LETTER

Fourier Magnitude-Based Privacy-Preserving Clustering on Time-Series Data*

Hea-Suk KIM[†], ***Nonmember and*** **Yang-Sae MOON**[†a)], ***Member***

SUMMARY Privacy-preserving clustering (*PPC* in short) is important in publishing sensitive time-series data. Previous PPC solutions, however, have a problem of not preserving *distance orders* or incurring privacy breach. To solve this problem, we propose a new PPC approach that exploits Fourier magnitudes of time-series. Our magnitude-based method does not cause privacy breach even though its techniques or related parameters are publicly revealed. Using magnitudes only, however, incurs the distance order problem, and we thus present magnitude selection strategies to preserve as many Euclidean distance orders as possible. Through extensive experiments, we showcase the superiority of our magnitude-based approach.

key words: *time-series data, clustering, privacy-preserving, Fourier magnitude, distance order*

1. Introduction

The aim of privacy-preserving data mining (PPDM) [1] algorithms is to extract relevant knowledge from a large amount of data while protecting at the same time sensitive information. In this paper we address the problem of privacy-preserving clustering (*PPC* in short) on sensitive time-series data [5], [6]. Typical examples are as follows: (1) drivers do not wish to disclose their exact speed recorded in the vehicle monitoring system, but they still allow clustering of driving patterns [8]; (2) patients with heart disease do not want to disclose their private electrocardiogram (ECG) data, but they still allow clustering of patient ECG data.

PPC solutions can be classified into (1) secure multiparty computation(SMC)-based solutions [3] and (2) distortion-based solutions [1], [6], [7]. In this paper we focus on the distortion-based approach, in which the data providers distort original time-series and publish the distorted time-series to third parties. A simple distortion-based solution is random data perturbation [1], [8], but it does not preserve distance orders and shows bad clustering accuracy [6]. Other distortion methods [6], [7] were proposed to overcome this problem, but they may cause privacy breach if their distorting techniques or parameters are publicly revealed.

To solve the privacy breach problem, we propose a new PPC approach that exploits Fourier magnitudes of time-series. In this *magnitude-based* approach, the data providers publish a few Fourier magnitudes of a time-series to third parties, and the third parties perform clustering by using those magnitudes only. Without the corresponding phase information, attackers cannot reconstruct the original time-series from the Fourier magnitudes. Thus, our magnitude-based method does not cause privacy breach even though its distorting techniques or related parameters are publicly revealed. However, it incurs the distance order problem since it uses Fourier magnitudes only instead of Fourier coefficients. To discuss this problem, we present a notion of *distance-order preservation*, which represents how many time-series preserve their relative Euclidean distance orders before and after the distortion.

To preserve as many Euclidean distance orders as possible, we present magnitude selection strategies. The first strategy, called *sequential selection*, is simply choosing the first few magnitudes as in the *coefficient-based method* [6]. We then propose two greedy strategies that select magnitudes based on the given sample time-series. The first greedy strategy is *local selection*, which first computes the degree of distance-order preservation for each individual magnitude and then greedily selects the magnitudes having larger degrees. The second greedy strategy is *global selection*, which first selects the first magnitude using the local selection and then repeatedly selects the next one by investigating which one is the best in preserving distance orders if it is combined with the previously selected magnitudes.

Experimental results show that our magnitude-based method is comparable to the coefficient-based method both in distance-order preservation and clustering accuracy, and the global selection is superior to the local selection as well as the sequential selection.

2. Related Work

PPC solutions can be categorized into (1) SMC-based and (2) distortion-based solutions. SMC-based solutions [3] provide secure mining algorithms for the distributed environment, and they are orthogonal to our PPC problem. On the other hand, distortion-based solutions [1], [6]–[8] are generally used for the centralized environment that consists of multiple data providers and one or more third parties. A simple distortion-based solution is using random data perturbation [1], [8], but it may incur bad clustering accuracy [6]. Geometric transformation and rotation perturbation [7] can be used for distorting a set of time-series, but these solutions

Manuscript received October 13, 2009.
Manuscript revised February 3, 2010.

[†] The authors are with the Department of Computer Science, Kangwon National University, Korea.

