Similarity matching of time series based on key point alignment dynamic time warping

Yangzheng Li, Zhigang Chen+, Xiaoheng Deng
School Of Computer Science And Engineering
Central South University
Changsha, China
+ Corresponding author mail: czg@csu.edu.cn

***Abstract*—Similarity measurement is an important basis in time series analysis. Among them, dynamic time warping distance (DTW) is considered to be the most effective distance measurement method. However DTW's huge computational overhead is difficult to meet the application requirements in the era of big data. Previous optimization methods often focus on reducing unnecessary calculation objects and do not involve warping distance calculation itself. After studying many related optimization algorithms, we propose a DTW matching algorithm based on key structure point alignment. By extracting the key structure points of time series and calculating the warping alignment relationship between the key structure points, the constraint range of the cumulative distance matrix of the approximate optimal warping distance from the path is mapped, which greatly reduces the amount of calculation of the distance cumulative matrix, then approximate warping distance can be calculated quickly. The experimental results show that the calculation speed of our method is significantly improved compared with the traditional algorithm in similarity matching, and it also has a good performance in classification accuracy.**

Keywords-time series; data mining; dynamic time warping distance

I. INTRODUCTION

Time series is a concept derived from data mining, which generally refers to an ordered set of the same statistical index values arranged according to their time order. The analysis of time series has made important applications in the fields of finance, medical treatment, meteorology, geology and so on[1-4].

The similarity measurement is an important basis of time series analysis. Comparing and judging the similarities and differences of two groups of time series can further realize the classification and clustering of time series, thus it can be used as an important basis for time series analysis. In related research, distance is usually used as the measure of similarity between time series. The smaller the distance is, the more similar the time series is. Euclidean distance is the most classical time series distance measurement, which is simple and fast to calculate, but the calculation method of point-to-point alignment can not be used for the comparison of unequal time series, nor can it solve the problems of distortion, scaling and drift on the time axis, so it is rarely used in practical application.

Another common classical distance measure, dynamic time warping (DTW)[5], has been proposed for many years and still plays an important role. It aligns each series point through dynamic programming to find the minimum cumulative distance, allows warping alignment in time axis, and solves the limitation of Euclidean distance, but the calculation cost of cumulative distance matrix is large, The time complexity of DTW is $O(n^2)$, which is hard to apply to the time series analysis of big data.

Many other distance measurement algorithms have also been proposed. For example, the symbolic editing distance based on time series (EDR)[6], the longest common subsequence (LCSS)[7], the most similar subsequence (TSW)[8], etc. they are also algorithms based on dynamic programming, which has considerable time complexity compared with DTW. In addition, there are non dynamic programming methods, although the complexity is lower, but the accuracy is often insufficient. The maximum shifting correlation distance[9] has only the time complexity of $O(n)$. It finds the maximum Pearson correlation coefficient of the two time series through the sliding window to obtain the approximate distance. The fragment alignment distance[10] uses the approximate derivative and the number of continuous segments of each segment of the sequence to represent a segment of subsequence, and calculates the distance in the way of approximate diagonal alignment. Since the complex alignment is not considered, the computational complexity is reduced to linear. The fluctuation features distance[11] considers the trend change of time series and calculates the distance by weighting the change value. The MPdist[12] uses the method of matrix representation of the sequence, divides the sequence into multiple subsequence groups, puts forward the closest subsequence pair to form a sequence, and takes a large enough value as the distance result.

