DTEC: Distance Transformation Based Early Time Series Classification

Liuyi Yao * Yaliang Li† Yezheng Li ‡ Hengtong Zhang* Mengdi Huai §
Jing Gao* Aidong Zhang§

Abstract

In many time-sensitive applications, knowing the classification results as early as possible while preserving the accuracy is extremely important for further actions. Shapelet-based early classification methods are popular due to their natural interpretability. However, most of the existing shapelet-based methods ignore the distance information between the shapelets and the time series. The distance information, though may contain some noise, can reflect more information between the shapelets and the time series. Some existing works adopt the distance information, but are not robust to the noise in the distance information. To tackle this challenge, we present a novel distance transformation based early classification (DTEC) framework, which transfers the original time series into the distance space. Upon the distance space, a probabilistic classifier is trained, and a novel classification criterion confidence area is proposed in order to overcome the noise brought by the training phase and the dataset. The effectiveness of the proposed framework is validated on three time series benchmarks as well as the extensive datasets selected from UCR time series archive.

1 Introduction

In many time-sensitive scenarios, people tend to know the results as early as possible so that actions can be taken in time. One important application is the acute disease detection, especially heart failure detection. Every year, about 735,000 Americans have a heart attack and about 610,000 people die of heart disease[1]. The medical professionals point out that when the heart attack occurs, the more time that passes without treatment, the greater the damage to the heart muscle[2]. Therefore, the earlier the patients' abnormal heart activities are detected, more timely the medical intervention can be conducted. Hence, it is non-trivial to develop classification methods that can provide early prediction, while not sacrifice the classification accuracy.

Stimulated by the urgent demands of our daily life, early time series classification gains much attention from researchers in the recent years. The main goal of early classification is to make the classification as early as possible with the satisfactory accuracy. In [29], Xing et. al. point out that early time series classification is a tradeoff between accuracy and earliness. Furthermore, Ghalwash et. al. in [7] regard earliness, interpretability, and uncertainty estimation as three highly desirable properties for the early time series classification methods. Some existing works [28, 29, 25] sacrifice one property for the others, and cannot guarantee all three properties at the same time. The most popular methods are shapelet-based methods [30, 6, 7, 11], whose common characteristic is that the threshold needs to be carefully designed to determine whether a time series contains a specific shapelet. However, in the case that there are no obvious distinguish patterns between different classes, it is hard to determine the threshold. Therefore, the distance information between the shapelets and the time series becomes more valuable.

Sangnier et. al. adopt the distance (similarity) information between shapelets (landmark) and the time series in [21]. However, this method still relies on the user-defined classification criterion (i.e., when the model can make classification decision) and is as well not robust to the noise existing in the distance transformation process. Some other works are designed to find discriminative timestamps [20], and at each detected discriminative timestamps, classifiers are trained to estimate the class probability. However, when the starting timestamp of training time series is not the same as the new coming time series (i.e., the new coming time series is not aligned with the training time series), the discriminative timestamps of the new coming time series would not coincide with that detected in the training time series. Thus it might account for the low accuracy or

*University at Buffalo, {liuyiyao, hengtong, jing}@buffalo.edu
†Alibaba Group, yaliang.li@alibaba-inc.com
‡University of Pennsylvania, yezheng@sas.upenn.edu
§University of Virginia, {mh6ck, aidong}@virginia.edu
[1] https://www.cdc.gov/heartdisease/facts.htm
[2] https://www.cdc.gov/heartdisease/heart_attack.htm

decision delay in the new coming time series.

To tackle the above challenges, we propose the **D**istance **T**ransformation based **E**arly **C**lassification (DTEC) framework. DTEC transfers time series into the distance space with the shapelets as its bases, and trains a probabilistic classifier (i.e., multi-class logistic regression) upon the distance space. By transforming time series into distance space, the distance information between the time series and shapelets can be fully explored. In order to eliminate the potential noise in the process of distance transformation and classifier training, DTEC devises a new classification criterion called confidence area. Distinguished from the existing classification criterion that outputs the result only if the probability is higher than the user provided threshold, the confidence area takes the sustained strength of the class probability into account. Therefore, the perturbation brought by noise can be alleviated, which allows a more reliable classification result. Besides, to obtain the earliness and interpretability properties, a two-step feature (shapelet) selection is embedded in the training phase: The first step is the candidate shapelets extraction which forms the bases of the distance space; The second step is the earliness-aware sparse group lasso which is applied to select important and early shapelets in the decision procedure. Moreover, because the timestamps examined by the classifier are determined by the distance variation which only depends on the new coming time series, the timestamp alignment issue mentioned above is addressed.

2 Methodology

In this section, we first define the early classification problem, followed by the introduction of the proposed DTEC framework. The rest of this section illustrates each component of the framework in detail.

2.1 Problem Setting. A time series with length T is denoted as $\mathbf{S} = \{s_1, s_2, \ldots, s_T\}$. The subsequence is a subset of the full time series containing consecutive timestamps, and the subsequence of S ranging from timestamp i to timestamp j is notated as $\mathbf{S}_{i,j} = \{s_i, s_{i+1}, \ldots, s_j\}$. The early classification aims to classify $\mathbf{S}$ accurately as early as possible.

2.2 Framework. Figure 1 illustrates the proposed framework. It contains two components, the training phase and the decision phase. The training phase can be offline. In the training phase, the candidate shapelets are first extracted from the training dataset to form the bases of the distance space. Next, the training time series are transferred into the distance space. Upon the distance space, a probabilistic classifier (i.e., the decision function) is trained. In the decision phase, the already observed subsequence in the new coming time series is first transferred to distance space and then fed into the decision function. If it is ready to make decisions according to the classification criterion, the system outputs the result. Otherwise, the system waits for the next timestamp and repeat the decision phase. The following sections introduce each component in detail.

