Subsequence Matching of Stream Synopses under the Time Warping Distance

Su-Chen Lin[1], Mi-Yen Yeh[2], and Ming-Syan Chen[1]

[1] Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan
[2] Institute of Information Science, Academia Sinica, Taipei, Taiwan
sclin@arbor.ee.ntu.edu.tw, miyen@iis.sinica.edu.tw, mschen@cc.ee.ntu.edu.tw

Abstract. In this paper, we propose a method for online subsequence matching between histogram-based stream synopsis structures under the dynamic warping distance. Given a query synopsis pattern, the work continuously identifies all the matching subsequences for a stream as the histograms are generated. To effectively reduce the computation time, we design a Weighted Dynamic Time Warping (WDTW) algorithm which computes the warping distance directly between two histogram-based synopses. Our experiments on real datasets show that the proposed method significantly speeds up the pattern matching by sacrificing a little accuracy.

1 Introduction

Subsequence matching is a popular application in a data stream environment such as sensor network monitoring and financial data analysis. When given a query sequence, users would have great interests in continuously monitoring similar subsequences when a data stream keeps evolving. Therefore, a real-time and space-saving approach is required.

The similarity measurement is an important factor of the subsequence matching. Compared with the Euclidean distance, the dynamic time warping (DTW) [1] distance is more robust since it offers elastic scaling and shifting capabilities in time axis. To match two sequences of length M and N respectively, an intuitive solution is to compute the DTW distances of all possible matchings. However, the time complexity of this method is $O(MN^3)$ since it costs $O(MN)$ to obtain a DTW distance.

Various types of DTW algorithms on subsequence matching have been proposed [2,3]. However, it is impractical for stream applications to preprocess the whole data in advance as these methods did. Hence, Sakurai et al. [4] develop an online subsequence matching algorithm named SPRING, which is based on the DTW algorithm with relaxed boundary constraints. Given a query sequence of length M, SPRING spends $O(M)$ time to identify a matching subsequence at each data point of a stream. However, for a stream of length N, the total time cost of SPRING is $O(MN)$, which is heavy especially when N and M are large.

Searching for a better solution, we notice a growing interest in synopsis techniques [5] which meet the real-time requirement with a small accuracy loss in

stream applications. Similarly, synopsis structures for speeding up the DTW algorithm are discussed. Keogh et al. proposed the PDTW algorithm which used a piecewise aggregate approximation (PAA) approach with equal-width histogram-based synopses to speeding up DTW [6]. In addition, synopsis structures with arbitrary-width histograms are designed for better approximation accuracy. Examples include the adaptive piecewise constraint approximation (APCA) [7] and the Haar wavelet reconstruction [8]. Chan et al. proposed a Haar wavelet-based approximation method under the time warping distance, but the method cannot deal with subsequences and has much overestimation [9].

For the above reasons, we present a new subsequence matching method under the dynamic time warping distance for data streams that are summarized with an arbitrary-width histogram-based synopsis structure. Each histogram contains a value and a timestamp indicating the end of this histogram as shown in Fig. 1(a). Given a query sequence Q with length M that summarized with m histograms, we want to continuously report subsequences of stream with distances to Q not greater than the threshold ϵ. We propose the Weighted Dynamic Time Warping (WDTW) algorithm that derives these matching subsequences. In order not to overestimate the warping distance, which could happen when one histogram of a stream synopsis matches multiple histograms of the other one, we have designed a method to lower the overestimated distance. The WDTW algorithm is shown to have $O(m)$ complexity in both time and space at the coming of each histogram of the stream. After processing n histograms, the total time complexity of our method is $O(mn)$, where $m \ll M$ and $n \ll N$ for synopsis streams.

To evaluate the WDTW algorithm, we conduct two experiments using a real dataset of time series. For comparisons, we implemented two other subsequence matching methods, referred to as Synopsis-DTW and MicroCell-DTW. The experimental results show that, when compared with MicroCell-DTW, our method and Synopsis-DTW have far low computational time costs. However, our method has only a little trade-off in accuracy, which is not true for Synopsis-DTW.

