Classifying of Time Series using Local Sequence Alignment and Its Performance Evaluation

Keiichi Tamura, *Member, IAENG*, Takumi Ichimura

***Abstract*—Time series classification is the task of predicting the class label of an unclassified time series. In the era of big data, time series classification is one of the best-known grand challenges because of its many fields of application and difficulty. There are three important things that we need to consider in time series classification; representation, similarity measurement, and assignment strategy. Representation for time series is a technique that converts time series to feature vectors representing the characteristics of time series. In the last decade, Symbolic Aggregate approXimation (SAX), which is a state-of-the-art feature expression for time series, has attracted the attention of many data mining researchers, because huge number of good sequence data mining algorithms are available once time series are converted to SAX sequences. In this paper, we propose a novel method for time series classification using a hybrid SAX-based symbolic representation, which is called a moving average convergence divergence (MACD)-histogram-based SAX (MHSAX) proposed in our previous work. The proposed time series classification method includes the MHSAX and a nearest neighbor (1-NN) classifier utilizing the local sequence alignment technique. To evaluate the proposed time series classification method, we implemented it and conducted experiments using all 85 data sets in the UCR Time Series Classification Archive. The experimental results show that the proposed time series classification method outperforms not only other distance-based 1-NNs, but also other state-of-the-art methods.**

***Index Terms*—Time series classification, SAX, MACD histogram, Local sequence alignment**

I. Introduction

TIME series are temporal data representing as sequences of data observed periodically or observations collected at regular intervals. Time series are ubiquitous in any domain of natural science involving temporal data measurements, such as engineering, medical science, astronomy, and sociology [1], [2]. In last few decades, data mining techniques for time series have been active research topics, because knowledge discovery on time series data is beneficial for the broad range of applications. Data mining researchers and practitioners have been studying a wide range of time series data mining techniques, from basic methods, such as classification, clustering, prediction, frequent pattern and motif extraction, similarity search, and anomaly detection, to large-scale time series management, parallel processing, and time series indexing structures [3], [4], [5], [6].

There are several different types of time series; in this study, we focus on a simple time series that is a sequence of primitive items (e.g., real numbers, integer values, or symbols), including sensor-monitored values, stock prices, currency exchange rates, radio waves, electrocardiogram values, event streams, earthquake waves, and biomedical signals. With the emergence of the Internet of Things (IoTs), the use of time series data generated by sensor devices has been attracting attention. Time series data from devices of IoTs are usually simple time series. For example, temperatures observed by temperature sensors are series of real values. Large scale time series databases have been constructed on the Internet. Developing new data mining techniques for a large scale time series databases is one of the most important key challenges in the era of big data.

Time series classification is the task that identifies the class label of an unlabeled time series using training data whose class labels are known in advance [7]. Time series data mining researchers and practitioners have been tackling developing methods for time series classification, because it has a broad range of applications from science to engineering, including biological analysis, electroencephalogram, image and motion recognition, and financial analysis. Each data point in a time series is simple; however, in contract to multivariate data, time series data are sequence data, such as strings; therefore, there is a specific need to capture time-variable features. In the design of time series classifiers three points should be considered: the feature expression for the time series, the definition of the distance function, and the classification strategy.

High-level symbolic representation is one of the most robust techniques for the feature expression of a time series. In high-level symbolic representation, a time series is encoded into a sequence of symbols in order to eliminate the influence of noise. Techniques for the symbolic representation of time series allow a rich variety of sequence algorithms (e.g., subsequence search, frequent sequence mining, and sequence alignment) to be applied to time series data. This has motivated researchers to utilize well-known string algorithms to improve the performance of time series data mining. In particular, symbolic Aggregate approXimation (SAX) [8] is one of the best-studied high-level symbolic representations for time series because it can compress time series and provide a variety of measurement metrics. SAX is the first symbolic representation for time series that allows for dimensionality reduction and indexing with a lower-bounding distance measure.

In our previous work [9], we proposed nearest neighbor (1-NN) SAX-based time series classification methods that utilize a moving average convergence divergence (MACD)-histogram-based SAX (MHSAX) representations. Moving average convergence divergence (MACD) histogram [10] is the acceleration of time that represents the latent features of time series. MHSAX is a hybrid representation of SAX representations of a time series and its MACD histograms. MHSAX adequately captures not only the local variation,

K.Tamura is with Graduate School of Information Sciences, Hiroshima City University, 3-4-1, Ozuka-Higashi, Asa-Minami-Ku, Hiroshima 731-3194 Japan, corresponding e-mail: ktamura@hiroshima-cu.ac.jp

T.Ichimura is with Department of Management and Systems, Prefecture University of Hiroshima, 1-1-71, Ujina-Higashi, Minami-ku, Hiroshima 734-8559, Japan, corresponding e-mail: ichimura@pu-hiroshima.ac.jp

but also the global variation in time series. To evaluate the method, we implemented it and experiments were conducted by using the UCR time series classification archive [11]. The experimental results showed that the proposed time series classification methods outperform other distance-based 1-NNs. Moreover, subsequent work [12] proposed a new measurement for MHSAX.

MHSAX is a superior high-level representation; however, there is room for further improvement in the accuracy of the calculation of the distance between time series. In this paper, we propose a novel method for time series classification involving the MHSAX representation. The proposed time series classification method includes the MHSAX representation and a nearest neighbor (1-NN) classifier utilizing the local sequence alignment technique. The main contributions of this study are as follows.

- The proposed time series classifier is a novel 1-NN SAX-based classifier utilizing the local sequence alignment technique, which is used in the bioinformatics field and is useful for distinguishing dissimilar sequences that are suspected to contain regions of similar sequence motifs within their larger sequences. The Smith-Waterman algorithm [13] is a general local alignment method based on dynamic programming. We modified this algorithm so that it measures the distance between two MHSAX representations using the algorithm.
- To evaluate the proposed time series classification method, experiments were conducted by using the whole data sets of the UCR Time Series Classification Archive [11]. The proposed time series classification method is compared with other distance-based 1-NNs. The experimental results showed that the proposed time series classification method outperforms other distance-based 1-NNs. In addition, the proposed time series classification method shows good performance compared with our previous method and the other latest SAX-based methods.

The rest of this paper is organized as follows. In Sections II and III, related work and MHSAX are respectively summarized and described briefly. In Section IV, a novel method for time series classification is proposed. In Section V, the experimental results are shown, and we discuss the method's performance. We conclude the paper in Section VI.

II. Related Work

There are three major approaches to classify time series: distance-based, feature-based, and model-based approaches [14]. The distance-based approach defines the distance function, measuring the distance between time series, and classifies the time series with reference to the mutual distance. The feature-based approach discovers the signature subsequences of a time series and classifies the time series according to whether the time series includes these signature subsequences. The model-based approach attempts to apply statistical model analysis to time series classification.

