Clustering Unsynchronized Time Series Subsequences with Phase Shift Weighted Spherical *k*-means Algorithm

Tiantian Yang
School of Digital Media, Jiangnan University, Wuxi, China
Email: taliayoung@163.com

Jun Wang
School of Digital Media, Jiangnan University, Wuxi, China
Email: wangjun_sytu@hotmail.com

***Abstract*—Time series have become an important class of temporal data objects in our daily life while clustering analysis is an effective tool in the fields of data mining. However, the validity of clustering time series subsequences has been thrown into doubts recently by Keogh et al. In this work, we review this problem and propose the phase shift weighted spherical *k*-means algorithm (PS-WSKM in abbreviation) for clustering unsynchronized time series. In PS-WSKM, the phase shift procedure is introduced into the clustering process so that the phase problem is solved effectively. Meanwhile, the subsequences weights are assigned to subsequences to make the algorithm more robust. Experimental results on ECG datasets show that our approach is effective for the problem of unsynchronized time series subsequences clustering, which makes contributions to a wide range of applications, particularly in intelligent healthcare.**

***Index Terms*—time series clustering, unsynchronized time series subsequences, phase shift weighted spherical *k*-means algorithm**

I. Introduction

Time series data is an important kind of temporal data, which has initiated various research and development attempts in the fields of data mining. Clustering is one of the most frequently used methods in the fields of machine learning [1-4, 11-16]. Recently, time series clustering has aroused great interest among researchers. However, Keogh et al. declared that clustering time series subsequence is meaningless [5]. In [5], E. Keogh et al. conducted several clustering experiments with some of the commonly used clustering algorithms, such as k-means, hierarchical, EM, SOMs and other variants of k-Means, and found that the center subsequences obtained by the clustering algorithms are seriously distorted. This work invalidated the contributions of dozens of previously published papers. The reason for this phenomenon comes from the fact that the phase problem is not effectively handled in the clustering process and each subsequence has the equal contributions to the center subsequences. As a result, the final center subsequences are seriously distorted and the clustering algorithms lose effectiveness.

In order to further study the problems posed by Keogh and link up the clustering techniques with the time series applications, in this work, we integrate the principle of phase shift into the clustering process and then propose a novel clustering algorithm, i.e., phase shift weighted spherical k-means algorithm (PS-WSKM in abbreviation). We aim at providing an effective and robust approach for the problem of clustering times series subsequences.

The rest of the paper is organized as follows. In section 2, we discuss the principle of unit vectors. In section 3, we propose the phase shift weighted spherical k-means algorithm PS-WSKM and investigate its properties. Section 3 reports the experimental results.

II. Properties of Unit Vectors

A. Definitions

Definition 1: Subsequence: Given a time series T of length m, a subsequence C of T is a sampling of length $s \le m$ of contiguous position from T, that is, $C = t_p, \ldots, t_{p+n-1}$ for $1 \le p \le m\text{-}s+1$.

Definition 2: Optimal phase shift: Given subsequence X and C of length s, the optimal phase shift τ_{opt} of X relative to C is defined as

$$\tau_{opt}(X,C) = \arg\min_{\tau} d\left(C, X^{(\tau)}\right) = \arg\min_{\tau} \sqrt{\sum_{i=1}^{s}\left(c_{i+\tau} - x_i\right)^2}$$

τ=1, 2, …, s

in which $X^{(\tau)}$ denotes the subsequence resulting from shifting X with phase shift τ. If $i+\tau > s$, the subsequence wraps around to its end and uses the value at $i+\tau\text{-}s$..

B. Some Properties of Unit Vectors

In this section, we will show some important properties of the unit vectors. Suppose we are given n unit vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\boldsymbol{R}^s$ and their weights $w_1, \ldots, w_n$, w_i>0, i = 1, 2, …, n. The weighted mean vector of the unit vectors can be computed as:

doi:10.4304/jcp.9.5.1103-1108

$$\mathbf{m} = \frac{\sum_{i=1}^{n} w_i \mathbf{x}_i}{\sum_{i=1}^{n} w_i} \tag{1}$$

Note that the weights of $\mathbf{x}_i$'s should be nonnegative real numbers. In addition, the mean vector **m** need not have a unit norm. One may capture its direction via the following calculation:

$$\mathbf{c} = \frac{\mathbf{m}}{\|\mathbf{m}\|} \tag{2}$$

The weighted mean vector **c** with unit norm computed by Eq.(2) may be thought as the vector that is closest in cosine similarity (in an average sense) to all the unit vectors in dataset X. This provides us a solid theoretical foundation for the proposed algorithm.