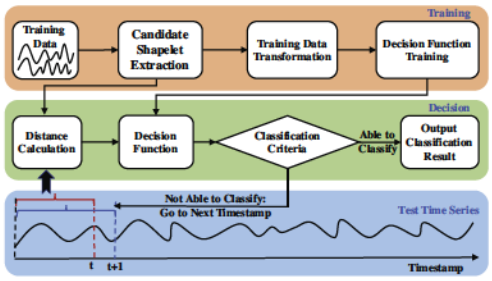


Figure 1: DETC Framework

2.3 Candidate Shapelet Extraction. Shapelets refer to the time series subsequences, which are in some sense maximal representative of a class [31, 16, 9, 26]. In early classification, the traditional ways to find shapelets examine all the frames available in the dataset [30, 21], which is time-consuming. Another drawback of the traditional shapelet extraction is that thresholds need to be carefully designed to determine whether a time series contains the shapelet [6], which may not be desirable when some classes do not contain distinguishable patterns. We propose a new way to select shaplelets, which doesn't heavily depend on the design of the threshold. The proposed shapelet selection method contains two parts. The first one is the coarse-grained selection, namely candidate shapelet extraction, introduced in this section. The other part is the fine-grained shapelet selection embedded in the decision function training procedure by adding earliness-aware sparse group lasso regularization, presented in Section 2.5.

There are three steps in the candidate shapelet extraction: reference time series selection, subsequence sampling, and candidate shapelet selection.

Reference Time Series Selection. The naive way to select reference time series is to randomly select several time series from each class. Alternatively, to avoid missing important patterns, we cluster the time series in the same class using the methods mentioned in [32, 24, 1], and select several representatives from each cluster as the reference time series.

Subsequence Sampling. Next step is to sample subsequences from the reference time series set. Let $\mathcal{L} = \{L_1, L_2, \ldots, L_n\}$ denote the length set, where

each element is the length of the subsequence to be sampled. For each reference time series, we sample N subsequences for every length L_i in $\mathcal{L}$. The class of the sampled subsequence is the same as the time series that the subsequence is sampled from. The earliness of the sampled subsequence is the ratio of the timestamp where the subsequence appears and the length of entire time series, i.e., $E(\mathbf{S}_{i,j}) = \frac{j}{\ell(\mathbf{S})}$, where $\mathbf{S}_{i,j}$ is the subsequence sampled from S, and $\ell(\mathbf{S})$ is the length of time series $\mathbf{S}$. The smaller the earliness is, the earlier the subsequence appears.

Candidate Shapelet Selection. Before introducing the last step, we give the definition of the distance of subsequences of possibly different lengths:

(2.1)
$$d\left(\mathbf{S}_{i,j}, \mathbf{S}_{k,l}\right) = \begin{cases} \text{if } j-i \geq l-k\text{:} \\ \min\limits_{m=i,\ldots,j-(l-k)} \left\|\mathbf{S}_{m,m+(l-k)} - \mathbf{S}_{k,l}\right\|_2; \\ \text{if } j-i < l-k\text{:} \\ \min\limits_{m=k,\ldots,l-(j-i)} \left\|\mathbf{S}_{m,m+(j-i)} - \mathbf{S}_{i,j}\right\|_2. \end{cases}$$

If we assume the length of $\mathbf{S}_{i,j}$ is shorter than $\mathbf{S}_{k,l}$, the distance of the two subsequences is the minimum ℓ_2 distance between $\mathbf{S}_{i,j}$ and all length $j-i+1$ subsequences of $\mathbf{S}_{k,l}$.

With the distance function defined above, we can calculate the distance between every sampled subsequence and all the time series in the dataset. For every sampled subsequence, the distances can be divided into two lists. One is the in-class distance list, which contains the distances to the time series that share the same class with the subsequence. The other is the out-class distance list, which records the distances to the time series whose classes are not the same as the subsequence. For subsequence $\mathbf{S}_{i,j}$, if the third quartile of in-class distance list is less than the first quartile of out-class distance list, then $\mathbf{S}_{i,j}$ is selected as the candidate shapelet. This preliminary selection procedure is reasonable as a good shapelet should have small in-class distance and large out-class distance.

The selected subsequences form the candidate shapelet set. For notation simplicity, the candidate shapelet set is denoted as $\mathcal{C} = \{C_h\}_{h=1}^{H}$, where $\mathbf{C}_h$ is the h-th selected candidate shapelet and H is the total number of subsequences in the candidate shapelet set. And the earliness vector of the candidate shapelet set is denoted as $\mathbf{E} = [e_1, e_2, \ldots, e_H]$, where e_h is the earliness of the candidate shapelet $\mathbf{C}_h$.

2.4 Data Transformation and Label Assignment. Data transformation is related to both training phase and decision phase. In the training phase, the time series are transferred offline, while in the decision phase, the distance transformation is applied online, i.e., when a new timestamp comes, the system updates the distance between the observed subsequence and the candidate shapelet set.

Compared with the traditional methods that use a threshold to determine whether a time series contains the shapelet, we project the time series into the distance space with candidate shapelets as its bases, in order to fully explore the distance information between the time series and the candidate shapelet set. The projection procedure is $\mathcal{T} : \mathbf{S_{1,T}} \xrightarrow{\mathcal{C}} \mathbf{D}$, where $\mathbf{S_{1,T}} \in \mathbb{R}^{1\times T}$ and $\mathbf{D} \in \mathbb{R}^{(T-\ell_{\mathcal{C}}+1)\times H}$. $\ell_{\mathcal{C}}$ is the maximum subsequence length in the candidate shapelet set, and $D_{i,j} = d\left(\mathbf{S}_{1,\ell_{\mathcal{C}}+i-1}, C_j\right)$.

Change-Point. According to Eqn. (2.1), the distance matrix is non-increasing over time: $\mathbf{D}_{i,h} \geq \mathbf{D}_{i+1,h}, \forall i = \ell_{\mathcal{C}}, \ldots, T-1; h = 1, 2, \ldots, H$. Thus, we can define the change point as: The timestamp t is a change-point if $\exists h$, s.t. $\mathbf{D}_{t,h} < \mathbf{D}_{t-1,h}$. Only the change-points are taken into consideration, because if one timestamp is not a change-point, the transferred distance remains the same since the last change-point.