[*] This study was supported by the Research Grant from Kangwon National University.

a) E-mail: ysmoon@kangwon.ac.kr (corresponding author)

DOI: 10.1587/transinf.E93.D.1648

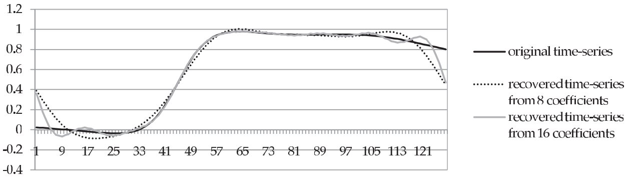


Fig. 1 Reconstructed time-series from Fourier coefficients.

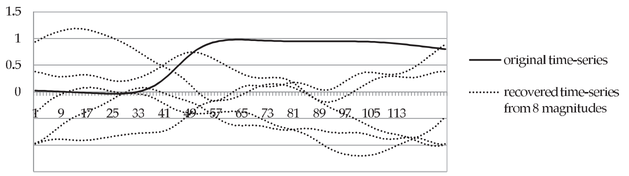


Fig. 2 Reconstructed time-series from Fourier magnitudes.

may cause privacy breach at the worse case when their transforming or perturbing parameters are disclosed to attackers.

Recently, Mukherjee and Chen [6] proposed the *coefficient-based* method that exploited a few Fourier coefficients instead of a whole time-series. This method, however, may cause privacy breach if coefficient positions are revealed. Figure 1 shows an example of reconstructing the original time-series, which is a Chlorine time-series [5] of length 128. As shown in the figure, the reconstructed time-series from 8 or 16 coefficients are very similar to the original time-series.

3. Fourier Magnitude-Based Approach

Discrete Fourier transform (DFT) converts a time-series to a sequence of frequencies, each of which is a function of magnitude and phase [6]. For a given time-series $X(=\{x_0,\ldots,x_{n-1}\})$, we can obtain its coefficient sequence $X^c(=\{x^c_0,\ldots,x^c_{n-1}\})$ and magnitude sequence $X^m(=\{x^m_0,\ldots,x^m_{n-1}\})$ as $x^c_k = \frac{1}{n}\sum_{i=0}^{n-1} x_i e^{\frac{-2\pi ik}{n}\cdot j}$ and $x^m_k = \|x^c_k\|$, where $k = 0,\ldots,n-1$.

Our magnitude-based method exploits Fourier magnitudes x^m_k's. That is, it selects a few magnitudes from a time-series and publishes those magnitudes to third parties. If magnitudes and phases are given, we can get their coefficients, and vice versa. Without phases, however, the exact coefficients cannot be obtained from the given magnitudes. It means that we cannot recover the time-series without the phase information. Thus, the magnitude-based method does not incur the privacy breach problem.

Figure 2 shows an original time-series and its example time-series reconstructed from Fourier magnitudes for the same Chlorine time-series of Fig. 1. As shown in the figure, many different time-series can be reconstructed due to missing phases, and attackers cannot choose a specific one since they do not know the exact phases. This explains why our magnitude-based method does not cause privacy breach.

Distance orders represent the relative orders among distances between time-series. As the distance measure, we use the Euclidean distance since it is one of the most widely used distance functions [6]–[8]. In general, preserving both the absolute distances between time-series and their privacy is difficult. However, preserving the relative orders among distances is enough for providing higher accuracy in many mining algorithms [4]. Based on this observation, we use the notion of distance order preservation for assuring clustering accuracy.

Definition 1: Suppose time-series O, A, and B form a record $[O,(A,B)]$, and they are distorted to time-series O^d, A^d, and B^d, respectively. We say that the *distance order of* (A,B) *with respect to* O is *preserved*, or simply the *distance order of* $[O,(A,B)]$ is *preserved*, if $D(O,A) \leq D(O,B) \Rightarrow D(O^d,A^d) \leq D(O^d,B^d)$ or $D(O,A) \geq D(O,B) \Rightarrow D(O^d,A^d) \geq D(O^d,B^d)$ holds, where $D(\cdot,\cdot)$ is the Euclidean distance function.

