International Journal of Computer Applications in Technology

ISSN online: 1741-5047 - ISSN print: 0952-8091
https://www.inderscience.com/ijcat

A novel shape-based time series classification with SAX-Ensemble

Mariem Taktak, Slim Triki

DOI: 10.1504/IJCAT.2023.10056447

Article History:

Received:	18 December 2021
Last revised:	04 May 2022
Accepted:	10 June 2022
Published online:	23 May 2023

A novel shape-based time series classification with SAX-Ensemble

Mariem Taktak*

Higher Institute of Applied Sciences and Technologies of Sousse,
Sousse, Tunisia
Email: mariem.taktak@gmail.com
*Corresponding author

Slim Triki

National Engineering School of Sfax,
Sfax, Tunisia
Email: slim.triki@enis.tn

Abstract: Since the first publication of the Symbolic-Aggregate Approximation (SAX), a lot of extensions with novel SAX-distance measure are published. Each of them attempts to integrate additional statistical features in order to improve original SAX average-based feature. Each SAX-feature has its own distance function which quantifies the (dis)similarity between two Time Series (TS). However, none of them can fit the overall shape-characteristics of a TS and give the superiority to an individual SAX-based classifier. In order to combine the prediction of each single SAX-based classifier, we propose a collection of several SAX-features to compose a shape-based ensemble for TS classification. The proposed SAX-Ensemble scheme is applied on a multiple domain representation of the TS where the diversity of collected SAX-features make the setting of the SAX-discretisation parameters a challenging task especially for a long TS data or a large training data set. In order to avoid a time-consuming of either grid search or expensive optimisation algorithm, we instead apply a data-aware or data-agnostic parameters setting technique. Experimental results on real TS database show that the performance of the proposed SAX-Ensemble with data-aware technique exceeded the SAX-based classifiers with more flexible and realistic parameters estimation.

Keywords: time series data; symbolic aggregate approximation; shape-based classification.

Reference to this paper should be made as follows: Taktak, M. and Triki, S. (2023) 'A novel shape-based time series classification with SAX-Ensemble', *Int. J. Computer Applications in Technology*, Vol. 71, No. 1, pp.64–77.

Biographical notes: Mariem Taktak holds her Engineering degree and PhD degree both in Computer Science from National Engineering School of Sfax, Tunisia in 2011 and 2019, respectively. Now, she is a Temporary Assistant Professor at the Higher Institute of Sciences and Technologies of Sousse, Tunisia. Her main research focuses on the time series data mining.

Slim Triki is an Associate Professor in the National Engineering School of Sfax, Tunisia. He has obtained the Electromechanical Engineering degree from the National Engineering School of Sfax, Tunisia in 1999 and Master degree and PhD degree in Industrial and Human Control from the University of Valenciennes et du Hainaut Cambrésis, France in 2001 and 2005, respectively. His research interests include supervision of manufacturing systems, fault detection and isolation of hybrid dynamical systems and temporal data mining.

1 Introduction

The major importance of the symbolic representation by SAX is its ability to be exploited within a classification process (Lin et al., 2007). This is achieved by using the MIDIST() function, which is a proper distance measure that guarantees no false dismissal. SAX converts a TS data to a symbolic representation which is achieved in three steps as illustrated in Figures 1(a), 1(b) and 1(c). In the first step, the *z*-normalised time series is divided into ω equal-length subsequences. Then, the average value of each subsequence is computed which results in a PAA representation. Finally, each entry of the PAA vector is mapped to a symbol according to a discretisation of a Gaussian distribution. Figure 2 shows an illustration of the discretisation process for an alphabet size $\alpha = 4$ (i.e., by defining 3 breakpoints $\{\beta_1,\beta_2\ \beta_3\}$ to divide the Gaussian curve into 4 equal-sized areas). SAX can also be combined with a sliding window to pre-process a very long TS data. This technique is illustrated in Figure 3.

Figure 1 From the TS data to a several variations of SAX representations

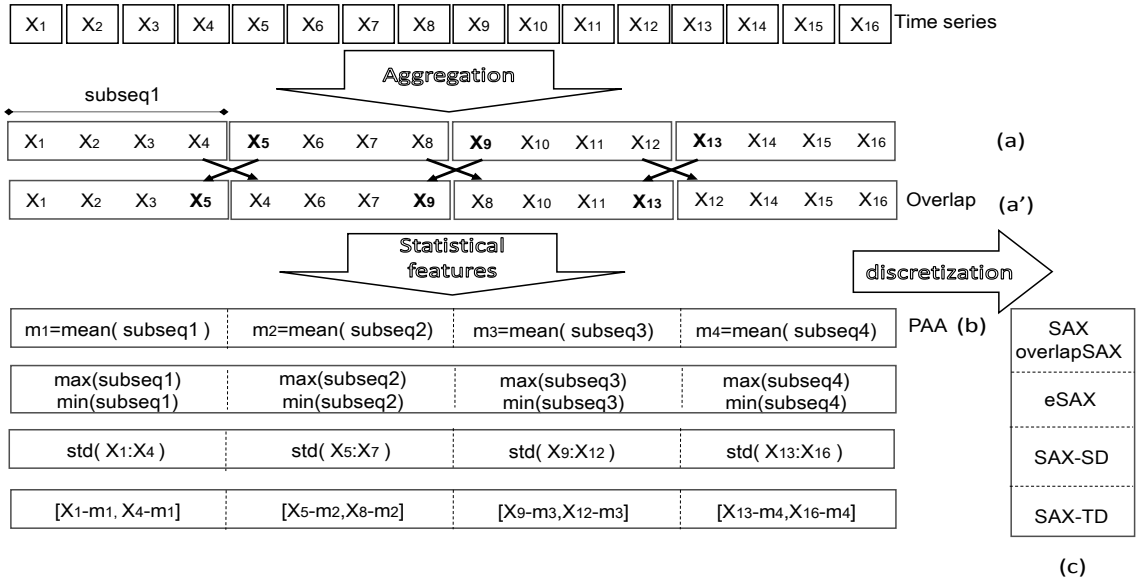


Figure 2 Example of PAA's value discretisation (the number of breakpoints $\{\beta_1,\beta_2,\beta_3\}$ is related to the SAX alphabet size $\{'a','b','c','d'\}$)

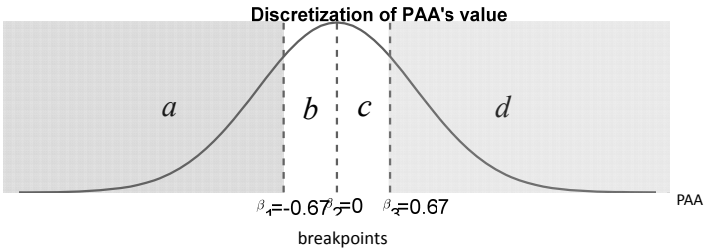
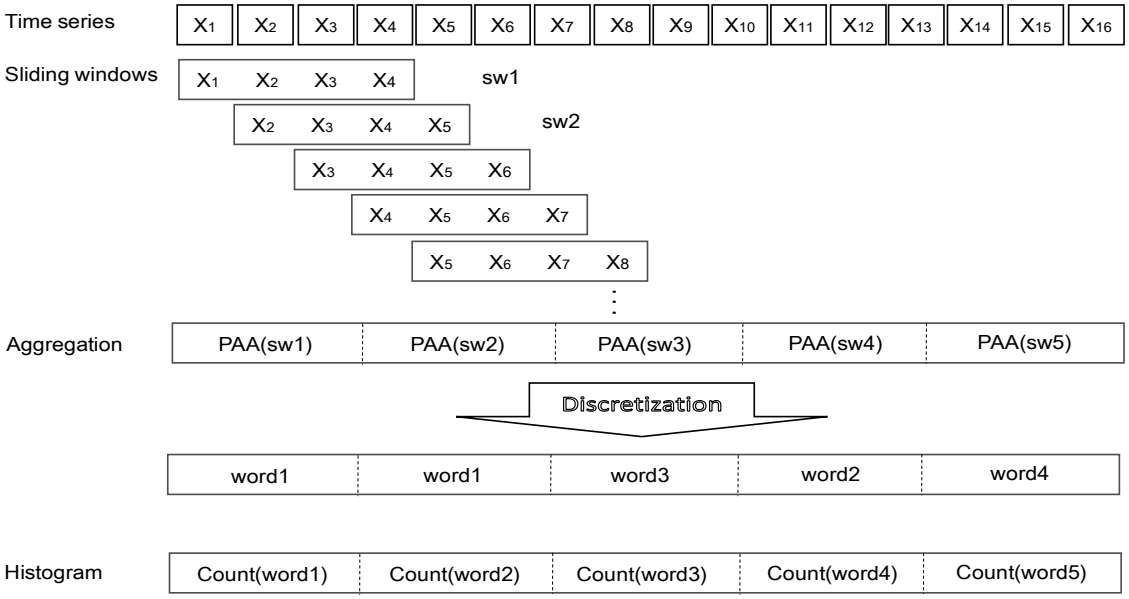


Figure 3 Illustration of the SAX with sliding window technique



Since its first apparition in 2007, SAX has attracted the attention of several researchers in the data mining area, who reported some drawbacks and attempted to improve it and proposed a novel SAX-based classifier algorithm. Owing to the temporal correlation of the data, the high dimension and the different lengths of the TS in the database, the need to extract key signal features is of direct influence in the TS data classification efficiency. However, the original version of the SAX extracts only the simple statistical mean value over a subsequence of data. To circumvent this drawback, several researchers reported the need of additional statistical features to extract more important information such as:

- Maximum and minimum values as proposed in *ESAX* (Lkhagva et al., 2006),
- Trend information by applying linear regression as proposed in *1d-SAX* (Simon et al., 2013),
- Standard deviation as proposed in *SAX-SD* (Chaw and Hayato, 2016),
- The range between the first value (res. last value) and the mean value of each subsequence as proposed in *SAX-TD* (Sun et al., 2014),
- Trend information by computing the sample difference (delta value) as proposed in *DDlcss* (Gorecki, 2014),
- Trend information by swapping the end points of each subsequence with the end points of the neighbouring subsequence as proposed in *Overlap-SAX* (Fuad, 2020).
- Trend information by adjusting the range between local maximum value (res. local minimum value) and the mean value of each subsequence as proposed in *SAX-BD* (He et al., 2020).

