Privacy preserving similarity detection for data analysis

Iraklis Leontiadis, Melek Önen, Refik Molva, M.J. Chorley, G.B. Colombo

Privacy preserving similarity detection for data analysis

Iraklis Leontiadis, Melek Önen, Refik Molva
Networking and Security Department
EURECOM, Sophia-Antipolis, France
{leontiad,onen,molva}@eurecom.fr

M.J. Chorley, G.B. Colombo
School of Computer Science & Informatics
Cardiff University
{m.j.chorley, fg.colombo}@cs.cardiff.ac.uk

***Abstract*—Current applications tend to use personal sensitive information to achieve better quality with respect to their services. Since the third parties are not trusted the data must be protected such that individual data privacy is not compromised but at the same time operations on it would be compatible. A wide range of data analysis operations entails a similarity detection algorithm between user data. For instance clustering on big data groups together objects based on the heuristic that similar objects are likely to be put under the same cluster. Similarity decisions are important for numerous applications such as: online social networks, recommendations systems and behavioral advertisement. In this paper we propose a mechanism that protects user privacy and preserves data similarity results although encrypted. We analyze the security of the scheme and we further demonstrate its correctness and feasibility through a real life experiment where "personality traits" by users are collected for a 4square application.**

Keywords*—*Information security, privacy, data analysis, similarity detection

I. Introduction

Most of today's ICT applications tend to leverage user information more and more to achieve better content delivery. In particular recommendation systems collect data about users and their interactions with their environment in order to deliver the most appropriate and personalized content to them. The leveraged information spanning users' social relations and personal interests consist of highly sensitive data and hence raises the problem of privacy; a naive solution on the aforementioned problem could be to encrypt data before analyzing them. This would not solve the problem as operations after encryption would not be feasible. A more suitable solution could be to encrypt data homomorphically thus statistical properties on data after encryption are preserved. Even though this solution seems approachable the current homomorphic encryption schemes fall short of giving a solution for a global analysis system applied to some large scale dataset. Moreover anonymization techniques do not offer the appropriate security guarantees for individual data privacy and also have been vulnerable to attacks [1].

One of the basic building blocks in the vast majority of data analysis scenarios is similarity detection. By analyzing users' dataset, a recommendation engine can discover similar profiles and thus recommend a newly arrived user some content that was already consumed by other existing "similar users". Online advertisers sought to increase their revenues by inspecting the online behavior of users. That implies an outsourcing of personal sensitive information by online retailers to the advertisers.

The aforementioned applications imply a privacy violation risk. Since the input to the data analysis operations is personal sensitive private information and operations performed over them violate user privacy. As such, users and companies either tend not to submit their data for further analysis to untrusted parties or they give limited access on it due to individual privacy violation risks [1, 2, 3, 4]. Radical solutions include a restriction either on the available data analysis operations from the analyzer perspective or an outsource of aggregate information instead of individual data. But this will degrade very much the accuracy on data analysis and also this method is not always feasible.

We analyze a well-known similarity detection algorithm, namely the cosine similarity and combine it with some obfuscation mechanisms in order to achieve a privacy preserving similarity computation solution.

In this paper we present a privacy preserving protocol for similarity detection. Cosine similarity can recognize similar vectors based on the formed angle between the vectors.This new privacy preserving mechanism first maps users' data into vectors and applies some geometrical transformations that on the one hand preserve the angle between any pair of vectors and assures the confidentiality of the content of their coordinates. The accuracy of the proposed solution is then evaluated with the study on users' personality characteristics.

In section II we present related work in the area of privacy preserving data analysis. In III there is the problem description. We give our solution in section IV and in section V we argue for the security of the scheme. The evaluation of solution with real world data is analyzed in VI.

II. Related Work

We proceed into a taxonomy of previous solutions in the area of privacy preserving data analysis. We start with more generic solutions and we further describe previous work in the context of specific privacy preserving similarity detection algorithms for clustering.