It is a hard work to cluster time series subsequences that are not strictly synchronized and many solutions have been proposed. One straightforward solution is to adjust the phase while the algorithm runs so that the cosine similarity between two subsequences is maximized. This procedure requires finding the optimal phase shift between two subsequences. However, for two subsequences with length s, brute force search for the optimal phase shift between them involves $O(s^2)$ computation complexity. This will become the speed bottleneck when the algorithm runs on large datasets. In the following, we will show that, using the convolution theorem, the time complexity to find the optimal phase shift τ_{opt} between **x** and **c** is $O(s\log s)$, which provides us an efficient approach for finding the optimal phase shift.

Let **x** and **y** be two normalized vectors whose length equals to 1 in the Euclidean space, r_{xy} be the consine similarity and d_{xy} be Euclidean distance between **x** and **y**, respectively, then we could have the following relationship:

$$1 - r_{xy} = \frac{1}{2} {d_{xy}}^2 \tag{3}$$

On the other hand, the cosine similarity between two subsequences $X^{(\tau)} = \{x_1, x_2, \ldots, x_s\}$ and $C = \{c_1, c_2, \ldots, c_s\}$ can be computed as follows:

$$r_{x,c}(\tau) = \sum_{i=0}^{s} x_i c_{i+\tau} \tag{4}$$

where s is the length of the two subsequences. Obviously, the cosine similarity between two subsequences is similar in nature to the convolution of two discrete series. Whereas convolution involves reversing the series, then shifting it and multiplying by another one, the cosine similarity defined in Eq.(4) only involves shifting it and multiplying, without the reversing step.

Theorem 1: Assuming X and C are subsequences of length s, the time complexity to find the optimal phase shift τ_{opt} between X and C is $O(s\log s)$.

Proof: According to the convolution theorem, under suitable conditions, the Fourier transform of a convolution is the pointwise product of Fourier transforms. Let F denotes the Fourier operator and F^{-1} as the inverse Fourier transform, $F\{X\}$ and $F\{C\}$ are the Fourier transforms of time series X and C, the consine similarity defined in (4) can be computed as follows:

$$r_{x,c}(\tau) = F^{-1}\{F\{X\} \cdot F^*\{C\}\}(\tau)$$

in which $F^*(C)$ denotes the complex conjugate of the Fourier transform of C. With Fast Fourier Transforms (FFT), $r_{x,c}(\tau)$ s for different values of τs can be computed together, thus the time complexity of computing $r_{x,c}(\tau)$ s for different τ values is identical to that of FFT, which is $O(s\log s)$. On the other hand, according to the relationship between consine similarity and Euclidean distance revealed in Eq.(3), the optimal phase τ_{opt} of X relative to C can be computed with

$$\tau_{opt}(X, C) = \arg\max_{\tau} r_{x,c}(\tau)$$

Thus the Theorem is proved. €

The properties discussed above provide us useful theoretical tools to develop an effective clustering algorithm for clustering unsynchronized time series data.

III. Phase Shift Weighted Spherical k-Means Algorithm

Let X be the set of subsequences with the length of s, $\mathrm{B} = [\beta_1, \beta_2, \ldots, \beta_n]$ be the weight vector, $V=[v_1, \ldots, v_s]$ be the center subsequence, $d(x_i, v_j)$ be the Euclidean distance between subsequence x_i and v_j, and r be a parameter for the weight β_i. For the jth cluster, the learning criterion can be defined as follows:

$$J_j\left(X, v_j, \mathrm{B}\right) = \sum_{i=1}^{n} \frac{1}{\beta_i^r} d^2\left(x_i, v_j\right) + \varepsilon \sum_{i=1}^{n} \frac{1}{\beta_i^r} \tag{5a}$$

$$\beta_i > 0, \ \sum_{i=1}^{n} \beta_i = 1 \tag{5b}$$

The penalty term $\varepsilon \sum_{i=1}^{n} \frac{1}{\beta_i^r}$ in Eq.(5) is introduced into the objective function to avoid zero division errors. We can minimize Eq.(5) by iteratively solving the following optimization problems:

Problem P_1: Fix $\mathrm{B} = \hat{\mathrm{B}}$ and $X = \hat{X}$, solve the reduced problem $J(\hat{X}, v, \hat{\mathrm{B}})$;

Problem P_2: Fix $V = \hat{V}$ and $X = \hat{X}$, solve the reduced problem $J(\hat{X}, \hat{v}_j, \mathrm{B})$;

Problem P_3: Fix $\mathrm{B} = \hat{\mathrm{B}}$ and $V = \hat{V}$, solve the reduced problem $J(X, \hat{v}_j, \hat{\mathrm{B}})$.

To solve problem P_1 and find the center that makes Eq.(5) minimized, the cluster center v_j can be computed as follows:

$$v_j = \frac{\sum_{i=1}^{n} \frac{x_i}{\beta_i^r}}{\sum_{i=1}^{n} \frac{1}{\beta_i^r}}, \ l = 1, \ldots, s \tag{6}$$

To solve problem P_2 and find the fuzzy feature weight B that makes the objective function minimized under constraints Eq.(5b), we use Lagrange multipliers. By using Lagrange multipliers, we have

$$\beta_i = \frac{\left(d^2\left(x_i, v_j\right)+\varepsilon\right)^{\frac{1}{r+1}}}{\sum_{i}^{n}\left(d^2\left(x_i, v_j\right)+\varepsilon\right)^{\frac{1}{r+1}}} \tag{7}$$

Unlike problem P_1 and problem P_2, problem P_3 is defined on a discrete domain and the function $J(X, v_j, \hat{\mathrm{B}})$ is uncontinuous. Thus the partial derivatives cannot be used here. However, we can shift the phase for each subsequence x_i so that $d^2\left(x_i, \hat{v}_j\right)$ is minimized. In this way, the phase problem involved in time series data is solved in this step. For fixed $\hat{\mathrm{B}}$ and $\hat{v}_j$,

$$J_j(\hat{X}, \hat{v}_j, \mathrm{B}) = \sum_{i=1}^{n}\frac{1}{\hat{\beta}_i^r}\left(d^2\left(x_i, \hat{v}_j\right)+\varepsilon\right) \tag{8}$$

is also minimized.

In the following, we will extend the above procedure to the case of multiple clusters. Let $W = [w_1, w_2, \ldots, w_k]$ be the weights of each cluster, $\mathrm{B} = [\beta_{ji}]_{c\times n}$ be the weights of each subsequence in each cluster, V be center subsequences of clusters, the objective function can be formulated as follows:

$$Q_U\left(X, V, W, \mathrm{B}\right) = \sum_{j=1}^{k}\frac{1}{w_j^{\alpha}}J_j\left(X, v_j, \mathrm{B}\right) = \sum_{j=1}^{k}\frac{1}{w_j^{\alpha}}\sum_{i\in C_j}\frac{1}{\beta_{ji}^{r}}\left(d^2\left(x_i, v_j\right)+\varepsilon\right) \tag{9a}$$

subject to

$$\sum_{i\in C_j}\beta_{ji} = 1, \beta_{ji} > 0,\ j = 1, 2, \ldots, k \tag{9b}$$

$$w_j > 0,\ \sum_{j=1}^{k} w_j = 1 \tag{9c}$$

It can be illustrated as the combination of several clusters with objective functions Eq.(5) weighted by $\frac{1}{w_j^{\alpha}}$. Its minimization implies that the weighted quadratic sum of the within-cluster distance should be minimized. We have the following iteration equations:

$$\beta_{ji} = \begin{cases} \dfrac{\left(\dfrac{1}{w_j^{\alpha}}\left(d^2\left(x_i, v_j\right)+\varepsilon\right)\right)^{\frac{1}{r+1}}}{\sum_{i\in C_j}\left(\dfrac{1}{w_j^{\alpha}}\left(d^2\left(x_i, v_j\right)+\varepsilon\right)\right)^{\frac{1}{r+1}}} & i\in C_j, \\ 0 & i\notin C_j \end{cases} \tag{10a}$$