Each SAX-feature (see Figure 1) has its own SAX-based distance measure used to quantify the (dis)similarity between TS data. It is clear that there is no one SAX-feature that can fit all shape-characteristics of a TS and give the superiority to one of these SAX-based classifiers. Hence, we propose a combination of several SAX-features through simple ensemble schemes for an accurate and tolerable time-cost classification via (dis)similarity measure. Ensembles become popular in recent TS mining research (Asmita et al., 2021) and are very competitive on classification problems. For example, Elastic Ensemble (Lines and Bagnall, 2015) combines 11 one-nearest neighbour (1NN) classifiers based on distance measure with Euclidean and several elastic distances like DTW, LCSS and ERP. Other ensemble methods like COTE (Bagnall et al., 2015) or Hive-COTE (Lines et al., 2016) incorporate more than 30 different classifiers for TS classification. Unfortunately, the accuracy improvement comes at a price of expensive computation resources. Typically, SAX can significantly reduce the computational cost due to its dimensionality reduction property through discretisation of the TS data. However, the main challenge of any SAX-based classifier is the task of finding the best discretisation parameters ω and α. A grid search technique is usually adopted to choose the best pair (ω,α) providing the minimum classification error rate by cross validation on the training data. A few works adopt optimisation algorithm due to its complexity in space and time. Both techniques provide good results under assumption of an ideal training data. In practice, we cannot always expect the training data to be so perfectly labelled. In addition, the computational cost of a typical grid search through Leave-One-Out Cross Validation (LOOCV) algorithm increases with the size of the training data. Furthermore, both techniques are data-agnostic in the sense that parameters are found independently to the temporal and spectral properties of the TS data. As a solution to this issue, a data-aware method was proposed in Chaw and Hayato (2017). In this data-aware technique, segment size ω is estimated with adapted Shannon sampling theorem, while alphabet size α is estimated based on the most frequent average value among segments. In addition, we study here an alternative approach to the grid search with LOOCV which uses a supervised SAX-feature selection algorithm. Finally, multiple domain representations of the TS data are proposed to improve the classification of the SAX ensemble scheme.

In summary, the main contributions of this paper are as follows:

- First, we propose a novel SAX-Ensemble classification method which integrates multiple SAX-based distance measure applied on a multiple domain representation of TS data.
- Second, the proposed ensemble scheme includes two additional trend-based SAX distance so as to enhance diversity through enrichment of the trend shape feature. Furthermore, each distance measure is weighted by a TS complexity factor since in many domains the different classes may have different complexities.
- Third, SAX-discretisation parameters are defined either by a data-aware technique or a data-agnostic through supervised SAX-feature selection technique.
- Finally, we perform extensive experiments on UCR time series data sets and compare our proposed SAX-Ensemble with the candidate SAX-based algorithms to verify the performance of our algorithm in the classification of the TS based on their shape.

The rest of this paper is organised as follows. Section 2 provides the background and definition of the candidate SAX-based distance function. In Section 3, we briefly review the two categories of the SAX-based methods for the TS data classification, namely shape-based and structure-based. In Section 4, we introduce our SAX-Ensemble scheme with different proposed configurations. In Section 5, we present our experimental evaluation and discuss our results. Finally, we present our conclusion and future work.

2 Definition and background

We give now the necessary definition of both, taxonomy and SAX-based distance used in this paper.

Definition 1: *A time series* T_n *is a sequence of real-valued numbers* X_i^n *:* $T_n = \left[X_1^n, X_2^n, \ldots, X_N^n\right]$ *where N is the length of* T_n.

Definition 2: *A z-normalised time series T is series with zero mean and one standard deviation:* $T_{z-norm}=\left[\frac{X_1-\mu}{\sigma},\ldots,\frac{X_n-\mu}{\sigma}\right]$ *where* $\mu=\frac{1}{N}\sum_{i=1}^{N}X_i$ *and* $\sigma=\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(X_i-\mu\right)^2}$.

Definition 3: *A subsequence T_{ij} from time series T is a continuous subset of the values from X_i to X_j:* $T_{ij}=\left[X_i:X_j\right]$.

Definition 4: *A Piecewise Aggregate Approximation (PAA) of a time series T of length N is a reduced time series $\overline{T}$ of length ω $(\omega\ll N)$ obtained by dividing T into ω-dimensional equal-sized subsequence which mapped by their average value:* $\overline{T}=\left[\overline{X}_1,\overline{X}_2,\ldots,\overline{X}_\omega\right]$ *where* $\overline{X}_i=mean\left[X_{\frac{N}{\omega}(i-1)+1}:X_{\frac{N}{\omega}i}\right]$.

Definition 5: *The MINDIST() distance measure between two time series T_1 and T_2 with equal length N and of the same SAX words number ω is defined as:* $MINDIST\left(\hat{T}_1,\hat{T}_2\right)=\sqrt{\frac{N}{\omega}}\sqrt{\sum_{i=1}^{\omega}\left(dist\left(\hat{x}_{1i},\hat{x}_{2i}\right)\right)^2}$ *where the SAX symbolic form of a time series T is represented as:* $\hat{T}=\left[\hat{x}_1,\ldots,\hat{x}_\omega\right]$ *and dist() function is a precomputed distance table (Lin et al., 2012).*

Definition 6: *The SAX_TD() distance measure between two time series T_1 and T_2 is defined as:* $SAX\_TD\left(T_1,T_2\right)=MINDIST\left(\hat{T}_1,\hat{T}_2\right)+TD\left(T_1,T_2\right)$ *where the TD() is a trend distance (Vineeth et al., 2014).*

Definition 7: *The trend distance TD() between the i-th segment of two time series T_1 and T_2 is calculated as:*

$TD\left(\overline{X}_i^1,\overline{X}_i^2\right)=\sqrt{\left(\Delta\overline{X}_i^1\left(t_s\right)-\Delta\overline{X}_i^1\left(t_e\right)\right)^2+\left(\Delta\overline{X}_i^2\left(t_s\right)-\Delta\overline{X}_i^2\left(t_e\right)\right)^2}$

where t_s and t_e are the start and end point of the i-th time segment, and $\Delta\overline{X}_i^j\left(t_s\right)=X^j_{\frac{N}{\omega}(i-1)+1}-\overline{X}_i^j$, $\Delta\overline{X}_i^j\left(t_e\right)=X^j_{\frac{N}{\omega}i}-\overline{X}_i^j$.

Definition 8: *The SAX_SD() distance measure between two time series T_1 and T_2 is defined as:* $SAX\_SD\left(T_1,T_2\right)=MINDIST\left(\hat{T}_1,\hat{T}_2\right)+SD\left(T_1,T_2\right)$ *where the SD( ) function is the standard deviation distance (Chaw and Hayato, 2016).*

Definition 9: *The standard deviation of the i-th segment in a time series T is defined as:* $sd\left(\overline{X}_i\right)=std\left[T_{\frac{N}{\omega}(i-1)+1:\frac{N}{\omega}i}\right]$ *and the standard deviation distance between the i-th segment of two time series T_1 and T_2 is defined as:* $SD\left(\overline{X}_i^1,\overline{X}_i^2\right)=\sqrt{\left(sd\left(\overline{X}_i^1\right)-sd\left(\overline{X}_i^2\right)\right)^2}$.

Definition 10: *For a subsequence $\left[X_i:X_{i+r-1}\right]$ of length r, the slope s and intercept b of the line obtained from linear regression are:* $s=\frac{\overline{x'y}-\overline{x'}\,\overline{y}}{\overline{x'^2}-\overline{x'}^2}$, $b=\overline{y}-s\left(\overline{x'}-i\right)$ *where* $\overline{x'}=\frac{\sum_{k=i}^{i+r-1}k}{r}=\frac{2i+r-1}{2}$, $\overline{y}=\frac{\sum_{k=i}^{i+r-1}X_k}{r}$, $\overline{x'y}=\frac{\sum_{k=i}^{i+r-1}kX_k}{r}$ *and* $\overline{x'^2}=\frac{\sum_{k=i}^{i+r-1}k^2}{r}=\frac{(i+r-1)(i+r)\left[2(i+r)-1\right]}{6r}-\frac{(i-1)i(2i-1)}{6r}$.