Data perturbation Several techniques have been proposed in order to obfuscate data such that when users submit their data to the data analyzer individual data privacy is being protected but specific data mining algorithms can be applied on it. Privacy preserving data mining by adding noise on data has been first proposed in [5, 6]. The solution has been proposed

for privacy preserving decision trees as a solution to derive association rules from databases. In [7] the authors proposed geometrical transformation for data clustering. Transformation though are data dependent and do not scale for multidimensional data.

Anonymization Data anonymization asks for unlinkability on data records and users. K-anonymity [8] has been proposed as a solution to protect the release of data to an untrusted party such that the personal private information for each data record cannot be distinguished from $k-1$ other users. Suppression and generalization are two techniques to achieve k-anonymity. By generalization [9] specific attributes are generalized in order to protect user anonymity. For instance instead of releasing the exact data of birth only the month and the year is released. With suppression [10] no data is released. Solutions for data anonymity imply an information loss through out the described techniques and operation after the release of the data are inconsistent.

Data separation In [11] cryptographic tools are being used to protect user data privacy when the id3 tree is constructed for association rules. The id3 tree is a widely known technique for data classification. The categorical data of a set of records is being constructed by choosing the attributes than containing the higher information gain. Information gain is expressed as conditional entropy and the problem of id3 construction is approximated by finding the attributes that information gain is maximized. The authors assume that data are split horizontally, thus the data analyzer in order to compute the conditional entropy of two users should separately and privately obtain the data from both. It turns out that information gain for an attribute between two users is expressed as $(u_1+u_2)\cdot\log(u_1+u_2)$. The problem has been addressed as a secure multi-party computation of this expression for two users.

Privacy preserving data classification on horizontally partitioned data has been addressed in [12, 13] as well. The solution is based on a privacy preserving protocol for sum computation based on randomization and privacy preserving union set computation. Those two functionalities can securely be used by an untrusted party to infer the global confidence of an attribute in order to infer the association rules that will classify the data. In [14] privacy preserving clustering on vertically partitioned data is addressed by submitting only the similarities on objects and not the real data. However how the users are computing the similarities while at the same time preserving their privacy is not clearly addressed. Vaidya et al. tackle this issue by constructing a protocol for secure dot product computation without the use of a trusted party. However the communication cost for computing all the dot products between users is high [15]

Our Contributions As opposed to previous solutions we propose a scheme that is data independent and assures higher level of privacy. Previous solutions that are based on geometrical transformations do not scale for multidimensional data [7] and also there is no concrete security analysis with respect to the leakages of the protocol for example. We did not tackle our similarity problem with respect to data anonymization as anonymization does not fully assure data confidentiality. Moreover, data separation techniques in which data are split in between different sites are not always a real world scenario in which each user holds its data in its entire form.

III. Problem Statement

A. Similarity and privacy

We assume a set of n users. Each user $\mathcal{U}_i$ holds its personal sensitive private data $\mathcal{D}_i$. An untrusted data analyzer $\mathcal{A}$ seeks to obtain $\mathcal{D}_i$ from each user $\mathcal{U}_i$, where $0 \leq i \leq \mathcal{N}$. We consider each $\mathcal{D}_i$ as a multidimensional vector of size m: $\mathcal{D}_i = (d_1, d_2, d_3, \cdots, d_m)$. After the data collection, $\mathcal{A}$ is applying a similarity detection algorithm $\mathcal{F}$ over any pair of data vectors in order to identify similarities between them in order to further form clusters. During the detection of similarities in between data the privacy of users should not be compromised. As such we are looking for an obfuscation mechanism $\phi : \mathbb{R}^m \rightarrow \mathbb{R}^m$ such that for any two vectors $\mathbf{x}, \mathbf{y}$:

$$\mathcal{F}(D_i, D_j) = \mathcal{F}(\phi(D_i), \phi(Dj))$$

where ϕ will preserve the privacy of individual data and at the same time similarity detection through cosine computation will be compatible.