However, due to its excellent universality and matching accuracy, DTW is still difficult to replace. In order to solve the computational cost of DTW, researchers have proposed many methods, such as the following boundary distance[13,14], early

This work is supported by the Intelligent software and hardware system of medical process assistant and its application belong to "2030 Innovation Megaprojects" - New Generation Artificial Intelligence (Project no. 2020AAA0109605), and Major special project of Changsha science and technology plan (Project no. kh2103016).
DOI reference number: 10.18293/SEKE2022-021

abandonment[15]. At the ACM SIGKDD conference, Thanawin et al.[16] proposed the UCR suite, which integrates important DTW acceleration methods in the past. These methods try to skip the calculation of part of the warping distance, so as to save the overall calculation time, and do not involve the calculation of the warping distance itself. Another idea is to reduce the dimension of time series, and then calculate the distance by dynamic programming as usual. Pr-DTW[17] method represents the segmented time series by weighting multiple statistical indicators, and calculates the distance after greatly reducing the amount of data. Soft-DTW[18] proposes a method to find the approximate warping path by subdividing the warping distance matrix step by step, which does not need to calculate the complete warping distance matrix.

Based on these studies, this paper proposes a distance measurement algorithm based on key point alignment. The cumulative distance is calculated by finding the near optimal path through the key points, which significantly reduces the computational complexity. The algorithm in this paper is used for the measurement of 1-NN classifier for experimental test. The results show that compared with the traditional algorithm, the algorithm in this paper significantly reduces the time cost and maintains good matching accuracy.

II. PRE KNOWLEDGE

A. Classical dynamic time warping method

Set the time series as $X\{x_1,\ x_2,\ x_3, \ldots\ldots x_m\}$, and $Y\{y_1,\ y_2,\ y_3,\ \ldots\ldots y_n\}$, define the DTW distance between the two time series as *DTW* (*X*, *Y*), and construct an matrix $\boldsymbol{D}$ of size $m \times n$, calculate the value of each matrix element $\boldsymbol{D}[i][j]$ ($i \in [1, m]$, $j \in [1, n]$) according to the following method:

$$\boldsymbol{D}[1][1] = d(1,1) \tag{1}$$

$$\boldsymbol{D}[i][j] = d(i,j) + \min \begin{cases} \boldsymbol{D}[i-1][j] \\ \boldsymbol{D}[i][j-1] \\ \boldsymbol{D}[i-1][j-1] \end{cases} \tag{2}$$

$$d(i,j) = |x_i - y_j| \text{ or } (x_i - y_j)^2 \tag{3}$$

Then $\boldsymbol{D}[m][n]$ is *DTW* (*X*, *Y*). In the process of calculating *DTW* (*X*, *Y*), connect the minimum values of $\boldsymbol{D}[i][j]$ to obtain a path of warping distance accumulation:

$$path = \{(1,1), \ldots\ldots (i,j) \ldots\ldots, (m,n)\} \tag{4}$$

where the two values in each binary represent two elements from two time series, and *i* and *j* represent their time axis positions in their respective series. When *path* can make the cumulative distance function $\sum_{i,j\ in\ path}^{len(path)} d(x_i, y_j)$ gets the minimum value, that is, *DTW* (*X*, *Y*), at this time, the *path* is called the optimal alignment path of these two time series, and each pair of binary in the *path* is called the "alignment" relationship.

B. Constraint range

The calculation of *DTW* (*X*, *Y*) needs to traverse all elements in the calculation matrix. Its complexity is $O(m \times n)$, can be recorded as $O(n^2)$. Global constraint is an idea of optimizing DTW distance calculation, which reduces the amount of calculation by limiting the cumulative range of warping distance. For $X\{x_1,\ x_2,\ x_3, \ldots\ldots x_m\}$ and $Y\{y_1,\ y_2,\ y_3,\ \ldots\ldots y_n\}$, the element that far from the diagonal in the warping distance matrix, such as $\boldsymbol{D}[1][n]$, reflect the alignment between the first element in *X* and the last element in *Y*. This means that time series is aligned with the time axis in a very distorted state.

This alignment usually does not conform to the actual situation. The value on the matrix is often too large to be added to the final result. In fact, this part of the calculation can be omitted. In the methods of Itakura constraint[19] and Sakoe-Chuba constraint[20], as shown in Figure 1, the calculation range is limited near the diagonal of the warping distance matrix, as the dark part in the figure.