Definition 11: *The trend distance TD2() between the i-th segment of two time series T_1 and T_2 is calculated as:* $TD2\left(\overline{X}_i^1,\overline{X}_i^2\right)=\sqrt{\left(s_i^1\left(t_s\right)-s_i^2\left(t_e\right)\right)^2+\left(b_i^1\left(t_s\right)-b_i^2\left(t_e\right)\right)^2}$ *where s_i^j and b_i^j are, respectively, the slope and intercept of the i-th segment in the time series T_j*

Definition 12: *For a subsequence $\left[X_i:X_{i+r-1}\right]$ of length r, the trend p and the offset o of the line obtained from linear interpolation is:* $p=\frac{X_{i+r-1}-X_i}{r}$ *and* $o=\overline{X}_i-p\left(X_i-i\right)$ *where* $\overline{X}_i=mean\left(\left[X_i:X_{i+r-1}\right]\right)$.

Definition 13: *The trend distance TD3() between the i-th segment of two time series T_1 and T_2 is calculated as:* $TD2\left(\overline{X}_i^1,\overline{X}_i^2\right)=\sqrt{\left(p_i^1\left(t_s\right)-p_i^2\left(t_e\right)\right)^2+\left(o_i^1\left(t_s\right)-o_i^2\left(t_e\right)\right)^2}$ *where p_i^j and o_i^j are, respectively, the trend and offset of the i-th segment in the time series T_j.*

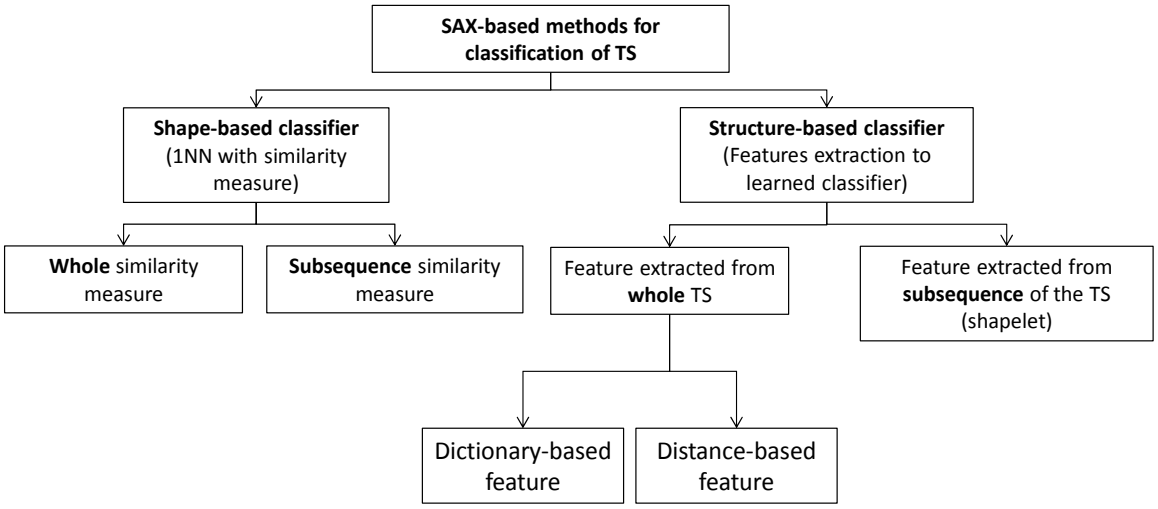
Definition 14: *The SAX_BD() distance measure between two time series T_1 and T_2 is given by the following equation:* $SAX\_BD\left(T_1,T_2\right)=MINDIST\left(\hat{T}_1,\hat{T}_2\right)+BD\left(T_1,T_2\right)$ *where BD() is the boundary distance function.*

Definition 15: *The boundary distance between the i-th segments of two time series T_1 and T_2 is defined as:* $BD\left(\overline{X}_i^1,\overline{X}_i^2\right)=\sqrt{\left(\Delta X_i^1-\Delta X_i^2\right)^2+\left(\Delta x_i^1-\Delta x_i^2\right)^2}$ *where* $\Delta X_i^j=\max\left[T^j_{\frac{N}{\omega}(i-1)+1:\frac{N}{\omega}i}\right]-\overline{X}_i^j$ *and* $\Delta x_i^j=\min\left[T^j_{\frac{N}{\omega}(i-1)+1:\frac{N}{\omega}i}\right]-\overline{X}_i^j$.

3 SAX-based methods for TS classification

SAX-based methods for TS classification can be divided in two categories: the shape-based and the structure-based as depicted in the flowchart of the Figure 4.

Figure 4 Flowchart of the SAX-based algorithm for the TS data classification



3.1 Structure-based classifier

Based on higher-level structural information, SAX offers an appropriate alternative to determine similarity between long TS data. Lin et al. (2012) used the Bag-of-Patterns (BoP) to represent TS data instances, and then use the 1NN classifier. Bag-of-Patterns are obtained by transforming TS data into SAX symbols and then using a sliding window to scan the symbols and adopt a histogram-based representation of unique words. The 1NN-BoP addresses the scalability in classification of TS data while maintaining the lazy nearest neighbour classifier. To provide an interpretable TS data classification, Senin and Malinchik (2013) proposed SAX-VSM as an alternative to 1NN classifier. Like BoP, SAX-VSM also takes the advantage of the SAX and uses the sliding window technique to convert TS data into a set of Bag-of-Words (BoW). During the training phase, SAX-VSM builds Term Frequency-Inverse Document Frequency (TF-IDF) BoW for all classes and then, cosine similarity is used to do the classification. Alternatively, Kate (2016) showed that the TS classification performance improves if instead of using similarity distances (especially DTW distance) with 1NN technique features with a more powerful technique such as Support Vector Machine (SVM) are used. Moreover, this distance-based feature can be easily combined with the SAX method under the BoP representation. Another structure-based TS data classification using SAX is Representative Pattern Mining (RPM) (Wang et al., 2016). RPM focuses on finding the most representative subsequence (i.e., shapelets) for the classification task. After conversion of TS into sequences of SAX symbols, RPM takes advantage of grammatical inference techniques to automatically find recurrent and correlated patterns of variables lengths. This pool of patterns is further refined so that the most representative patterns that capture the properties of a specific class are selected. Other grammatical-based classifier was proposed in Daoyuan et al. (2016) and named Domain Series Corpus (DSCo). As its name suggest, DSCo applies the SAX with sliding window technique on training TS data to build a corpus and subsequently each class is summarised with an *n*-gram Language Model (LM). The classification is performed by checking which LM is the best fit for the tested TS. Recently, Thach et al. (2019) proposed an ensemble of multiple sequence learner algorithm called SAX-SEQL. In summary, SAX-SEQL uses a multiple symbolic representation which combines SAX with Symbolic Fourier Approximation to find a set of discriminative sub-sequences by employing a brunch-and-bound feature search strategy.

3.2 Shape-based classifier

It is well known that a key feature that distinguishes TS from other kinds of data is its shape. That is, distance measure in combination with 1NN classifier is shown to be the simple and excellent method for shape-based classifier. Most common distance measures for matching TS are Euclidean, DTW and LCSS. While Euclidean allows for only a one-to-one point comparison, DTW and LCSS allow for more elasticity and instead compare one-to-many points. In shape-based category, SAX; ESAX; Overlap-SAX; SAX-TD, SAX-SD and SAX-BD are usually used within 1NN classifier where the (dis)similarity measure is defined as; (i) MINDIST() function for SAX, ESAX and Overlap-SAX, (ii) SAX_TD() function for SAX-TD, (iii) SAX_SD() function for SAX-SD and (iv) SAX_BD() function for SAX-BD. Another recent subsequence similarity measure based on DTW named shapeDTW was proposed in Zhao and Itti (2018). Aiming for a many-to-many point comparison, shapeDTW attempts to pair locally similar subsequence and to avoid matching points with distinct neighbourhood subsequence. Hence, shapeDTW represents local subsequence around each temporal point by a shape descriptor and then, uses DTW to align two sequences of descriptors. The SAX representation can be used as shape descriptors to map a local subsequence within the 1NN shapeDWT classifier. We will focus in this work on the simple 1NN classifier with SAX-based distance measure. In particular, our proposed SAX-Ensemble is a collection of a 1NN SAX-based classifiers among them: 1NN SAX, 1NN ESAX, 1NN SAX-TD, 1NN SAX-SD and 1NN SAX-BD.

4 Proposed SAX-ensemble for shape-based TS classification

4.1 Representation ensemble

Figure 5 shows an overview of our proposed SAX-Ensemble shape-based TS data classification. From each original TS in the database, two additional representations are extracted; (i) a periodogram in the frequency domain and (ii) a derivative in the temporal domain. The use of a periodogram representation helps SAX-based distance to detect similar but time-shifted TS data. In some extensions of SAX-based distance like SAX-TD and SAX-SD the trend information is estimated in quantitative value per segment. Instead, in the second additional derivative representation, we estimate the derivative value per sample using simple first-order difference. This helps to further gain insight about local variation of the TS data. In the proposed SAX-Ensemble scheme, each TS representation is used to learn a set of seven 1NN base classifiers whose individual decisions are combined using five combining rules:

1 Majority vote with equal weight to all base classifiers (VOTE): the most voted class label is the ensemble's output.

2 Majority vote with accuracy as weight to each base classifier (Weighted VOTE): accuracy is estimated by cross-validation on the training set and a weight of 0 is assigned to all base classifiers that are less accurate than the most accurate base classifier by a factor of δ. The most voted class label from the selected base classifiers is the ensemble's output.

3 Minimum distance (MinDIST): choose the class label of the base classifier reporting the minimum distance among the nearest distances from all the base classifier.