Label Assignment. In the training set, labels are needed at each timestamp. we assume that the label at each timestamp is gradually close to the label at the last timestamp, since in the distance space, the distance matrix $\mathbf{D}$ is non-increasing. Thus, the label $\mathbf{Y}_{t,j}$, which is the probability that time series $\mathbf{S}$ belongs to the j-th class at each timestamp t, is assigned as: If $t = 1$: $\mathbf{Y}_{1,j} = \frac{1}{K}$, where K is the number of total classes; If t is a change-point: $\mathbf{Y}_{t,K_S} = \frac{1}{K} + \frac{K-1}{K}\frac{\sum_{j=1}^{H}(\mathbf{D}_{1,j}-\mathbf{D}_{t,j})}{\sum_{j=1}^{H}(\mathbf{D}_{1,j}-\mathbf{D}_{T,j})}$ and $\mathbf{Y}_{t,j\neq K_S} = \frac{1}{K-1}(1-\mathbf{Y}_{t,K_S})$, where K_S is the class index that $\mathbf{S}$ belongs to.

2.5 Decision Function. We adopt the multinomial logistic regression as the decision function, for its ability to output the probability of each class. If there are K classes in total, the decision function at timestamp t is defined as follows:

(2.2)
$$Pr(\mathbf{S}_{1,t} = K) = \frac{1}{1+\sum_{k=1}^{K-1}\exp(\mathbf{W}_{k,1} + \mathbf{W}_{k,2:H+1}\mathbf{D}'_{t,:})};$$
$$Pr(\mathbf{S}_{1,t} = k, k \neq K) = Pr(\mathbf{S}_{1,t} = K)\exp\left(\mathbf{W}_{k,1} + \mathbf{W}_{k,2:H+1}\mathbf{D}'_{t,:}\right),$$

where $\mathbf{W}$ is the parameter matrix and $\mathbf{W} \in \mathbb{R}^{K-1,H+1}$. $\mathbf{D}'_{t,:}$ denotes the t-th row vector in distance matrix $\mathbf{D}'$.

Loss Function. By dividing $Pr(\mathbf{S_{1,t}} = \mathbf{K})$ and taking the logarithm to both sides of Eqn. (2.2), we get: $\forall k \neq K$, $\ln\frac{Pr(\mathbf{S}_{1,t}=k)}{Pr(\mathbf{S}_{1,t}=K)} = \mathbf{W}_{k,1} + \mathbf{W}_{k,2:H+1}\mathbf{D}'_{t,:}$. With the help of the above equation, the final loss function is summarized in the following equation:

(2.3)
$$\sum_{q=1}^{Q} ||\mathbf{D}^{(q)}\mathbf{W}' - \mathbf{L}^{(q)}||_F^2 + \lambda_1||\mathbf{W}diag(\mathbf{E})||_{2,1} + (1-\lambda_1)||\mathbf{W}||_1,$$

where $\mathbf{D}^{(q)}$ is the distance matrix of q-th time series $\mathbf{S}^{(q)}$ in the training set; $\mathbf{L}^{(q)} \in \mathbb{R}^{N_q \times (K-1)}$, and $\mathbf{L}^{(q)}_{i,k} = \ln \frac{\mathbf{Y}^{(q)}_{i,k}}{\mathbf{Y}^{(q)}_{i,K}}$, where N_q denotes the number of change points in q-th time series $S^{(q)}$; $diag(\mathbf{E})$ denotes the diagonal matrix whose main diagonal elements are the earliness vector $\mathbf{E}$; $||\cdot||_{2,1}$ denotes the $\ell_{2,1}$ norm, and $||\mathbf{W}diag(\mathbf{E})||_{2,1} = \sum_{i=1}^{H+1} ||(\mathbf{W}diag(\mathbf{E}))_{:,i}||_F$; For computation convenience, we add element 1 in the first place of each row vector in $\mathbf{D}^{(q)}$, and add element 0 in the first place of the earliness vector $\mathbf{E}$ as the corresponding data to the intercept $\mathbf{W}_{:,1}$.

The first part of Eqn. (2.3) is the distance between the groundtruth label and the predicted label. The next two parts are earliness-aware sparse group lasso. As every column in the parameter $\mathbf{W}$ represents one candidate shapelet, the $\ell_{2,1}$ norm is applied to select candidate shapelet based on the earliness and the classification task. The l_1 norm aims to make $\mathbf{W}$ sparse within each column vector. In general, the earliness-aware sparse group lasso selects important early candidate shapelet based on the classification task.

Efficient Optimization. It is observed that there are three different norms in Eqn. (2.3), so we adopt the idea of alternating direction method of multipliers (ADMM) [5] to split the loss function into several subproblems that are easier to solve.

First we introduce auxiliary variables $\mathbf{A}$ and $\mathbf{B}$, and rewrite the optimization problem as follows:

(2.4)
$$\min_{\mathbf{W},\mathbf{A},\mathbf{B}} \sum_{q=1}^{Q} ||\mathbf{D}^{(q)}\mathbf{W}' - \mathbf{L}^{(q)}||_F^2 + \lambda_1 ||\mathbf{A}diag(\mathbf{E})||_{2,1} + (1-\lambda_1)||\mathbf{B}||_1,$$
$$\text{s.t. } \mathbf{W} = \mathbf{A} \text{ and } \mathbf{W} = \mathbf{B}.$$

The argumented Lagrangian of the optimization is:

(2.5)
$$\begin{aligned}\mathcal{L} = &\sum_{q=1}^{Q} ||\mathbf{D}^{(q)}\mathbf{W}' - \mathbf{L}^{(q)}||_F^2 + \lambda_1 ||\mathbf{A}diag(\mathbf{E})||_F^2 + (1-\lambda_1)||\mathbf{B}||_1 \\ &+ \rho trace(\mathbf{U}^T(\mathbf{W}-\mathbf{A})) + \frac{\rho}{2}(||\mathbf{W}-\mathbf{A}||_F^2) \\ &+ \rho trace(\mathbf{V}^T(\mathbf{W}-\mathbf{B})) + \frac{\rho}{2}(||\mathbf{W}-\mathbf{B}||_F^2),\end{aligned}$$

where $\mathbf{U}$, $\mathbf{V}$ are the dual variables associated with the constrain and $\rho > 0$ is the penalty parameter.