In Definition 1, the distance order of $[O,(A,B)]$ is the same as that of $[O,(B,A)]$, and we thus use the notation $[O,(A,B)]$ only in the paper. The reason why we use the form of $[O,(A,B)]$ is that many clustering algorithms use the operation of comparing one point with other points. For example, the k-means algorithm compares a representative point of each cluster with other points; and the hierarchical algorithm compares a leaf point (or a group of points) with other neighbor points (or groups of points). Thus, preserving distance orders of $[O,(A,B)]$'s is important in preserving clustering accuracy in many algorithms. To exploit the distance order preservation as a metric of preserving clustering accuracy, we now quantify its measure as follows:

Definition 2: Suppose $\mathbb{S}$ is a set of $[O,(A,B)]$'s of time-series. We define the *degree of distance order preservation* of $\mathbb{S}$, simply denoted by *ddop* of $\mathbb{S}$, as the ratio of the number of distance order preserved $[O,(A,B)]$'s to the number of all $[O,(A,B)]$'s in $\mathbb{S}$. That is, $ddop = \frac{\text{the number of distance order preserved } [O,(A,B)]\text{'s in } \mathbb{S}}{\text{the number of all } [O,(A,B)]\text{'s in } \mathbb{S}}$.

To preserve as many distance orders as possible, we use the *ddop* in selecting Fourier magnitudes.

4. Magnitude Selection Strategies

In this section we propose selection strategies that choose f magnitudes among $n\,(\gg f)$ magnitudes obtained from a time-series of length n.

The simplest strategy is to select the first f magnitudes from total n magnitudes. We call it the *sequential selection*. The sequential selection, however, has a problem of not considering *ddop* in selecting magnitudes. It just assumes that most energy is concentrated on the first few coefficients, but this assumption is not true for many types of time-series. According to the experiment [6], energy concentration varies depending on types of time-series. To solve this problem, we need to investigate which magnitudes closely preserve distance orders.

We next propose two greedy selection strategies: *local selection* and *global selection*. The local selection first computes *ddop* of each individual magnitude and then greedily selects the magnitudes having larger degrees. Algorithm 1 shows the local selection. In the procedure *LocalIndex* (), we first randomly choose sample $[O, (A, B)]$'s from the given database (Line 1). We then compute *ddop* for each magnitude (Line 2). We finally store indexes of f magnitudes having large *ddop* (Line 3). Those stored f indexes are used in the main algorithm to extract f magnitudes from a time-series.

Algorithm 2 shows the global selection that selects the first magnitude using the local selection and then repeatedly selects the next one by considering the previously selected magnitudes. In the procedure *GlobalIndex* (), we first randomly choose sample $[O, (A, B)]$'s (Line 1). We then repeatedly select the next j-th magnitude based on the previously selected 1-st to $(j-1)$-th magnitudes (Lines 2-8). Like in the local selection, those stored f indexes are used in the main algorithm to extract f magnitudes. The global selection is a little bit complex, but it will be better than the local selection in preserving distance orders since it uses a global approach instead of a local approach.

We now analyze the number of records $[O, (A, B)]$'s in two greedy strategies. If we let the number of time-series be m, the number of cases of selecting O becomes m. After then, for each O we need to consider $\binom{m-1}{2}$ cases, i.e., $\frac{(m-1)(m-2)}{2}$ cases, since we select two time-series for A and B from the rest $(m-1)$ time-series. As a result, for the given m time-series, we can generate total $\frac{m(m-1)(m-2)}{2}$ records. This large number makes it difficult to use all the $[O, (A, B)]$'s in the strategies, and we thus reduce the number of $[O, (A, B)]$'s through the random sampling. To determine the sample size s, we use Eq. (1) developed by Cochran [2]. In Eq. (1), Z, D, and N are the upper $100 \cdot (a/2)$ percentile point of standard normal distribution for desired confidence level $(1 - a)$, the desired level of precision, and the population size, respectively.

$$s = \frac{s_0}{1 + \frac{(s_0-1)}{N}}, \text{ where } s_0 = \frac{Z^2 \cdot 0.5^2}{D^2}. \tag{1}$$

Algorithm 1 *LocalSelection* $(S = \{s_0, \ldots, s_{n-1}\}, f)$

```
1: if LocalIndex () is not called yet then LocalIndex ();
2: Extract $n$ Fourier magnitudes from $S$ through DFT;
3: Select $f$ magnitudes from $n$ ones by the order of $Lidx[j]$;
```

Procedure *LocalIndex* ()

```
1: Randomly choose a set $\mathbb{S}$ of sample $[O, (A, B)]$'s from the database;
2: for $i := 1$ to $n$ do Compute $ddop[i]$ from $\mathbb{S}$ using $i^{th}$ mag.;
3: for $j := 1$ to $f$ do $Lidx[j]$ := the index of $j^{th}$ largest $ddop[i]$;
```

