Fast and Accurate Temporal Data Classification using Nearest Weighted Centroid

Saeid Soheily-Khah
SKYLADS, R&D LAB
Paris, France
email:saeid@skylads.com

***Abstract*—Classification is an important task in the field of machine learning and data mining due to its wide applications. Among many algorithms to classify temporal data, the k-nearest neighbor with the dynamic time warping (DTW) is performant. However, it is very time consuming as it needs to compare time series query with all training samples. Since the computational complexity of the DTW is quadratic, these comparisons pose some problems for large data, either it is not applicable or it takes a lot of time to classify the data. To obtain fast accurate results by reducing the data size, classification based on nearest centroid could be one solution. In this paper, to speed up the nearest neighbor classification, and specially to make it applicable for huge datasets, we propose a fast accurate k-nearest weighted centroid classifier. Using generalized k-means-based clustering within each class, we obtain a small number of weighted centroids per class. A wide range of datasets is used to evaluate the efficiency of the proposed classifier, which outperforms the alternative ones.**

***Index Terms*—Temporal data classification, dynamic time warping (DTW), nearest neighbor, nearest centroid.**

I. Introduction

Recent years, the problem of temporal data classification has attracted great interest in domains as diverse as medicine, finance and industry. Many algorithms have been proposed for the time series classification problem, including decision trees [1], neural networks [2], Bayesian classifiers, support vector machines [3], nearest neighbors [4], [5], etc. The k-nearest neighbors is very simple to understand and works well in practice, but it is a lazy learning algorithm. It needs to compare each time series with the whole training samples. More exactly, to classify new data, all the training data are needed. Therefore, because of the time complexity of DTW, the algorithm for large dataset either it is not applicable or it takes a lot of time to classify the data.

Data reduction [8], [9], by discard a large fraction of data, offers a possibility to speed up the nearest neighbor classification. If we are careful in choosing which data we discard, we can maintain high accuracy, in some cases actually improving the accuracy, with significantly reducing the classification time complexity. In such a way, classification based on nearest centroid (or prototype) could be one solution [10], [11]. This allows us to condense and abridge huge datasets into smaller datasets as small as a single instance per class. Earlier studies have shown that, summarizing a set of data by averaging condense better than choosing just the most important data point in the set (e.g. medoid[1]) [12].

Averaging a set of time series under time warp is a challenging task, as it faces the problem of multiple sequence alignments [13]–[15]. Finding the solution to the multiple sequence alignment problem under time warp has been shown to be an NP-complete problem [16], with a complexity of $O(T^N)$, that increases exponentially with the number of time series N, and the time series length T. The complexity of this approach prevents its use. To avoid search by dynamic programming, one can obtain the global average (or centroid) of a set of time series by combining progressively pairwise time series averaging. Such progressive approaches suffer of early error propagation through the set of pairwise centroid combinations. The iterative approaches proceed similarly to the progressive ones, but mainly reduce the error propagation by repeatedly refining the centroid and realigning it to the initial time series [17]. In general, the main progressive and iterative approaches are of heuristic nature with no guarantee of optimality and are limited to the DTW. Even, if the provided approximations are accurate for globally similar time series, they are generally poor for time series which share local characteristics with distinctive global behaviors.

This paper introduces a fast and accurate k-nearest centroid classifier through a tractable centroid estimation based on an extension of the standard DTW, proposed in [18], to consider both global and local temporal differences. The centroid estimation is formalized as an optimization problem. The developed solution allows to estimate not only the centroid, but also its weight vector that indicates representativeness of the centroid elements. A very diverse collection of public datasets is used to evaluate the efficiency of the proposed classification, which outperforms the alternative ones. In the next section, we review the well-known related works. Then, we present the problem statement, prior to introduce our solution. Section V presents the experiments conducted for comparison purposes and discuss the results obtained. Lastly, Section VI concludes the paper.

The main contributions of this work are to i) propose a fast accurate nearest weighted centroid classifier under the extended DTW to capture both global and local differences,

[1]A medoid is a time series in a set that minimizes sum of the distances to all other time series within the same set.