The above problem can be solved as follows:

(2.6)
$$\begin{aligned}\mathbf{W}_{t+1} &= \arg\min_{\mathbf{W}} \mathcal{L}(\mathbf{W}, \mathbf{A}_t, \mathbf{B}_t, \mathbf{U}_t, \mathbf{V}_t), \\ \mathbf{A}_{t+1} &= \arg\min_{\mathbf{A}} \mathcal{L}(\mathbf{W}_{t+1}, \mathbf{A}, \mathbf{U}_t, \mathbf{B}_t, \mathbf{V}_t), \\ \mathbf{B}_{t+1} &= \arg\min_{\mathbf{A}} \mathcal{L}(\mathbf{W}_{t+1}, \mathbf{A}_{t+1}, \mathbf{U}_t, \mathbf{B}, \mathbf{V}_t), \\ \mathbf{U}_{t+1} &= \mathbf{U}_t + (\mathbf{W}_{t+1} - \mathbf{A}_{t+1}), \\ \mathbf{V}_{t+1} &= \mathbf{V}_t + (\mathbf{W}_{t+1} - \mathbf{B}_{t+1}),\end{aligned}$$

where t denotes the t-th iteration.

Updating W. The relevant terms of $\mathbf{W}$ in $\mathcal{L}$ are: $\sum_{q=1}^{P} ||\mathbf{D}^{(q)}\mathbf{W}' - \mathbf{L}^{(q)}||_F^2 + \rho trace((\mathbf{U}^t)'(\mathbf{W} - \mathbf{A}^t)) + \frac{\rho}{2}(||\mathbf{W} - \mathbf{A}^t||_F^2) + \rho trace((\mathbf{V}^t)'(\mathbf{W} - \mathbf{B}^t)) + \frac{\rho}{2}(||\mathbf{W} - \mathbf{B}^t||_F^2)$. The above terms can be rewritten as: $\sum_{q=1}^{Q} ||\mathbf{D}^{(q)}\mathbf{W}' - \mathbf{L}^{(q)}||_F^2 + \frac{\rho}{2}||\mathbf{W} - \mathbf{A}^t + \mathbf{U}^t||_F^2 + \frac{\rho}{2}||\mathbf{W} - \mathbf{B}^t + \mathbf{V}^t||_F^2$. The closed form solution of $\mathbf{W}$ is:

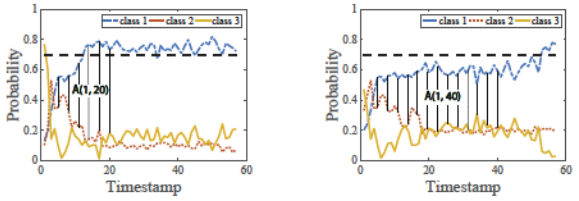
(2.7)
$$\begin{aligned}\mathbf{W}^{t+1} = &\sum_{q=1}^{Q} \left(\mathbf{L}'\mathbf{D}^{(q)} + \frac{1}{2}\rho\mathbf{A}^t - \frac{1}{2}\rho\mathbf{U}^t + \frac{1}{2}\rho\mathbf{B}^t - \frac{1}{2}\rho\mathbf{V}^t\right) \\ &\times \left((\mathbf{D}^{(q)})'\mathbf{D}^{(q)} + 2\rho\mathbf{I}\right)^{-1}.\end{aligned}$$

Updating A. The relevant terms of $\mathbf{A}$ in $\mathcal{L}$ are: $\lambda_2||\mathbf{A}diag(\mathbf{E})||_{2,1} + \rho trace((\mathbf{U}')^t(\mathbf{W}^{t+1} - \mathbf{A})) + \frac{\rho}{2}(||\mathbf{W}^{t+1} - \mathbf{A}||_F^2)$. The above equation can be simplified as: $\lambda_2||\mathbf{A}diag(\mathbf{E})||_{2,1} + \frac{\rho}{2}||\mathbf{W}^{t+1} - \mathbf{A} + \mathbf{U}^t||_F^2$. The closed form solution of $\mathbf{A}$ is: $\mathbf{A}^{t+1}_{:,i} = \left(1 - \frac{2\lambda_2 \mathbf{E}_i}{\rho||\mathbf{C}||_2}\right)_+ \mathbf{C}$, where $\mathbf{E}_i$ is the i-th element in the earliness vector $\mathbf{E}$; $\mathbf{C} = \mathbf{W}_{:,i} + \mathbf{U}_{:,i}$; and $(\cdot)_+$ is defined as: $(x)_+ = \begin{cases} x & \text{if } x > 0, \\ 0 & \text{if } x \le 0. \end{cases}$

Updating B. The relevant terms in $\mathcal{L}$ are: $(1-\lambda_1)||\mathbf{B}||_1 + \rho trace(\mathbf{V}^T(\mathbf{W}-\mathbf{B})) + \frac{\rho}{2}(||\mathbf{W}-\mathbf{B}||_F^2)$. Similar to updating $\mathbf{A}$, the closed form to update $\mathbf{B}$ is:

(2.8)
$$\mathbf{B}_{i,j} = \begin{cases} (\mathbf{W}_{i,j} + \mathbf{V}_{i,j}) - \frac{1-\lambda_1}{\rho} & \text{if } (\mathbf{W}+\mathbf{U})_{i,j} \ge \frac{1-\lambda_1}{\rho}; \\ (\mathbf{W}_{i,j} + \mathbf{V}_{i,j}) + \frac{1-\lambda_1}{\rho} & \text{if } (\mathbf{W}+\mathbf{U})_{i,j} \le -\frac{1-\lambda_1}{\rho}; \\ 0 & \text{otherwise.} \end{cases}$$

2.6 Classification Criterion. The last step of the decision phase is figuring out when to make classification decision, i.e., at the current timestamp, whether the system can output the classification result.