$j = 1, 2, \ldots, k$

$$w_j = \frac{\left(\sum_{i\in C_j}\dfrac{1}{\beta_{ji}^{r}}\left(d^2\left(x_i, v_j\right)+\varepsilon\right)\right)^{\frac{1}{\alpha+1}}}{\sum_{j=1}^{k}\left(\sum_{i\in C_j}\dfrac{1}{\beta_{ji}^{r}}\left(d^2\left(x_i, v_j\right)+\varepsilon\right)\right)^{\frac{1}{\alpha+1}}},\ j = 1, 2, \ldots, k \tag{10b}$$

$$v_j = \frac{\sum_{i\in C_j}\dfrac{1}{\beta_{ji}^{r}}x_i}{\sum_{i\in C_j}\dfrac{1}{\beta_{ji}^{r}}},\ j = 1, 2, \ldots, k \tag{10c}$$

Now we discuss the phase adjustment and cluster assignment procedure. For each subsequence, we find the cluster center which has the maximal similarity with it and the corresponding optimal phase shift τ_{opt} required to produce this maximal similarity. Then we assign the subsequence to this cluster and adjust the phase by the optimal phase shift τ_{opt}.

We state the process of the algorithm PS-WSKM as follows:

Algorithm: PS-WSKM

Input: periodic time series T, period s, cluster number k

Output: the vector of the subsequences weights B, partition matrix U, cluster weights W and the center subsequence V.

Step 1: Segment the time series T with period s to obtain the set of the subsequences $X^{(0)}$.

Step 2: Randomly generate a set of initial cluster weights $W^{(0)}$ and subsequences weights in each cluster $B^{(0)}$. Randomly choose k subsequences as initial centers $V^{(0)}$. Set $t = 0$.

Step 3: Let $\hat{\mathrm{B}} = \mathrm{B}^{(t+1)}$, $\hat{W} = W^{(t+1)}$ and $\hat{V} = V^{(t+1)}$. For each subsequence $X_i^{(t)}$, find the center subsequence $V_l^{(t+1)}$ which has the maximal similarity with it and the corresponding optimal phase shift τ_{opt} required to produce this similarity. Assign the subsequence to this cluster with $u_{li}^{(t+1)} = 1$ and $u_{ji}^{(t+1)} = 0$, for $j = 1, \ldots, k$ and $j \neq l$. Adjust the phase of $X_i^{(t)}$ with $X_i^{(t+1)} = \{X_i^{(t)}\}^{(\tau opt)}$.

Step 4: Let $\hat{X} = X^{(t+1)}$, $\hat{W} = W^{(t)}$ and $\hat{v} = v^{(t)}$, compute $B^{(t+1)}$ using equation Eq.(10a).

Step 5: Let $\hat{W} = W^{(t)}$, $\hat{\mathrm{B}} = \mathrm{B}^{(t+1)}$ and $\hat{X} = X^{(t+1)}$, compute $v^{(t+1)}$ using equation Eq.(10c).

Step 6: Let $\hat{X} = X^{(t+1)}$, $\hat{\mathrm{B}} = \mathrm{B}^{(t+1)}$ and $\hat{v} = v^{(t+1)}$, compute $W^{(t+1)}$ using equation Eq.(10b).

Step 7: Compute $Q_{U(t+1)}(X^{(t+1)}, v^{(t+1)}, W^{(t+1)}, B^{(t+1)})$ using equation Eq.(10a). If $|\ Q_{U(t+1)}(X^{(t+1)}, v^{(t+1)}, W^{(t+1)}, B^{(t+1)}) - Q_{U(t)}(X^{(t)}, v^{(t)}, W^{(t)}, B^{(t)})| < \varepsilon$, output $(U^{(t+1)}, v^{(t+1)}, W^{(t+1)}, B^{(t+1)})$ and stop. Otherwise, $t = t+1$ and goto Step 3.

The newly introduced weighting vector W is used to determine discords clusters. According to Eq.(10b), the cluster weight w_j reflects the weighted within-cluster distance of cluster j. If most of the subsequences within cluster j are close to their cluster center and the size of cluster j is much smaller than others, the w_j will have small value. This shows that the cluster with small size and small within-cluster distance is more likely to be a discord cluster.

IV. EXPERIMENTAL STUDY

Electrocardiograms (ECG) are typical periodic time series we encounter in our daily life. Up to now, there are many test datasets available from Internet. In this work, we test our algorithm using ECG datasets.