4 Sum (SUM): sum the output score value given by the base classifier for each class label, the label which accumulated higher score is the ensemble's output.

5 Product (PROD): multiplicate the output score value given by the base classifier for each class label, the label which accumulated higher score is the ensemble's output

6 Maximum (MAX): choose the label of the base classifier reporting the maximum score among all the base classifiers.

In order to use the three last combination rules, each 1NN base classifier should output a label with a score. As 1NN classifier does not, naturally, provide such a score a Non-Conformity (NC) measure is calculated for each class label y according to the following equation (Vineeth et al., 2014):

$$NC(D,T)=\frac{\sum_{j=1}^{k} d_{T_j}^{y}}{\sum_{j=1}^{k} d_{T_j}^{\neg y}} \tag{1}$$

where D is a data set including all training TS data, T is the TS data with unknown label, $\sum_{j=1}^{k} d_{T_j}^{y}$ is the sum of distances of the k-nearest TS from D to the TS data T with the same label and $\sum_{j=1}^{k} d_{T_j}^{\neg y}$ is the sum of distances of the k nearest TS from D to the TS data T with different classes label. Each 1NN base classifier, in our SAX-Ensemble, include a SAX-based distance measure. In addition to the above-mentioned 1NN SAX-distance-based classifier (1NN SAX, 1NN ESAX, 1NN SAX-TD, 1NN SAX-SD and 1NN SAX-BD), we propose two additional version for the 1NN SAX-TD base classifier. To distinguish them, we will use the following notation: 1NN SAX-TD1 (original), 1NN SAX-TD2 (version 2) and 1NN SAX-TD3 (version 3). While 1NN SAX-TD1 uses the trend distance as given in SAX-TD distance (see Definition 6), trend distance in the two others is slightly different. In version 2, the *slope* and *intercept* are obtained from linear regression and used subsequently to estimate trend distance (see Definition 10 and 11). However, we use linear interpolation to estimate trend distance in the base classifier of version 3 (see Definition 12 and 13). Furthermore, a correction complexity coefficient is used in conjunction with each SAX-based distance measure in an analogue way as proposed in the Complexity Invariant Distance (CID) (Batista et al., 2014). Specifically, the SAX-LZ indicator as given in Yin et al. (2019) is used to quantify complexity of a TS data and the correction coefficient (CF) between two TS data T_1 and T_2 is defined by the following equation:

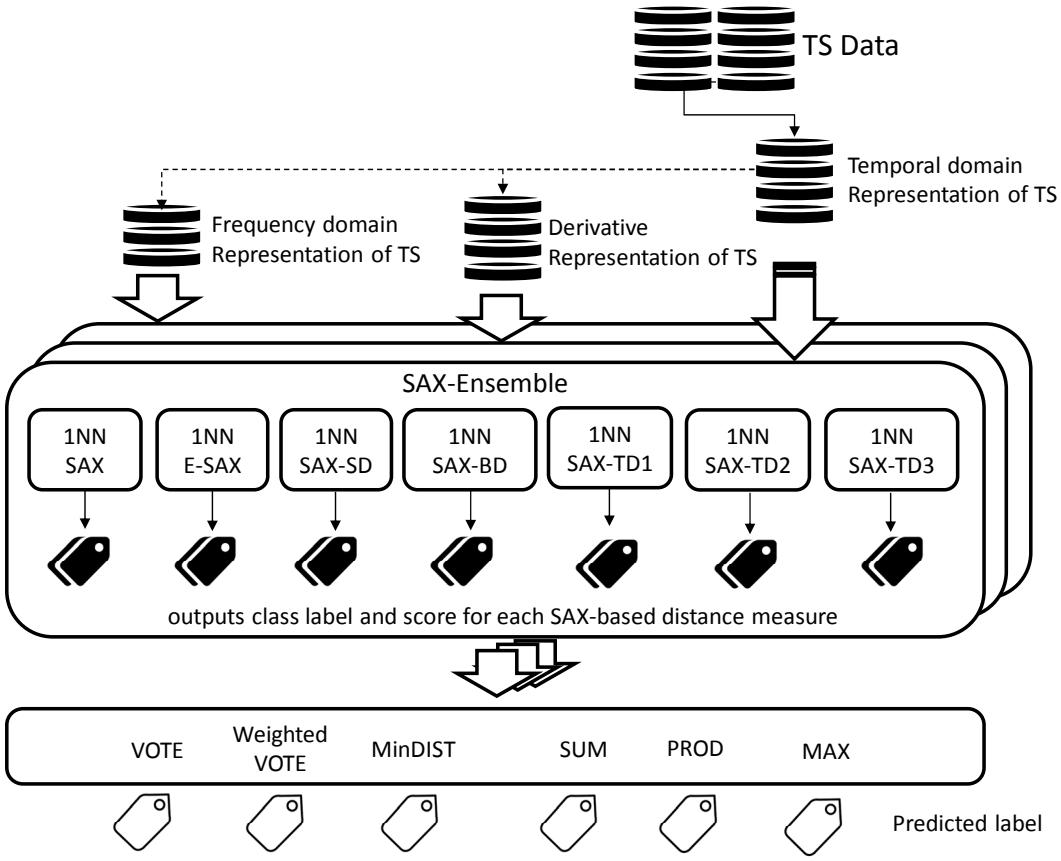
$$\mathrm{CF}(T_1,T_2)=\frac{\max\left(\text{SAX-LZ}(T_1),\text{SAX-LZ}(T_2)\right)}{\min\left(\text{SAX-LZ}(T_1),\text{SAX-LZ}(T_2)\right)} \tag{2}$$

In our proposed SAX-Ensemble, only the original SAX symbolic representation of the TS data is used to compute CF for each SAX-based distance within base classifier.

4.2 SAX-Ensemble parameters estimation

As already reported, the setting of the SAX-discretisation parameters is at the core of any SAX-based classification method. In order to avoid classical parameters setting that should be applied for each base classifier, we propose two techniques for our SAX-Ensemble classification scheme. The first proposed is the data-aware parameters selection technique described in Chaw and Hayato (2017). To the best of our knowledge, this technique has never been used in the literature with ensemble of SAX-based classifier. The second is the data-agnostic technique which selects the best SAX-discretisation parameters in a single pass on the training data set.

Figure 5 Schematic diagram of the proposed SAX-Ensemble shape-based TS classification



4.2.1 Data-aware parameters estimation

The main idea behind the data-aware technique is to find SAX-discretisation parameters according to the temporal and spectral properties of the data itself. Chaw and Hayato (2017) prove that the number of PAA segments (i.e., segment size) can be defined based on Shannon's sampling theorem:

$$f_{sampling} = \frac{\text{number of PAA segment}}{\text{duration of the TS}} = 2f_0 \tag{3}$$

where f_0 is the fundamental frequency of the TS calculated using the Average Magnitude Difference (AMD) function. Subsequently, the alphabet size is estimated based on the skewness of the TS distribution between the mean and the most frequent mean value (i.e., the mode) among the segments. More formally, the alphabet size is interpolated within the integer value from 2 up to 20 by the following formula:

$$\alpha = floor\left\{20 - \left|mode - \min\left(\left[\beta_1, \beta_{19}\right]\right)\right| \cdot \left(\frac{\beta_{19} - \beta_1}{20 - 2}\right)\right\} \tag{4}$$

where $\{\beta_1,\ldots,\beta_{19}\}$ are the set of breakpoints which divide the Gaussian curve within the minimum and maximum value of the alphabet size (i.e., within 2 and 20), the function *floor*(X) rounds each element of X to the nearest integer less than or equal to that element.

4.2.2 Data-agnostic parameters estimation

Our proposed data-agnostic parameters estimation is close, in principle, to the classical grid-search technique which selects the parameters combination based on the average 1NN-classification accuracy determined by cross validation on training set. The main difference between the data-agnostic and grid-search method is the algorithm used inside the selection process. Precisely, the parameters estimation within data-agnostic method is based on a supervised SAX-feature selection algorithm. Hence, we initially need to represent a TS as a vector of features. For the sake of simplicity, we will exploit the PAA representation to construct the feature vector. Formally, given a time series $T = \left[X_1,\ldots,X_N\right]$, from a training data set D of size $n \times N$, the feature vector constructed using PAA will be simply:

$$\text{Feature-}T = \left[\text{Feature}\left(T_{PAA_1}\right),\ldots,\text{Feature}\left(T_{PAA_\omega}\right)\right] \tag{5}$$

where $T_{PAA_i} = \left[X_{\frac{N}{\omega}(i-1)+1} : X_{\frac{N}{\omega}i}\right]$ and each feature vector's entry is defined as:

$$\text{Feature}\left(T_{PAA_i}\right)=\begin{bmatrix}\text{mean}\left(T_{PAA_i}\right)\\ \min\left(T_{PAA_i}\right)\\ \max\left(T_{PAA_i}\right)\\ std\left(T_{PAA_i}\right)\\ T_{PAA_i}\left(end\right)-\text{mean}\left(T_{PAA_i}\right)\\ T_{PAA_i}\left(1\right)-\text{mean}\left(T_{PAA_i}\right)\\ slope\left(T_{PAA_i}\right)\\ intercept\left(T_{PAA_i}\right)\\ \text{SAX-LZ}\left(\hat{T}\right)\\ \text{ESAX-LZ}\left(\hat{T}\right)\end{bmatrix}^T \tag{6}$$

Then, a Fisher score (Duda et al., 2012) is computed for all concatenated feature vector of the TS T in D, denoted as $F\text{-}D=\left[\text{Feature-}T_1,\ldots,\text{Feature-}T_n\right]^T$, to decide how well the feature vector entry in F-*D* separates a class from another. Let $y\in\{1,2,\ldots,c\}^n$ be a vector of class labels, the Fisher score of a column-wise feature vector f in $F\text{-}D$ is defined as:

$$FS\left(f,y\right)=\frac{\sum_{k=1}^{c}n_k\left(\mu_k^f-\mu^f\right)}{\sum_{k=1}^{c}n_k\left(\sigma_k^f\right)^2} \tag{7}$$

where

- n_k is the number of TS in D having the class label *k*,
- μ^f is the overall mean of the elements in f,
- μ_k^f aand σ_k^f are the mean and standard deviation of the elements in f labelled with the *k*-th class.