However, this global constraint method lacks flexibility. If the alignment path is outside the constraint range, there will be a large error between the calculation result and the optimal distance. When the alignment path is in the constraint range, there is still a large computational overhead.

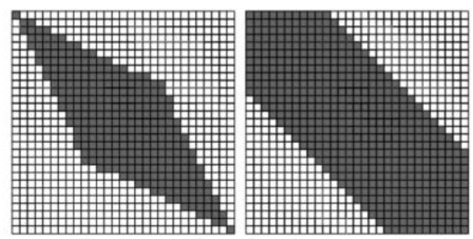

Figure 1. Itakura constraint and Sakoe-Chuba constraint

III. METHOD

In fact, to get the final DTW distance value, only needs to calculated the value on the optimal alignment path in the matrix. Because the value on the path only depends on the minimum of the three candidate cumulative distance values. The problem is that the optimal alignment path cannot be known until the complete warping matrix is calculated, but the near optimal alignment path can be found through some methods. By constraining the calculation range of the waping distance matrix near the near optimal path, a very close DTW distance can be obtained.

A. Find key structural points

Due to the continuity of the time series, a few key structural points in the time series can reflect the approximate trend shape of the whole time series image. Therefore, a set $X'\{(k_1, x_{k1}), (k_2, x_{k2}), (k_3, x_{k3}), \ldots\ldots (k_p, x_{kp})\}$ can be used to represent time series $X\{x_1,\ x_2,\ x_3, \ldots\ldots x_m\}$ approximantely. Where k_i represents the time axis position of each key structure point, *p* is the number of key stuctural points in the time series image, and *p* is much smaller than *m*.

In order to find the key points of the time series, firstly use the PAA[21] method to process the time series data to reduce the noise jitter of the time series image and improve the operation efficiency. Then the extreme points in all time series image points should be screened. The left and right derivatives of the extreme points are opposite, as shown follows:

$$\{x_i | (x_i - x_{i-1})(x_{i+1} - x_i) < 0\} \quad (5)$$

And add those non extreme points x_i with large turning points, as shown follows:

$$\{x_i || arctan(x_i - x_{i-1}) - arctan(x_{i+1} - x_i)| > \gamma\} \quad (6)$$

Where γ is the threshold, its value is π/6 in this paper.

Some time series data may have local jitter, resulting in the aggregation of key points in a small section. In order to ensure that the key points reflect the overall morphological trend of the time series, these points need to be further filtered to reduce local aggregation. When the distance between a newly added key point and the previous key point is lower than the threshold γ_c, it will not be added to the key points set. And γ_c can be calculated as equation (7):

$$\gamma_c = 0.1 * \sqrt{(\max(x) - \min(x))^2 + m^2} \quad (7)$$

In addition, the first and last points of each time series are specified as key points.

Assuming that there are time series $X\{x_1, x_2, x_3, \dots\dots x_m\}$ and $Y\{y_1, y_2, y_3, \dots\dots y_n\}$, $X'\{(k_1, x_{k1}), (k_2, x_{k2}), \dots\dots, (k_i, x_{ki}), \dots\dots (k_p, x_{kp})\}$ and $Y'\{(l_1, y_{l1}), (l_2, y_{l2}), \dots\dots, (l_j, y_{lj}), \dots\dots (l_q, y_{lq})\}$ are obtained after key point filtering. Where k_i and l_j respectively represent the time axis position of key points in the corresponding time series, and p and q are the number of key points in the corresponding time series.

B. Key structural points alignment

Elements of the distance accumulation matrix $\boldsymbol{K}$ of the key point sequence X' and Y' is calculated as follows, and the size is $p \times q$:

$$\boldsymbol{K}[1][1] = dk(1,1) \quad (8)$$

$$\boldsymbol{K}[i][j] = dk(i,j) + \min \begin{cases} \boldsymbol{K}[i-1][j] \\ \boldsymbol{K}[i][j-1] \\ \boldsymbol{K}[i-1][j-1] \end{cases} \quad (9)$$

$$dk(i,j) = |x_{ki} - y_{lj}| * (1 + |k_i/k_p - l_j/l_q|) \quad (10)$$

When calculating the distance accumulation matrix of key points, in addition to calculating the difference of time series elements corresponding to key points, it can be multiplied by the correction value of time axis position to obtain a more reasonable alignment relationship. After calculating the cumulative distance matrix $\boldsymbol{K}$ of key points, the optimal alignment path of key points $path_{kp}$ can be obtained.