978-1-5386-9576-0/18/$31.00 ©2018 IEEE
DOI 10.1109/ICCAIRO.2018.00024

ii) present a tractable meaningful centroid estimation based on the extended dynamic time warping to condense large datasets, iii) demonstrate that the proposed classifier speed up the nearest neighbor classification and outperforms the alternative methods through a deep analysis and experiments.

II. Related works

A commonly used technique to glean benefits of nearest neighbor classification, with reducing the time complexity, is to use nearest centroid classifiers [19]. The idea that the average of a set of objects is more representative than any individual object in the set is not new and has been proposed a century ago in [20]. By exploiting this idea in machine learning, the nearest centroid classifier generalizes the nearest neighbor classifier by replacing the set of neighbors with their centroid (i.e., average) to make it faster and, less intuitively, more accurate. By considering a small number of centroids rather than just one centroid, we can improve the accuracy. Here, we review some major approaches for averaging time series under DTW, to obtain a suitable centroid in the context of data reduction in classification. These methods are mainly derived from the multiple sequence alignment methods to address the challenging problem of aligning more than two time series [12], [21]–[23].

Gupta et al. in [24], proposed a progressive approach to estimate a global centroid by combining pairwise averaging, called "*NonLinear Alignment and Averaging Filters* (NLAAF)". Pairs of time series are first selected randomly, and then aligned according to the dynamic time warping to estimate their centroids. The same process is iterated on the centroids estimated,until one sequence is obtained as a global centroid. To avoid the bias induced by random selection, Niennattrakul et al. proposed a framework of shape averaging, called "*Prioritized Shape Averaging* (PSA)" based on hierarchical clustering [25], [26][2]. The pairwise time series averaging is guided by the dendrogram obtained thorough hierarchical clustering. Although this hierarchical averaging aims to remove the bias induced by random selection, growth length of the average sequence remains a problem. In addition, local averaging strategies may let an initial approximation error propagate throughout the averaging process. If the averaging process has to be repeated, the effects may dramatically alter the quality of the result. This is why a global approach is desirable, where time series would be averaged all together, with no sensitivity to their order of consideration.

A direct manner to estimate the centroid is proposed in Abdulla et al. [27], where a dynamic time warping between each time series and a reference one, generally the time series medoid, is first performed. The whole time series are then described in the representation space defined by the reference, where the global centroid can be estimated. This approach is iterated by Petitjean et al. [14] on the centroid estimated until its stabilization, called "*Dtw Barycenter Averaging* (DBA)". The aim is to minimize the sum of squared DTW distances from the average sequence to the set of time series. In particular, the DBA under time warp is a global approach that can average a set of time series all together [28].

Very recently, a majorize-minimize (MM) and stochastic sub-gradient (SSG) mean algorithm proposed in [29], [30]. Sub-gradient methods are nonsmooth optimization techniques [31] that operate very similar to the gradient descent methods, but they replace the gradient with a sub-gradient. Since, the authors claim that their algorithms are equivalent to the DBA algorithm, we have ignored them in our experiments. However, DBA is a batch and SSG a stochastic optimization method: While the DBA algorithm computes an exact sub-gradient on the basis of all sample time series, the SSG algorithm estimates a sub-gradient on the basis of a single randomly picked sample time series. Furthermore, the SSG algorithm is not a descent method, and during the optimization, the value of loss function can increase. In contrast, the DBA is a descent method [28]. While DBA converges to solutions after a finite number of updates, for SSG necessary conditions of optimality is needed to conclude local minimality.

As a summery, almost all the progressive and iterative methods are heuristic, with no guarantee of optimality. Lastly, even if the provided approximations are accurate for globally similar time series, they are in general poor for time series that share local characteristics with distinctive global behaviors. To circumvent these problems, we proposed in [18], [32] a fast tractable centroid estimation that captures both global and local temporal features under weighted dynamic time warping. It formalizes the multiple time series averaging as an optimization problem and propose a solution yielding a local optimum. We have shown that centroid estimation under WDTW outperforms all the existing averaging techniques. In this paper, we propose to use the developed solution of centroid estimation to speed up time series nearest neighbor classification.

III. Definition and problem statement

Given a large time series training samples, we want to propose the most fast accurate possible nearest centroid classifier, by condensing data into a small number of centroids that capture both global and local temporal features.