In our experiments, we split ECGs into subsequences without overlapping with each other, each of which contained one cycle. Then we normalized the subsequences to make each of them has the length of 1. The resulted subsequences were the input of our algorithm.

A. Robustness Experiments on PS-WSKM

This experiment was conducted to verify the robustness of the proposed algorithm PS-WSKM when the cluster number is set to 1 and compared it with Pk-means (k=1) [6].

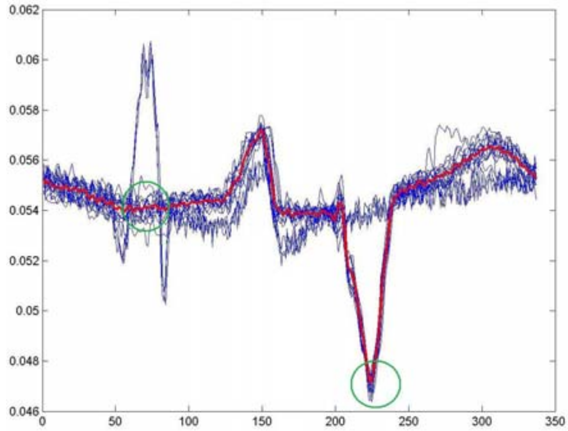
In order to intuitively show how the discords affect the center subsequence found by the algorithm, we created a synthetic dataset by introducing known number of discords into the *xmitdb_x108_0* dataset. In our dataset, the discords were more than 10% percent such that the negative effects of the discords to the whole dataset cannot be ignored. We run the proposed PS-WSKM (k=1) and Pk-means on it respectively and plot the resulting subsequences and the center subsequences in Fig 1(a) and Fig 1(c), respectively. Meanwhile, we also run the both algorithms on the original *xmitdb_x108_0* dataset without discords involved and the results are plotted in Fig 1(b) and Fig 1(d). In Fig 1, the subsequences after phase adjustment by both algorithms are drawn in blue and their center subsequences are highlighted in red. The primary differences between the center subsequences are highlighted by green circles. One point needs to be mentioned here is that the phases of the four resulted center subsequences in Fig 1 are not synchronized. This is due to the random initialization of the algorithms. However, this does not prevent us coming to our conclusions.

Comparing Fig 1(a) with Fig 1(b), we can observe that the center subsequence generated by PS-WSM changes slightly after discords are introduced. This implies that PS-WSKM is robust to the discords in the dataset. In contrast, the center subsequence generated by Pk-means is sensitive to the discords in the dataset when we compare Fig 1(c) with Fig 1(d). The robustness of PS-WSKM comes from the introduction of the weight vector B. Recall the center update equation in Eq.(6) and we can easily infer that the discords with larger β_is have less contribution to the center subsequence computation. However, for Pk-means, each subsequence makes identical contribution to the center subsequence such that the center subsequence is influenced greatly if enough discords are introduced.

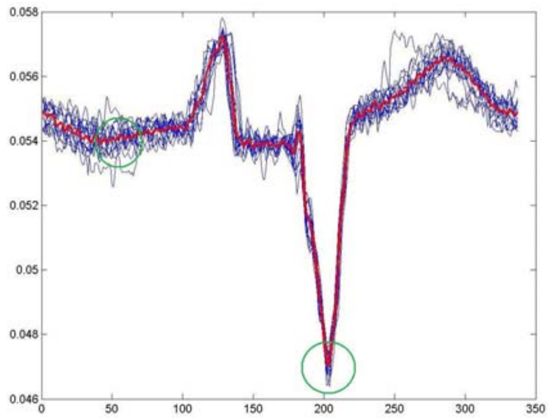
B. Multiple-cluster Datasets Experiments on PS-WSKM

To evaluate the performance of PS-WSKM, we conducted two experiments on the dataset containing multiple clusters.

