\mathbf{D}_{1j}^{(t)}$ matrix to $\mathbf{R}_j^{(t)}$ and forwards the result to the second node, that in turn adds its matrix $\mathbf{D}_{2j}^{(t)}$ and then forwards the outcome to the third node in the chain; this process continues up to the last node of the ring, that sends back the "blind" partial sum of the matrices to the server;

978-1-4673-1017-8/14/$31.00 ©2014 IEEE

3) the server collects the partial sums of all the rings, it subtracts from each of them the corresponding random matrix $\mathbf{R}_j^{(t)}$ and employs the result to compute the centroids in accordance to eq.(2);
4) unless the convergence condition is reached, the centroids and a new random matrix are sent back to the first user of each ring, and the algorithm returns to step 2.

To better understand benefits and drawbacks of the newly proposed scheme, we next compare the communication cost of both architectures and explore what happens in the ring-based solution when the rings are error-prone.

As regards the communication cost, we measure it by the number of connections required to support all server-user, user-user and server-server communications: within the first scheme, the cost is $C = NS + S - 1$; within the second setting, the cost is $C' \simeq 2 \cdot \frac{N}{N_{node}} \rfloor + (N_{node} - 1) \cdot \frac{N}{N_{node}} \rfloor$. It is straightforward to conclude that in the first setting the minimum cost is $C_{min} = C|_{S=2} = 2N + 1$, whereas in the second the cost C' ranges from $C'_{min} = C'|_{N_{node}=N} = N + 1$ to $C'_{max} = C'|_{N_{node}=2} = \frac{3}{2}N$, and even in the worst case, C' is far lower than C_{min}. Unfortunately, the ring-based architecture is much more vulnerable: we therefore complement the previous considerations examining the feasibility of this scheme in a scenario where users are unstable, i.e., there is a non-null probability p that the i-th user cannot provide its data array, where we assume p not to depend on the user and the users to fail independently one from the other.

To achieve some insights on the FCM behavior, we first ran the algorithm with a decreasing number n of available users data: for each n, we computed the new centroids position on a significant number of trials and evaluated the average distances from the "correct" centroids, i.e., those determined when the entire data set was available. We considered acceptable the loss of fewer than N' data, where N' corresponds to the first value that violates the condition $\max_j E[||\mathbf{c}'_j - \mathbf{c}_j||] < \delta$, with $E[\cdot]$ indicating the average, $\mathbf{c}'_j$ the j-th centroid determined with n data and δ the accuracy error. We repeated the evaluation for several data sets that display different features in terms of sparseness and cardinality. As expected, we found that for a given δ the minimum acceptable number N' of data greatly varies: yet, it was quite interesting to observe that the FCM algorithm is quite robust against data losses: in the most pessimistic condition we ran into, it could bear $N' \simeq 0.2N$ losses still guaranteeing a relative accuracy error in determining the centroids lower than $\delta = 10^{-2}$; in the most favorable case, N' could be as high as $0.7N$. Accordingly, we can determine the probability P_{fail} that the decentralized ring-based algorithm fails, defined as the probability that fewer than N' data arrays are available, as a function of p and of the number of nodes N_{node} in each ring:

$$P_{fail} = \sum_{i=\frac{N'}{N_{node}}}^{N_{ring}} \binom{N_{Ring}}{i} P_{ring}^i \cdot (1 - P_{ring})^{(N_{ring} - i)}, \quad (4)$$

where $P_{ring} = 1 - (1 - p)^{N_{node}}$ is the ring failure probability, given that the generic ring is out of order if at least one of its nodes fails. Fig.2 summarizes our preliminary findings: it reports P_{fail} as a function of N_{node}, when $N = 1000$ and $N' = 200$. Depending on p, this figure indicates that we have to work with fairly small values of N_{node}, i.e., small rings, if we want to confine P_{fail} to sufficiently low values. Yet, these results refer to very conservative δ and N' values: fitting δ and N' to each specific data set, we could easily get very low P_{fail} values even with higher p values and much larger rings, therefore with appealingly low communication costs.

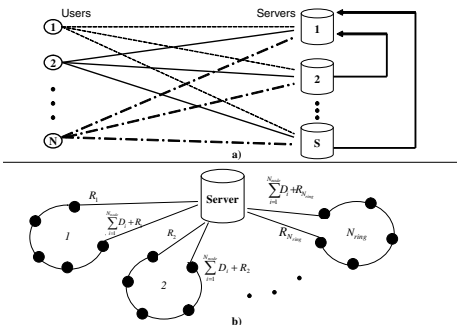

Fig. 1. Alternative architectures for privacy preserving profiling: a) N users communicating to S servers; b) N users organized in N_{ring} rings.

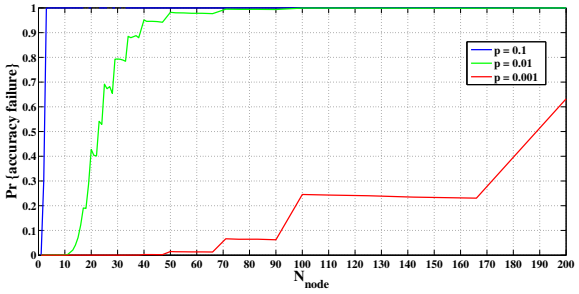

Fig. 2. P_{fail} for $p = \{10^{-1}, 10^{-2}, 10^{-3}\}$, when $N = 1000$ and $N' = 200$.

REFERENCES

[1] J. C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algoritms",*Plenum Press,* , 1981, New York .
[2] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", *Journal of Cybernetics*, No.3, 1973, pp.32-57.
[3] C. Barcellona, G. Di Bella, J. Golic, P. Cassarà, I. Tinnirello, "Multi-Party Metering: An Architecture for Privacy-Preserving Profiling Schemes", *Sustainable Internet and ICT for Sustainability (SustainIT), 2013*, pp.1,6, 30-31 Oct. 2013.
[4] M. Ito, A. Saito, T. Nishizeki, "Secret Sharing Schemes Realizing General Access Structure", *in Proc. of the IEEE Global Telecomunication Conf. Globecom*, vol. 87, pp. 99-102, 1987.
[5] Chris Clifton, Murat Kantarcioglou, Xiadong Lin, and Michael Y. Zhu, "Tools for privacy preserving distributed data mining", *SIGKDDExplorations 4*, no. 2, 2002.