B. Cosine similarity

Cosine similarity is a widely used distance metric for numerical data. Cosine similarity[16] depicts the geometrical similarity of two objects in an Euclidean space by measuring the angle θ formed by their vector representation in a n dimensional Euclidean space. The dot product $< \mathbf{a} \cdot \mathbf{b} >$ of two vectors **a,b** is $< \mathbf{a} \cdot \mathbf{b} >= ||a|| \cdot ||b|| \cos\theta$, where $||a|| = \sqrt{\sum_{i=0}^{n} a_i{}^2}$ is the norm of vector **a** and a_i denotes the coefficients of this vector. Thus,

$$\cos\theta = \frac{< \mathbf{a} \cdot \mathbf{b} >}{||a|| \cdot ||b||}$$

and the more similar the data the closer the angle between their corresponding vectors is and the closer to 1 their cosine. The cosine similarity is our similarity detection function $\mathcal{F}$.

C. Correctness and Privacy

Definition 1 (Privacy Preserving Data Analysis(PPDA)) *In a Privacy preserving data analysis scheme a set of n users $\mathcal{U}_i$ are perturbating their data and afterwards the data are sent to the data analyzer $\mathcal{A}$ for analysis. PPDA consists of 2 polynomial time algorithms $PPDA =$ Encrypt,Analyze:*

Encrypt$(\mathcal{D}_i) \rightarrow \bar{\mathcal{D}}_i$: *It takes as input user data and it outputs the encryption of it.*

Analyze$(\bar{\mathcal{D}}_i, \bar{\mathcal{D}}_j) \rightarrow \mathcal{F}(\bar{\mathcal{D}}_i, \bar{\mathcal{D}}_j)$: *It takes as input two encrypted data vectors and it outputs the result of a data analysis algorithm $\mathcal{F}(\bar{\mathcal{D}}_i, \bar{\mathcal{D}}_j)$.*

Definition 2 (Correctness) *A PPDA scheme is correct if for all pairwise combinations of data $\mathcal{D}_i$, $\mathcal{D}_j$ the analyzer executes* Analyze(Encrypt$(\mathcal{D}_i)$) *and it obtains $\mathcal{F}(\bar{\mathcal{D}}_i, \bar{\mathcal{D}}_j) = \mathcal{F}(\mathcal{D}_i, \mathcal{D}_j)$,*

Definition 3 (Confidentiality) *Let $\Upsilon =$ (Encrypt, Analyze) be a PPDA scheme. Υ is defined as confidential if any adversary cannot recover $\mathcal{D}_i$ from $\bar{\mathcal{D}}_i$*

Intuitively, the security guarantee we require from a PPDA scheme is that given encrypted vectors $\bar{\mathcal{D}}_i$ an adversary cannot learn any information about the plaintext $\mathcal{D}_i$ although it may learn the outpu of the function $\mathcal{F}$.

IV. SOLUTION

A. Idea of Solution

The idea of the solution is to apply some transformations to original vectors which on the one hand preserve the angle between any pair of them and on the other hand assure privacy. Since rotation in a two dimensional space preserves angles, we apply this transformation to two-dimension vectors named as sub-vectors which originate from the data vector. Additionally, these sub-vectors are further randomly scaled and thus obfuscated while still not having an impact on the angle.

The reason why rotation and scaling are combined is that random scaling alone raises some security problems. Indeed, if only random scaling is applied then an adversary can discover whether the coordinates of that vector are similar or not. Hence thanks to the rotation, the adversary cannot discover similarities between one vector's coordinates. The mapping of vectors into subvectors also decreases the probability of discovering the original vector since the scaling factor differs from subvector to subvector.

B. Preliminaries

Vector scaling

Vector scaling with a scaling factor s is defined by a multiplication operation between the vector v and the identity matrix S in which the main diagonal has been substituted with the scale factor s.

$$v \cdot S = v \cdot \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

Vector Rotation Vector rotation with an angle θ is defined by a matrix multiplication between the vector v and the rotation matrix $\mathcal{R}_\theta$: $v \cdot R = v \cdot \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$

C. Protocol description

We now describe the details of the protocol with respect to Definition 1.