2 Preliminaries

Dynamic Time Warping (DTW) is a widely used distance measurement in time series applications. It can compute the distance between two series of different lengths since it solves the problem of shifting and scaling in the time axis. In essence, the DTW distance between two series $X = \{x_i | 1 \leq i \leq N\}$ and $Y = \{y_j | 1 \leq j \leq M\}$ is computed as follows. First, the distance between two data points is defined as $d(i,j) = \|x_i - y_j\|$. If x_i matches y_j, we define it as a *matching pair* (i,j). A sequence of all matching pairs from (1,1) to (N,M) between series X and Y is called a warping path W. Then, a warping distance for the path W is $Distance(W) = \sum_{(i,j)\in W} \|x_i - y_j\|$.

Obviously, there exists multiple warping paths between X and Y. The DTW distance between X and Y is defined as the warping path with the smallest $Distance(W)$, which we defined as the optimal warping path. The dynamic

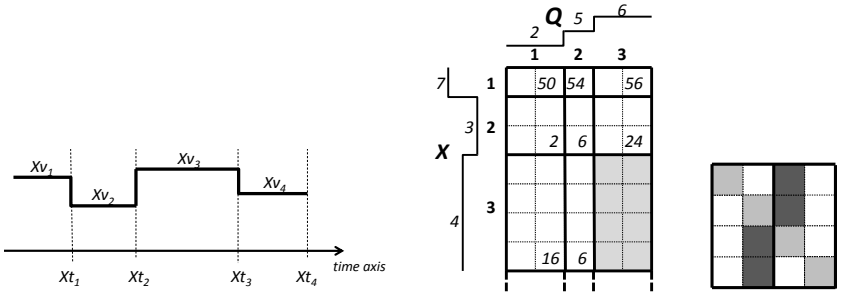


Fig. 1. (a)Histogram-based synopsis stream, (b)Accumulated distance matrix, (c)The dark shaded micro cells are overestimated distance area in directly accumulating distance

programming technique can solve the optimal warping path problem in $O(MN)$ time complexity. Please refer to [1] for more details.

3 Subsequence Matching over Stream Synopses

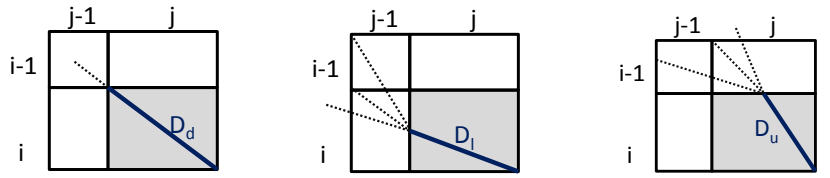
3.1 Problem Definition

Given a data stream of length N, an online stream is summarized as a sequence of histograms $X = \{x_1, x_2, .., x_n\}$, each of which has a height xv_i and an endpoint xt_i as shown in Fig. 1(a). We can denote a stream synopsis as $X = \{\langle xv_1, xt_1\rangle, ..., \langle xv_n, xt_n\rangle\}$. $X[t_s : t_e]$ is defined as a synopsis subsequence which starts from time t_s and finishes at t_e, where both t_s and t_e have to be endpoints of histograms of X. For ease of exposition, we denote a synopsis subsequence as $X[[i] : [j]]$ to mean that it starts from the i^{th} to the j^{th} histograms of X. Given the notations, we now define the subsequence matching problem.

Definition 1 (Synopsis Subsequence Matching). *Given an online running stream synopsis X, a query synopsis subsequence Q, and a threshold ϵ, the goal of synopsis subsequence matching is to locate all the subsequences $X[t_s : t_e]$ that satisfy $Dtw(X[t_s : t_e], Q) \leq \epsilon$*

3.2 WDTW: A Weighted Algorithm for Dynamic Time Warping

To solve the above issue, we propose *Weighted Dynamic Time Warping* (WDTW) method. When the distance between the stream synopsis $X = \{x_1, x_2, ..., x_n\}$ and the query sequence $Q = \{q_1, q_2, ..., q_m\}$ is derived, an accumulated distance matrix of $n \times m$ *cells* will be created to keep the histogram mapping information sequentially. As Fig. 1(b) shows, each cell is divided by the bold-solid lines according to width of each histogram. The cell (i, j) is constructed by the i^{th} histogram of X and j^{th} histogram of Q. The size of each cell is different due to



(a) Case1: diagonal cell (b) Case2: left cell (c) Case3: up cell

Fig. 2. Three alternatives for each matching procedure

the different width of each histogram. Therefore, each cell can be further divided into multiple square *micro cells* with side lengths equal to one time unit. For example, in Fig. 1(b), the shaded cell $(3, 3)$ contains 4×2 micro cells. Later we will show that in fact only the latest two rows of these cells need to be kept.

