Individualized Warping Window Size for Dynamic Time Warping

Krisztian Buza [1] **Ladislav Peška** [1 2]

Abstract

In this paper, we present our work towards learning individualized warping window sizes (WWS) for time series classification based on dynamic time warping (DTW). DTW is one of the most popular distance measures for time series classification and it was shown that its warping window size is crucial for the final accuracy of the model. WWS is therefore considered as an important parameter of DTW. In contrast to the previous works, in which static WWS was used, i.e., the WWS size was selected for the entire dataset, we propose a hubness-aware approach to select WWS for each instance *individually*. We evaluate our approach on publicly available real-world datasets and show that the classification accuracy using individualized WWS is significantly higher than the accuracy in case of static WWS.

1. Introduction

Time series classification (TSC) is the common theoretical background of various recognition tasks such as speech recognition, handwriting recognition on a touch screen, or the diagnosis of diseases based on medical time series (ECG, EEG). Distance-based approaches for TSC range from nearest neighbor (Xi et al., 2006) and its recent versions, such as hubness-aware classifiers (Radovanović et al., 2010; Tomašev et al., 2015), through distance-based features (Buza et al., 2015; Kate, 2016) and kernel approaches (Meszlényi et al., 2016; Xue et al., 2017) to convolutional neural networks using DTW-based features (Meszlényi et al., 2017). See also Abanda et al. (2019) for a recent survey on distance-based TSC.

One of the most popular distance measures for TSC is Dynamic time warping (DTW), originally introduced for speech recognition (Sakoe & Chiba, 1978). When comparing two time series, DTW aims at matching local patterns while allowing for shifts and elongations between the two time series. The amount of allowed shifts and elongations is controlled by the *warping window size* (WWS), a parameter of DTW.

Originally, the warping window was introduced for computational reasons (in order to calculate an approximation of the actual DTW distance with an order of magnitude less computations) and it was considered a "necessary evil" (Ratanamahatana & Keogh, 2005). However, after surveying "more than 500 papers", Ratanamahatana & Keogh (2005) concluded that a relatively narrow warping window is "necessary for accurate DTW". This resulted in the wide-spread use of a default WWS of 5% or 10%.

Recently, the role of the warping window size has been examined more thoroughly and the analysis showed that the appropriate selection of WWS may be crucial for the accuracy of DTW-based classifiers (Dau et al., 2018). All the aforementioned works considered a *static* WWS: the same warping window size was used for the entire dataset, i.e., for all the time series of the dataset.

In contrast, we argue that an individualized selection of the warping window size may be beneficial. In particular, we propose to select the appropriate WWS for each time series separately, similarly to the case of individualized selection of the number of nearest neighbors (Buza et al., 2010). In this paper, we propose a hubness-aware approach for the individualized selection of WWS. We perform experiments on publicly available real-world time series datasets and show that the proposed individualized WWS leads to significantly better results on various challenging datasets.

The rest of this paper is organized as follows. In order to make sure that our work is self-contained, Section 2 reviews dynamic time warping. This is followed by the description of our approach (Section 3) and experiments (Section 4). We draw conclusions in Section 5.

2. Dynamic Time Warping

In simple cases (e.g. Euclidean distance), the distance of two time series $T_1 = (x_1^{(1)}, \ldots, x_{L_1}^{(1)})$ and $T_2 = (x_1^{(2)}, \ldots, x_{L_2}^{(2)})$

[1]Telekom Innovation Laboratories, Department of Data Science and Engineering, Faculty of Informatics, ELTE - Eötvös Loránd University, Budapest, Hungary [2]Department of Software Engineering, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic. Correspondence to: Krisztian Buza <chrisbuza@yahoo.com>.

Time Series Workshop at the 36th International Conference on Machine Learning, Long Beach, USA, 2019.

is calculated in a way that each $x_i^{(1)}$ is compared to $x_i^{(2)}$ and the results of these comparisons are aggregated.

However, when observing a phenomenon several times, the corresponding characteristic patterns may not appear at the exactly same time position, and events' duration may also vary slightly. In order to address these challenges, DTW allows for shifts and elongations, i.e., $x_i^{(1)}$ is compared to $x_j^{(2)}$ where i may be different from j.