We get the $path_{kp}$ reflecting the alignment relationship of key points. Due to the morphological continuity of time series, the alignment relationship between key points will be affected by key points. We can approximate the alignment path of the original time series by mapping the alignment path of the key points.

C. Mapping to original distance matrix

Construct the cumulative distance matrix $\boldsymbol{D}$ of the original time series, with the size of $m \times n$. The corresponding points in the $path_{kp}$ obtained in the previous step are mapped into the warping distance matrix of the original time series.

For each (i, j) in $path_{kp}$ indicates that the key point (k_i, x_{ki}) from X' is aligned with (l_j, y_{lj}) from Y', and mapped to the original distance matrix $\boldsymbol{D}$ is $[k_i][l_j]$. Each group of alignment relations in the $path_{kp}$ are mapped to $\boldsymbol{D}$ in turn to obtain a series of points on the near optimal path of the original time series, as shown in Figure 2.

Connecting them, as shown in the red part of the matrix in Figure 3, we can get a path similar to the optimal alignment path (the gray part of the matrix). As shown in Figure 4, expand the connection path outward by r areas range. The greater the value of r, the closer the result to the optimal distance can be obtained. In this paper, r = 1, and the final distance calculation constraint range can be obtained.

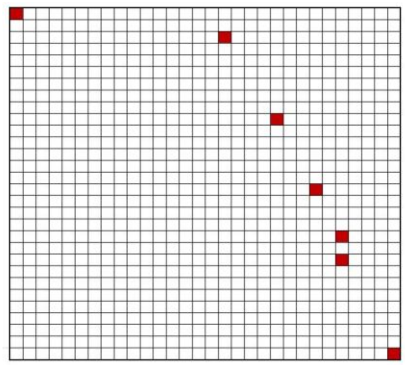

Figure 2. Mapping to distance matrix

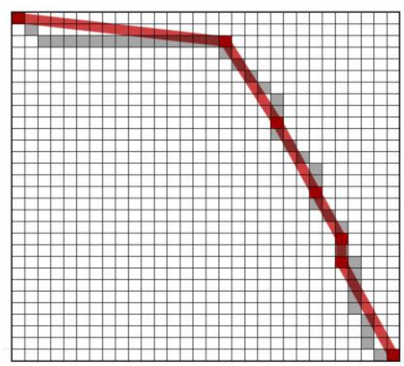

Figure 3. Approximate alignment path

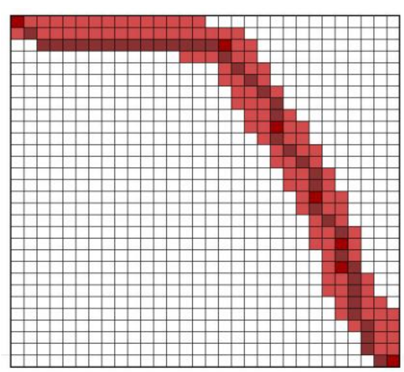

Figure 4. Constraint range

D. Calculate distance under the constraint range

Through the above methods, we get a constraint range close to the optimal warping path, which is based on the key structure alignment of the time series. Compared with the global constraint, it is more suitable for the actual alignment of the time series.