Similar to DTW, WDTW also works in a dynamic programming way. Intuitively, the warping distance at each cell is directly derived from three neighbor cells of various sizes in the matching procedure. However, this will result in overestimated distance. Using Fig. 1(c) as an example, if we directly compute the accumulating warping distance of the cell (3,3) from the cell (3,2), all the light and dark shaded micro cells will be counted in. In fact, the distance contributed by the dark shaded micro cells are redundant and should be eliminated.

To lower the overestimated distance, our WDTW works as follows. First, we define $D(i, j)$ as the minimum accumulated distance between $X[[s] : [i]]$ and $Q[[1] : [j]]$, $s = 1, 2, ..., i$,

$$D(i,j) = \begin{cases} 0, & \text{if } j = 0, \\ \infty, & \text{if } i = 0, j \neq 0, \\ \min\{D_d(i,j), D_l(i,j), D_u(i,j)\}, & \text{otherwise.} \end{cases} \quad (1)$$

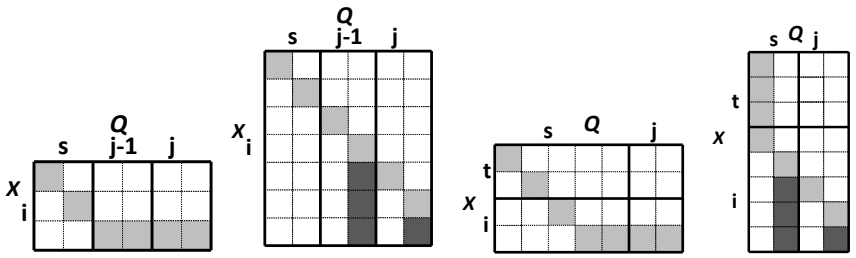
where $D_d(i, j)$, $D_l(i, j)$, and $D_u(i, j)$ denote the minimum distances for cell (i, j) with warping paths through the cell $(i - 1, j - 1)$, $(i, j - 1)$, and $(i - 1, j)$ respectively. We now discuss how to compute these distances case by case.

Case 1: Minimum Distance Path from the Diagonal Cell

In this case, the warping path comes from the cell $(i-1, j-1)$ to the current cell (i, j) as Fig. 2(a) shows. The path implies that the histogram x_{i-1} matches the histogram q_{j-1} and x_i matches q_j. The distance $D_d(i, j)$ can de obtained from sum of $D(i - 1, j - 1)$ and the distance of the cell (i, j).

The sub-optimal path of the current cell (i, j) passes $\max\{lx_i, lq_j\}$ micro cells, where lx_i and lq_j are the length of histogram x_i and q_j respectively. The micro cells passed by the sub-optimal path are called *steps* in the rest of this paper. Consequently, we can obtain the accumulated distance $D_d(i, j) = D(i - 1, j - 1) + d_{i,j} \times \max\{lx_i, lq_j\}$, where $d_{i,j} = \|xv_i - qv_j\|$. For example in Fig.1(b), the $D(3, 3)$ in the shaded area can be computed as: $D_d(3, 3) = D(2, 2) + d_{3,3} \times \max\{lx_3, lq_3\} = 6 + (4 - 6)^2 \times 4 = 22$. ■

Case 2: Minimum Distance Path from the Left Cell



(a) $lx_i \leq \sum_{k=s}^{j} lq_k$ (b) $lx_i > \sum_{k=s}^{j} lq_k$ (c) $E_v(i, j-1) = 0$ (d) $E_v(i, j-1) \neq 0$

Fig. 3. (a)(b) are two kinds of cell combinations, and (c)(d) are examples used in case 2. The shaded areas are passed by warping paths and the dark shaded areas stand for E_v for each cell.