(a) Incorrect Classification Case (b) Classification Delay Case

Figure 2: Two Problems

The common way outputs the classification result when the probability is higher than a threshold. However, we observe two issues that occur during the classification procedure, which might be triggered by the noise in the distance transformation and the classifier training. The first issue is the incorrect classification at the initial timestamps. Due to the lack of information, it is possible that high probability is assigned to the incorrect class. As shown in Figure 2a, if we set the threshold as 0.7 (the black dash-line), the time series is incorrectly classified as class 3. The other issue is the classification

delay. If the highest probability reaches the threshold at a very late stage, then the model would be unable to classify the time series early. An example is shown in Figure 2b. Though the probability that the time series belongs to class 1 is always higher than the other two classes after some initial timestamps, the model is unable to output the classification result until the last few timestamps when the highest probability reaches the threshold.

To tackle these two observed problems, we propose a new classification criterion called confidence area. Confidence area measures the sustained strength of the class probability so that not only the class probability value but also the lasting time of the highest probability are taken into consideration. Let p_i^k denote the probability that the time series belongs to class k at timestamp i. The confidence area of class k at timestamp t, denoted as $A(k,t)$, is defined as:

(2.9)
$$A(k,t) = \begin{cases} \sum_{i=t^k}^{t} \left(p_i^k - \max\left\{p_i^j\right\}_{j\neq k} \right) & \text{If } p_i^k \geq \max\left\{p_i^j\right\}_{j\neq k} \\ 0 & \text{Otherwise,} \end{cases},$$

where t^k is the timestamp that $P_i^k \geq \max\left\{p_i^j\right\}_{j\neq k}$, $\forall t^k \leq i \leq t$, and if $i < t^k$, $p_i^k < \max\left\{p_i^j\right\}_{j\neq k}$. If the probability of class k is the highest probability from timestamp t^k to t, $A(k,t)$ is defined as the area between the largest and the second largest class probability. Otherwise, $A(k,t)$ is zero. The grid area in Figure 2 shows the calculation of the confidence area. In Figure 2a, the black grid area is the confidence area of class 1 at timestamp 20, $A(20,1)$. And in Figure 2b, the black grid area is the confidence area of the class 1 at time stamp 40, $A(40,1)$. If the confidence area of class k is greater than the predefined threshold, the system outputs the classification result as class k at that timestamp.

Confidence area can effectively avoid the problems that the classic classification criteria has. As mentioned above, the two cases in Figure 2 suffer from incorrect classification (Figure 2a) and classification delay (Figure 2b) issues under the traditional classification criterion (if set the threshold as 0.7). These two issues can be solved by using confidence area: for incorrect classification case, though the value of class 3 is very high at the initial stage, the lasting time of class 3 staying in the highest probability is short. Thus, the system doesn't output the result as class 3, because the confidence area of class 3 is very small. For classification delay case, though the value of class 1 is not very high, the lasting time that class 1 owns the highest probability is long, which means the system will output class 1 as the classification result once the confidence area of class 1 reaches the threshold.

3 Experiment

3.1 Experiment Setup. In this section, experiment setup is presented to ensure the fairness of the comparison between DTEC and the baseline methods.

Datasets. Three datasets, CBF, TwoLeadECG and Coffee are selected from UCR time series archive[3] to show the performance of DTEC in detail. Besides, we also validate DTEC on the first 35 datasets from UCR time series archive which are also used in [20].

Performance Measures. In early classification scenario, the error rate and the the total timestamps used to classify(i.e., earliness) are non-ignorable when evaluating the performance. The **error rate** is defined as the portion of time series records that are incorrectly classified. The lower the error rate, the better the performance. The **earliness** is defined as the average percentage of the time used, which is formulated as $Earliness = \frac{1}{Q}\sum_{q=1}^{Q}\frac{t_q}{T_q}$, where Q is the number of time series, t_q and T_q are the decision time and the total length of the q-th time series respectively. The smaller the earliness is, the better the performance is.

Nevertheless, when considering the error rate and the earliness separately, we probably face a dilemma that how to choose between two models, one with low error rate but taking more time and the other one with high error rate and taking less time. To solve this dilemma, we propose a new measurement called the F_α score, which is defined as: $F_\alpha = \frac{1}{\frac{1-\alpha}{\text{Error Rate}}+\frac{\alpha}{\text{Earliness}}}$. The smaller the F_α score is, the better the model performs. α is a user-defined parameter ranging from 0 to 1, which reflects the importance degree of the error rate and the earliness. When $\alpha = 0.5$, the error rate and earliness are equally important. When $\alpha \geq 0.5$, the earliness is more important than the error rate. Otherwise, the error rate gains more importance. By introducing α, we can customize the model comparison based on the demand of the real-world task.

Baseline Methods. The baselines in the experiment are **ECDIRE**, **RelClass**, **ECTS** and **EDSC**. **ECDIRE** [20] is a three steps early classification method. In the first step, the discriminative classes and timestamps are selected. The second step is the prediction reliability control step. And in the last step, a set of classifiers are trained for each timestamp selected in the first step. The code of ECDIRE is downloaded from the authors' website[4]. **RelClass** [21] applies early classification method upon proximity space representation. The differences between the proposed method DTEC and RelClass are that DTEC provides a two-

[3] http://www.cs.ucr.edu/~eamonn/time_series_data/

[4] http://www.sc.ehu.es/ccwbayes/members/umori/ECDIRE/ECDIRE.html

Table 1: Performance Comparison on CBF, TwoLeadECG and Coffee

Method	CBF			TwoLeadECG			Coffee		
	Error Rate	Earliness	$F_\alpha score$	Error Rate	Earliness	$F_\alpha score$	Error Rate	Earliness	$F_\alpha score$
ECDIRE	0.11	0.2855	0.1588	0.19	0.6938	0.2983	**0.04**	0.8214	0.0763
RelClass	0.36	**0.2308**	0.2813	0.28	0.8363	0.4195	0.11	**0.3844**	0.1711
ECTS	0.15	0.7150	0.3306	0.27	0.6443	0.3805	0.25	0.8394	0.3853
EDSC	0.16	0.3185	0.2130	0.12	0.4685	0.1911	0.25	0.5423	0.3422
DTEC	**0.08**	0.4883	**0.1375**	**0.05**	**0.4672**	**0.0903**	**0.04**	0.6299	**0.0752**

steps shapelet selection. Moreover, in DTEC, the classification criteria also takes the bias of the training procedure into consideration. The authors of RelClass provide their code on their website[5]. **ECTS** [28, 29] is an 1-NN based early classification framework and **EDSC** [30] is the first interpretable early classification method.