The distance-based approach has been well-studied, and many studies have reported that 1-NN is the simplest and yet most stable algorithm. The early studies were based on the Euclidean distance; however, the Euclidean distance is not robust against slight gaps between time series and differences in their shapes. To address this problem, the dynamic time warping (DTW) distance was proposed [15]. DTW improves the performance of time series classification dramatically. Ding et al. [3] reported that 1-NN with DTW, in general, performs well, and the difference between its performance and that of other subsequent distance metrics is small.

Shapelets [16], [17] are one of the most well-known techniques for feature-based and model-based approaches. Shapelets are segments of time series that identify class efficiently. They are extracted by evaluating the class prediction qualities of numerous candidates extracted from the series segments. Since SAX was proposed, researchers have focused on the feature-based approach using SAX. SAX-VSM [18] is a state-of-the-art algorithm based on SAX and the "bag of words" model. Each class is represented by a feature vector and the feature vector is weighted by TF*IDF weighting. An unlabeled time series is assigned to a class in which the unlabeled time series has the highest feature score.

Recently, some state-of-the-art methods have been proposed. Silva et al. [19] proposed recurrence plots for time series feature representation. To measure the distance between two time series, they use Campana-Keogh (CK-1) distance, which is a Kolmogorov complexity-based distance for estimating image similarity. Gormes et al. [20] proposed a novel feature-based method in which frequent sequences of symbols (motifs) are defined as features that are included only in a specific class. Decision trees are then constructed using the extracted motifs. Kamath et al. [21] proposed a feature construction algorithm based on genetic programming. In addition, Wang et al. [22] introduced a completely new method in which deep learning techniques are applied.

The proposed time series classification method uses the SAX-based approach. The SAX-based method is limited in terms of discrimination capability because it cannot capture the local variation in a time series. MACD histograms facilitate the recognition of local variation in a time series; therefore, MHSAX improves the class identification rate of the time series. The method most similar to ours was presented in [23], where Zhao et al. proposed a new DTW-based method named shapeDTW. DTW can capture the global variation; however, it does not necessarily achieve locally sensible matches. To address this issue, shapeDTW attempts to pair locally similar subsequences and to avoid matching points with distinct neighborhoods. In contrast to conventional methods, our method encodes time series into SAX-based high-level symbolic representations because noise can then be ignored.

III. MACD-Histogram-based SAX

In this section, SAX, MACD histogram and MHSAX are described more in detail.

A. Symbolic Aggregate approximation

High-level symbolic representation of time series, once introduced, has attracted much attention by enabling applications of numerous sequence mining algorithms, bioinformatics, and text mining to temporal data. Symbolic Aggregate Approximation (SAX) algorithm [8] is one of the most active techniques for high-level symbolic representation for time series and being used widely by researchers to analyze time

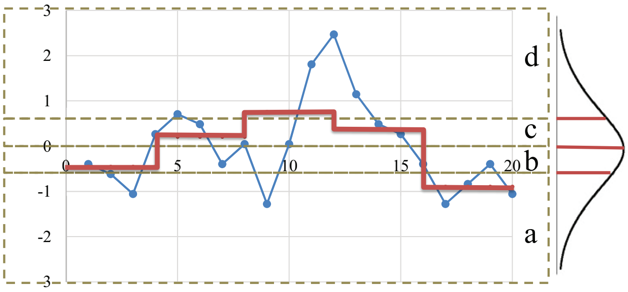

Fig. 1. Example of SAX

series and streaming data. There are two aspect of SAX: compression of a time series and conversion of the time series into symbols. SAX reduces the length of the time series and transforms the compressed time series into a symbolic string. After SAX was proposed, it enthralled time series researchers because it is a simple and intuitive representation. Moreover, the lower bound of the distance between SAX representations of two different time series can be calculated and this allows conventional sequence algorithms to be utilized efficiently.

SAX is a symbolization technique to convert a time series to a sequence string. There are three main steps in SAX: (1) normalization, (2) compression using piecewise aggregate approximation (PAA) [24], and (3) discretization. In the normalization step, each time series is normalized such that the mean and standard deviation are zero and one, respectively. In the compression using the PAA step, a compressed time series is created, where the length is reduced from n to l, where $l \leq n$. In the discretization step, each value of the compressed time series is converted into a discrete symbol from a set of α symbols.

TABLE I
BREAKPOINTS

α	3	4	5	6	7	8
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15
β_2	0.43	0	-0.25	-0.43	-0.57	-0.76
β_3		0.67	0.25	0	-0.18	-0.32
β_4			0.84	0.43	0.18	0
β_5				0.97	0.57	0.32
β_6					1.07	0.76
β_7						1.15

The details of a SAX representation of a the time series are as follows. Let the i-th time series in a time series data set TS be $T_i = (t_{i,1}, t_{i,2}, \cdots, t_{i,n})$. In this study, T_i is a simple time series, where each value is a primitive value such as an integral value or real number. In the normalization step, for each value of T_i, $t_{i,j}$ is normalized to the value

$$c_{i,j} = \frac{t_{i,j} - avg}{sd}, \tag{1}$$

where $avg = (\sum_{i=1}^{m}\sum_{j=1}^{n} t_{i,j})/(n \times m)$ and $sd = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}(t_{i,j} - avg)^2/(n \times m)}$. Let the i-th normalized time series of n lengths be

$$C_i = (c_{i,1}, c_{i,2}, \cdots, c_{i,n}). \tag{2}$$

In the compression using the PAA step, C_i is divided into l frames, where each frame has the same length $w = n/l$. The average of the values in each frame represents the frame. Thus, C_i of length n is compressed into a time series of length l. Let the PAA representation of C_i be $P_i = (p_{i,1}, p_{i,2}, \cdots, p_{i,l})$, where the j-th value of P_i is defined as follows:

$$p_{i,j} = \frac{1}{w} \sum_{k=w\times(j-1)+1}^{w\times j} c_{i,k}. \tag{3}$$

In the discretization step, the codomain of the real number is first divided into α regions, where the boundaries between areas are determined by equalizing the area of each region under the $N(0,1)$ Gaussian curve. The number α is called the cardinality. The boundaries are called breakpoints, and an ordered list of the breakpoints' values is denoted by

$$\begin{aligned} B = (\beta_0, \beta_1, \cdots, \beta_{\alpha-1}, \beta_\alpha), \\ where \beta_i < \beta_{i+1}, \beta_0 = -\infty, and\ \beta_\alpha = +\infty). \end{aligned} \tag{4}$$

If there are α regions, the area of the region between the breakpoints β_{i-1} and β_i is $1/\alpha$.