The warping path from the cell $(i, j-1)$ to cell (i, j), as shown in Fig. 2(b), denotes that x_i matches the consecutive histograms q_{j-1} and q_j. We use a combined calculation of the consecutive histograms in a row to lower the overestimated distance of $D_l(i, j)$. Without loss of generality, we assume that the start histogram of these consecutive ones is q_s. Our major insight is that the consecutive histograms that match the same histogram x_i can be combined. In other words, the consecutive cells (i, s) to (i, j), where $s < j$, will be considered as a cell combination, where the sub-optimal warping path Ω goes from the upmost-leftmost micro cell of (i, s) to the bottommost-rightmost micro cell of (i, j).

Definition 2 (The Adjustable Distance in the Vertical Direction). *For the warping path passes from the cell (i, s) to the cell (i, j), where s is the smallest index of these consecutive ones as a combination, the value $E_v(i, j)$ is defined to represent the adjustable distance of cell (i, j) in the vertical direction as follows.*

$$E_v(i,j) = \begin{cases} 0 & ,if\ lx_i \leq \sum_{k=s}^{j} lq_k \\ \min\{d_{i,s}, ..., d_{i,j}\} \times (lx_i - \sum_{k=s}^{j} lq_k) & ,otherwise \end{cases} \quad (2)$$

Lemma 1. *The total distance of the sub-path, in a cell combination from (i, s) to (i, j), of any warping path that contains it is: $\sum_{k=s}^{j} (d_{i,k} \times lq_k) + E(i, j)$.*

Proof. Based on case 1, the sub-optimal path of the cell combination passes $\max\{lx_i, \sum_{k=s}^{j} lq_k\}$ steps (micro cells). If $\sum_{k=s}^{j} lq_k \geq lx_i$ as Fig.3(a) shows, the distance of the sub-path would be $\sum_{k=s}^{j} (d_{i,k} \times lq_k)$ and $E_v(i, j)$ would be 0. If $lx_i > \sum_{k=s}^{j} lq_k$ as Fig.3(b) shows, in addition to the diagonal steps, the path has to pass further $lx_i - \sum_{k=s}^{j} ly_k$ vertical steps. In order to obtain the optimal path, the vertical steps must be in the cell which has minimum $d_{i,k}$, where k is from s to j. In this condition, the adjustable distance in vertical direction is $d_{min} \times lx_i - \sum_{k=s}^{j} ly_k = E_v(i, j)$, which is the dark shaded area in Fig.3(b) for example. Therefore, the total distance of the sub-path is $\sum_{k=s}^{j} (d_{i,k} \times lq_k) + E_v(i, j)$. □

In sum, Lemma 2 shows how to calculate the value of $D_l(i, j)$.

Lemma 2. *The accumulated distance $D_l(i,j)$, where the path comes from the cell $(i,j-1)$, can be obtained with following equation in O(1) time.*

$$D_l(i,j) = D(i,j-1) - E_v(i,j-1) + d_{i,j} \times lq_j + E_v(i,j) \tag{3}$$

Proof. Without loss of generality, for the sub-path from the cell (t,s) to the cell (i,j), where $t \leq i$ and $s < j$, there are two cases of $E_v(i,j-1)$. When $E_v(i,j-1) = 0$ as Fig. 3(a) and 3(c) show, no adjustable distance in the vertical direction should be distributed to the next cell. Hence, $E_v(i,j)$ is also equal to 0, and Eq. (3) is obtained in this case.