The calculation of the DTW distance (Sakoe & Chiba, 1978) is implemented as filling the entries of an $L_1 \times L_2$ matrix. Each entry of the matrix corresponds to the distance between a prefix of T_1 and a prefix of T_2. In particular, the value in the i-th row and j-th column, denoted as $d_{i,j}$, corresponds to the distance between $T_1' = (x_1^{(1)}, \ldots, x_i^{(1)})$ and $T_2' = (x_1^{(2)}, \ldots, x_j^{(2)})$, and it is calculated as follows:

$$d_{i,j} = d_e(x_i^{(1)}, x_j^{(2)}) + \min\{d_{i,j-1}, d_{i-1,j}, d_{i-1,j-1}\} \quad (1)$$

where the terms of the minimum correspond to the cases of elongation in T_1, T_2 or matching the next elements in both time series; $d_e(x_i^{(1)}, x_j^{(2)})$ is the elementary distance between observations $x_i^{(1)}$ and $x_j^{(2)}$. In our current work

$$d_e(x_i^{(1)}, x_j^{(2)}) = |x_i^{(1)} - x_j^{(2)}|,$$

but we note that other variants exist, such as the squared difference between $x_i^{(1)}$ and $x_j^{(2)}$.

The entries $d_{i,j}$ of the matrix can be calculated in a column-wise fashion, i.e., in this order: $d_{1,1}$, $d_{2,1}$, $\ldots$, $d_{L_1,1}$, $d_{1,2}$, $d_{2,2}, \ldots, d_{L_1,2}, \ldots d_{L_1,L_2}$. The first entry of the matrix, $d_{1,1}$, is initialized as $d_{1,1} = d_e(x_1^{(1)}, x_1^{(2)})$. In the cases, where some of the terms $d_{i,j-1}$, $d_{i-1,j}$, $d_{i-1,j-1}$ are undefined (i.e., if $i - 1 = 0$ or $j - 1 = 0$), they are excluded from Eq. (1). Finally, the DTW distance of time series T_1 and T_2 equals to d_{L_1,L_2}.

An example for the calculation of DTW is shown in Fig. 1.

In most applications, it may not be reasonable to allow for arbitrarily large shifts and elongations, therefore, the calculations are usually restricted to the entries around the diagonal of the matrix, i.e.,

$$d_{i,j} \text{ is calculated } \Leftrightarrow |i - j| \leq w,$$

where w is the *warping window size* which controls the amount of allowed shifts and elongations.

With *full DTW* we refer to the case when the above restriction is *not* made (or equivalently: when w is set to the maximum of the length of the two time series that are compared).

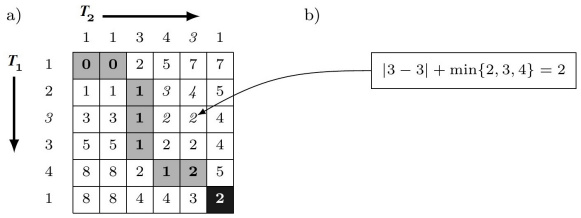

Figure 1. Example for the calculation of the DTW-matrix. a) The DTW-matrix. The time series T_1 and T_2 are shown on the left and top of the matrix. The marked entries of the matrix correspond to the mapping between the both time series. b) The calculation of the value of an entry.

3. Our Approach

Our approach for the individualized selection of the appropriate WWS is based on how frequently time series appear as good and bad neighbors. Following (Tomašev et al., 2015), we define good (bad, resp.) k-nearest neighbors as follows: a time series T is a good (bad, respectively) k-nearest neighbor of another time series T' if T is a k-nearest neighbor of T' and their class labels are the same (different, respectively). Note that the nearest neighbor relationship is asymmetric, i.e., if T is one of the k-nearest neighbors of T', it does *not* mean that T' is also one of the k-nearest neighbor of T. Therefore a time series may appear more or less than k-times as nearest neighbor of *other* time series. Roughly speaking, time series that appear surprisingly often as one of the (good/bad) k-nearest neighbors of other time series, are called (good/bad) *hubs*. In order to quantify this phenomenon, given a time series T, we use $GN_k(T)$ and $BN_k(T)$ to denote how many times T appears as one of the good/bad k-nearest neighbor of other time series. As the set of k-nearest neighbors depends on WWS, we use $GN_{k,w}(T)$ and $BN_{k,w}(T)$ to denote $GN_k(T)$ and $BN_k(T)$ calculated with DTW using warping window size w.

For a time series T, we determine the individualized WWS as follows: we consider warping window sizes $0, 1, \ldots, w_{max}$, where w_{max} is 10% of the length of T. For each WWS, we determine $\mathcal{N}_{k,w}^{\mathcal{D}}(T)$, which denotes the set of k-nearest neighbors of T in the training set D calculated using DTW with warping window size w. For each $T' \in \mathcal{N}_{k,w}^{\mathcal{D}}(T)$, we calculate $GN_{k,w}(T')$ and $BN_{k,w}(T')$. Finally, we select w_{best} that maximizes

$$\sum_{T' \in \mathcal{N}_{k,w}^{\mathcal{D}}(T)} GN_{k,w}(T') - BN_{k,w}(T'). \quad (2)$$

In case of ties, i.e., if several WWS have the same score in terms of Formula (2), out of the warping window sizes with maximal score, the lowest one is selected.