For cardinality is α, there are α symbols for mapping a symbol to each region. Let a set of symbols be $\Sigma = \{\Sigma_1, \cdots, \Sigma_\alpha\}$. The value of $p_{i,j}$ is mapped to a symbol according to

$$s_{i,j} = \Sigma_k, \quad iif \quad \beta_{k-1} \leq p_{i,j} < \beta_k. \tag{5}$$

Let a sequence of the assigned symbols be $S_i = (s_{i,1}, s_{i,2}, \cdots, s_{i,l})$. This sequence is called a SAX string. A SAX string, where T_i is encoded on the condition that the cardinality is α and the size of window is w, is denoted by $SAX(w,\alpha)[T_i]$. The j-th element of $SAX(w,\alpha)[T_i]$ is also denoted by $SAX(w,\alpha)[T_i]_j$. Fig. 1 shows an example of a SAX representation of a time series. The blue and red lines show a normalized time series and its PAA representation, respectively. The domain is divided into four regions so that the cardinality is four. Each region is assigned a symbol "a," "b," "c," or "d." The time series is hence converted to SAX string "bcdcca."

B. MACD Histogram

The MACD and its histogram are defined as the velocity and the acceleration of a time series. These criteria are used for the technical analysis of stock prices, which provides the indicator of stock trading. A series of stock prices is referred to as a time series; therefore, the chances of profiting from trading a stock can be determined by analyzing the time series of stock prices. In the theory, an object moves in a two-dimensional space and a time series is regarded as trajectories of its two-dimensional positions. Velocity and acceleration of the object are calculated using the observed changes in position.

MACD is the difference between the two types of exponential moving averages (EMAs). The EMA is a type of weighted moving average known as an exponentially weighted moving average. The weighting for each older value in a time series decreases exponentially. The definition of the EMA for the t-th element of T_i is

$$\begin{aligned} ema(ws)[T_i]_t &= \gamma \times t_{i,t} + (1-\gamma) ema[T_i]_{t-1} \\ &= \sum_{k=0}^{ws} (\gamma(1-\gamma)^k t_{i,(t-k)}), \end{aligned} \tag{6}$$

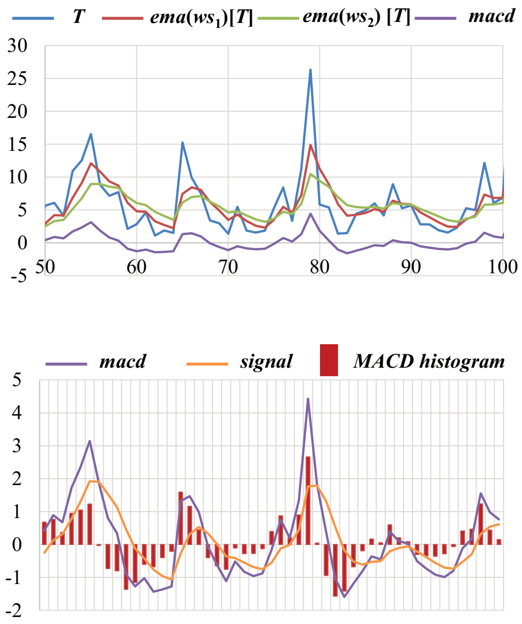

Fig. 2. Example of MACD histogram

where ws is the size of the sliding window, and $\gamma = 2/(ws-1)$. Suppose that $t = k$; this implies that the average is calculated using the $(k - ws)$-th to the k-th element. Fig. 2 shows a simple example. The blue line is an time series and EMAs are the moving average of T

Let the time series of the EMA values of T_i under ws be $ema(ws)[T_i]$. The difference between $ema(ws_1)[T_i]_t$ and $ema(ws_2)[T_i]_t$ is called the MACD, where $ws_1 \neq ws_2$:

$$macd(ws_1, ws_2)[T_i]_t = ema(ws_1)[T_i]_t - ema(ws_2)[T_i]_t, \ ws_1 < ws_2. \quad (7)$$

The MACD is considered to be a derivative value of the EMA and is a velocity. In Fig. 2, the MACD of T is the difference between the long term EMA and the short term EMA. A value in the MACD is the difference between the red line and the green line.

The EMA of the MACD, where the size of window is ws_3, is called the MACD signal. The difference between the signal and the MACD is called the MACD histogram.

$$signal(ws_1, ws_2, ws_3)[T_i]_t = ema(ws_3)[macd(ws_1, ws_2)[T_i]]_t, \quad (8)$$

$$histogram(ws_1, ws_2, ws_3)[T_i]_t = macd(ws_1, ws_2)[T_i]_t - signal(ws_1, ws_2, ws_3)[T_i]_t, \quad (9)$$

The MACD histogram is a derivative value of the MACD and is regarded as the acceleration of the time series. In Fig. 2, the purple line and the orange lines are the MACD and the MACD signal of T. The red bars are the values of the MACD histogram.

C. Definition

By utilizing SAX strings of time series and analyzing techniques for sequences, it is possible to capture the characteristics of the whole time series data. Discretization brings robustness to noise. However, depending on the use of discretization the degree of change in value may be ignored.

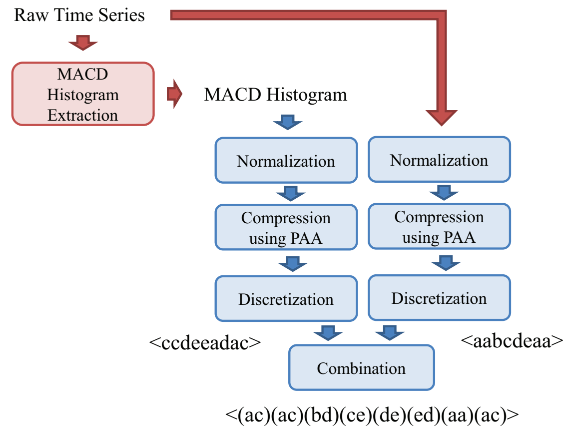

Fig. 3. MHSAX

To compensate for this drawback, the MACD histogram of time series is embedded in SAX. In MHSAX approach, a time series is defined as a sequence of a pair of values of time series and its MACD histogram.

The MHSAX sequence of a time series is a hybrid high-level symbolic representation of SAX. A MHSAX string is a sequence that merges two different types of SAX sequences: the SAX sequence of a time series and the SAX sequence of the MACD histograms of the time series. Let the SAX of T_i and $histogram(ws_1, ws_2, ws_3)[T_i]$ be $SAX(w, \alpha)[T_i]$ and $SAX(w, \alpha)[histogram(ws_1, ws_2, ws_3)[T_i]]$, respectively. For brevity, $SAX(w, \alpha)[T_i]$ is denoted by $OSAX(p)[T_i]$, and $SAX(w, \alpha)[histogram(ws_1, ws_2, ws_3)[T_i]]$ is denoted by $MSAX(p)[T_i]$, where p is the set of parameters $\{w, \alpha, ws_1, ws_2, ws_3\}$.