Let $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ be a set of time series $\mathbf{x}_i = (x_{i1}, ..., x_{iT})$ $i \in \{1, ..., N\}$ of length T, and $\mathbf{c} = (c_1, ..., c_T)$ be a centroid time series with an associated weight vector $\mathbf{w} = (w_1, ..., w_T)$ that indicates the importance of each element of $\mathbf{c}$. An alignment $\boldsymbol{\pi}$ of length $|\boldsymbol{\pi}| = m$ between two time series $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as the set of m couples of aligned elements of time series $\mathbf{x}_i$ to elements of $\mathbf{x}_j$: $\boldsymbol{\pi} = ((\pi_1(1), \pi_2(1)), (\pi_1(2), \pi_2(2)), ..., (\pi_1(m), \pi_2(m)))$, where the applications π_1 and π_2 obey to boundary and monotonicity conditions. In the following, we first give the definition of the weighted dynamic time warping measure, prior to the formalization of centroid estimation problem under this metric. We will denote $\mathbb{A}$ as the set of all alignments between two time series. The "*Weighted Dynamic Time Warping* (WDTW)

[2]Also called agglomerative hierarchical clustering (AHC).

between the time series $\mathbf{x}$ and the weighted time series $(\mathbf{c}, \mathbf{w})$ is defined by:

$$\min_{\boldsymbol{\pi}\in\mathbb{A}} \frac{1}{|\boldsymbol{\pi}|} \sum_{(t',t)\in\boldsymbol{\pi}} f(w_t)\,\varphi(x_{t'}, c_t) = \min_{\boldsymbol{\pi}\in\mathbb{A}} C(\boldsymbol{\pi}) = C(\boldsymbol{\pi}^*) \quad (1)$$

where $f : (0, 1] \rightarrow \mathbb{R}^+$ is a non-increasing function and $\varphi : \mathbb{R}\times\mathbb{R} \rightarrow \mathbb{R}^+$ is a positive, real-valued, dissimilarity function.

The above definition generalizes DTW to the case where different instants are weighted. The fact that the function f is non-increasing guarantees that the most important instants (i.e. the instants with the higher weights) of the centroid $\mathbf{c}$ should be privileged in the optimal alignment that minimizes the cost function C. Considering the metric defined above as dissimilarity measure, we obtained the following optimization problems aims to estimate an accurate centroid for each class as well as its weight vector, subject to: $\sum_{t=1}^{T} w_t = 1$ and $w_t > 0,\ \forall t$:

$$\arg\min_{\mathbf{c},\mathbf{w}} g(\mathbf{c}, \mathbf{w}) \;=\; \arg\min_{\mathbf{c},\mathbf{w}} \sum_{i=1}^{N} \mathrm{WDTW}(\mathbf{x}_i, (\mathbf{c}, \mathbf{w}))$$

IV. Solutions and algorithm

Given $\mathbf{w} = (w_1, ..., w_T)$ and $\boldsymbol{\Pi}^* = \{\boldsymbol{\pi}^*_{\mathbf{x}} \,/\, \mathbf{x} \in \mathbf{X}\}$, the centroid $\mathbf{c}$ that minimize $g(\mathbf{c}/\mathbf{w}, \boldsymbol{\Pi}^*)$, $\forall t,\ 1 \le t \le T$, is given by:

$$c_t = \frac{\displaystyle\sum_{\mathbf{x}\in\mathbf{X}} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t)\in\boldsymbol{\pi}^*_{\mathbf{x}}} x_{t'}}{\displaystyle\sum_{\mathbf{x}\in\mathbf{X}} \frac{|N(t, \mathbf{x})|}{|\boldsymbol{\pi}^*_{\mathbf{x}}|}} \quad (2)$$

where $\boldsymbol{\pi}^*_{\mathbf{x}}$ denotes the optimal alignment for $\mathbf{x} \in \mathbf{X}$ and $|N(t, \mathbf{x})| = \{t' \,/\, (t', t) \in \boldsymbol{\pi}^*_{\mathbf{x}}\}$ denotes the number of time instants of $\mathbf{x}$ aligned to time t of $\mathbf{c}$. The solution can thus be obtained by equating the partial derivative of the Lagrangian of $g(\mathbf{c}/\mathbf{w}, \boldsymbol{\Pi}^*)$ with respect to $\mathbf{c}$ to 0 and solving for $\mathbf{c}$.