Under the constraint range, the calculation of distance matrix can be reduced, the result is closer to the optimal distance, and no need to traverse the entire distance matrix, but only the part of the matrix within the constraint range. The pseudo code for calculating the near optimal warping distance under the constraint range is as follows:

Algorithm 1 Constrainted warping distance

Input: Constraint *C* , series *X* and *Y*

Output: Distance

```
dist(1, 1) = d(1,1)
for (i, j) in C
  if (i-1, j) in C
        d1 = d(i-1, j)
   if (i, j-1) in C
        d2 = d(i, j-1)
   if (i-1, j-1) in C
        d3 = d(i-1, j-1)
  dist(i, j) = min(d1, d2, d3)
return dist(C.end)
```

IV. EXPERIMENTAL

UCR[22] is a representative data set in time series research. It contains 128 sub data sets from different sources, and the amount of data is very abundant. So we chose to run the experiment on the UCR dataset. In the experiment we take the measurement algorithm in this paper as the distance measurement of 1-NN classifier, preprocess it with PAA reduction method and Z-score standardization, and test its classification performance. We mainly investigate its running time and classification error rate, reflecting the calculation efficiency and matching accuracy of the algorithm respectively. We selected four algorithms DTW, EDR, LCSS, and TSW as the control. TABLE I lists the basic information of the data sets used.

In PAA reduction, we can control the processed time series length by modifying the window *w* size of PAA. The larger *w* is, the smaller the processed length is. When *w* = 1, the original time series is returned. We counted the time taken by each algorithm to complete a similarity search on the arrowhead training set for the same data set. Figure 5 shows the change of each algorithm time with the PAA window *w*. Among them, KPDTW is the algorithm of this paper, and the rest are four control algorithms. In the chart, as the *w* value increases, that is, the time series length decreases. We can see the time of the control algorithm decreases sharply, and the time of the algorithm in this paper decreases gently, which is lower than that of the control algorithm. Conversely, as the *w* value decreases, that is, the time series length increases, the time of the control algorithm increases sharply, while the time growth of the algorithm in this paper is relatively flat. This shows that the algorithm in this paper has better time complexity performance when the amount of data increases.

TABLE I. DATA SET INFORMATION

Name of data set	Size of train sets	Size of test sets	length	Number of classes
Adiac	390	391	176	37
ArrowHead	36	175	251	3
BME	30	150	128	3
CBF	30	900	128	3
Coffee	28	28	286	2
CricketX	390	390	300	12
ECG200	100	100	93	2
FaceAll	560	1690	131	14
GestureMidAirD1	208	130	Vary	26
GesturePebbleZ1	132	172	vary	6
Gunpoint	50	150	150	2
Lightning7	70	73	319	7
OliveOil	30	30	570	4
Plane	105	105	144	7
ShapesAll	600	600	512	60
Symbols	25	995	398	6
ToeSegmentation1	40	228	277	2
Trace	100	100	275	4
TwoPatterns	1000	4000	128	4
UMD	36	144	150	3
UWaveGestureLibraryX	896	3582	315	8
Wafer	1000	6164	152	2
WordSynonyms	267	638	270	25
Yoga	300	3000	426	2

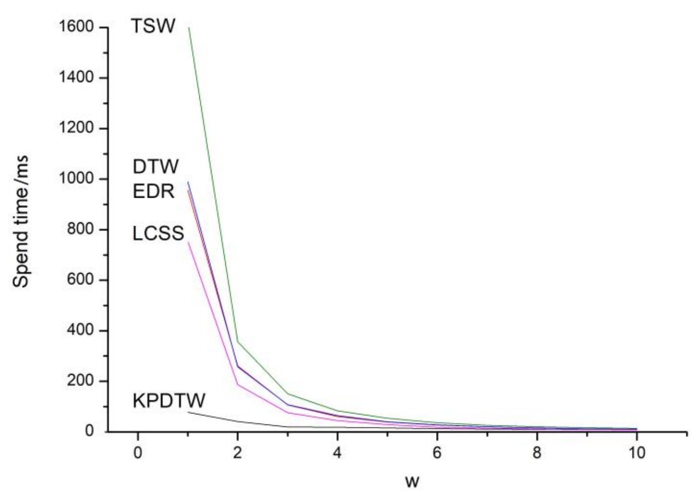

Figure 5. The relationship between spend time of each algorithm and *w* (ArrowHead dataset)

By running the classification test under 24 data sets, Figure 6 shows the cumulative running time of each algorithm. It can be seen that with the increase of the running data set, the cumulative time growth of the control algorithm is significantly higher than that of the algorithm in this paper which is expressed in KPDTW, especially in the data set with longer length, the running time will rise sharply. After running the classification tasks of 24 data sets, the algorithm in this paper saves several times the time overhead compared with the control algorithm.