MHSAX is a sequence that alternates elements of $OSAX(p)[T_i]$ and $MSAX(p)[T_i]$. In particular, the sequence is $(OSAX[T_i]_1, MSAX(w, \alpha)[T_i]_1,\ OSAX[T_i]_2,\ MSAX(w, \alpha)\ [T_i]_2, \cdots, OSAX[T_i]_l,\ MSAX(w, \alpha)[T_i]_l)$, where $l = n/w$. The MHSAX sequence of T_i is denoted by $MHSAX(p)[T_i]$. Suppose that $OSAX(p)[T_i] =< aabcdeaa >$ and $MHSAX(p)[T_i] =< ccdeedac >$. We resequence alternately, and then, $MHSAX(p)[T_i] =< (ac)(ac)(bd)(ce)(de)(ed)(aa)(ac) >$.

Fig. 3 shows the process for generating a MHSAX sequence from a time series. First, the MACD histogram of a time series is extracted. The SAX sequences of the time series and its MACD histogram are generated through three SAX generation steps. Finally, the MHSAX sequence is generated by merging these two different types of SAX sequences.

IV. Proposed Method

In this section, we propose a novel method for time series classification utilizing the local alignment technique.

A. Problem Definition

Suppose that there are k classes in a time series data set and CL is given as a set of class labels $CL = \{CL_1, CL_2, \cdots, CL_k\}$. Time series classification is defined as a task that maps a time series T_u, which is unlabeled, to a class label $cl \in CL$. The mapping function is a classifier TC, which is written as $TC : T_u \rightarrow cl,\ cl \in CL$. Given a training data set that consists of pairs: a time series and its class label, the classifier TC learns patterns of the training data set and then outputs the predicted class label of an imputed time series.

B. Local Sequence Alignment

In the distance-based approach for classifying time series. The definition of distance between time series is a critical factor to classify time series. In this study, a time series is represented as a MHSAX sequence. Therefore, the distance between two MHSAX sequences needs to be defined. In our previous work, we measured the distance between two MHSAX sequences using the extended Levenshtein distance, because they are regarded as strings. The Leveshtein distance is known as the edit distance and measures how many operations are required to transform one MHSAX sequence into another MHSAX sequence.

The edit distance measure how dissimilar two MHSAX sequences are. Suppose that there are three MHSAX sequences in Table II. The distances between $MHSAX(p)[T_1]$ and $MHSAX(p)[T_2]$, and between $MHSAX(p)[T_2]$ and $MHSAX(p)[T_3]$ are 3/5 and 2/5, respectively. In this case, our previous method detect T_2 is similar to T_3. T_1 and T_2 have a characteristic pattern $< *(aa)(aa)(aa)* >$, though. The majority of difficult time series classification problems distinguish different class time series by identifying characteristic patterns. The extended Levenshtein distance is unsuitable for these types of time series classification problems.

To consider characteristic patterns, the distance between two MHSAX sequences is calculated using local sequence alignment scores. Local sequence alignment is known as the Smith-Waterman algorithm, and it can extract the locally most similar subsequences. The Smith-Waterman algorithm was originally proposed for determining similar patterns between two sequences of gene sequences. The local sequence alignment technique help to distinguish time series which are partly different. The Smith-Waterman algorithm also has shift invariance. Therefore, when two sequences are similar but differ in phase or when there are regions of the sequences that are aligned and others are not, the Smith-Waterman algorithm can calculate the similarity score between time series.

Suppose that there are two time series T_l and T_k and their MHSAX representations are $MHSAX(p)[T_l]$ and $MHSAX(p)[T_k]$. The score matrix for the local sequence alignment is defined as

$$SM_{i,0} \leftarrow 0 \quad i = 0, \cdots, l_l/2, SM_{0,j} \leftarrow 0 \quad j = 0, \cdots, l_k/2,$$

$$SM_{i,j} \leftarrow max \begin{cases} 0 \\ SM_{i-1,j} - 1 \\ SM_{i,j-1} - 1 \\ SM_{i-1,j-1} + f(a,b,c,d) \end{cases}, \tag{10}$$

where $a \leftarrow MHSAX(p)[T_l]_{2i-1}$, $b \leftarrow MHSAX(p)[T_k]_{2j-1}$, $c \leftarrow MHSAX(p)[T_l]_{2i}$, $d \leftarrow MHSAX(p)[T_k]_{2j}$.

TABLE II
EXAMPLE OF MHSAX

Symbols	Contents
$MHSAX(p)[T_1]$	$< (ac)(aa)(aa)(aa)(ba) >$
$MHSAX(p)[T_2]$	$< (ba)(bb)(aa)(aa)(aa) >$
$MHSAX(p)[T_3]$	$< (ac)(bc)(ab)(bb)(aa) >$

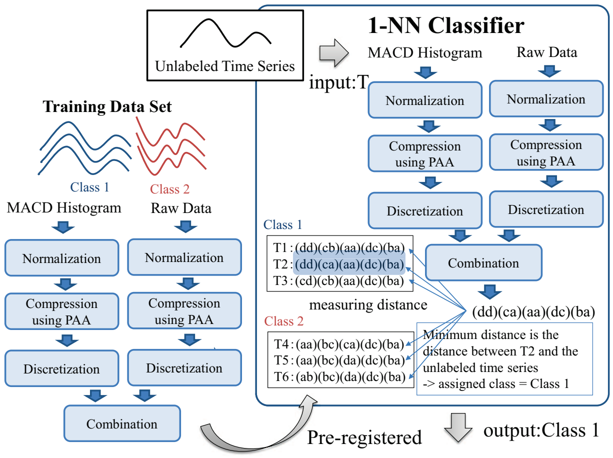

Fig. 4. Processing steps of proposed time series classification method.

$$f(s_1, s_2, s_3, s_4) \leftarrow \begin{cases} 1, & if \;\; s_1 = s_2 \;\&\; s_3 = s_4 \\ e_1, & if \;\; s_1 \neq s_2 \;\&\; s_3 = s_4 \\ e_2, & if \;\; s_1 = s_2 \;\&\; s_3 \neq s_4 \\ -1, & otherwise. \end{cases} \tag{11}$$

The parameter e_1 and e_2 are user given parameters. The default values are $e_1 = 0$ and $e_2 = 0$. This means if either the raw data of a time series or the MACD histogram of the time series is different, score is set to 0.