The solution for weight $\mathbf{w}$ is obtained by equating the partial derivative of the Lagrangian of $g(\mathbf{w}/\mathbf{c}, \boldsymbol{\Pi}^*)$, with respect to $\mathbf{w}$ to 0 and solving for $\mathbf{w}$ as:

$$w_t = \frac{A_t^{\frac{1}{1+\alpha}}}{\sum_{t=1}^{T} A_t^{\frac{1}{1+\alpha}}},\ \text{with}\ A_t = \sum_{\mathbf{x}\in\mathbf{X}} \frac{1}{|\boldsymbol{\pi}^*_{\mathbf{x}}|} \sum_{(t',t)\in\boldsymbol{\pi}^*_{\mathbf{x}}} (x_{t'} - c_t)^2 \quad (3)$$

in which $f(w_t) = w_t^{-\alpha}$ is the introduced non-increasing warping function, where $\alpha \in \mathbb{R}^+$ controls the influence of the weighting scheme.

Algorithm 1 summarizes the steps required to solve the centroid estimation problem using WDTW. For algorithm 2, the main difference with the standard nearest centroid classifiers is that, the weight vector $\mathbf{w}_i$ for each centroid $\mathbf{c}_i$ should be learned firstly by algorithm 1. The weights associated to the centroids reflect the elements that are shared locally in a set of time series. In addition, in the classifying step for the proposed algorithm, we classify each $\mathbf{x} \in \mathbf{X}_{test}$ according to the proposed WDTW.

Algorithm 1 *Weighted Centroid Estimation*

Input: $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$
Initialization: $\mathbf{c}^{(0)}$ randomly selected from $\mathbf{X}$, $\mathbf{w}^{(0)} = (\frac{1}{T}, ..., \frac{1}{T})$, $p = 0$
$\boldsymbol{\Pi}^{*(0)}$: optimal alignments between $\mathbf{X}$ and $(\mathbf{c}^{(0)}, \mathbf{w}^{(0)})$
repeat
  $p \leftarrow p + 1$
  Update $c_t^{(p)}$, $\forall t, 1 \le t \le T$ using Eq. 2
  Update $w_t^{(p)}$, $\forall t, 1 \le t \le T$ using Eq. 3
  Update $\boldsymbol{\Pi}^{*(p)}$: optimal alignments between $\mathbf{X}$ and $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)})$
until $(\mathbf{c}^{(p)}, \mathbf{w}^{(p)}) \approx (\mathbf{c}^{(p-1)}, \mathbf{w}^{(p-1)})$
Return $\mathbf{c}^{(p)}, \mathbf{w}^{(p)}$

Algorithm 2 *Nearest Weighted Centroid Classification*

Input: $\mathbf{X}_{train} = \{\mathbf{x}_1, ..., \mathbf{x}_{N_{train}}\}$,
$\mathbf{Y}_{train}$ = labels,
$\mathbf{X}_{test} = \{\mathbf{x}_1, ..., \mathbf{x}_{N_{test}}\}$,
k = class number
for i : 1 to k **do**
  $(\mathbf{c}_i, \mathbf{w}_i)$ = **Weighted Centroid Estimation**$(\mathbf{X}_{train}|\mathbf{Y}_{train} == i)$
end for
for $\forall \mathbf{x}_j \in \mathbf{X}_{test}$ **do**
  classify $\mathbf{x}_j$ by nearest weighted centroid using Eq. 1
end for
Return class labels for $\forall\, \mathbf{x} \in \mathbf{X}_{test}$

As we explained in the introduction, the nearest centroid classifier could be one solution to reduce the classification time complexity, and by considering a small number of centroids rather than one, we can improve the accuracy of the classifier. While for some datasets, a single centroid may capture the nature of the class, for other datasets, it may require a small number of centroids per class. Using clustering within each class to obtain small number of centroids one can outperform the accuracy of nearest neighbor classifier. Generalized k-means-based clustering proposed in [18] allows us to improve the accuracy of the classifier, when we need more than one centroid per class. In the following for our experiments, we consider situations with just one centroid (prototype) per class for all temporal data classification approaches.