When $E_v(i,j-1) \neq 0$, as the dark shaded area shown in Fig. 3(b) and 3(d), the vertical steps in the previous cells can be distributed to the cell (i,j). With Lemma 1 and Definition 2, $D_l(i,j)$ can be obtained as follows.

$$\begin{aligned} D_l(i,j) &= D(t-1,s-1) + \sum_{k=t}^{i-1} (d_{k,s} \times lx_k) + \sum_{k=s}^{j} (d_{i,k} \times lq_k) + E_v(i,j) \\ &= D(i,j-1) - E_v(i,j-1) + d_{i,j} \times lq_j + E_v(i,j) \end{aligned}$$ □

Based on Lemma 2, the information required to calculate $D_l(i,j)$ is only stored in cell (i,j) and $(i,j-1)$. Since the computation time of $E_l(i,j)$ and $E_l(i,j-1)$ is constant, $D_l(i,j)$ is obtained in constant time. We give an example of this case. In Fig. 1(b), $E_v(3,3) = \min\{d_{3,2}, d_{3,3}\} \times (lx_3 - lq_2 - lq_3) = 1 \times (3-2) = 1, D_l(3,3) = D(3,2) - E_v(3,2) + d_{3,3} \times lq_3 + E_v(3,3) = 6-3+4\times 2+1 = 12$. ■

Case 3: Minimum Distance Path from the Up Cell

In this case, the warping path passes through cell $(i-1,j)$ as Fig.2(c) shows. Since this case is similar to case 2 despite the consecutive cells are in a column, $D_u(i,j)$ can also be derived in constant time. In Fig. 1(b), $E_h(3,3) = 0$, $D_u(3,3) = D(2,3) - E_h(2,3) + d_{3,3} \times lx_3 + E_h(3,3) = 24 - 0 + 1 \times 4 + 0 = 28$. ■

After deriving the distance values of the three cases, the minimum accumulated distance can be obtained. Continuing the previous example, $D(3,3) = \min\{D_d(3,3), D_l(3,3), D_u(3,3)\} = \min\{22, 12, 28\} = 12$.

We now describe how WDTW identifies matching subsequences. For each synopsis histogram x_i arriving at time t_i, WDTW computes the i^{th} row of the accumulated distance matrix $D(i,j)$ where $j = 1, 2, ..., m$. Then, the warping distance of the most similar subsequence, $X[t_s : t_i]$, to Q is $Dtw(X[t_s : t_i], Q) = D(i,m)$. To get the start time t_s, each cell (i,j) keeps the start time index where the warping distance comes from in the matrix $S(i,j)$. Therefore, when constructing the warping path, the start and end time of the most similar subsequence are kept in the last cell (i,m) of this path. For example, in Fig. 1(b), $t_s = S(3,2) = S(2,1) = t_2$. Hence, $X[t_2 : t_3]$ is reported at t_3 if $Dtw(X[t_2 : t_3], Q) \leq \epsilon$.

Notice that as we compute $D(i,j)$, $S(i,j)$ is also obtained. Since the computation cost of each $D(i,j)$ and $S(i,j)$ is $O(1)$, the computation time of m histograms is $O(m)$. Also, only the information of cell$(i-1,j-1)$,$(i,j-1)$, and $(i-1,j)$ is used. Therefore, WDTW only needs to keep the latest two rows of cells, i.e., $O(2 \times m) = O(m)$ in space.

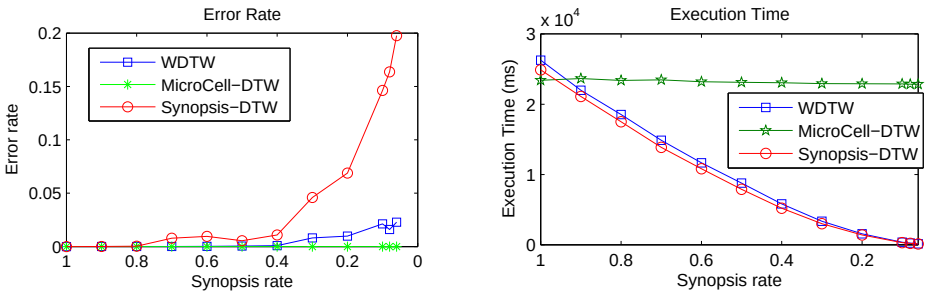


Fig. 4. Error rate and execution time at different synopsis rates

4 Performance Evaluation

4.1 Experiment Setup and Performance Metrics

The real datasets are downloaded from the UCR Time Series Classification / Clustering Archive [10]. We choose time series in the *posture* dataset. The first pattern in each series was used as the query patterns with length $M = 1024$, while the rest part of the series as data streams with length $N = 32768$. The synopsis histograms in the following experiments were built by Haar wavelet decomposition with varied synopsis rate [8], which is defined as a ratio of the number of histograms to the number of points in the original stream.