The distance is defined as follows:

$$dist(MHSAX(p)[T_l], MHSAX(p)[T_k]) = \\ 1 - max(SM)/max(l_l/2, l_k/2), \tag{12}$$

where $max(SM)$ returns maximum value in SM.

Let us consider the above example again. Under the local sequence alignment score, the distance between $MHSAX(p)[T_1]$ and $MHSAX(p)[T_2]$, and between $MHSAX(p)[T_2]$ and $MHSAX(p)[T_3]$ are 3/5 and 1/5, respectively. In this case, T_2 is more similar to T_1 than T_3.

C. Algorithm

The main algorithm of the proposed classifier is based on the 1-NN classifier [25], which is the simplest yet most robust technique for distance-based time series classification. The 1-NN classifier assigns an unlabeled time series to the class label of its closest neighbor. The processing steps for the proposed time series classification are as follows (Fig. 4).

1) Each time series in the training data set is encoded to an MHSAX sequence.
2) An unlabeled time series is encoded to an MHSAX sequence.
3) For all pairs of time series in the training data set and the unlabeled time series, the distance based on the

TABLE III
RESULTS OF ERROR RATES

Name	Number of Classes	Number of Train Data	Number of Test Data	Length	Euclidean 1-NN	Best Warping Window DTW 1-NN	DTW 1-NN	MHSAX	Proposed Method
50Words	50	450	455	270	0.369	0.242	0.31	**0.193**	0.195
Adiac	37	390	391	176	0.389	0.391	0.396	**0.286**	0.299
ArrowHead	3	36	175	251	0.2	0.2	0.297	**0.085**	0.097
Beef	5	30	30	470	0.333	0.333	0.367	**0.133**	**0.133**
BeetleFly	2	20	20	512	0.25	0.3	0.3	0.1	**0.05**
BirdChicken	2	20	20	512	0.45	0.3	0.25	**0**	**0**
Car	4	60	60	577	0.267	0.233	0.267	0.133	**0.116**
CBF	3	30	900	128	0.148	0.004	**0.003**	0.034	0.04
ChlorineConcentration	3	467	3840	166	**0.35**	**0.35**	0.352	0.369	0.371
CinC_ECG_torso	4	40	1380	1639	0.103	**0.07**	0.349	0.114	0.121
Coffee	2	28	28	286	**0**	**0**	**0**	**0**	**0**
Computers	2	250	250	720	0.424	0.38	0.3	0.272	**0.268**
Cricket_X	12	390	390	300	0.423	**0.228**	0.246	0.248	0.233
Cricket_Y	12	390	390	300	0.433	**0.238**	0.256	0.243	0.243
Cricket_Z	12	390	390	300	0.413	0.254	0.246	0.248	**0.238**
DiatomSizeReduction	4	16	306	345	0.065	0.065	**0.033**	0.058	0.055
DistalPhalanxOutlineAgeGroup	3	139	400	80	0.218	0.228	0.208	**0.2025**	0.205
DistalPhalanxOutlineCorrect	2	276	600	80	0.248	0.232	0.232	**0.231**	**0.231**
DistalPhalanxTW	6	139	400	80	0.273	0.272	0.29	**0.245**	0.2625
Earthquakes	2	139	322	512	0.326	0.258	0.258	0.208	**0.204**
ECG	2	100	100	96	0.12	0.12	0.23	0.11	**0.09**
ECG5000	5	500	4500	140	0.075	0.075	0.076	**0.064**	**0.064**
ECGFiveDays	2	23	861	136	0.203	0.203	0.232	0.132	**0.105**
ElectricDevices	7	8926	7711	96	0.45	0.376	0.399	0.324	**0.322**
Face(all)	14	560	1690	131	0.286	**0.192**	**0.192**	0.218	0.211
Face(four)	4	24	88	350	0.216	0.114	0.17	0.034	**0.022**
FacesUCR	14	200	2050	131	0.231	0.088	0.095	0.045	**0.039**
Fish	7	175	175	463	0.217	0.154	0.177	**0.051**	**0.051**
FordA	2	1320	3601	500	0.341	0.341	0.438	0.326	**0.266**
FordB	2	810	3636	500	0.442	0.414	0.406	0.350	**0.297**
Gun-Point	2	50	150	150	0.087	0.087	0.093	**0**	**0**
Ham	2	109	105	431	0.4	0.4	0.533	0.361	**0.333**
HandOutlines	2	370	1000	2709	0.199	0.197	0.202	0.164	**0.162**
Haptics	5	155	308	1092	0.63	0.588	0.623	0.512	**0.509**
Herring	2	64	64	512	0.484	0.469	0.469	**0.343**	**0.343**
InlineSkate	7	100	550	1882	0.658	0.613	0.616	0.552	**0.550**
InsectWingbeatSound	11	220	1980	256	0.438	**0.422**	0.645	0.453	0.472
ItalyPowerDemand	2	67	1029	24	**0.045**	**0.045**	0.05	0.048	0.048
LargeKitchenAppliances	3	375	375	720	0.507	**0.205**	**0.205**	0.32	0.322
Lightning-2	2	60	61	637	0.246	**0.131**	**0.131**	0.163	0.147
Lightning-7	7	70	73	319	0.425	0.288	0.274	0.219	**0.164**
MALLAT	8	55	2345	1024	0.086	0.086	**0.066**	0.108	0.118
Meat	3	60	60	448	0.067	0.067	0.067	**0.033**	**0.033**
MedicalImages	10	381	760	99	0.316	**0.253**	0.263	0.315	0.359
MiddlePhalanxOutlineAgeGroup	3	154	400	80	0.26	0.253	0.25	**0.235**	0.24
MiddlePhalanxOutlineCorrect	2	291	600	80	**0.247**	0.318	0.352	0.265	0.258
MiddlePhalanxTW	6	154	399	80	0.439	0.419	0.416	0.393	**0.388**
MoteStrain	2	20	1252	84	0.121	0.134	0.165	0.114	**0.107**
Non-InvasiveFetalECGThorax1	42	1800	1965	750	**0.171**	0.185	0.209	0.413	0.430
Non-InvasiveFetalECGThorax2	42	1800	1965	750	**0.12**	0.129	0.135	0.290	0.304
OliveOil	4	30	30	570	0.133	0.133	0.167	**0.066**	0.1
OSULeaf	6	200	242	427	0.479	0.388	0.409	0.119	**0.107**
PhalangesOutlinesCorrect	2	1800	858	80	**0.239**	**0.239**	0.272	0.265	0.258
Phoneme	39	214	1896	1024	0.891	0.773	0.772	**0.708**	0.724
Plane	7	105	105	144	0.038	**0**	**0**	**0**	**0**
ProximalPhalanxOutlineAgeGroup	3	400	205	80	0.215	0.215	0.195	0.175	**0.170**
ProximalPhalanxOutlineCorrect	2	600	291	80	0.192	0.21	0.216	0.265	**0.161**
ProximalPhalanxTW	6	205	400	80	0.292	0.263	0.263	0.235	**0.23**
RefrigerationDevices	3	375	375	720	0.605	0.56	0.536	**0.453**	**0.453**
ScreenType	3	375	375	720	0.64	0.589	0.603	**0.538**	**0.538**
ShapeletSim	2	20	180	500	0.461	0.3	0.35	0.038	**0.016**
ShapesAll	60	600	600	512	0.248	0.198	0.232	0.12	**0.101**
SmallKitchenAppliances	3	375	375	720	0.659	**0.328**	0.357	0.389	0.402
SonyAIBORobotSurface	2	20	601	70	0.305	0.305	0.275	**0.244**	0.251
SonyAIBORobotSurfaceII	2	27	953	65	0.141	0.141	0.169	0.157	**0.136**
StarLightCurves	3	1000	8236	1024	0.151	0.095	0.093	**0.057**	0.075
Strawberry	2	370	613	235	0.062	0.062	0.06	**0.044**	**0.044**
SwedishLeaf	15	500	625	128	0.211	0.154	0.208	0.088	**0.0832**
Symbols	6	25	995	398	0.1	0.062	0.05	**0.034**	**0.043**
SyntheticControl	6	300	300	60	0.12	0.017	**0.007**	0.046	0.06
ToeSegmentation1	2	40	228	277	0.32	0.25	0.228	0.166	**0.149**
ToeSegmentation2	2	36	130	343	0.192	0.092	0.162	0.061	**0.053**
Trace	4	100	100	275	0.24	0.01	**0**	**0**	**0**
TwoLeadECG	2	23	1139	82	0.253	0.132	0.096	0.070	**0.034**
TwoPatterns	4	1000	4000	128	0.09	0.002	**0**	0.00025	0.002
uWaveGestureLibrary_X	8	896	3582	315	0.261	0.227	0.273	**0.219**	0.233
uWaveGestureLibrary_Y	8	896	3582	315	0.338	**0.301**	0.366	0.305	0.352
uWaveGestureLibrary_Z	8	896	3582	315	0.35	0.322	0.342	**0.296**	0.31072
UWaveGestureLibraryAll	8	896	3582	945	0.052	**0.034**	0.108	0.036	0.044
Wafer	2	1000	6174	152	0.005	0.005	0.02	**0.003407**	0.0037
Wine	2	57	54	234	0.389	0.389	0.426	**0.259**	**0.259**
WordSynonyms	25	267	638	270	0.382	0.252	0.351	**0.246**	0.258
Worms	5	77	181	900	0.635	0.586	0.536	**0.436**	**0.480**
WormsTwoClass	2	77	181	900	0.414	0.414	0.337	0.276	**0.254**
Yoga	2	300	3000	426	0.17	0.155	0.164	0.095	**0.094**