V. Experimental study

In this section, we evaluate and compare the effectiveness of the proposed method for temporal data classification. For the comparison, first the classification error rate and subsequently, the required run time are evaluated. Our experiments are conducted on 24 public datasets[3]. The experimentations are first carried out on standard well-known datasets, which define a favorable case for the classification and centroid estimation task as time series behave similarly within classes (e.g. CBF, CC and GUNPOINT). We then consider more complex datasets (e.g. BME, UMD and CONSSEASON). They are composed of time series that behave differently with in the same classes while sharing several local characteristics.

In particular, we have two challenging case for classification: 1) complex datasets, where for each class we have different behaviors within the class, and 2) large datasets due to the time complexity. In the following, we first describe the validation process, prior to present and discuss the results obtained.

[3] BME, UMD, CONSSEASON in [32], the rest in cs.ucr.edu/~eamonn/time_series_data/

A. Validation process

We compare here the proposed classification based on nearest weighted centroid algorithm with some well-known averaging techniques (i.e. NLAAF, PSA and DBA) used for nearest centroid classifiers, the standard nearest neighbor classification under DTW, and the nearest neighbor classification using different kernels (i.e., k_{DTAK}, k_{GA} [33]–[35]). We focus on these methods because they constitute the most frequently used variants of temporal classification. For our comparison, we rely on the classification time consumption, while the main contribution of this work is to speed up temporal classification. To verify the accuracy of the method, we have confidence in the classification error rate. The lower the error rate, the better the agreement is. Lastly, we consider a goodness criteria which is the ratio of "$fastness$" to "$accuracy$". For this, we define "$fastness$" as the normalized time consumption (so that all values are in (0,1)), and (1 - error rate) is the "$accuracy$". The lower value is the better classifier.

B. Results for time series classification

Here, we evaluate the relevance of the proposed nearest weighted centroid in a classification context. The classification error rate and time consumption for different methods are represented in Tables I and II. Results in bold correspond to the best values.

Classification error rate Based on the classification error rate (Table I), for k=1, in overall nearest neighbor classification methods lead to the best results, followed by the proposed nearest weighted centroid approach (WDTW). Using a small number of weighted centroids per class obtained by generalized k-means based clustering within each class, presented in [18], we can beat the nearest neighbor classifier. For example, for dataset OSULEAF, if we suppose k=2 and we consider two weighted centroids per class, using generalized k-means based clustering, we beat even all the nearest neighbors classifiers by error rate = 0.295.

The experiments show that the proposed nearest weighted centroid classifier significantly outperforms all the alternative nearest centroid classifications. To validate our claim, in Table I, we ignored the nearest neighbor classifiers (grey colors) and considered the major well-known nearest centroid classification methods. As one can note, classification using WDTW yields the best classification error rate overall (21 datasets out of 24). Table I shows the results for k=1, where we defined just one centroid per class.

Complexity and time considerations Table II shows the time consumption for different classifiers. As one can note, the nearest centroid using DBA is the fastest method, similar to the proposed nearest weighted centroid classifier. The small difference is related to the weight computation for each centroid. While we significantly improve the accuracy, one can ignore this non-significantly difference in run time.

As we mentioned in Section II, the main drawback of NLAAF and PSA lies in the growth of their resulting lengths. While nearest centroid classifier normally should be faster than nearest neighbor ones, the nearest centroid classifiers