In TABLE II, the classification error rates of different data sets of this algorithm and four control algorithms under 1-NN classifier are listed. The lower the error rate, the more accurate the matching result is. The algorithm with the best performance will be expressed in bold. The results show that the algorithm in this paper has excellent performance in five algorithms, achieves low error rate in 24 data sets, and the average ranking is slightly better than DTW algorithm.

Figure 7 shows the error rate comparison between the algorithm KPDTW in this paper and the control algorithm. It can be seen that the algorithm in this paper is highly close to DTW.

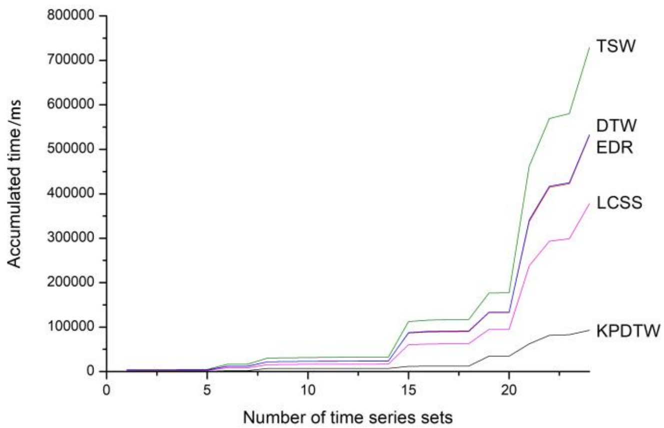

Figure 6. The accumulated time of each algorithms

TABLE II. CLASSIFICATION ERROR RATE OF ALGORITHM

Name of data set	Error rate				
	KPDTW	DTW	EDR	LCSS	TSW
Adiac	0.399	**0.391**	0.859	0.849	0.847
ArrowHead	0.251	**0.210**	0.268	0.229	0.234
BME	0.060	**0.053**	0.167	0.193	0.227
CBF	**0.002**	0.004	0.009	0.013	0.007
Coffee	0.036	**0.000**	0.071	0.071	0.071
CricketX	**0.267**	0.297	0.464	0.428	0.490
ECG200	**0.120**	0.150	0.310	0.170	0.140
FaceAll	**0.015**	0.025	0.050	0.045	0.090
GestureMidAirD1	**0.346**	0.362	0.477	0.777	0.685
GesturePebbleZ1	0.181	**0.175**	0.199	0.649	0.333
Gunpoint	**0.067**	0.087	0.160	0.080	0.093
Lightning7	**0.288**	**0.288**	0.341	0.301	0.301
OliveOil	0.167	**0.133**	0.833	0.833	0.833
Plane	**0.000**	**0.000**	0.028	**0.000**	**0.000**
ShapesAll	0.145	0.198	0.230	0.130	**0.100**
Symbols	**0.053**	0.062	0.179	0.104	0.124
ToeSegmentation1	**0.145**	0.170	0.195	0.215	**0.145**
Trace	**0.010**	**0.010**	0.030	0.230	0.110
TwoPatterns	**0.000**	0.002	0.050	0.090	0.095
UMD	**0.028**	**0.028**	0.139	0.250	0.243
UWaveGestureLibraryX	0.297	**0.227**	0.343	0.300	0.350
Wafer	0.005	0.005	0.055	**0.000**	0.005
WordSynonyms	0.280	**0.262**	0.455	0.340	0.345
Yoga	0.188	**0.156**	0.274	0.178	0.188
Average ranking	**1.542**	1.583	4.167	3.375	3.417