TABLE IV
SUMMARY OF RESULTS

Metric	EQ 1-NN	BWW DTW 1-NN	DTW 1-NN	MHSAX	Proposed Method
Number of Best Solutions	7	16	11	31	**47**
Average of Error Rates	0.288	0.237	0.256	0.198	**0.195**
Average of Rankings	3.976	2.835	3.505	2.011	**1.882**

local sequence alignment score between the MHSAX sequences is calculated. The class label of the nearest time series is assigned to the unlabeled time series.

V. EXPERIMENTS

In the experiments, we used the UCR Time Series Classification Archive [11], which is the largest available time series classification benchmark data set. Table III shows the details of each data set in the UCR Time Series Classification Archive. This archive includes 85 types of labeled time series data sets with a variety of lengths, class numbers, and data sizes. Each data set is divided into two types of data sets: a training data set, and a test data set. For each data set, we constructed a 1-NN classifier using the training data set and we measured the error rates of classification using the test data set. To evaluate the proposed time series classification method, we conducted three experiments.

A. Experiment 1

In this experiment, the proposed time series classification method was compared with four types of 1-NN classifiers: *EQ 1-NN*, *BWW DTW 1-NN*, *DTW 1-NN*, and *MHSAX*. The *EQ 1-NN* classifier utilizes the Euclidean distance and the *BWW DTW 1-NN*, and *DTW 1-NN* classifiers employ the DTW distance. The *MHSAX* classifier is our previous method, which is a 1-NN classifier with MHSAX using the edit distance as the distance measurement. The MACD's parameters for the proposed time series classification method were $ws_1 = 3,\ ws_2 = 5,$ and $ws_3 = 4$. Moreover, the least error rates were found by varying the following parameters: the PAA window size $w \in \{1, 2, 3, 4, 5\}$ and the cardinality $\alpha \in \{3, 4, \cdots, 14\}$.

Table III shows the classification error rates for each method. The values in the table are described on the UCR Time Series Classification Archive web site. Underlined values indicate the lowest error rate. Table IV summarizes the results. The proposed time series classification method obtains the lowest error rates of the tested methods for 47 out of 85 data sets. In addition, its average error rates and average rankings are the smallest. Moreover, MHSAX-based 1-NN classifiers is superior to DTW-based classifiers. Wang et al. [26] reported the classification performance of 9 distance measures and several variants thereof. They found DTW performs exceptionally well in comparison to other distance measurements. This indicates MHSAX-based 1-NN classifiers have the best performance.

B. Experiment 2

We compared the proposed time series classification method with two other SAX-based classifiers BOW [27] and SAX-VSM [18]. Table V compares their performance on 42 data sets from the UCR Time Series Classification Archive. The proposed time series classification method achieves good performance compared with BOW and SAX-VSM. We also compared our proposed time series classification method with shapeDTW [23]. In [23], 84 data sets (not including the *StarLightCurves* data sets) in the UCR Time Series Classification Archive were used. Table VI compares the performance of the methods. Both our previous method and the proposed time series classification method are superior to shapeDTW.