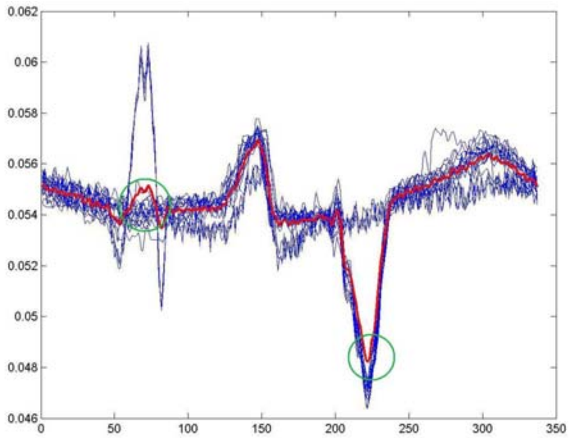
The first part of PS-WSKM experiment was conducted on MIT-BIH datasets labeled 102, 104, 106 and 108. Fig 2 shows the results obtained by PS-WSKM running on these datasets when the optimal k values are properly assumed. As can be seen from Fig 2, PS-WSKM can discover the center subsequence of each cluster successfully, which agree with the clustering results obtained by Pk-means.



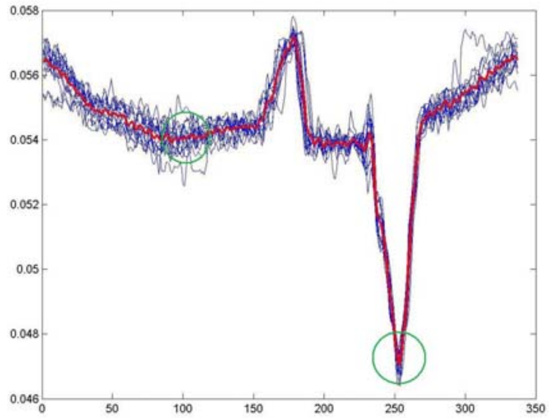
(a)



(b)



(c)



(d)

Fig.1 the center subsequences obtained by different algorithms

(a) center subsequences resulted from PS-WSKM (k=1) running on the dataset with discords introduced (b) center subsequences resulted from PS-WSM running on the original *xmitdb_x108_0* dataset (c) center subsequences resulted from Pk-means running on the dataset with discords introduced (d) center subsequences resulted from Pk-means running on the original *xmitdb_x108_0* dataset

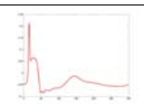
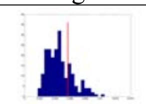
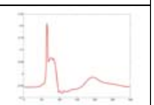
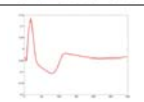
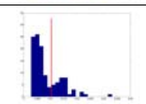
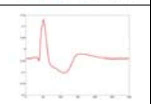
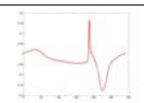
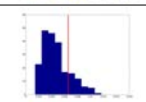
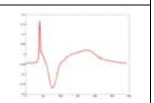
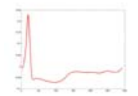
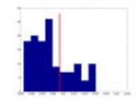
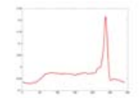
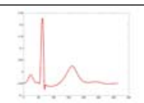
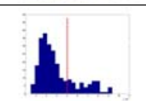
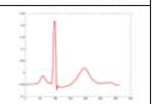
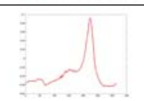
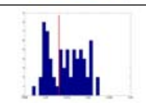
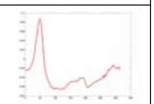
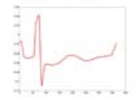
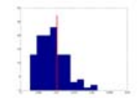
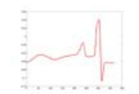
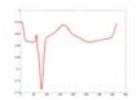
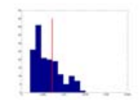
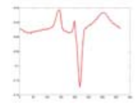
	PS-WSKM (r=1, α=3)			Pk-means
	Cluster weight	Center subsequence of each cluster	Distribution of the subsequences weights	Center subsequence of each cluster
102	0.6994			
	0.3006			
104	0.4084			
	0.5916			
106	0.7177			
	0.2823			
108	0.4363			
	0.5637			

Fig.2 Comparison of PS-WSKM and Pk-means results on several MIT-BIH datasets

IV Conclusions

The effectiveness of clustering time series subsequences has been thrown into doubts by Keogh et al. recently. In this work, we investigate this problem by introducing subsequences' weights and a phase shift procedure into the clustering process. We proposed PS-WSKM to cluster time series subsequences so that the unsynchronized time series subsequences can be clustered effectively.

On one hand, this work has opened up new opportunities for applying clustering techniques to the unsynchronized time series subsequences clustering tasks. On the other hand, the introduction of subsequences weights makes the algorithm more robust. However, we only use the raw time series data. In our future work, we will integrate the dimensionality reduction or feature extraction techniques into the clustering process. In this way, the learning algorithm will become more smart and effective.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants 61300151 and by the Natural Science Foundation of Jiangsu Province under Grant BK20130155.