Finally, the combination of SAX-discretisation parameters which leads to a higher cumulative score, $CumSC=\sum_{f\in D}FS\left(f,y\right)$, is selected. It is important to note that we use the Fisher score for its fast computation.

5 Experimental evaluations

We conducted a pre-evaluation of the SAX-Ensemble classification algorithm with random selected data from the UCR-archive database (Dau et al., 2019). The average accuracy results in Table 1 showed that only majority vote tend to provide better results and more complex combination rule fail to reach even the 50% classification accuracy. Hence, we will report only the results from majority vote (VOTE and weighted VOTE) in the next evaluation on 100 databases from the UCR-archive benchmark. In machine learning, the generally accepted statistic for comparing several classifiers over data sets is the non-parametric Nemenyi post-hoc test (Demšar, 2006). For *k* classifier and *n* data sets, if χ_F^2 denotes the Friedman statistic and R_j denotes the average rank of the *j-th* classifier over the *n* data sets, then Nemenyi statistic test is computed as:

$$F_F=\frac{\left(n-1\right)\chi_F^2}{n\left(k-1\right)-\chi_F^2} \tag{8}$$

where $\chi_F^2=\frac{12n}{k\left(k+1\right)}\left[\sum_j R_j^2-\frac{k\left(k+1\right)^2}{4}\right]$

Table 1 Results of a SAX-Ensemble pre-evaluation

acc (vote)	acc (mindist)	acc (sum)	acc (prod)	acc (max)
71,74%	51,29%	37,80%	38,06%	37,02%

If the result of the above test is to reject the null hypothesis (i.e., the average rank of all classifiers is the same on data sets), Demšar recommends grouping classifiers into *cliques*, within which there is no significant difference in rank. This allows the average ranks and groups of not significantly different classifiers to be plotted on an order line in a graph called Critical Difference (CD) diagram. From this diagram, the better the classifier performs the smaller average rank is. However, for comparison between two classifiers, we use Wilcoxon signed-rank test which is the recommended statistical significance test for such comparisons (Japkowicz and Shah, 2011). The SAX-Ensemble classification scheme with the three forms of representation and majority vote combination rule is abbreviated as 'VOTE_x_fft_dx'. Furthermore, we add a term to indicate the used SAX-discretisation parameters selection technique, 'auto' for data-aware and 'sup' for data-agnostic technique. Finally, a value of the threshold factor δ for filtering the classifiers is added in the end the SAX-Ensemble abbreviation when a weighted vote rule is used. In our experiment, the factor is fixed to 0.75 to filter as many single classifiers as possible that perform worse in training.

In a first experimental evaluation, our SAX-Ensemble scheme adopting data-agnostic parameters estimation technique is evaluated in comparison with the baseline SAX-based classifier including 1NN SAX, 1NN ESAX, 1NN SAX-TD and 1NN SAX-BD. Figure 6(a) shows the Critical Difference of average ranks on classification accuracy conducted on 100 TS database from the UCR archive where the critical value for $p\text{-value}<0.01$ is 0.89. It is important to note that He et al. (2020) and Sun et al. (2014), provide the best SAX-discretisation parameters of the 1NN SAX-based algorithms through a grid search technique applied on testing set. In order to avoid this unrealistic practice, we consider that TS in the test data set is unseen then we only use training set with our data-agnostic technique. The CD diagram of Figure 6(a) shows that 1NN SAX-BD is ranked first while our 'VOTE_x_fft_dx_sup' ensemble is ranked third. However, there is no significance difference between 'VOTE_x_fft_dx_sup' and the 1NN SAX-TD classifier which is ranked second since they are grouped into the same

clique. We believe that the 1NN SAX-based classification accuracy is biased by the SAX-discretisation parameters found from the unseen testing data set. That is, results of the first experiment cannot lead to a fair comparison with our proposed SAX-Ensemble. For a realistic and fair comparison, we conduct a second experiment using parameters setting with data-aware technique for all evaluated algorithm. Table 2 shows the classification accuracy of our proposed SAX-Ensemble and other 1NN SAX-based algorithms on 100 TS database. From the CD diagram of the Figure 6(b), we can see that 'VOTE_x_fft_dx_auto_$\delta = 0.75$' ensemble classifier is ranked first. However, there is no significant difference between the two first ranked classification algorithms which are grouped into the same *clique*. Subsequently, we conduct an experiment to evaluate the influence of the threshold factor on the classification accuracy of the SAX-Ensemble scheme with majority vote combination rule. For this reason, we set the factor δ to the following three values: 0.7, 0.75 and 0.8. Figure 6(c) shows the CD diagram where there is no significant difference in the average ranking between ensemble classifiers based on VOTE and Weighted VOTE combination rules with a factor value above 0.7 using data-aware estimation of the SAX-discretisation parameters. In Figure 7(a), we present the average ranks of each SAX-Ensemble classification algorithm and their average training times. The 'VOTE_x_fft_dx_auto', denoted as VOTE_auto in the Figure 7, is faster than competitor SAX-ensemble scheme. In addition, the 'VOTE_x_fft_dx_auto' preserves its efficiency even when classifying large database with long TS data as one can see from Figure 7(b). Finally, we conduct experiments with and without additional forms of the TS data in order to observe the influence of including the first derivative and the periodogram on the classification results of the SAX-Ensemble. Figure 6(d) shows the CD diagram where we can see that SAX-Ensemble with additional forms significantly outperforms SAX-Ensemble without additional forms of the TS data. From these results, we suggest that 'VOTE_x_fft_dx_auto' is the best SAX-Ensemble scheme in term of scalability and efficiency. In order to test the significance of the 'VOTE_x_fft_dx_auto' SAX-Ensemble against others candidate classifiers, we use the Wilcoxon signed rank test where results are displayed in Figure 8. By a visual inspection, the more dots there are below the black slash the better performs 'VOTE_x_fft_dx_auto'. However, by a statistical evaluation, a *p*-value less than or equal to 0.05 indicates a significant improvement. We can see from Figure 8 that none of the 1NN SAX-based classifier has an equivalent performance with the SAX-Ensemble 'VOTE_x_fft_dx_auto' classification algorithm.

Table 2 Classification accuracy of the evaluated algorithms

TS Name	VOTE x_fft_dx auto	VOTE_x_fft_dx_ delta0,75	SAX	ESAX	SAXTD	SAXBD	SAXSD
ACSF1	0,50	0,50	0,13	0,30	0,54	0,35	0,15
Adiac	0,58	0,71	0,04	0,05	0,61	0,29	0,27
ArrowHead	0,82	0,79	0,39	0,40	0,86	0,42	0,70
Beef	0,73	0,73	0,50	0,53	0,73	0,57	0,73
BeetleFly	0,75	0,75	0,50	0,50	0,70	0,60	0,65
BirdChicken	0,80	0,80	0,50	0,50	0,60	0,80	0,70
BME	0,87	0,87	0,48	0,73	0,75	0,55	0,89
Car	0,75	0,75	0,23	0,23	0,77	0,47	0,75
CBF	0,54	0,56	0,33	0,33	0,37	0,41	0,66
Chinatown	0,86	0,86	0,27	0,27	0,91	0,67	0,95
CinCECGTorso	0,96	0,92	0,92	0,93	0,92	0,94	0,94
Coffee	0,96	0,96	0,54	0,54	0,79	0,71	0,96
Computers	0,57	0,57	0,53	0,49	0,54	0,56	0,57
CricketX	0,68	0,68	0,64	0,65	0,61	0,60	0,64
CricketY	0,62	0,61	0,57	0,58	0,56	0,53	0,60
CricketZ	0,66	0,65	0,63	0,64	0,63	0,61	0,65
DiatomSizeReduction	0,91	0,92	0,51	0,51	0,95	0,66	0,93
DistalPhalanxOutlineAgeGroup	0,72	0,71	0,43	0,42	0,65	0,63	0,70
DistalPhalanxOutlineCorrect	0,75	0,76	0,64	0,64	0,74	0,61	0,77
DistalPhalanxTW	0,64	0,65	0,37	0,50	0,59	0,54	0,55
Earthquakes	0,74	0,74	0,68	0,59	0,69	0,73	0,62
ECG200	0,91	0,90	0,87	0,91	0,88	0,87	0,92
ECGFiveDays	0,88	0,90	0,84	0,84	0,83	0,76	0,87
EOGHorizontalSignal	0,30	0,31	0,08	0,08	0,27	0,15	0,21
EOGVerticalSignal	0,45	0,44	0,40	0,41	0,41	0,40	0,43
EthanolLevel	0,30	0,33	0,25	0,25	0,30	0,26	0,29
FaceAll	0,73	0,73	0,69	0,69	0,68	0,64	0,73
FaceFour	0,81	0,82	0,80	0,80	0,78	0,80	0,81
FacesUCR	0,77	0,77	0,75	0,77	0,73	0,65	0,80
FiftyWords	0,61	0,61	0,58	0,60	0,61	0,60	0,63