TABLE V
COMPARISON WITH SAX-BASED METHODS

	BOW	SAX-VSM	Proposed Method
50Words	0.316	0.374	**0.195**
Adiac	0.325	0.417	**0.299**
Beef	0.267	0.233	**0.133**
CBF	0.048	**0.01**	0.04
ChlorineConcentration	0.405	**0.341**	0.371
CinC_ECG_torso	0.164	0.344	**0.121**
Coffee	0.036	**0**	**0**
Cricket_X	0.305	0.308	**0.233**
Cricket_Y	0.313	0.318	**0.243**
Cricket_Z	0.295	0.297	**0.238**
DiatomSizeReduction	0.111	0.121	**0.055**
ECG	0.11	0.14	**0.09**
ECGFiveDays	0.164	**0.001**	0.105
Face(all)	0.238	0.245	**0.211**
Face(four)	0.102	0.114	**0.022**
FacesUCR	0.137	0.109	**0.039**
Fish	0.029	**0.017**	0.051
Gun-Point	0.407	0.013	**0**
Haptics	0.63	0.584	**0.509**
InlineSkate	0.629	0.593	**0.55**
ItalyPowerDemand	**0.044**	0.089	0.048
Lightning-2	0.328	0.213	**0.147**
Lightning-7	0.37	0.397	**0.164**
MALLAT	**0.098**	0.199	0.118
MedicalImages	0.401	0.516	**0.359**
MoteStrain	0.177	0.125	**0.107**
OliveOil	0.233	0.133	**0.1**
OSULeaf	0.153	0.165	**0.107**
SonyAIBORobotSurface	0.409	0.306	**0.251**
SonyAIBORobotSurfaceII	0.154	**0.126**	0.136
SwedishLeaf	0.125	0.278	**0.083**
Symbols	0.088	0.109	**0.043**
SyntheticControl	**0.017**	**0.017**	0.06
Trace	**0**	**0**	**0**
TwoPatterns	0.01	**0.004**	0.034
TwoLeadECG	0.248	0.014	**0.002**
uWaveGestureLibrary_X	0.242	0.323	**0.233**
uWaveGestureLibrary_Y	**0.352**	0.364	**0.352**
uWaveGestureLibrary_Z	0.325	0.356	**0.31**
Wafer	0.01	**0.001**	0.003
WordSynonyms	0.371	0.44	**0.258**
Yoga	0.145	0.151	**0.094**

TABLE VI
COMPARISON WITH SHAPEDTW

Metric	EQ 1-NN	BWW DTW 1-NN	DTW 1-NN	shapeDTW 1-NN	MHSAX	Proposed Method
Number of Best Solutions	1	1	2	31	31	**37**
Average of Error Rates	0.288	0.237	0.256	0.214	0.198	**0.195**
Average of Rankings	4.607	3.392	4.047	3.202	2.547	**2.380**

C. Experiment 3

In Eqn. (11), there two user given parameters e_1 and e_2. In this experiment, we discuss performance regarding to parameters e_1 and e_2. In Experiment 1 and Experiment 2, these parameters are set to the default value, which is zero. When we use the default value, this means that local

TABLE VII
PARAMETER STUDIES (A)

Name	Default	Sensitive
50Words	**0.195604**	0.221978
Adiac	**0.299233**	0.304348
ArrowHead	0.097143	**0.08**
Beef	0.133333	**0.066667**
BeetleFly	**0.05**	0.1
BirdChicken	**0**	0.1
Car	**0.116667**	**0.166667**
CBF	**0.04**	0.226667
ChlorineConcentration	0.371094	**0.358594**
CinC_ECG_torso	0.121014	**0**
Coffee	**0**	**0**
Computers	**0.268**	0.292
Cricket_X	**0.233333**	0.284615
Cricket_Y	**0.24359**	0.297436
Cricket_Z	**0.238462**	0.317949
DiatomSizeReduction	**0.055556**	0.062092
DistalPhalanxOutlineAgeGroup	**0.205**	0.21
DistalPhalanxOutlineCorrect	**0.231667**	0.245
DistalPhalanxTW	**0.2625**	0.265
Earthquakes	0.204969	**0.189441**
ECG	0.09	**0.07**
ECG5000	0.064222	**0.060222**
ECGFiveDays	0.105691	**0.00813**
ElectricDevices	**0.322786**	0.371807
Face(all)	**0.211243**	0.230769
Face(four)	0.022727	**0.011364**
FacesUCR	**0.039024**	0.054146
Fish	**0.051429**	**0.051429**
FordA	0.26687	**0.18717**
FordB	0.297305	**0.244774**

TABLE VIII
PARAMETER STUDIES (B)

Name	Default	Sensitive
Gun-Point	**0**	**0**
Ham	**0.333333**	0.361905
HandOutlines	**0.162**	0.171
Haptics	**0.50974**	0.594156
Herring	**0.34375**	0.375
InlineSkate	**0.550909**	0.650909
InsectWingbeatSound	0.472727	**0.452525**
ItalyPowerDemand	0.048591	**0.041788**
LargeKitchenAppliances	**0.322667**	0.370667
Lightning-2	**0.147541**	0.245902
Lightning-7	**0.164384**	0.315068
MALLAT	**0.11855**	0.123241
Meat	**0.033333**	0.05
MedicalImages	**0.359211**	0.389474
MiddlePhalanxOutlineAgeGroup	0.24	**0.23**
MiddlePhalanxOutlineCorrect	**0.258333**	0.26
MiddlePhalanxTW	**0.388471**	0.390977
MoteStrain	**0.107827**	0.111022
Non-InvasiveFetalECGThorax1	**0.430534**	0.609669
Non-InvasiveFetalECGThorax2	**0.304326**	0.397964
OliveOil	**0.1**	**0.1**
OSULeaf	**0.107438**	0.136364
PhalangesOutlinesCorrect	**0.258741**	**0.258741**
Phoneme	**0.724156**	0.768987
Plane	**0**	**0**
ProximalPhalanxOutlineAgeGroup	0.170732	**0.156098**
ProximalPhalanxOutlineCorrect	0.161512	**0.14433**
ProximalPhalanxTW	**0.23**	0.2475
RefrigerationDevices	**0.453333**	0.477333
ScreenType	0.538667	**0.530667**

TABLE IX
PARAMETER STUDIES (C)

Name	Default	Sensitive
ShapeletSim	**0.016667**	0.166667
ShapesAll	**0.101667**	0.165
SmallKitchenAppliances	**0.402667**	0.456
SonyAIBORobotSurface	**0.251248**	0.234609
SonyAIBORobotSurfaceII	0.136411	**0.129066**
StarLightCurves	**0.075765**	**0.075765**
Strawberry	0.044046	**0.035889**
SwedishLeaf	**0.0832**	0.1056
Symbols	**0.043216**	0.063317
SyntheticControl	**0.06**	0.203333
ToeSegmentation1	0.149123	**0.122807**
ToeSegmentation2	**0.053846**	0.092308
Trace	**0**	0.03
TwoLeadECG	0.034241	**0.031607**
TwoPatterns	**0.002**	0.02075
uWaveGestureLibrary_X	**0.233948**	0.295366
uWaveGestureLibrary_Y	**0.352038**	**0.352038**
uWaveGestureLibrary_Z	**0.31072**	0.363763
UWaveGestureLibraryAll	**0.044389**	0.064210
Wafer	0.003731	**0.000649**
Wine	**0.259259**	**0.259259**
WordSynonyms	**0.258621**	0.302508
Worms	**0.480663**	0.524862
WormsTwoClass	**0.254144**	0.309392
Yoga	0.094333	**0.092**

alignment scores are one if only both of raw value and its MACD value are same. In time series classification, there are problems of distinguishing time series where the value fluctuates drastically. Even if, the difference is large or small, local alignment scores are zero. To consider this, the followings are more sensitive parameter settings.