TABLE I: Classification error rate (k=1)

	k-nearest neighbor			k-nearest centroid			k-nearest
	kernel	kernel	standard	NLAAF	PSA	DBA	weighted centroid
	k_{DTAK}	k_{GA}	DTW	DTW	DTW	DTW	WDTW
ADIAC	0.425	0.407	0.404	0.567	0.527	0.517	**0.465**
BEEF	0.500	0.500	0.500	0.600	**0.533**	0.567	**0.533**
BME	0.127	0.200	0.140	0.360	0.360	0.360	**0.333**
CAR	0.267	0.200	0.283	0.583	0.467	0.367	**0.337**
CBF	0.013	0.183	0.011	0.106	0.220	0.036	**0.031**
CC	0.023	0.177	0.023	0.113	0.377	**0.027**	**0.027**
COFFEE	0.179	0.179	0.179	**0.464**	**0.464**	**0.464**	**0.464**
CONSSEASON	0.089	0.022	0.111	0.350	0.389	0.133	**0.022**
ECG200	0.220	0.110	0.210	**0.260**	**0.260**	**0.260**	**0.260**
FACEFOUR	0.148	0.102	0.182	0.511	0.250	**0.136**	0.148
FISH	0.166	0.103	0.177	0.503	0.457	0.349	**0.343**
GUNPOINT	0.087	0.020	0.093	0.397	0.387	0.286	**0.213**
LIGHTING2	0.131	0.246	0.115	**0.377**	**0.377**	**0.377**	**0.377**
LIGHTING7	0.274	0.274	0.315	0.399	0.384	0.410	**0.383**
MEDICALIMAGES	0.275	0.259	0.275	0.611	**0.533**	0.564	0.536
OLIVEOIL	0.133	0.133	0.133	0.367	0.433	0.167	**0.133**
OSULEAF	0.368	0.306	0.430	0.794	0.842	0.521	**0.498**
PLANE	0.000	0.009	0.000	0.152	0.162	**0.009**	**0.009**
SWEDISHLEAF	0.222	0.266	0.216	0.550	0.511	0.326	**0.296**
SYMBOLES	0.050	0.051	0.055	0.109	0.084	0.054	**0.044**
TRACE	0.000	0.010	0.000	0.060	0.028	0.020	**0.000**
TWOPATTERNS	0.000	0.010	0.000	0.093	0.090	**0.060**	0.070
UMD	0.118	0.035	0.132	0.446	0.458	0.430	**0.386**
YOGA	0.317	0.323	0.367	0.430	0.530	0.467	**0.333**

TABLE II: Classification time consumption (k=1)

	k-nearest neighbor			k-nearest centroid			k-nearest
	kernel	kernel	standard	NLAAF	PSA	DBA	weighted centroid
	k_{DTAK}	k_{GA}	DTW	DTW	DTW	DTW	WDTW
ADIAC	358.6	547.6	581.4	231.2	246.5	**131.2**	142.8
BEEF	23.7	29.5	26.8	79.2	88.2	**13.9**	15.0
BME	6.2	8.1	8.9	10.8	12.7	**3.3**	3.4
CAR	134.9	188.2	161.1	912.3	1049.8	**44.7**	45.9
CBF	43.6	49.8	55.7	50.4	55.3	**14.9**	15.0
CC	41.1	46.8	45.8	9.7	11.6	**2.4**	2.6
COFFEE	5.2	7.2	8.1	8.2	7.9	**0.7**	**0.7**
CONSSEASON	58.6	79.9	81.4	358.7	378.9	**2.2**	2.4
ECG200	9.7	11.8	11.5	9.4	10.4	**0.2**	**0.2**
FACEFOUR	22.4	40.9	33.0	49.8	54.0	**15.7**	16.9
FISH	584.9	955.1	865.1	1022.2	1107.8	**106.8**	107.4
GUNPOINT	15.3	27.3	20.4	39.3	41.2	**2.1**	2.2
LIGHTING2	191.8	297.2	225.1	7682.4	9666.5	**11.3**	12.6
LIGHTING7	50.9	76.8	70.4	132.2	155.6	**16.9**	17.2
MEDICALIMAGES	282.5	448.9	375.1	853.2	959.8	**20.7**	21.6
OLIVEOIL	32.2	45.8	45.2	26.2	29.3	**16.2**	16.4
OSULEAF	775.8	1244.9	1238.8	10877.3	20858.4	**107.7**	110.1
PLANE	29.2	37.0	27.6	8.6	8.5	**4.7**	4.8
SWEDISHLEAF	470.7	737.6	640.2	255.7	272.2	**40.8**	41.6
SYMBOLES	328.8	590.6	555.0	714.0	880.6	**256.3**	258.4
TRACE	74.4	95.4	101.2	202.3	283.9	**10.3**	10.4
TWOPATTERNS	67.8	106.1	84.2	48.1	41.2	**7.7**	7.9
UMD	9.8	12.6	14.3	12.3	13.9	**2.8**	3.0
YOGA	137.4	224.6	230.9	285.9	377.7	**33.4**	35.9

using NLAAF and PSA are highly time-consuming, because of the progressive increase of the centroid length during the pairwise combination process. The resulting centroid lengths for the remaining methods are equal to the length of the initial centroid (e.g., medoid).