Encryption During the encryption phase each user $\mathcal{U}_i$ holds a vector $\mathcal{D}_i = (d_1, d_2, d_3, \ldots, d_m)$ of size m. It generates subvectors of 2 dimensions $d_i^{(k,l)} = \begin{pmatrix} d_k \\ d_l \end{pmatrix}$. If m is odd then $(m+1)/2$ are constructed, otherwise if m is even then we have $m/2$ subvectors. In general we have $\lceil m/2 \rceil$ subvectors. Afterwards each user choses a random scaling factor for each subvector and it scales each subvector $d_i^{(k,l)}$ with the random scaling factor s_i: $S_i^j = s_i^j \cdot d_i^{(k,l)}$, if k and l are new coefficients of the subvector. That is, if any of the coefficients of the subvector $d_i^{(k,l)}$ has been previously selected to form a vector then the old random scale factor s_i must be used for $d_i^{(k,l)}$. Then the intermediate vector $\mathcal{S}_i$ is further rotated with a rotation matrix $\mathcal{R}_\theta$, where θ is the rotation angle: $\bar{d}_i^{k,l} = S_i^j \cdot \mathcal{R}_\theta = S_i^j \cdot \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$.

In the end each user $\mathcal{U}_i$ sends $\bar{\mathcal{D}}_i = (\bar{d}_i^{(1,2)}, \bar{d}_i^{(2,3)}, \cdots, \bar{d}_i^{(k,l)}), \forall k, l \in [0 \cdots m] s.t : \|\{k,l\}\| = m$ to the data analyzer $\mathcal{A}$. In hereafter we will write d_i^j to denote the j^{th} subvector of user $\mathcal{U}_i$ and $\bar{d}_i^j$ for the j^{th} encrypted subvector of user $\mathcal{U}_i$. As such the obfuscated mechanism ϕ consists of random scalings and rotations by an angle θ: $\phi(d_i) = s_i^j \cdot d_i^{k,l} \cdot \mathcal{R}_\theta$.

Analyze The analyzer then performs the similarity detection function $\mathcal{F}$ over the encrypted data: $\forall \mathcal{U}_i, \mathcal{U}_j, i \neq j$:

$$\mathcal{F}(\bar{d}_i, \bar{d_j}) = \begin{cases} \cos(\bar{d}_i^{1,2}, \bar{d}_j^{1,2}) \\ \vdots \\ \cos(\bar{d}_i^{\lceil m/2 \rceil}, \bar{d}_j^{\lceil m/2 \rceil}) \end{cases}$$

D. Correctness

Theorem 2 *The PPDA scheme presented above is correct.*

Proof: It is known that $\cos(a,b) = \frac{<a \cdot b>}{\|a\| \cdot \|b\|} = \frac{a^T \cdot b}{\|a\| \cdot \|b\|}$. For the proof of the theorem we need to prove the following three lemmas:

Lemma 1 *The transpose of an orthogonal matrix A, A^T is equal to its inverse A^{-1}*

Proof: It is known that:

$$A \cdot A^{-1} = I_A \tag{1}$$

where I_A it's the identity matrix of A. Also we obtain:

$$A \cdot A^T = \begin{bmatrix} A_{1,1}^T \cdot A_{1,1} & \cdots & A_{1,m}^T \cdot A_{1,m} \\ A_{n,1}^T \cdot A_{n,1} & \cdots & A_{n,m}^T \cdot A_{n,m} \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \vdots & \vdots \\ 0 & \cdots & 1 \end{bmatrix} = I_A \tag{2}$$

From (1), (2) we have that for any orthogonal matrix A, $A^T = A^{-1}$ ■

Lemma 2 *The multiplication two vectors a, b with a rotation matrix $\mathcal{R}$ preserves its cosine similarity.*

Proof: $\cos(Ra, Rb) = \frac{<Ra \cdot Rb>}{\|Ra\| \cdot \|Rb\|} = \frac{(Ra)^T \cdot Rb}{\|Ra\| \cdot \|Rb\|} = \frac{a^T R^T \cdot Rb}{\|a\| \cdot \|b\|} = \frac{a^T R^{-1} \cdot Rb}{\|Ra\| \cdot \|Rb\|} = \frac{a^T \cdot b}{\|a\| \cdot \|b\|} = \cos(a,b)$ where $\|Ra\| = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \|a\|$ and $\|Rb\| = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \|b\|$ ■