We compared our algorithm with two methods: MicroCell-DTW and Synopsis-DTW. MicroCell-DTW divided each histogram into multiple one-time-unit histograms. It computed the warping distance based on micro cells, not the cells, and the matching problem was solved using the SPRING method [4]. On the other hand, Synopsis-DTW computed warping distance directly on the cells as WDTW did, but was regardless of handling the overestimated distance.

4.2 Experimental Results

The first experiment examined the accuracy of WDTW and Synopsis-DTW. The subsequences produced by MicroCell-DTW were regarded as the benchmark since it produced the correct warping distance based on the synopsis histograms. The error rate is defined as the ratio of the sum of false alarms and misses to the number of correct subsequences and the reported subsequences by either WDTW or Synopsis-DTW. For each missed or false alarmed subsequence, the penalty is the ratio of the time interval of the false alarmed/missed part to the whole length of itself.

The results were shown in Fig. 4(a). When the synopsis rate decreased, the error rate of Synopsis-DTW increased significantly. In contrast, the error rate of WDTW increased much slightly. For example, the error rate of WDTW is 2.2% while that of Synopsis-DTW is up to 19.7% when the synopsis rate is 0.06. This shows the importance of dealing with the overestimated distance.

The second experiment examined the speed of the three algorithms. The results are in Fig. 4(b). MicroCell-DTW was independent of the synopsis rate since it examined each micro cells of a stream, where the number is equal to the original stream length. Under the synopsis rate of 0.06, MicroCell-DTW can only process 0.85 histograms each millisecond. In contrast, WDTW can process 141 histograms under the same synopsis rate, which is 165 times faster than MicroCell-DTW did. In other words, WDTW processed the online subsequence matching 165 times faster than MicroCell-DTW did. On the other hand, WDTW and Synipsis-DTW had almost the same computation cost, which decreased along with the synopsis rates. This shows that the processing time of the overestimated distance is very small. Concluded from Fig. 4(a) and 4(b), WDTW processes the online subsequence matching efficiently while sacrificing only a very little accuracy.

5 Conclusion

We presented WDTW, an efficient online subsequence matching algorithm under dynamic time warping in a streaming environment. Once a various-width synopsis histogram of a stream is generated, according to the query sequence, WDTW reports all the matching subsequences. The experimental results show that when compared with MicroCell-DTW and Synopsis-DTW, WDTW has a low computation cost to meet the time and space constraints of streams, with a little trade-off in accuracy.

References

1. Sakoe, H.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing (1978)
2. Chen, Y., Nascimento, M.A., Chin, B., Anthony, O., Tung, K.H.: Spade: On shape-based pattern detection in streaming time series. Proc. of ICDE (2007)
3. Han, W.S., Lee, J., Moon, Y.S., Jiang, H.: Ranked subsequence matching in time-series databases. Proc. of VLDB (2007)
4. Sakurai, Y., Faloutsos, C., Yamamuro, M.: Stream monitoring under the time warping distance. Proc. of ICDE (2007)
5. Aggarwal, C.C., Yu, P.S.: A survey of synopsis construction in data streams. Advances in Database Systems (2009)
6. Keogh, E., Pazzani, M.: Scaling up dynamic time warping for datamining applications. Proc. of the SIGKDD (2000)
7. Keogh, E., Chakrabarti, K., Mehrotra, S., Pazzani, M.: Locally adaptive dimensionality reduction for indexing large time series databases. Proc. of SIGMOD (2001)
8. Burrus, C.S., Gopinath, R.A., Guo, H.: Introduction to wavelets and wavelet transforms: A primer. Prentice Hall (1997)
9. Chan, F.K.P., chee Fu, A.W., Yu, C.: Haar wavelets for efficient similarity search of time-series: with and without time warping. IEEE TKDE (2003)
10. Keogh, E., Xi, X., Wei, L., Ratanamahatana, C.A.: Ucr time series classification/clustering archive. http://www.cs.ucr.edu/∼eamonn/time_series_data/