Classification goodness criteria Lastly, for a global comparison considering both error rate and run time index, we define a goodness criteria as a ratio of "$fastness$" to "$accuracy$". The "$fastness$" (normalized time consumption), should be lower and the "$accuracy$" (1 - error rate), should be higher. Table III presents the classification average ranking on different methods, according to the defined goodness criteria. The best performance belongs to the proposed nearest weighted centroid classifier with the average rank of **1.25**, followed by the classifier using using DBA, with the average rank of

1.75, followed by the nearest neighbor classifier using k_{DTAK}. The last group, the classifiers which use nearest centroid with NLAAF and PSA yield the lowest performances.

TABLE III: Classification average ranking (k=1)

	k-nearest neighbor			k-nearest centroid			k-nearest
	kernel	kernel	standard	NLAAF	PSA	DBA	weighted centroid
	k_{DTAK}	k_{GA}	DTW	DTW	DTW	DTW	WDTW
AVERAGE RANK	3.46	5.17	4.83	5.65	5.90	1.75	**1.25**

VI. CONCLUSION

The nearest neighbor classification under time warp poses some problems for large training data due to its time complexity. Recently different classification based on nearest centroid introduced as one solution. This work proposed a new nearest weighted centroid classification approach, as each centroid could be a good representativeness of a class. The experiments on a wide range public datasets show the benefits of the method proposed. Firstly, we have shown that the proposed nearest weighted centroid classification is more accurate and outperforms all the nearest centroid classifiers which used the well-known existence averaging techniques. Using generalized k-means-based clustering within each class, to obtain small number of centroids per class, even we outperform the accuracy of nearest neighbor classifiers. Later, we proposed a goodness criteria to evaluate the classifiers by considering both classification error rate and run time complexity. Finally, using this evaluation measure, we have shown that the proposed nearest weighted centroid classifier outperforms the other alternative methods. In the near future, we will improve the time and space complexity of classification tasks, in the context of comparison of time series, by using segment of interests, which are instances with higher weights, instead of considering the whole time series. The second direction to further explore includes finding a symmetric variant of the WDTW that is originally introduced as an asymmetric method.

REFERENCES

[1] J. Rodríguez , C.J. Alonso: Interval and dtw-based decision trees. ACM-SAC (2004)

[2] A. Nanopoulos and R. Alcock R. and Y. Manolopoulos: Feature-based classification of time-series data. Nova Science Publishers Inc. (2001)

[3] Y. Wu, C.T. Yeh, W.C. Hung, C.Y. Tang: Gaze direction estimation using support vector machine with active appearance model. Multimedia Tools Appl (2014)

[4] T.M Cover, P.E. Hart: Nearest neighbor pattern classification". IEEE (1967)

[5] B.V. Dasarathy: Nearest Neighbor (NN) Norms. IEEE Computer Society P. (1991)

[6] A. Bagnal and J. Lines: An experimental evalution of nearest neighbor time series classification. Tech. Rep. CMP-C14-01 (2014)

[7] X. Xi, E. Keoge, C. Shelton, L. Wei and C. Ratanamahatana: Fast time series classification using numerosity reduction. Machine learning (2006)

[8] D.R. Wilson and T.R. Martinez: Instance Pruning Techniques. ICML (1997)

[9] E. Pekalska and R. Duin and P.Paclik: Prototype Selection for Dissimilarity-Based Classifiers. Pattern Recognition, pp. 189-208 (2006)

[10] A. Dabney: Classification of microarrays to nearest centroids. Bio-inf. (2005)

[11] A. Dabney, J.D. Storey: Optimality driven nearest centroid classification (2007)

[12] F. Petitjean, G. F., G.I. W., A. N., Y. C. and E. Keogh: Dynamic time warping averaging of time series allows faster and more accurate classification. ICDM (2014)

[13] V. Hautamaki, P. Nykanen and P. Franti: Time-series clustering by approximate prototypes. 19th International Conference on Pattern Recognition (2008).