Lemma 3 *The random scaling of two vectors a, b with different random scaling factors r_1 and r_2 preserves its cosine similarity*

Proof: $\cos(r_1 a, r_2 b) = \frac{<r_1 a \cdot r_2 b>}{\|r_1 a\| \cdot \|r_2 b\|} = \frac{(r_1 a)^T \cdot r_2 b}{r_1 \|a\| \cdot r_2 \|b\|} = \frac{r_1 a^T \cdot r_2 b}{r_1 \|a\| \cdot r_2 \|b\|} = \frac{a^T \cdot b}{\|a\| \cdot \|b\|} = \cos(a,b)$ ■

From lemma 1,2 and 3 we have that multiplication of a random vector and random scaling is a correct encryption mechanism. The proof of lemma 2 is based on lemma 1: the rotation matrix $\mathcal{R}$ is orthogonal and as such $\mathcal{R}^{-1} = \mathcal{R}^T$. Furthermore the rotation doesn't change the vector norms.

■

V. Security

Theorem 1 *The PPDA scheme presented above is secure according to definition 3.*

Proof: The security of the scheme is based on the randomness of the scale factor $\mathcal{S}_i$ and on the rotation matrix $\mathcal{R}_\theta$. The data analyzer cannot recover the original vector $\mathcal{D}_i$ of a user $\mathcal{U}$ unless it performs brute force guesses for the scaling factor and the Rotation matrix. ■

We observe security leakages when the obfuscated mechanism does not entail both random scalings and rotations. If each user only selects random scaling as the encryption mechanism then an attacker by obtaining a good guess for a coefficient of a user's vector it can recover the specific two dimensional vector by computing the inverse of the guessed element and multiplying it by the encrypted coefficient.

On the other hand, thanks to rotations the aforementioned problem is mitigated but the following one is appearing if used alone: if two users with secret vectors $\mathcal{D}_i$, $\mathcal{D}_j$ respectively have the same value at the same position of their vectors then only by encrypting with a rotation matrix $\mathcal{R}_\theta$ of angle θ , the corresponding encrypted vectors would have the same value at this position. This violates the security definition 3. So in order for the cosine similarity to be securely preserved after the encryption of the vectors, both random scaling and rotations must by applied.

To be more precise with our security analysis, we consider two categories of adversaries: we define external adversaries as data analyzers and internal adversaries as users themselves.

External adversaries Data analyzers do not know the rotation matrix and as such the aforementioned attacks cannot happen as long as the angle θ of the rotation matrix is kept secret. The data analyzers cannot identify common values in a specific data vector because the rotation with an unknown angle adds an additional security level for the vectors.

Internal adversaries We consider as internal adversaries the users that know the rotation matrix $\mathcal{R}_\theta$. In such a scenario the user that has a good guess for the coefficient of another user can reveal only the coefficients that are involved in a common random scaling factor per vector. That is if $\mathcal{U}_i$ has 5 coefficients and it defines cosine similarity in between the ((1,2),(3,4),(1,5)) coefficients then an adversary with a good guess for the first coefficient can recover only the second and fifth coefficient and nothing more, since for the second subvector the user would choose different a random scaling factor.

VI. Evaluation

We demonstrate the correctness of our protocol through an experimental evaluation setup. We obtain data originating from a personality experiment. We first cluster the data based on cosine similarity using a well known clustering algorithm. The same clustering algorithm is further applied over the encryption of the same data using ϕ which as already described combines rotation and random scaling. We proceed into an analysis of the data and next on the clustering algorithms that we use.

A. Data Set

The dataset contains an extract of the results from the Foursquare Personality Experiment[1] which uses the mobile social network Foursquare[2] combined with a standard personality test to allow the link between personality (as defined by the five-factor model [17]) and the places that people visit to be examined. To the best of our knowledge, this is the first time that it has been possible to correlate personality with place on such a granular level.