3.2 Result Analysis. The parameters in the baseline methods are set as suggested in the original paper. The threshold of the confidence area in DTEC is set as 6 and α in F_α is set as 0.5, which indicates the error rate shares equal importance with earliness. The following sections present the result analysis in detail.

Performance Comparison. Table 1 shows the results on three datasets. It is observed that under the F_α measurement, DTEC has the best performance on the three datasets. Detailed analysis is as follows:

Regarding the error rate and earliness, we find the dilemma mentioned above: On CBF dataset, RelClass has the best earliness with high error rate; DTEC has less than 10% error rate with more time to make the decision. The F_α provides a way to measure the error rate and the earliness together. By assigning the equal importance to the error rate and the earliness, DTEC has a smaller F_α score. Therefore, it has better performance. It is reasonable to choose DTEC on CBF dataset, because if a model, like RelClass, outputs the inaccurate classification result at the very early stage, it may mislead the further actions.

RelClass adopts the distances between landmarks (namely shapelet) and the time series over time, but it brings noise into the transferred space. Without appropriate mechanism to correct, the noise will affect the accuracy of the classification result. Besides, it only uses the distance in the last timestamps to train the classifier, which cannot model the propagation of each class over time and also may bring in some bias into the classifier. DTEC assigns labels to every change-points so that the classifier has more training data. Moreover, DTEC designs a confidence area classification criteria by considering the sustained strength of the class probability, to eliminate the noise brought by the training phase. These help DTEC outperform Relclass in terms of error rate and F_α score on the three datasets.

In addition to Relclass, we also observe the high error rate of EDSC and ETCS on Coffee Dataset. When using EDSC and ETCS, it is required to determine whether a time series belongs to one group or contains one specific shapelet. However, when the differences between classes are minor, it would lead to less clarity on the determination stage and generate classification errors, which is clearly reflected by the performance of the three datasets. Among the three datasets, CBF has the most distinguishable patterns between three classes, TwoLeadECG follows and the Coffee has the least obvious patterns. Compared with EDSC and ETSC, DTEC explores the distances between a time series and the candidate shapelets, so that the differences between two classes can be reflected in the distance space. Therefore, DTEC achieves the best performance on the Coffee dataset.

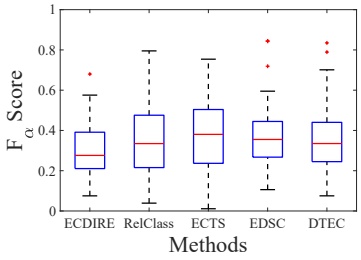
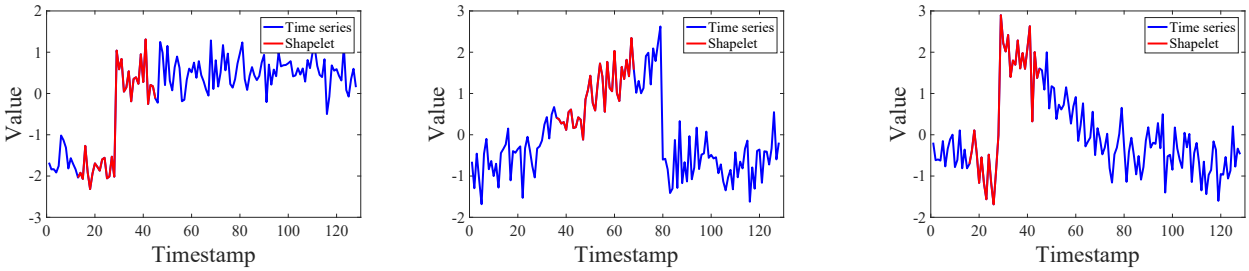


Figure 3: Performances on Extensive Datasets

ECDIRE is a newly proposed early classification method. It trains different classifiers for different timestamps. Nevertheless, when the starting timestamps in the test time series are not aligned with the training set, the discriminative timestamps found in the first step would result in missing of the best timestamp for decision making in the test time series. The performance of ECDIRE on dataset TwoLeadECG and Coffee reflects the case. On TwoLeadECG, ECDIRE makes more mistakes on classification than DTEC, and takes a longer time to classify. Moreover, on Coffee dataset,

[5] http://www.mayagupta.org/publications/Early_Classification_For_Web.zip.



(a) Class 1 (b) Class 2 (c) Class 3

Figure 4: Selected Shapelets on CBF Dataset

ECDIRE achieves the same accuracy with DTEC, but uses much more time to make the classification decision. It means DTEC can capture important feature earlier than ECDIRE while preserving the accuracy.

Result Analysis on Extensive Datasets. To further compare the performance of DTEC and baselines, we also conduct experiments on the first 35 datasets from UCR time series archive. We summarize the results in Figure 3. It is observed that on the extensive datasets, DTEC outperforms RelClass, ETSC, and EDSC, and has comparable results with the best baseline ECDIRE.

3.2.1 Analysis on Shapelet Selection. The goal of this section is to experimentally validate that DTEC can select important early features. To better illustrate the feature selection in DTEC, we choose the dataset CBF as the example, because CBF dataset has clear patterns between the three classes and the selected shapelet can be better visualized.