Table 2 Classification accuracy of the evaluated algorithms (continued)

TS Name	VOTE x_fft_dx auto	VOTE_x_fft_dx_ delta=0,75	SAX	ESAX	SAXTD	SAXBD	SAXSD
Fish	0,73	0,77	0,13	0,13	0,75	0,25	0,74
FordA	0,83	0,92	0,69	0,69	0,67	0,70	0,67
FordB	0,70	0,78	0,58	0,59	0,59	0,60	0,60
Fungi	0,87	0,87	0,76	0,88	0,84	0,83	0,83
GunPoint	0,93	0,93	0,49	0,66	0,87	0,60	0,80
GunPointAgeSpan	0,94	0,94	0,51	0,51	0,91	0,54	0,91
GunPointMaleVersusFemale	0,99	0,99	0,53	0,53	0,97	0,70	0,97
GunPointOldVersusYoung	0,96	0,90	0,48	0,48	0,94	0,51	0,93
Ham	0,57	0,55	0,48	0,50	0,56	0,57	0,50
HandOutlines	0,89	0,89	0,79	0,72	0,84	0,78	0,87
Haptics	0,41	0,40	0,22	0,22	0,40	0,30	0,41
Herring	0,66	0,58	0,59	0,59	0,48	0,47	0,53
HouseTwenty	0,86	0,86	0,75	0,78	0,73	0,76	0,71
InlineSkate	0,34	0,34	0,22	0,22	0,30	0,24	0,29
InsectEPGRegularTrain	0,84	0,78	0,66	0,65	0,65	0,72	0,70
InsectEPGSmallTrain	0,84	0,84	0,65	0,65	0,66	0,76	0,67
InsectWingbeatSound	0,61	0,61	0,55	0,56	0,56	0,54	0,57
ItalyPowerDemand	0,95	0,96	0,50	0,50	0,91	0,55	0,81
LargeKitchenAppliances	0,69	0,69	0,48	0,47	0,49	0,45	0,48
Lightning2	0,75	0,75	0,79	0,84	0,75	0,64	0,79
Lightning7	0,38	0,44	0,26	0,26	0,36	0,25	0,48
Meat	0,88	0,88	0,33	0,33	0,85	0,63	0,95
MedicalImages	0,67	0,68	0,51	0,63	0,66	0,59	0,63
MelbournePedestrian	0,75	0,75	0,20	0,37	0,81	0,46	0,78
MiddlePhalanxOutlineAgeGroup	0,56	0,56	0,50	0,50	0,47	0,45	0,49
MiddlePhalanxOutlineCorrect	0,79	0,80	0,53	0,55	0,75	0,60	0,69
MiddlePhalanxTW	0,58	0,58	0,27	0,27	0,51	0,44	0,49
MoteStrain	0,82	0,81	0,81	0,84	0,85	0,77	0,84
NonInvasiveFetalECGThorax1	0,84	0,84	0,67	0,75	0,78	0,74	0,81
NonInvasiveFetalECGThorax2	0,90	0,90	0,66	0,75	0,86	0,79	0,85
OliveOil	0,83	0,87	0,17	0,17	0,83	0,37	0,83
OSULeaf	0,60	0,60	0,51	0,53	0,52	0,53	0,54
PhalangesOutlinesCorrect	0,79	0,79	0,58	0,53	0,76	0,57	0,77
Phoneme	0,19	0,19	0,06	0,10	0,09	0,09	0,12
PigAirwayPressure	0,10	0,10	0,08	0,07	0,07	0,07	0,07
PigArtPressure	0,46	0,46	0,02	0,02	0,12	0,11	0,11
PigCVP	0,38	0,38	0,08	0,08	0,08	0,08	0,07
Plane	0,97	0,97	0,14	0,14	0,96	0,50	1,00
PowerCons	0,88	0,92	0,93	0,92	0,91	0,89	0,94
ProximalPhalanxOutlineAgeGroup	0,87	0,86	0,43	0,44	0,81	0,66	0,80
ProximalPhalanxOutlineCorrect	0,84	0,86	0,68	0,70	0,81	0,68	0,84
ProximalPhalanxTW	0,75	0,76	0,01	0,01	0,74	0,48	0,71
RefrigerationDevices	0,45	0,45	0,34	0,32	0,34	0,38	0,34
Rock	0,76	0,76	0,76	0,74	0,78	0,74	0,80
ScreenType	0,37	0,37	0,36	0,34	0,35	0,33	0,33
SemgHandGenderCh2	0,75	0,75	0,56	0,53	0,53	0,72	0,73
SemgHandMovementCh2	0,47	0,47	0,20	0,19	0,24	0,34	0,44
SemgHandSubjectCh2	0,63	0,63	0,16	0,23	0,28	0,49	0,62
ShapeletSim	0,72	0,72	0,53	0,57	0,53	0,52	0,61
ShapesAll	0,82	0,82	0,39	0,46	0,74	0,60	0,77
SmallKitchenAppliances	0,58	0,58	0,34	0,34	0,32	0,45	0,35
SmoothSubspace	0,43	0,74	0,33	0,72	0,83	0,64	0,98
SonyAIBORobotSurface1	0,75	0,80	0,69	0,70	0,67	0,76	0,69
SonyAIBORobotSurface2	0,89	0,89	0,86	0,88	0,86	0,83	0,85
Strawberry	0,97	0,96	0,64	0,64	0,95	0,68	0,94
SwedishLeaf	0,87	0,87	0,58	0,62	0,75	0,75	0,82
Symbols	0,91	0,91	0,16	0,16	0,84	0,56	0,83
SyntheticControl	0,66	0,68	0,17	0,17	0,36	0,39	0,38
ToeSegmentation1	0,81	0,83	0,65	0,68	0,66	0,64	0,66
ToeSegmentation2	0,85	0,84	0,86	0,85	0,85	0,86	0,85
Trace	0,91	0,91	0,39	0,48	0,59	1,00	0,88
TwoLeadECG	0,92	0,92	0,53	0,61	0,82	0,67	0,81
TwoPatterns	0,70	0,71	0,26	0,30	0,41	0,20	0,44
UMD	0,85	0,85	0,70	0,67	0,77	0,69	0,79
Wafer	1,00	1,00	1,00	1,00	1,00	0,99	1,00
Wine	0,56	0,56	0,50	0,50	0,48	0,50	0,57
WordSynonyms	0,61	0,61	0,55	0,57	0,57	0,55	0,58
Worms	0,65	0,62	0,48	0,51	0,52	0,53	0,52
WormsTwoClass	0,74	0,71	0,65	0,66	0,68	0,69	0,68
Yoga	0,80	0,80	0,63	0,65	0,81	0,67	0,84

Figure 6 The groups of evaluated classification algorithms that are not significantly different from each other at $p < 0.01$ using the Nemenyi test