When accessing the experiment, users sign in using their Foursquare account, allowing us to access the list of venues which they have 'checked in' to on the Foursquare service. We access only this list, storing the venues that the user has been to and the number of times they have visited each venue, but without accessing or storing the information about the individual checkins - we do not store *when* each visit to the venue occurred, nor the order in which venues were visited. Once users have accessed the system they then take a 44-item personality test [18, 19], revealing their five-factor personality scores. The five-factor model gives each person a score between 1 and 5 for each of the five personality traits: Openness, Conscientiousness, Extraversion, Agreeableness and Neuroticism.

The users participating in the study are a self-selecting group comprised of 173 people who both use Foursquare online location based tagging system and are willing to take part in a personality-based experiment.

B. Clustering

Clustering algorithms seek to group similar objects together. Similarity is measured with a distance metric which in our case is cosine similarity. Hierarchical clustering is a widely known approach for clustering. It constructs a binary tree of clustering objects that successively are merged under the same cluster with respect to the linkage metric. The linkage metric links clusters and objects together. It acts as an intergroup similarity measure. Two most popular linkage metrics are the *complete* metric which defines the maximum similarity between two objects as a verification to whether or not one object would be merged under the same cluster with another one and the *single* metric in which the minimum similarity is treated as the intergroup similarity metric. At the first step of the algorithm each object belongs to each own cluster. Then all the possible pairwise similarities between objects with respect to the defined distance metric are defined. Afterwards the algorithm iteratively merge clusters with respect to the linkage metric until there would be one cluster with the all the examined objects.

C. Simulation Setup and Results

We apply the hierarchical algorithm over the personality dataset with the complete linkage metric and based on cosine similarity.

The data consists of 173 different 5 dimensional vectors describing users' personality with respect to the 5 personality traits as previously described. We did not include venue