[14] F. Petitjean, A. Ketterlin and P. GanÇarski: A global averaging method for dynamic time warping, with applications to clustering. Pattern Recognition (2011)

[15] F. Zhou and F. De la Torre: Generalized time warping for multi-modal alignment of human motion. IEEE, Computer Vision and Pattern Recognition (2012)

[16] L. Wang, T. Jiang: On the complexity of multiple sequence alignment. JCB (1994)

[17] S. Soheily-Khah, A. Douzal-Chouakria, E. Gaussier: A comparison of progressive and iterative centroid estimation approaches under time warp Advanced Analysis and Learning on Temporal Data, Springer International Publishing, 144-156 (2016)

[18] S. Soheily-Khah, A. Douzal-Chouakria, E. Gaussier: Generalized k-means-based clustering for temporal data under weighted and kernel time warp. Pattern Recogn. Lett., Elsevier Science Inc., vol. 75, 63–69 (2016)

[19] R. Tibshirani, T. Hastie, B. Narasimhan and G. Chu: Diagnosis of multiple cancer types by shrunken centroids of gene expression. PNAS (2002)

[20] F. Galton: Vox populi. Nature, vol 75, no. 1949 (1907)

[21] C. Notredame, D.G. Higgins and J. Heringa: T-Coffee: A novel method for fast and accurate multiple sequence alignment. Journal of Molecular Biology (2000)

[22] S.H. Sze, Y. Lu and Q. Yang: A polynomial time solvable formulation of multiple sequence alignment. Journal of Computational Biology (2006)

[23] S. Soheily-Khah, A. Douzal-Chouakria and E. Gaussier: Progressive and Iterative Approaches for Time Series Averaging. ECML-PKDD (AALTD), Portugal (2015)

[24] L. Gupta, D. Molfese, R. Tammana, P. Simos: Nonlinear alignment and averaging for estimating the evoked potential. IEEE T. on Biomedical Engineering (1996)

[25] V. Niennattrakul, C. Ratanamahatana: On clustering multimedia time series data using k-means and dynamic time warping. MUE-IEEE, Pages 733–738 (2007)

[26] V. Niennattrakul, C. R.: Shape averaging under time warping. ECTI-CON (2009)

[27] Abdulla, W.H. and Chow, D. and Sin, G.: Cross-words reference template for DTW-based speech recognition systems. Proc. TENCON (2003)

[28] F. Petitjean, G. Forestier, G. Webb, A. Nicholson, Y. Chen and E. Keogh: Faster and more accurate classification of time series by exploiting a novel dynamic time warping averaging algorithm. Knowledge and Information Systems (KAIS), Springer, Vol 47, pages 1–26 (2016)

[29] D. Schultz and B. Jain: Nonsmooth analysis and subgradient methods for averaging in dynamic time warping spaces. arXiv preprint arXiv:1701.06393 (2017)

[30] D. Hunter and K. Lange: A tutorial on MM algorithms. The American Statistician, vol 58 (1), pages 30–37 (2004)

[31] A. Bagirov, N. Karmitsa, and M.M. Makela: Introduction to Nonsmooth Optimization. Springer International Publishing (2014)

[32] S. Soheily-Khah: Generalized k-means-based clustering for temporal data under time warp. Université Grenoble Alpes, Theses, https://tel.archives-ouvertes.fr/tel-01680370 (2016)

[33] H. Shimodaira, K. Noma, M. Nakai and Sh. Sagayama: Dynamic time-alignment kernel in support vector machine. NIPS, Vol. 14, Pages 921–928 (2002)

[34] C. Bahlmann, B. Haasdonk and H. Burkhardt: Online handwriting recognition with support vector machines-a kernel approach. IEEE (2002)

[35] M. Cuturi, J.-P. Vert, Øystein Birkenes and T. Matsui: A kernel for time series based on global alignments. ICASSP (2007)