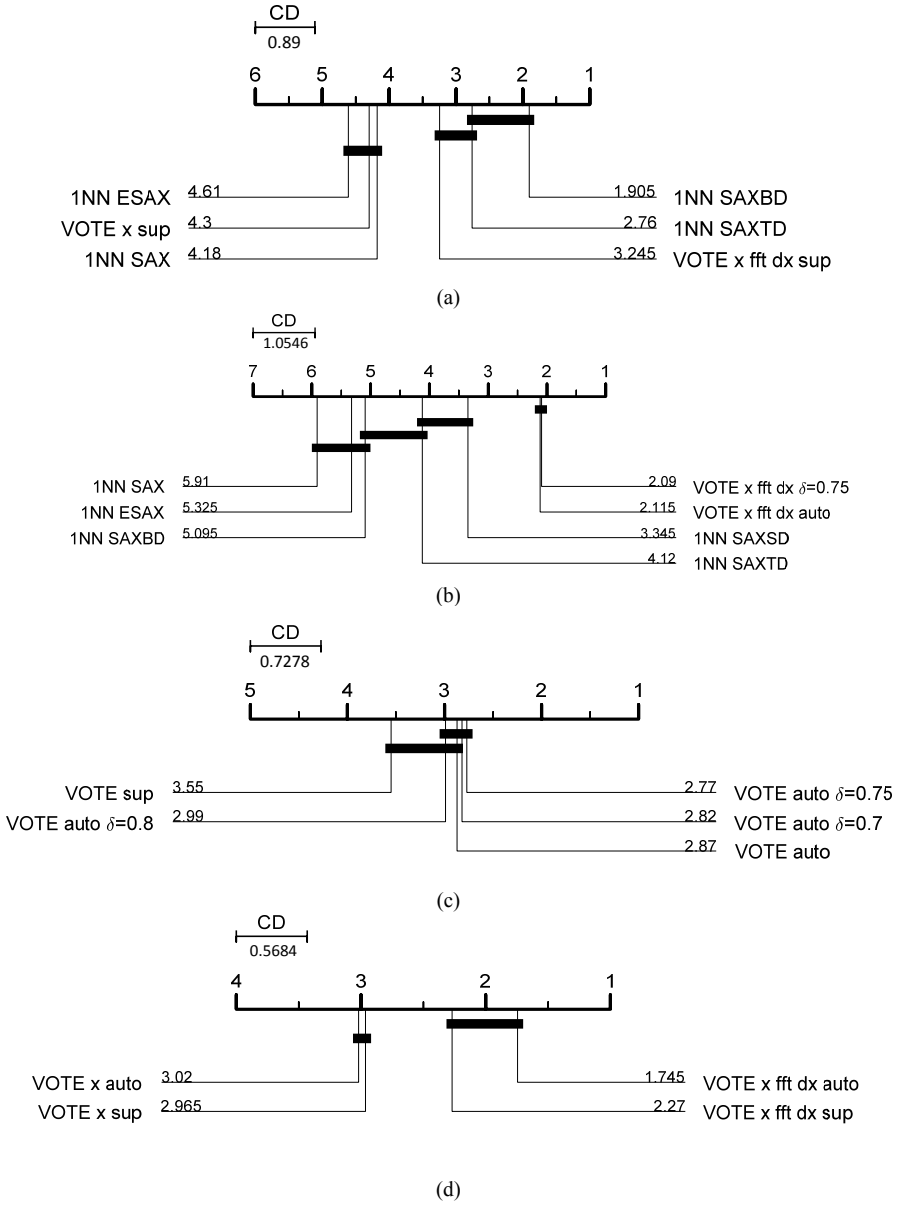


Figure 7 Average rank versus average training times (in logscale) of the SAX-Ensemble algorithms evaluated in Figure 6(b) when classifying (a) all of the one hundred TS database, (b) the three largest TS data sets NonInvasiveFetalECGThorax1, NonInvasiveFetalECGThorax2 and HandOutlines

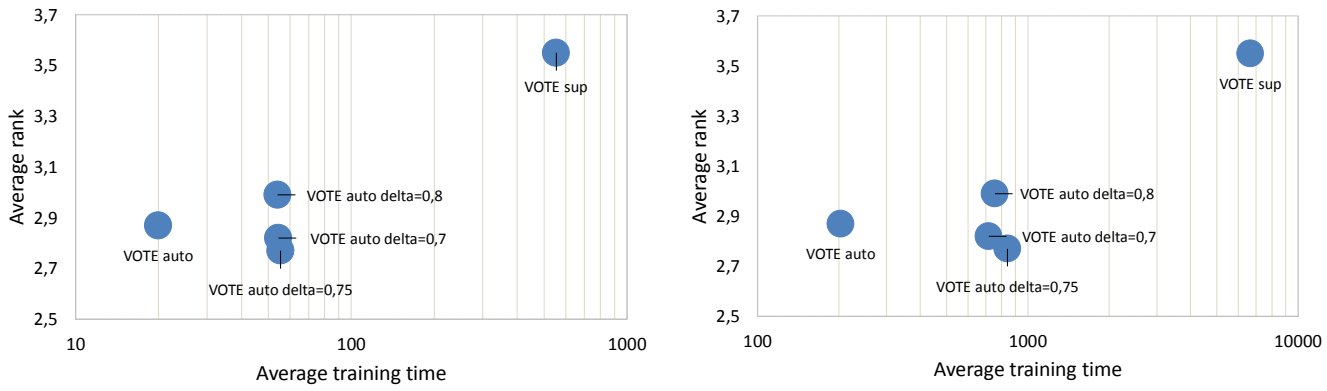
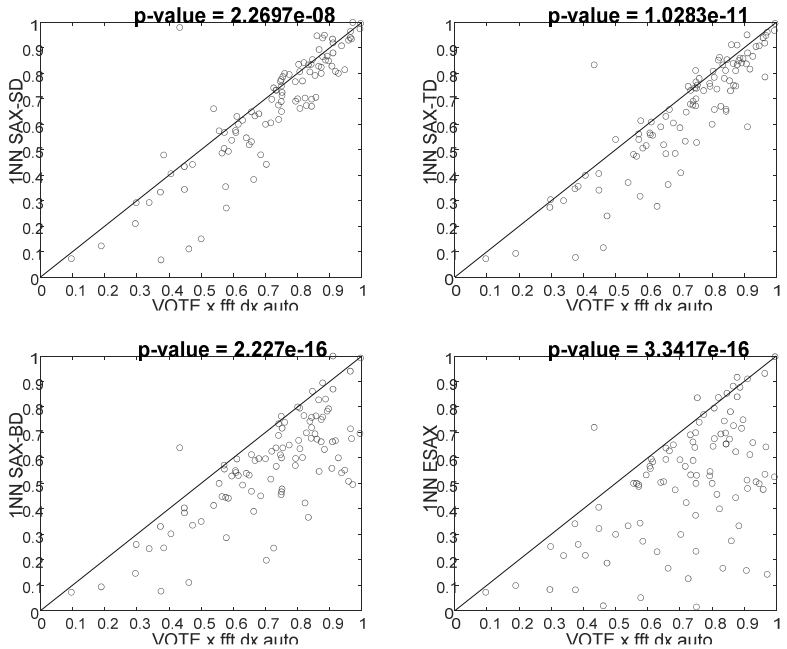


Figure 8 The SAX-Ensemble VOTE_x_fft_dx_auto algorithm is compared with 1NN SAX-SD, 1NN SAX-TD, 1NN SAX-BD and 1NN ESAX algorithms



Intuitively, we think that the proposed SAX-Ensemble scheme opens a new perspective for the classification of a weakly labelled TS data since the data-aware technique does not require the label information in the selection process. In order to observe the influence of the label information on the classification results, we conduct an experiment where we set the instances ratio in the training data set to a different value. The SAX-discretisation parameters are selected via data-aware/data-agnostic methods for SAX-Ensemble and via LOOCV on training set for the 1NN SAX-based candidate. Figure 9 shows the classification accuracy results of four TS databases for a ratio of labelled instances that vary from 0.75 to 0.15. From Figure 9, both SAX-Ensemble schemes show an almost stable accuracy with the decrease of the labelled instance in the training set. Particularly, the 'VOTE_x_fft_dx_auto' scheme provides a good classification accuracy with a small time-cost as one can see from Figure 10.

Figure 9 The 1NN classification accuracy (in %) of four TS databases with decreasing rate of the labelled data in the training set

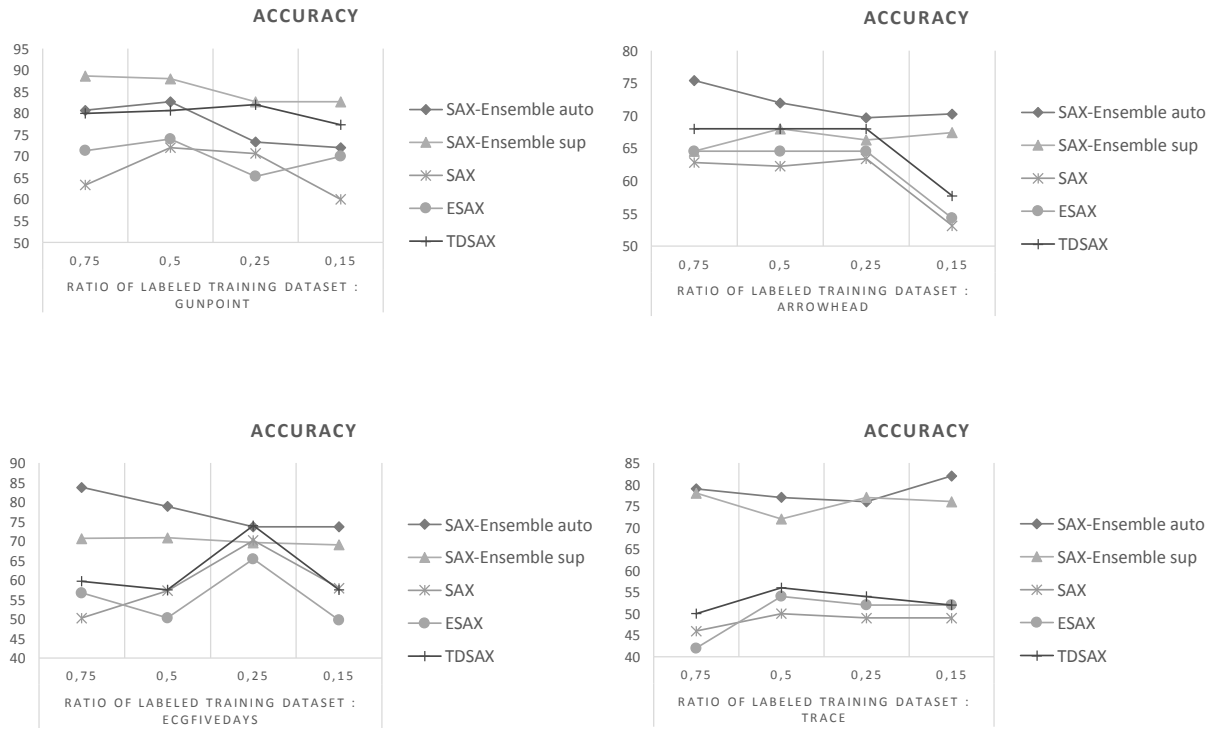
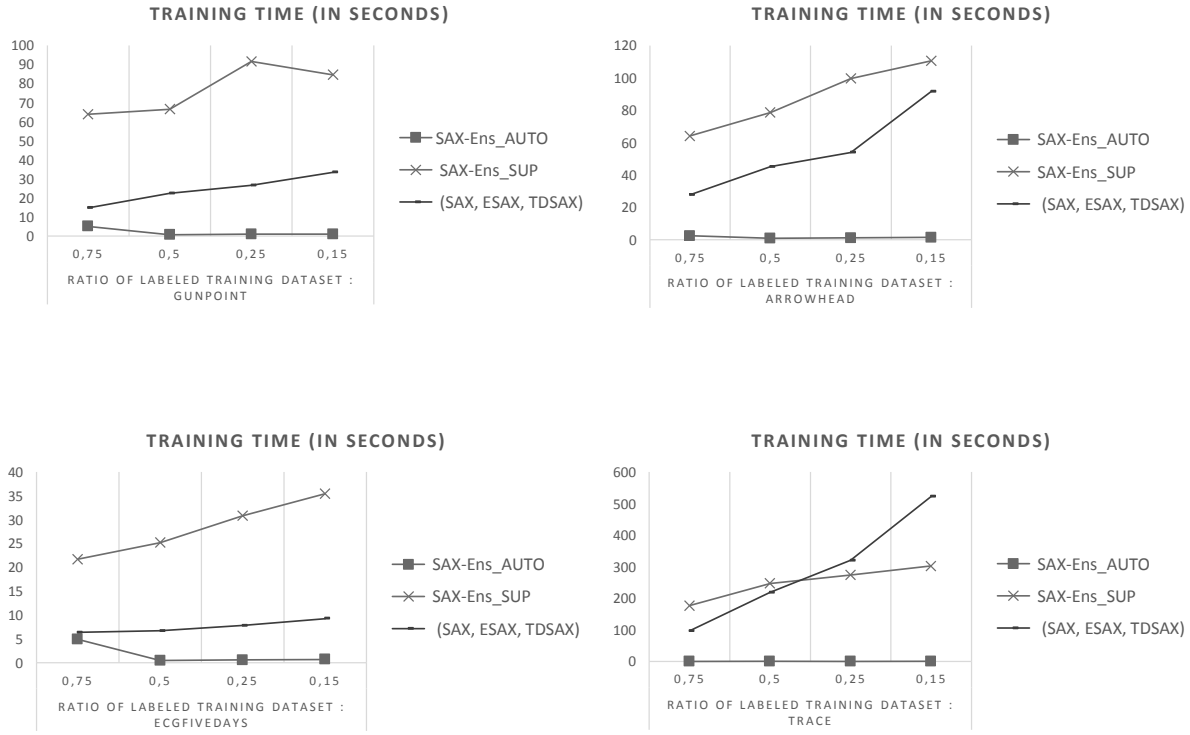