[1] http://www.cs.cf.ac.uk/recognition/foursqexp
[2] http://www.foursquare.com

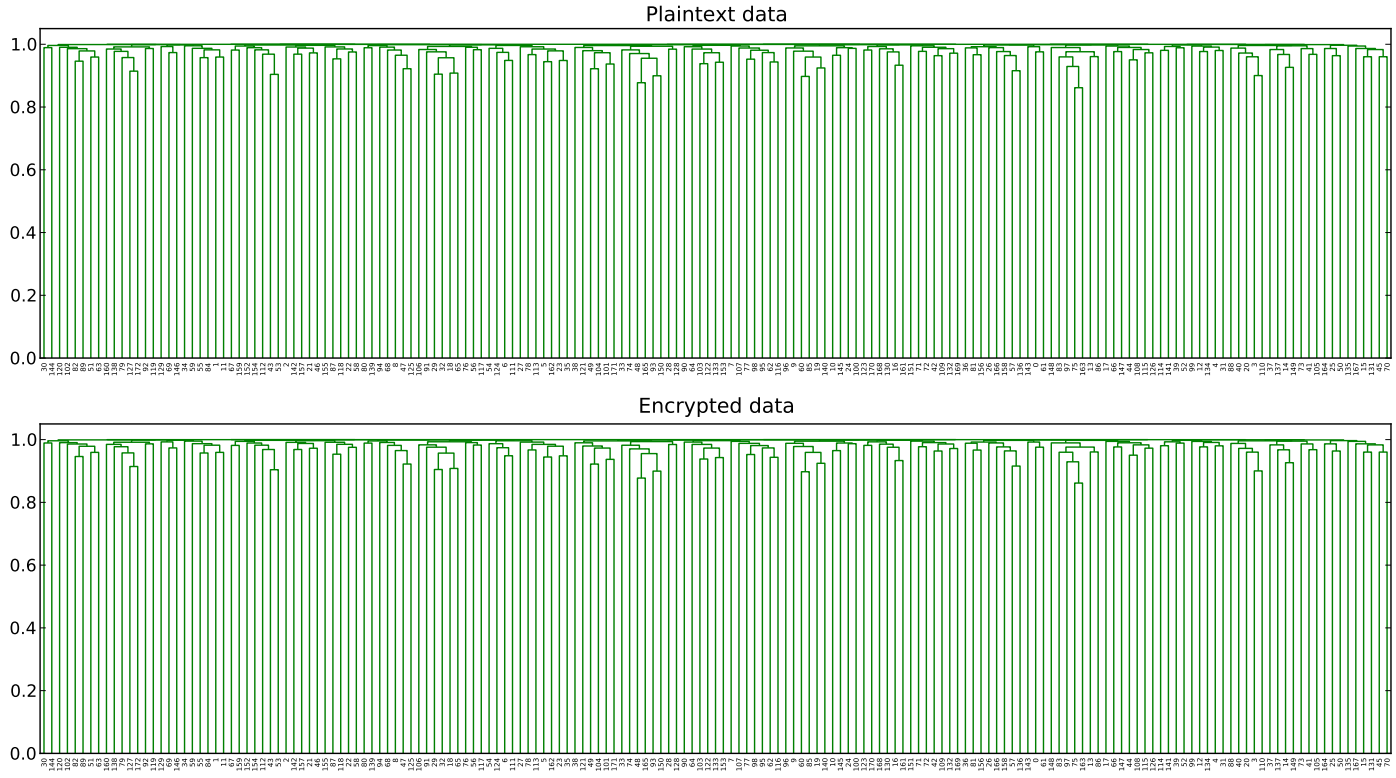

Fig. 1. Hierarchical Clustering

visits frequency since we believe that personality traits are considered much more sensitive data compared to location information and that users would be more interested in hiding such information. We consider similarity on 3 subvectors per user data: the subvectors are constructed with the $(1^{st}, 2^{nd}), (3^{rd}, 4^{th})$ and $(1^{st}, 5^{th})$ coordinates of the original vector respectively. Any pairwise subvector could be have chosen such that the union of the set of subvectors entail all the coefficients. The main similarity metric is computed as the average of the similarities between subvectors.

In order to protect their privacy, every user chooses a random scaling factor per two dimensions. After the random scaling process users apply the rotation operation to their partially obfuscated subvectors.

In figure 1 we plot the two dendograms of hierarchical clustering before and after the operation ϕ applied on data. The horizontal axis corresponds to cluster indexes that are formed by the algorithm and the vertical axis to the linkage similarity based on cosines. Clusters are connected with upside-down U-shaped lines. The clusters are exactly the same due to the correctness of the algorithm as has been previously proved. All the cosines between all the binomial coefficients of 2 over 173 elements has been computed. That results into a set of 14878 distances. For the linkage function we chose the *complete* option. Thus, two clusters will be merged together according to the maximum distance between their elements. Results shown in figure 1 demonstrate the correctness of our protocol: Geometrical transformation on data based on random scaling and rotation is compatible with cosine similarity for clustering and in addition preserves individuals' privacy.

D. Discussion

In our experiments we use as a single point of similarity an aggregate output of each three per user similarities. This is the average of cosine similarities. As such during the clustering the similarity between points depicts similarities between the averages. We could have demonstrated three different scenarios during the clustering process one for each subvector in order to check the correctness of our obfuscation mechanism but since this has been demonstrated once the other experiments wouldn't add extra knowledge to us. We also want to state that the aggregate function should not always be used for every case. This would imply an inconsistency on correctness since many inputs could evaluate the same average similarity. Suppose for instance that data consist of user interests on m items and for each user n similarities per two dimensions are computed. Then a single aggregate function on user n similarities might group together during clustering dissimilar objects that average the same similarities but on different inputs.

VII. Conclusion

The interplay between data analysis and privacy is emerging rapidly. Researchers from machine learning area have highlighted the merit of data analysis operations. However this exposure of personal sensitive data, facilitates privacy violations. Adversaries by gaining access to personal information can learn the real identity of users and overcome data legal regulations and restrictions. That postulates a mechanism that would shield individual data confidentiality. This would not be of significant value since data encryption to protect data confidentiality is more mature and well analyzed than 30 years before. The tricky approach is to allow operations on data by the security mechanism while at the same time personal sensitive information is not exposure to third parties.