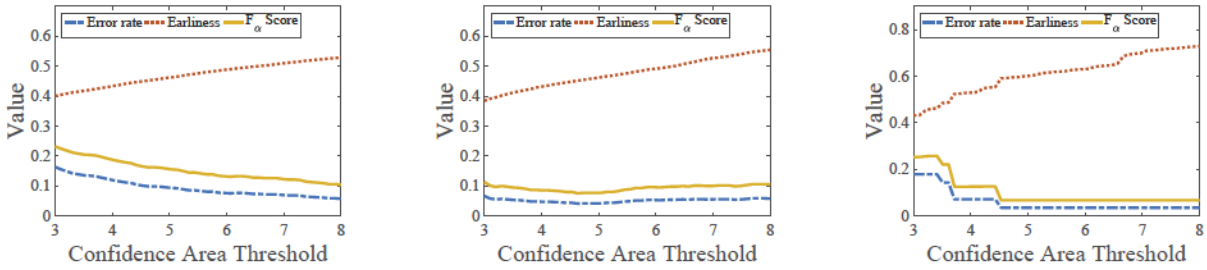
To intuitively show the shapelets selected by DTEC, the most important shapelets of each class are visualized in Figure 4, on which the blue line represents the time series and the red line represents the selected shapelet. From the figure, we can observe that the selected shapelet captures the characteristic of each class. In class 1 (Figure 4a), the most discriminative pattern is located between timestamp 20 and 40, and the selected shapelet capture that change. Similarly, in class 3 (Figure 4c), the shapelet successfully captures the pattern located from timestamp 20 to 70. Different from class 1 and 3, the shapelet selected in class 2 (Figure 4b) captures the gradually increasing trend which is the most important difference between the other classes.

One interesting finding is that for class 2 (Figure 4b), there is a sharp-decreasing pattern, but DTEC doesn't assign a very high weight to this kind of shapelet. There are three reasons for that. First, the gradually increasing shapelet is enough to distinguish class 2; Second, it appears later than the gradually decreasing pattern. The other reason is that the sharp decreasing pattern is also observed in some time series of class 1, and assigning a high weight to the shapelet containing sharp decreasing pattern would confuse class 2 with class 1. This finding directly demonstrates that DTEC can select early distinguishable shapelet.

3.2.2 Analysis on Confidence Area and Parameter Sensitivity. In this section, we explore the effectiveness of the classification criterion confidence area. We mainly focus on whether the confidence area can solve the problems mentioned in Section 2.6 as well as the parameter sensitivity of the confidence area.

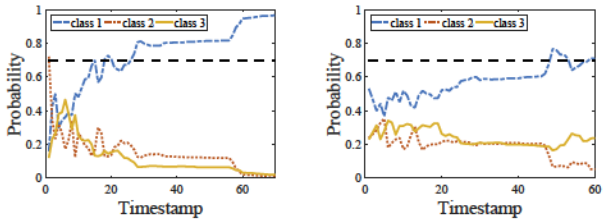
Confidence Area Analysis. Figure 6 shows the incorrect classification problem and classification delay problem under the traditional classification criterion in CBF dataset. In both cases, the true label is class 1. If we set the threshold of the traditional classification criterion as 0.7, in case 1 (Figure 6a), the class 3 probability is higher than 0.7 at timestamp 1, which leads to an incorrect classification result. In case 2 (Figure 6b), the probability of class 1 is always above the others and reaches 0.7 at the very late stage, which causes classification delay. The DTEC outputs the classification result as class 1 correctly at timestamp 28 in case 1. And in case 2, DTEC outputs class 1 at timestamp 41 in case 2, which is earlier than the time when the highest probability reaches 0.7. By considering the sustained strength of the class probability, DTEC can avoid the incorrect classification and the classification delay problem. Therefore, imposing confidence area as the classification criterion is meaningful.

Parameter sensitivity. In terms of the confidence area, intuitively, the higher the confidence area threshold is, the higher the confidence is, and the more time is used to make the classification. Ideally, the change degree of three measurements with respect to the confidence area threshold should be low so that the algorithm is not sensitive to the threshold. To evaluate the sensitivity, we change the confidence area threshold from 3 to 8. The results of the error rate, earliness and F_α score on three datasets are shown in Figure 5. From



(a) CBF Dataset (b) TwoLeadECG Dataset (c) Coffee Dataset

Figure 5: The Effect of Confidence Area Threshold



(a) Case 1 (b) Case 2

Figure 6: Two Problems Observed in CBF Dataset

the figure, it is observed that with the increase of the confidence area threshold, the changing trends of accuracy, earliness, and F_α score are the same as what we claimed above, but the changing range is very small. Therefore DTEC is not very sensitive to the confidence area threshold and best value for the confidence area threshold is between 5 and 7.

4 Related work

The high demand of accurate early classification in real life promotes the development of early time series classification methods [27, 22, 3, 19, 18]. In [28], Xing et. al. first propose the concept of MPL (Minimum Prediction Length) combined with 1-NN classification to classify time series early. Later, the shapelet is adopted to auxiliary classification in EDSC method [30], which makes the classification model interpretable. In [7], Ghalwash. et. al. point out that earliness, interpretability, and uncertainty estimation are three ideal properties for early classification methods, and extend the EDSC with uncertainty estimation available. Because of the interpretability, many shapelet based methods have been proposed [21, 11, 12, 6, 13]. Almost all shapelet based methods need to design the threshold to determine whether a time series contains a specific shapelet, while the distance information between time series and shapelet is ignored. Relclass [21] is the only one that utilizes the distance information and proposes the RelClass method. However, RelClass is not robust to the noise existing in the distance information.

Except for the shapelet based methods, Ghalwash. et. al. propose to train different hybrid HMM/SVM models on fixed length segments with different starting points [8]. However, this kind of method lacks the interpretability. In [20], Mori et. al. proposes ECDIRE to train different classifiers for different discriminative timestamps, and only examine the discriminative timestamps. However, ECDIRE depends a lot on the alignment of timestamps between the training dataset and the test dataset.

The nature of early classification methods, that accurate classification result can be provided as early as possible, has stimulated a lot of applications: early traffic classification on the network [4], early classification on motion recognition [14, 15, 17], early diagnosis [6, 2, 23], early odor detection [10], and so on.