Our approach is summarized in Algorithm 1.

Algorithm 1 Individualized warping window size selection for a time series T

Input: set of labelled training time series $\mathcal{D}$, time series T, number of nearest neighbors k

$w_{max} = 0.1 \cdot$ length(T)
$w_{best} = 0$
$score_{best} = 0$

for each $w \in 0, \ldots, w_{max}$ **do**
  $score = 0$
  # $\mathcal{N}^{\mathcal{D}}_{k,w}(T)$ = the set of k-nearest neighbors of T
  # among the time series in $\mathcal{D}$, calculated with
  # warping window size w
  for each $T' \in \mathcal{N}^{\mathcal{D}}_{k,w}(T)$ **do**
    $score = score + GN_{k,w}(T') - BN_{k,w}(T')$
    # $GN_{k,w}(T')$ and $BN_{k,w}(T')$ are calculated
    # on $\mathcal{D}$ with warping window size w
  end for
  if $score > score_{best}$ **then**
    $w_{best} = w$
    $score_{best} = score$
  end if
end for
return w_{best}

Radovanović et al. (2010) observed that bad hubs are responsible for a surprisingly large fraction of the total classification error. Within this framework, our approach can be seen as a method that searches for the warping window size w_{best} which minimizes the detrimental effect of bad hubs and maximize the positive effect of good hubs.

4. Experiments

We evaluated our approach in context of k-nearest neighbors classification on five publicly available time series dataset from the UCR repository[1]. While the simple nearest neighbors classifier with DTW using a static WWS may perform well in certain application scenarios, for the purpose of the evaluation of our approach, we selected datasets that are challenging for this method. In particular, we used the ArrowHead, Car, DistalPhalanxOutlineAgeGroup, DistalPhalanxTW and DodgerLoopDay datasets.

We performed experiments according to the 10×10-fold cross-validation protocol.[2] We classified the test time series with k-nearest neighbor using DTW distance with individualized selection of the warping window size, i.e., for each test time series, we selected the appropriate warping window size with Algorithm 1. We considered two settings of the number of nearest neighbors: $k = 5$ and $k = 10$. We measured the accuracy, i.e., the ratio of correctly classified time series, in each of the 10×10 folds and report the average accuracy and its standard deviation in Table 1.

We compared the performance of our approach, denoted as IWW, with that of k-nearest neighbor classification using DTW distance with a static WWS of 5% and 10% of the length of the time series.[3] We also report results for the case of calculating the full DTW matrix. In order to judge whether the difference is statistically significant, we used paired t-test with p-value of 0.05.

The results in Table 1 show that our approach significantly outperformed the baselines in the vast majority of the examined cases. There are only two exceptions: once our approach outperformed the baselines, but the difference was not significant (DistalPhalanxOutlineAgeGroup, $k = 5$), while in case of DistalPhalanxTW with $k = 10$, the baselines performed better, but the difference is not significant.

5. Conclusions and Outlook

In this work, we focused on the warping window size for DTW distance. We proposed an approach to select the warping window size for each time series *individually*. Our approach selects the warping window size in a way that minimize the detrimental effect of bad hubs, which were shown to account for a surprisingly large fraction of the overall classification error.

We implemented our experiments in Python and used Cython in oder to calculate DTW distances efficiently. Additional optimization may be required for large scale experiments and real-world applications, including the approximation of $GN_{k,w}(T)$ and $BN_{k,w}(T)$ values using nearest neighbor descent (Bratić et al., 2018).

In our future work, we plan to investigate further scoring functions based on $GN_{k,w}(T)$ and $BN_{k,w}(T)$ and perform experiments in context of other classifiers and applications (i.e., on additional datasets).

Acknowledgements. This work was supported by the project no. 20460-3/2018/FEKUTSTRAT within the Institutional Excellence Program in Higher Education of the Hungarian Ministry of Human Capacities.

[1] http://www.timeseriesclassification.com/

[2] With 10-fold cross-validation, we mean that the data is partitioned into 10 splits, out of which 9 serve as the training data and the remaining one is used as test data. The experiments are repeated 10-times, in each round of the 10-fold cross-validation, a different split is used as test data. With 10×10-fold cross-validation we mean that the above 10-fold cross-validation is repeated 10-times, each time using a different initial partitioning of the data.

[3] We note that we also performed experiments with static WWS of 1% and 0% (Manhattan distance). As these results did not change the conclusions, we omit them for simplicity.