In this paper we have presented a mechanism for privacy preserving clustering that is based on geometrical transformation of objects. Data are encrypted appropriately such that operations with respect to cosine similarity detection are compatible. We proceed into an analysis of the security risks of each operation and we conclude that the most secure way is a combination of random scalings and rotations. The rotation angle even if its known to the users it can only be used to reveal some coordinates of the vector only if the user has a good guess for one of the coordinates. Still this weakness is not of crucial importance for external adversaries (data analyzer) since they don't know the rotation matrix. Without scaling and only with rotation, similarities on the same position coordinates are possible to occur by both internal and external adversaries. This is mitigated by a random scaling factor, which is different per user and per subvectors with no common coefficients. We proceed into an experimental evaluation of a scheme in order to demonstrate its correctness. Personality traits have been obtained by 173 users and identical clustering results have been observed before and after the obfuscation proposed solution.

Acknowledgement This research has been funded by RECOGNITION grant 257756, an EC - FP7 Future Emerging Technologies project concerning Self-Awareness in Autonomic Systems.

References

[1] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, ser. SP '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 111–125.

[2] I. Rouf, H. Mustafa, M. Xu, W. Xu, R. Miller, and M. Gruteser, "Neighborhood watch: security and privacy analysis of automatic meter reading systems," in *Proceedings of the 2012 ACM conference on Computer and communications security*, ser. CCS '12. New York, NY, USA: ACM, 2012, pp. 462–473. [Online]. Available: http://doi.acm.org/10.1145/2382196.2382246

[3] M. Lisovich, D. Mulligan, and S. Wicker, "Inferring personal information from demand-response systems," *Security Privacy, IEEE*, vol. 8, no. 1, pp. 11–20, Jan.-Feb.

[4] S. McLaughlin, P. McDaniel, and W. Aiello, "Protecting consumer privacy from electric load monitoring," in *Proceedings of the 18th ACM conference on Computer and communications security*, ser. CCS '11. New York, NY, USA: ACM, 2011, pp. 87–98. [Online]. Available: http://doi.acm.org/10.1145/2046707.2046720

[5] R. Agrawal and R. Srikant, "Privacy-preserving data mining," 2000.

[6] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ser. PODS '01. New York, NY, USA: ACM, 2001, pp. 247–255.

[7] S. R. M. Oliveira and O. R. Zaïane, "Privacy preserving clustering by data transformation," *JIDM*, vol. 1, no. 1, pp. 37–52, 2010.

[8] L. Sweeney, "k-anonymity: a model for protecting privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, Oct. 2002.

[9] V. S. Iyengar, "Transforming data to satisfy privacy constraints," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 279–288.

[10] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," Tech. Rep., 1998.

[11] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in *CRYPTO*, 2000, pp. 36–54.

[12] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," *IEEE Trans. on Knowl. and Data Eng.*, vol. 16, no. 9, pp. 1026–1037, Sep. 2004. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2004.45

[13] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu, "Tools for privacy preserving distributed data mining," *ACM SIGKDD Explorations*, vol. 4, p. 2003, 2003.

[14] S. R. M. Oliveira and et al., "Privacy-preserving clustering by object similarity-based representation and dimensionality reduction transformation," in *IN PROC. OF THE WORKSHOP ON PRIVACY AND SECURITY ASPECTS OF DATA MINING (PSADM04) IN CONJUNCTION WITH THE FOURTH IEEE INTERNATIONAL CONFERENCE ON DATA MINING (ICDM04*, 2004, pp. 21–30.

[15] B. Goethals, S. Laur, H. Lipmaa, and T. Mielikäinen, "On private scalar product computation for privacy-preserving data mining," in *Proceedings of the 7th international conference on Information Security and Cryptology*, ser. ICISC'04. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 104–120.

[16] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press, 2008.

[17] L. R. Goldberg, "An Alternative "Description of Personality": the Big-Five Factor Structure." *Journal of Personality and Social Psychology*, vol. 59, no. 6, pp. 1216–29, Dec. 1990.

[18] O. P. John, L. P. Naumann, and C. J. Soto, "Paradigm shift to the integrative big five trait taxonomy," *Handbook of personality: Theory and research*, vol. 3, pp. 114–158, 2008.

[19] O. P. John, E. M. Donahue, and R. L. Kentle, "The big five inventoryversions 4a and 54," *Berkeley: University of California, Berkeley, Institute of Personality and Social Research*, 1991.