References

[1] T. W. Liao. Clustering of time series data—a survey, Pattern Recognition, vol.38, no.11, 2005, pp.1857-1874

[2] X. Zhang, J. Liu, Y. Du, et al. A novel clustering method on time series data, Expert Systems with Applications, vol. 38, no. 9, 2011, pp. 11891-11900.

[3] Pierpaolo D'Urso and Elizabeth Ann Maharaj. Autocorrelation-based fuzzy clustering of time series, Fuzzy Sets and Systems, vol.160, no.24, 2009, pp.3565-3589.

[4] E. A. Maharaj and P. D'Urso. Fuzzy clustering of time series in the frequency domain. Information Sciences, vol.181, no.7, 2011, pp.1187-1211

[5] Eamonn Keogh and Jessica Lin. Clustering of time-series subsequences is meaningless: implications for previous and future research, Knowledge and Information Systems, vol.8, no. 2, 2005, pp 154-177.

[6] Umaa Rebbapragada, Pavlos Protopapas and Carla E. Brodley, et al. Finding anomalous periodic time series. Machine Learning, vol.74, no.3, 2009, pp. 281-313.

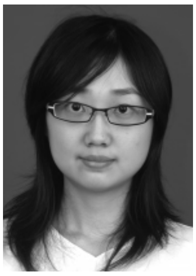
[7] E. Keogh, S. Lonardi, and B. Y. Chiu. Finding Surprising Patterns in a Time Series Database in Linear Time and Space. In Proceedings of the Fifth IEEE International Conference on Data Mining, pp 226-233, 2005

[8] MIT-BIH Arrhythmia Database Directory, http://www.physionet.org/physiobank/database/html/mitdbdir/intro.htm. 2010.

[9] L. Xu, A. Krzyzak, E. Oja. Rival Penalized Competitive learning for clustering analysis, RBF net and curve detection. IEEE Transactions on Neural Networks, vol. 4, no. 4, pp. 636-649, 1993

[10] Z. Bar-Joseph, G. Gerber, D. Gifford, et al. A new approach to analyzing gene expression time series data. In proceedings of the 6th Annual International Conference on Research in Computational Maolecular Biology. Washington, D.C., 2002, pp.39-48.

[11] J. Xie, Shuai Jiang, Weixin Xie, Xinbo Gao. An Efficient Global K-means Clustering Algorithm, Journal of Computers, vol. 6, no. 2, 2011, pp. 271-279
[12] J. Wu, Jie Xia, Jian-ming Chen, Zhi-ming Cui. Moving Object Classification Method Based on SOM and K-means, Journal of Computers, vol.6, no. 8, 2011, pp.1654-1661
[13] T. Li, Yan Chen, Jinsong Zhang. Logistics Service Provider Segmentation Based on Improved FCM Clustering for Mixed Data, Journal of Computers, vol.7, no.11, 2012, pp.2629-2633
[14] Hongfen Jiang, Junfeng Gu, Yijun Liu, et al. Study of Clustering Algorithm based on Fuzzy C-Means and Immunological Partheno Genetic, Journal of Software, vol. 8, no. 1, 2013, pp.134-141.
[15] Hongjie Jia, Shifei Ding, Hong Zhu, et al. A Feature Weighted Spectral Clustering Algorithm Based on Knowledge Entropy. Journal of Software, vol.8, no.5, 2013, pp.1101-1108.
[16] Jiashun Chen, Dechang Pi, Zhipeng Liu. An Insensitivity Fuzzy C-means Clustering Algorithm Based on Penalty Factor, vol.8, no.9, 2013, pp.2379-2384.



Tiantian Yang received her Ph.D. degree from Shanghai Institute of Microsystem and Information Technology. She is currently a lecture at the School of Digital Media, Jiangnan University. Her research interests include pattern recognition, digital image processing and MEMS.

Jun Wang received his Ph.D. degree from the Nanjing University of Science and Technology in January, 2011. He is currently an associate professor at the School of Digital Media, Jiangnan University. He has published nearly 20 papers in international/national authoritative journals. His research interests include pattern recognition, data mining and digital image processing.