Algorithm 2 *GlobalSelection* $(S = \{s_0, \ldots, s_{n-1}\}, f)$

```
1: if GlobalIndex () is not called yet then GlobalIndex ();
2: Extract $n$ Fourier magnitudes from $S$ through DFT;
3: Select $f$ magnitudes from $n$ ones by the order of $Gidx[j]$;
```

Procedure *GlobalIndex* ()

```
1: Randomly choose a set $\mathbb{S}$ of sample $[O, (A, B)]$'s from the database;
2: for $j := 1$ to $f$ do
3:   for $i := 1$ to $n$ do
4:     if $i$ is already in $Gidx[1..(j-1)]$ then $ddop[i] := -1$;
5:     else compute $ddop[i]$ from $\mathbb{S}$ using $Gidx[1..(j-1)]^{ths}$
           and $i^{th}$ magnitudes;
6:   end-for
7:   $Gidx[j]$ := the index of the largest $ddop[i]$;
8: end-for
```

Cochran's Eq. (1) is widely used in survey research to obtain the sample size for a large population. In our case, the number of possible $[O, (A, B)]$'s is very large, and we thus Eq. (1) to calculate the sample size with the given confidence level and interval. In the experiment we use 95% of confidence level and ±1.0% of confidence interval.

We also briefly analyze the computation overhead of greedy strategies. We only focus on the time complexity required to select magnitudes since performing DFT is commonly required in all strategies. First, *LocalIndex* () of the local selection computes $O(n)$ *ddop*'s for each $[O, (A, B)]$, and its time complexity is $O(nk)$ if k is the number of $[O, (A, B)]$'s (see Line 2 of Algorithm 1). Second, *GlobalIndex* () of the global selection computes $O(nk)$ *ddop*'s for each magnitude, and its complexity is $O(nfk)$ since we select f magnitudes (see Line 5 in Algorithm 2). In summary, for the given k sample $[O, (A, B)]$'s, *LocalIndex* () and *GlobalIndex* () incur $O(nk)$ and $O(nfk)$ additional computation overhead compared with the sequential selection.

5. Experimental Evaluation

We used UCR time-series data sets [5]. For each data set, we measured the *ddop* and the actual clustering accuracy. We experimented four privacy-preserving methods: the coefficient-based method [6], the sequential selection, the local selection, and the global selection. For simplicity, we denoted these methods by *CB*, *SS*, *LS* and *GS*, respectively.

To evaluate *ddop*, we first choose 4,858 to 9,604 samples of $[O, (A, B)]$ for each data set since the data sets consist of 28 to 6,136 time-series. We then obtain *ddop* for each privacy-preserving method. Figure 3 shows the relative trend of *ddop*'s that compare our methods with the previous CB [6] on different data sets. As shown in the figure, our SS, LS, and GS are generally worse than CB in preserving distance orders. This is an obvious result because our approach uses only magnitudes without the phase information. However, we note that the difference between CB and GS is very small, i.e., merely 14% on the average. It means that the clustering accuracy is not much worse even though we use magnitudes only. We also note that the results of GS are better than those of LS and SS.

We next discuss the actual accuracy preservation. As the accuracy measure, we use *F-measure* [6]. In general, the higher F-measure means the more accurate results. After executing the k-means algorithm for each of CB, SS, LS, and GS, we obtain their F-measures, respectively. Figure 4 shows the relative trend of F-measures on different

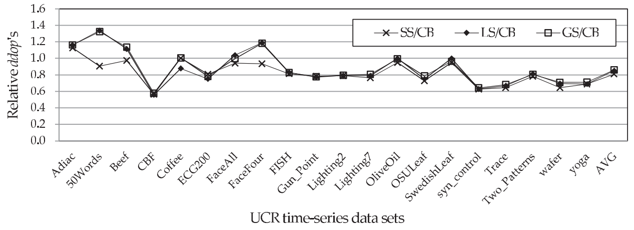


Fig. 3 Relative *ddop*'s on different time-series data sets.