Figure 10 The average training time (in logscale) of four TS databases with decreasing rate of the labelled data in the training set



6 Conclusion and future work

In this paper, a novel SAX-Ensemble scheme has been proposed for TS classification based on their shape. The conclusions can be drawn as follows:

1 To extract comprehensive shape information from TS data, multi-SAX features are integrated through an ensemble scheme. Consequently, the original SAX distance is enhanced by adding several SAX-based distance measures.

2 Although ensemble is well-known for its high demand of computing resources, SAX-Ensemble uses multiple domain representation of TS with data-aware SAX resolution which drastically reduces time-cost.

3 To combine classifiers from multiple domain representation, we suggest the use of the simple majority vote rule which enjoys better classification accuracy.

4 Results of extensive experiments on UCR archive database show that 'VOTE_x_fft_dx_auto' algorithm improves the results achieved by single 1NN with SAX-distance measure.

As future work, we plan to extend our SAX-Ensemble algorithm in order to design a hybrid ensemble scheme which integrates SAX-based structure information like bag-of-patterns or bag-of-features. Indeed, we intend to study an extended data-aware technique which leads to a multiple resolution of the TS through sampling scheme of the training set (e.g., Bagging).

References

Asmita, M., Nonita, S., Firas, H.A., Monika, M., Krishna, P.S. and Rajneesh, R. (2021) 'An ensemble approach to forecast COVID-19 incidences using linear and non-linear statistical models', *International Journal of Computer Applications in Technology*, Vol. 66, pp.415–426.

Bagnall, A., Lines, J. and Bostrom, A. (2015) 'Time-series classification with COTE: the collective of transformation-based ensembles', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 27, pp.2522–2535.

Batista, G.E.A.P.A., Keogh, E., Tataw, O.M. and Souza, V.M.A. (2014) 'CID: an efficient complexity-invariant distance for time series', *Data Mining and Knowledge Discovery*, Vol. 28, No. 3, pp.634–669.

Chaw, T.Z. and Hayato, Y. (2016) 'An improved symbolic aggregate approximation distance measure based on its statistical features', *Proceeding of the 18th International Conference on Information Integration and Web-based Applications and Services*, 28–30 November, Singapore, pp.72–80.

Chaw, T.Z. and Hayato, Y. (2017) 'Dynamic SAX parameter estimation for time series', *International Journal of Web Information Systems*, Vol. 13, No. 4, pp.387–404.

Daoyuan, L., Tegawendé, F.B., Jacques, K. and Yves, L.T. (2016) 'DSCo-NG: a practical language modeling approach for time series classification', *Proceedings of the 15th International Symposium Intelligent Data Analysis*, 13–15 October, Stockholm, pp.1–13.

Dau, H.A., Bagnall, A., Kamgar, K., Yeh, C., Zhu, Y., Gharghabi, S., Ratanamahatana, C.A. and Keogh, E. (2019) 'The UCR time series archive', *IEEE/CAA Journal of Automatica Sinica*, Vol. 6, pp.1293–1305.

Demšar, J. (2006) 'Statistical comparisons of classifiers over multiple data sets', *The Journal of Machine Learning Research*, Vol. 7, pp.1–30.

Duda, R.O., Hart, P.E. and Stork, D.G. (2012) *Pattern Classification*, John Wiley & Sons.

Fuad, M.M.M. (2020) 'Modifying the symbolic aggregate approximation method to capture segment trend information', *Proceedings of the 17th International Conference on Modeling Decisions for Artificial Intelligence*, 2–4 September, Spain, pp.230–239.

Gorecki, T. (2014) 'Using derivatives in a longest common subsequence dissimilarity measure for time series classification', *Pattern Recognition Letters*, Vol. 45, pp.99–105.

He, Z., Long, S., Ma, X. and Zhao, H. (2020) 'A boundary distance-based symbolic aggregate approximation method for time series data', *Algorithms*, Vol. 13, pp.284–304.

Japkowicz, N. and Shah, M. (2011) *Evaluating Learning Algorithms: An Classification Perspective*, Cambridge University Press.

Kate, R.J. (2016) 'Using dynamic time warping distances as features for improved time series classification', *Data Mining and Knowledge Discovery*, Vol. 30, No. 2, pp.283–312.

Lin, J., Keogh, E., Lee, W. and Lonardi, S. (2007) 'Experiencing SAX: a novel symbolic representation of time series', *Data Mining and Knowledge Discovery*, Vol. 15, No. 2, pp.107–144.

Lin, J., Khade, R. and Li, Y. (2012) 'Rotation-invariant similarity in time series using bag-of-patterns representation', *Journal of Intelligent Information Systems*, Vol. 39, No. 2, pp.287–315.

Lines, J. and Bagnall, A. (2015) 'Time series classification with ensembles of elastic distance measures', *Data Mining and Knowledge Discovery*, Vol. 29, No. 3, pp.565–592.

Lines, J., Taylor, S. and Bagnall, A. (2016) 'HIVE-COTE: the hierarchical vote collective of transformation-based ensembles for time series classification', *Proceedings of the 16th IEEE International Conference on Data Mining*, pp.1041–1046.

Lkhagva, B., Suzuki, Y. and Kawagoe, K. (2006) 'New time series data representation ESAX for financial applications', *Proceedings of the IEEE International Conference on Data Engineering*, pp.17–22.

Senin, P. and Malinchik, S. (2013) 'SAX-VSM: interpretable time series classification using SAX and vector space model', *Proceedings of the IEEE International Conference on Data Mining*, pp.1175–1180.

Simon, M., Thomas, G., René, Q. and Romain, T. (2013) '1d-SAX: a novel symbolic representation for time series', in Tucker, A. et al. (Eds): *International Symposium on Intelligent Data Analysis*, pp.273–284.

Sun, Y., Li, J., Liu, J., Sun, B. and Chow, C. (2014) 'An improvement of symbolic aggregate approximation distance measure for time series', *Neurocomputing*, Vol. 138, pp.189–198.

Thach, L.N., Severin, G., Iulia, I., O'Reilly, M. and Georgiana, I. (2019) 'Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations', *Data Mining and Knowledge Discovery*, Vol. 33, No. 4, pp.1183–1222.

Vineeth, N.B., Shen-Shyang, H., Vladimir, V. (2014) *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications*, Elsevier.

Wang, X., Lin, J., Senin, P., Oates, T., Gandhi, S., Boedihardjo, A.P., Chen, C. and Frankenstein, S. (2016) 'RPM: representative pattern mining for efficient time series classification', *Proceedings of the 19th International Conference on Extending Database Technology*, pp.1–12.

Yin, J., Xu, M. and Zheng, H. (2019) 'Fault diagnosis of bearing based on symbolic aggregate approximation and Lempel-Zif', *Measurement*, Vol. 138, pp.206–216.

Zhao, J. and Itti, L. (2018) 'shapeDTW: shape dynamic time warping', *Pattern Recognition*, Vol. 74, pp.171–184.