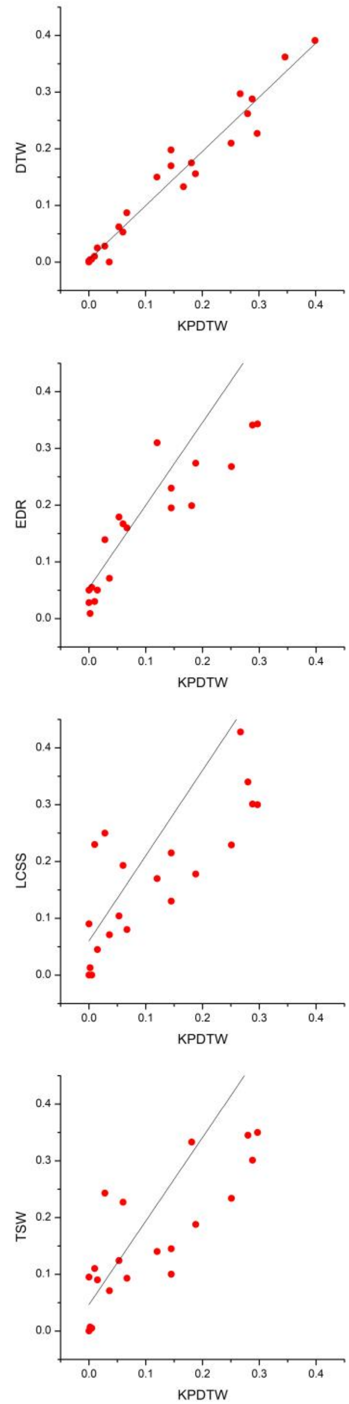

Figure.7 Error rate comparison of algorithms

V. RESULT ANALYSIS

As the most robust distance measurement method, DTW can ignore the distortion and stretching of time series in time axis through warping alignment, which has a good performance in many data sets. The algorithm in this paper is based on DTW and obtains the approximate optimal alignment relationship through the alignment of key points. The better the key point series fits the original time series, the closer the approximate distance is to the DTW distance, and the matching accuracy of the experimental results is also similar to DTW. Although the algorithm in this paper obtains the approximate distance, the approximate distance also reduces the influence of local noise, it has achieved better results in the matching accuracy under some data sets. Compared with DTW, the algorithm in this paper only sacrifices little accuracy, but the computational efficiency is significantly improved.

Let the length of the time series be n, the time complexity of extracting the time series of key points be $O(n)$, the number of key points be p, the time complexity of key point alignment and near optimal path generation be $O(p^2)$, the constraint range obtained is a fixed width, and its coverage is only linearly related to the data length. So the time complexity of calculating the near optimal distance under the constraint is $O(n)$, The final total time complexity of calculating the distance once is $O(p^2 + n)$.

In the worst case, that is, each point of the time series sequence is taken as the key point, it becomes the same time complexity $O(n^2)$ as DTW algorithm. Generally, considering that the number of key structure points is often much less than the length of the whole time series, generally $p << n$, the complexity of this algorithm is significantly lower than $O(n^2)$ of DTW. The experimental results verify this.

VI. Summary

In this paper, a time series matching algorithm based on the key points alignment is proposed. It constructs the approximate constraint range based on the alignment relationship of key points, which significantly reduces the computational overhead of distance matrix. The experimental results under multiple time series data sets show that compared with the traditional algorithm, the algorithm in this paper significantly improves the computing speed while maintaining good matching accuracy.

References

[1] Pal S H, Patet J N. Time-series data mining: A review[J]. Binary Journal of Data Mining & Networking, 2015, 5(1): 01-04.

[2] Kirlić A, Hadžić M. Big data and time series: A literature review paper[J]. Univerzitetska misao-časopis za nauku, kulturu i umjetnost, Novi Pazar, 2017 (16): 139-146.