Table 1. Classification accuracy (averaged over the 10×10 folds) $\pm$ its standard deviation for k-nearest neighbor in case of calculating full DTW, DTW with global warping widow sizes of 5% and 10% and individualized warping window sizes (IWW). The best approach is underlined. The symbols • and ○ denote whether the difference between IWW and the baselines is statistically significant (•) or not (○) based on paired t-test with p-value of 0.05.

DATASET	FULL DTW	WWS=5 %	WWS=10 %	IWW	
		$k = 5$			
ARROWHEAD	0.794±0.087	0.817±0.078	0.799±0.087	0.871±0.072	• / • / •
CAR	0.672±0.120	0.708±0.136	0.685±0.116	0.747±0.105	• / • / •
DISTALPHALANXOUTLINEAGEGROUP	0.819±0.049	0.817±0.047	0.818±0.049	0.824±0.042	○ / ○ / ○
DISTALPHALANXTW	0.725±0.053	0.727±0.052	0.727±0.052	0.753±0.052	• / • / •
DODGERLOOPDAY	0.398±0.125	0.448±0.127	0.411±0.126	0.537±0.121	• / • / •
		$k = 10$			
ARROWHEAD	0.792±0.083	0.819±0.075	0.800±0.084	0.870±0.069	• / • / •
CAR	0.612±0.139	0.680±0.134	0.624±0.141	0.697±0.134	• / • / •
DISTALPHALANXOUTLINEAGEGROUP	0.808±0.048	0.813±0.046	0.808±0.048	0.821±0.044	• / • / •
DISTALPHALANXTW	0.752±0.051	0.752±0.051	0.752±0.051	0.749±0.051	○ / ○ / ○
DODGERLOOPDAY	0.384±0.125	0.405±0.106	0.375±0.122	0.510±0.113	• / • / •

References

Abanda, A., Mori, U., and Lozano, J. A. A review on distance based time series classification. *Data Mining and Knowledge Discovery*, 33(2):378–412, 2019.

Bratić, B., Houle, M. E., Kurbalija, V., Oria, V., and Radovanović, M. Nn-descent on high-dimensional data. In *8th International Conference on Web Intelligence, Mining and Semantics*, pp. 20. ACM, 2018.

Buza, K., Nanopoulos, A., and Schmidt Thieme, L. Time-series classification based on individualised error prediction. In *13th International Conference on Computational Science and Engineering*, pp. 48–54. IEEE, 2010.

Buza, K., Koller, J., and Marussy, K. Process: projection-based classification of electroencephalograph signals. In *International Conference on Artificial Intelligence and Soft Computing*, pp. 91–100. Springer, 2015.

Dau, H. A., Silva, D. F., Petitjean, F., Forestier, G., Bagnall, A., Mueen, A., and Keogh, E. Optimizing dynamic time warpings window width for time series data mining applications. *Data Mining and Knowledge Discovery*, 32(4): 1074–1120, 2018.

Kate, R. J. Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30(2):283–312, 2016.

Meszlényi, R., Peska, L., Gál, V., Vidnyánszky, Z., and Buza, K. Classification of fmri data using dynamic time warping based functional connectivity analysis. In *24th European Signal Processing Conference (EUSIPCO)*, pp. 245–249. IEEE, 2016.

Meszlényi, R. J., Buza, K., and Vidnyánszky, Z. Resting state fmri functional connectivity-based classification using a convolutional neural network architecture. *Frontiers in Neuroinformatics*, 11:61, 2017.

Radovanović, M., Nanopoulos, A., and Ivanović, M. Time-series classification in many intrinsic dimensions. In *2010 SIAM International Conference on Data Mining*, pp. 677–688. SIAM, 2010.

Ratanamahatana, C. A. and Keogh, E. Three myths about dynamic time warping data mining. In *2005 SIAM International Conference on Data Mining*, pp. 506–510. SIAM, 2005.

Sakoe, H. and Chiba, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal processing*, 26 (1):43–49, 1978.

Tomašev, N., Buza, K., Marussy, K., and Kis, P. B. Hubness-aware classification, instance selection and feature construction: Survey and extensions to time-series. In *Feature Selection for Data and Pattern Recognition*, pp. 231–262. Springer, 2015.

Xi, X., Keogh, E., Shelton, C., Wei, L., and Ratanamahatana, C. A. Fast time series classification using numerosity reduction. In *23rd International Conference on Machine Learning*, pp. 1033–1040. ACM, 2006.

Xue, Y., Zhang, L., Tao, Z., Wang, B., and Li, F. An altered kernel transformation for time series classification. In *International Conference on Neural Information Processing*, pp. 455–465. Springer, 2017.