In our approach DTEC, the distance information is adopted to perform a two-phase early time series classification, and interpretable important features are extracted efficiently. Moreover, a new classification criterion confidence area is proposed to eliminate the effect of noise brought by the training phase.

5 Conclusion

In many applications, knowing the classification result as early as possible is critical because future actions can be taken in advance. Therefore, early time series classification is essential for time-sensitive applications. However, existing shapelet based early classification methods either ignore the distance information or are not robust. In this paper, we present a novel two-phase early time series classification framework DTEC. By transferring the time series into the distance space, a probabilistic classifier is trained to output the class label and its probability. Besides, a new classification criterion, named confidence area, is proposed to eliminate the noise and bias brought from the classifier and the dataset. Through experiments, it is demonstrated that DTEC can achieve high accuracy and preserve the earliness under the F_α score measure.

6 Acknowledgement

This work was supported in part by the US National Science Foundation under grants NSF IIS-1747614, NSF IIS-1553411, NSF IIS-1218393 and IIS-1514204. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah. Time-series clustering–a decade review. *Information Systems*, 53:16–38, 2015.

[2] E. Choi, A. Schuetz, W. F. Stewart, and J. Sun. Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, 24(2):361–370, 2017.

[3] A. Dachraoui, A. Bondu, and A. Cornuéjols. Early classification of time series as a non myopic sequential decision making problem. In *Proc. of ECML'16*, 2015.

[4] A. Dainotti, A. Pescapé, and C. Sansone. Early classification of network traffic through multi-classification. *Traffic Monitoring and Analysis*, pages 122–135, 2011.

[5] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, pages 17–40, 1976.

[6] M. F. Ghalwash, V. Radosavljevic, and Z. Obradovic. Extraction of interpretable multivariate patterns for early diagnostics. In *Proc. ICDM'13*, 2013.

[7] M. F. Ghalwash, V. Radosavljevic, and Z. Obradovic. Utilizing temporal patterns for estimating uncertainty in interpretable early decision making. In *Proc. KDD'14*, 2014.

[8] M. F. Ghalwash, D. Ramljak, and Z. Obradović. Early classification of multivariate time series using a hybrid hmm/svm model. In *Proc. BIBM'12*, 2012.

[9] J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme. Learning time-series shapelets. In *In Proc. of KDD'14*, 2014.

[10] N. Hatami and C. Chira. Classifiers with a reject option for early time-series classification. In *Proc. CIEL'13*, 2013.

[11] G. He, Y. Duan, R. Peng, X. Jing, T. Qian, and L. Wang. Early classification on multivariate time series. *Neurocomputing*, 149:777–787, 2015.

[12] G. He, Y. Duan, T. Qian, and X. Chen. Early prediction on imbalanced multivariate time series. In *Proc. CIKM'13*, 2013.

[13] I. Karlsson, P. Papapetrou, and H. Boström. Early random shapelet forest. In *Proc. of the International Conference on Discovery Science*, pages 261–276, 2016.

[14] K. Li, S. Li, and Y. Fu. Early classification of ongoing observation. In *Proc. of ICDM'14,*, 2014.

[15] S. Li, K. Li, and Y. Fu. Early recognition of 3d human actions. *TOMM*, 14(1s):20, 2018.

[16] J. Lines, L. M. Davis, J. Hills, and A. Bagnall. A shapelet transform for time series classification. In *In Proc. of KDD'12*, 2012.

[17] C. Ma, X. Weng, and Z. Shan. Early classification of multivariate time series based on piecewise aggregate approximation. In *International Conference on Health Information Science*, pages 81–88. Springer, 2017.

[18] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano. Early classification of time series from a cost minimization point of view. In *Proc. of NIPS'15 Time Series Workshop*, 2015.

[19] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano. Early classification of time series by simultaneously optimizing the accuracy and earliness. *IEEE Transactions on Neural Networks and Learning Systems*, 2017.

[20] U. Mori, A. Mendiburu, E. Keogh, and J. A. Lozano. Reliable early classification of time series based on discriminating the classes over time. *Data Mining and Knowledge Discovery*, pages 233–263, 2017.

[21] M. Sangnier, J. Gauthier, and A. Rakotomamonjy. Early and reliable event detection using proximity space representation. In *Proc. ICML'16*, 2016.

[22] T. Santos and R. Kern. A literature survey of early time series classification and deep learning. In *Proc. i-KNOW'16*, 2016.

[23] S. Somanchi, S. Adhikari, A. Lin, E. Eneva, and R. Ghani. Early prediction of cardiac arrest (code blue) using electronic medical records. In *Proc. KDD'15*, 2015.

[24] L. Ulanova, N. Begum, and E. Keogh. Scalable clustering of time series with u-shapelets. In *Proc. SDM'15*, 2015.

[25] W. Wang, C. Chen, W. Wang, P. Rai, and L. Carin. Earliness-aware deep convolutional networks for early time series classification. *CoRR*, abs/1611.04578, 2016.

[26] X. Wang, J. Lin, P. Senin, T. Oates, S. Gandhi, A. P. Boedihardjo, C. Chen, and S. Frankenstein. RPM: representative pattern mining for efficient time series classification. In *In Proc. of EDBT'16*, 2016.

[27] Z. Xing, J. Pei, and E. Keogh. A brief survey on sequence classification. *SIGKDD Explorations*, 12(1):40–48, 2010.

[28] Z. Xing, J. Pei, and S. Y. Philip. Early prediction on time series: A nearest neighbor approach. In *Proc. IJCAI'09*, 2009.

[29] Z. Xing, J. Pei, and P. S. Yu. Early classification on time series. *Knowl. Inf. Syst.*, pages 105–127, 2012.

[30] Z. Xing, J. Pei, P. S. Yu, and K. Wang. Extracting interpretable features for early classification on time series. In *Proc. SDM'11*, 2011.

[31] L. Ye and E. Keogh. Time series shapelets: a new primitive for data mining. In *Proc. KDD'09*, 2009.

[32] J. Zakaria, A. Mueen, and E. Keogh. Clustering time series using unsupervised-shapelets. In *Proc. ICDM'12*, 2012.