[3] Susto G A , Cenedese A , Terzi M . Time-Series Classification Methods: Review and Applications to Power Systems Data[J]. Big Data Application in Power Systems, 2018:179-220.

[4] Wu Y, Shen K, Chen Z, et al. Automatic measurement of fetal cavum septum pellucidum from ultrasound images using deep attention network[C]//2020 IEEE International Conference on image processing (ICIP). IEEE, 2020: 2511-2515.

[5] Gorecki T, Luczak M. Non-isometric transforms in time series classification using DTW[J]. Knowledge-Based Systems, 2014, 61(may): 98-108.

[6] Chen L, Tamer Özsu M, Oria V. Robust and fast similarity search for moving object trajectories [C]// Acm Sigmod International Conference on Management of Data. ACM, 2005, 491-502.

[7] Gorecki, Tomasz. Classification of time series using combination of DTW and LCSS dissimilarity measures[J]. Communications in statistics, B. Simulation and computation, 2018, 47(1-2):263-276.

[8] Hajihashemi Z, Popescu M. A multidimensional time series similarity measure with applications to eldercare monitoring[J]. IEEE Journal of Biomedical & Health Informatics, 2016, 20(3):953-962.

[9] Jiang G, Wang W, Zhang W. A novel distance measure for time series: Maximum shifting correlation distance[J]. Pattern recognition letters, 2019, 117(JAN.):58-65.

[10] Zhang M, Pi D. A New Time Series Representation Model and Corresponding Similarity Measure for Fast and Accurate Similarity Detection[J]. IEEE Access, 2017, 5:1-1.

[11] Chen H, Gao X. Similarity Measurement and Cluster Analysis of Time Series Based on Fluctuation Features[J]. Statistics & Decision, 2019.

[12] Gharghabi S, Imani S, Bagnall A, et al. Matrix profile xii: Mpdist: a novel time series distance measure to allow data mining in more challenging scenarios[C]//2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018: 965-970.

[13] Kim S W, Park S, Chu W W. An index-based approach for similarity search supporting time warping in large sequence databases[C]// Proceedings 17th international conference on data engineering. IEEE, 2001: 607-614.

[14] Izakian H , Pedrycz W , Jamal I . Fuzzy clustering of time series data using dynamic time warping distance[J]. Engineering Applications of Artificial Intelligence, 2015, 39(mar.):235-244.

[15] Keogh E, Wei L, Xi X, et al. Supporting exact indexing of arbitrarily rotated shapes and periodic time series under Euclidean and warping distance measures[J]. VLDB Journal. 2009, 18: 611-630.

[16] Rakthanmanon T, Campana B, Mueen A, et al. Searching and mining trillions of time series subsequences under time warping[C]// Acm Sigkdd International Conference on Knowledge Discovery & Data Mining. ACM, 2012.

[17] Cai Q, Chen L, Sun J. Piecewise statistic approximation based similarity measure for time series[J]. Knowledge-Based Systems, 2015, 85: 181-195.

[18] Cuturi M, Blondel M. Soft-dtw: a differentiable loss function for time-series[C]//International conference on machine learning. PMLR, 2017: 894-903.

[19] Geler Z , Kurbalija V , Ivanovic M , et al. Dynamic Time Warping: Itakura vs Sakoe-Chiba[C]// 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA). IEEE, 2019.

[20] Geler Z, Kurbalija V, Radovanovi M, et al. Impact of the Sakoe-Chiba band on the DTW time series distance measure for kNN classification[C]// International Conference on Knowledge Science, Engineering and Management. Springer, Cham, 2014.

[21] Keogh E , Chakrabarti K , Pazzani M , et al. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases[J]. Knowledge and Information Systems, 2001, 3(3):263-286.

[22] Chen Y, Keogh E, Hu B, Begum N, et al. The UCR Time Series Classification Archive. 2018. www.cs.ucr.edu/~eamonn/time_series_date