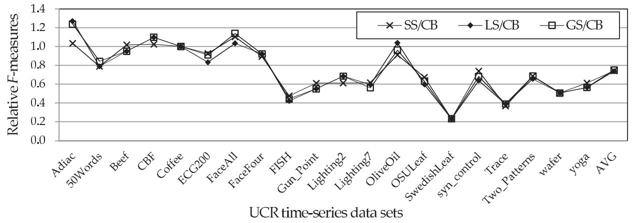
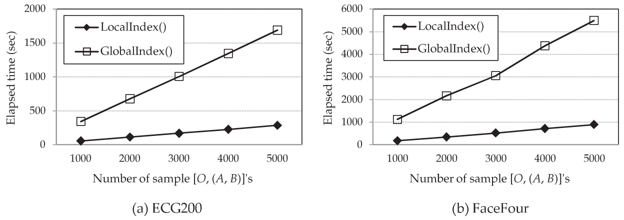


Fig. 4 Relative F-measures on different time-series data sets.



(a) ECG200

(b) FaceFour

Fig. 5 The elapsed times of *LocalIndex* () and *GlobalIndex* ().

data sets[†]. As shown in the figure, the relative F-measures are below 0.5 in FISH, SwedishLeaf, and Trace; in contrast, for many data sets including Adiac and CBF, the relative F-measures are above 0.8. It means that our methods provide a good result for many data sets, but does not for some data sets. Showing the F-measure difference by the characteristics of data sets is an interesting issue, and we leave it as the future work. In Fig. 4, the relative F-measures are in between 1.27 (Adiac) and 0.23 (SwedishLeaf), and their average is 0.75. That is, the F-measure difference between CB and our methods is 25% on the average. Also, GS is still superior to SS and LS.

To investigate the computation overhead of *LocalIndex* () and *GlobalIndex* (), we show their elapsed times in Fig. 5 by varying the number k of sample $[O, (A, B)]$'s. We use ECG200 and FaceFour data sets [5]. We note that their elapsed times linearly increase as the number of samples increases. This is because their time complexities are $O(nk)$ and $O(nfk)$, respectively, as we discussed in Sect. 4. This index selection process, however, can be seen as preprocessing steps, and we can ignore this overhead when we consider the whole process of publishing time-series.

6. Conclusions

In this paper we proposed the Fourier magnitude-based PPC on time-series data, which did not cause the privacy breach problem. We also presented a notion of *distance order preservation* and proposed magnitude selection strategies. We empirically showed that our magnitude-based approach could be comparable to the previous coefficient-based approach in clustering accuracy. These results indicate that our approach provides a higher degree of privacy-preservation as well as a comparable clustering accuracy.

[†] Different clustering methods provide different clustering results, and their accuracy is subjective. However, we use the same clustering method for the both original and distorted data sets. Thus, in our case, the smaller difference in clustering results of two data sets, the higher accuracy. This is why we use the *F-measure* to compare the clustering results of the original and distorted data sets.

References

[1] R. Agrawal and R. Srikant, “Privacy-preserving data mining,” Proc. Int'l Conf. on Management of Data, ACM SIGMOD, Dallas, Texas, pp.439–450, May 2000.

[2] W.G. Cochran, Sampling Techniques, 2nd ed., John Wiley & Sons, 1963.

[3] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M.Y. Zhu, “Tools for privacy preserving distributed data mining,” SIGKDD Explorations, vol.4, no.2, pp.28–34, Dec. 2002.

[4] A. Inan, M. Kantarcioglu, and E. Bertino, “Using anonymized data for classification,” Proc. 25th Int'l Conf. on Data Engineering, pp.429–440, Shanghai, China, April 2009.

[5] E.J. Keogh, X. Xi, L. Wei, and C.A. Ratanamahatana, The UCR Time Series for Classification/Clustering (http://www.cs.ucr.edu/~eamonn/time_series_data).

[6] S. Mukherjee, Z. Chen, and A. Gangopadhyay, “A privacy-preserving technique for euclidean distance-based mining algorithms using Fourier-related transforms,” VLDB Journal, vol.15, no.4, pp.293–315, 2006.

[7] S.R.M. Oliveira and O.R. Zanane, “Privacy-preserving clustering by data transformation,” Brazilian Symposium on Databases, pp.304–318, Amazonas, Brazil, Oct. 2003.

[8] S.L.F. Papadimitriou, G. Kollios, and P.S. Yu, “Time series compressibility and privacy,” Proc. 33rd Int'l Conf. on Very Large Data Bases, pp.459–470, Vienna, Austria, Sept. 2007.