$$e_1 = 0.5 \times (1 - \frac{|s_1 - s_2|}{\alpha}) \quad (13)$$

$$e_2 = 0.5 \times (1 - \frac{|s_3 - s_4|}{\alpha}) \quad (14)$$

Table VII, Table VIII, and Table IX show that the comparisons of the default parameter setting and the sensitive parameter setting. The results of the data sets of 23 are improved when the sensitive parameter setting is used. The results indicate that the sensitive parameter setting helps to improve classification performance for some data sets. Time series classification can be optimized for each application. The sensitive parameter setting is effective to improve classification performance.

VI. CONCLUSION

In the era of big data, time series classification is one of the best-known key challenges because of its many fields of application and difficulty. In this paper, we propose a novel method for time series classification using a hybrid SAX-based symbolic representation, which is called a moving average convergence divergence (MACD)-histogram-based SAX (MHSAX) proposed in our previous work. The proposed time series classification method includes the MHSAX and a nearest neighbor (1-NN) classifier utilizing the local sequence alignment technique. To evaluate the proposed time series classification method, we implemented it and conducted experiments using all 85 data sets in the UCR Time Series Classification Archive. The experimental results show that the proposed time series classification method outperforms our previous method. Its classification ability is superior to other state-of-the-art methods. Moreover, the sensitive parameter setting can improve classification performance.

In our future work, we are developing a new high-level symbolic representation for time series based on MHSAX. MHSAX is good representation, however, it is in the distance-based approach. Computation time for classifying time series is more heavy that other approaches. MHSAX needs its compressed representation to reduce computation time. MACD histogram shows the acceleration of time series; however, the type of movement of time series is not considered. We are going to develop more sensitive representation to differences in movement.

ACKNOWLEDGMENT

This work was supported by the MIC/SCOPE #162308002, JSPS KAKENHI Grant Number JP26330139, and Hiroshima City University Grant for Special Academic Research.

REFERENCES

[1] M. Last, A. Kandel, and H. Bunke, *Data Mining in Time Series Databases*, ser. Series in machine perception and artificial intelligence. World Scientific, 2004.

[2] P. Esling and C. Agon, "Time-series data mining," *ACM Comput. Surv.*, vol. 45, no. 1, pp. 12:1–12:34, Dec. 2012.

[3] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: Experimental comparison of representations and distance measures," *Proc. VLDB Endow.*, vol. 1, no. 2, pp. 1542–1552, Aug. 2008.

[4] S. Basu and M. Meckesheimer, "Automatic outlier detection for time series: An application to sensor data," *Knowl. Inf. Syst.*, vol. 11, no. 2, pp. 137–154, Feb. 2007.

[5] M. Shokoohi-Yekta, Y. Chen, B. Campana, B. Hu, J. Zakaria, and E. Keogh, "Discovery of meaningful rules in time series," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1085–1094.

[6] J. Lin, E. Keogh, S. Lonardi, J. P. Lankford, and D. M. Nystrom, "Visually mining and monitoring massive time series," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 460–469.
[7] P. Geurts, *Principles of Data Mining and Knowledge Discovery: 5th European Conference, PKDD 2001, Freiburg, Germany, September 3–5, 2001 Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, ch. Pattern Extraction for Time Series Classification, pp. 115–127.
[8] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: A novel symbolic representation of time series," *Data Min. Knowl. Discov.*, vol. 15, no. 2, pp. 107–144, Oct. 2007.
[9] K. Tamura and T. Ichimura, "Time series classification using macd-histogram-based sax and its performance evaluation," in *Proceedings of 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 2419–2424.
[10] J. J. Murphy and J. J. Murphy, *Technical analysis of the financial markets*. Fishkill, N.Y.: New York Institute of Finance, 1999.
[11] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: A survey and empirical demonstration," *Data Min. Knowl. Discov.*, vol. 7, no. 4, pp. 349–371, Oct. 2003.
[12] K. Tamura and T. Ichimura, "Mhsax-based time series classification using local sequence alignment technique," *Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2017*, pp. 286–291, 2017.
[13] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences." *Journal of molecular biology*, vol. 147, no. 1, pp. 195–197, Mar. 1981.
[14] Z. Xing, J. Pei, and E. Keogh, "A brief survey on sequence classification," *SIGKDD Explor. Newsl.*, vol. 12, no. 1, pp. 40–48, Nov. 2010.
[15] E. J. Keogh and M. J. Pazzani, "Scaling up dynamic time warping for datamining applications," in *Proceedings of KDD '00*, 2000, pp. 285–289.
[16] L. Ye and E. Keogh, "Time series shapelets: A new primitive for data mining," in *Proceedings of KDD '09*, 2009, pp. 947–956.
[17] J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme, "Learning time-series shapelets," in *Proceedings of KDD '14*, 2014, pp. 392–401.
[18] P. Senin and S. Malinchik, "Sax-vsm: Interpretable time series classification using sax and vector space model," in *Proceedings of 2013 IEEE 13th International Conference on Data Mining (ICDM)*, Dec 2013, pp. 1175–1180.
[19] D. F. Silva, V. M. A. de Souza, and G. E. Batista, "Time series classification using compression distance of recurrence plots," in *Proceedings of the IEEE International Conference on Data Mining (ICDM 2013)*, 2013, p. 687–696.
[20] E. F. Gomes, A. M. Jorge, and P. J. Azevedo, "Classifying heart sounds using sax motifs, random forests and text mining techniques," in *Proceedings of IDEAS '14*, 2014, pp. 334–337.
[21] U. Kamath, J. Lin, and K. De Jong, "Sax-efg: An evolutionary feature generation framework for time series classification," in *Proceedings of GECCO '14*, 2014, pp. 533–540.
[22] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," in *Proceedings of IJCAI'15*. AAAI Press, 2015, pp. 3939–3945.
[23] J. Zhao and L. Itti, "shapedtw: shape dynamic time warping," *CoRR*, vol. abs/1606.01601, 2016. [Online]. Available: http://arxiv.org/abs/1606.01601
[24] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, "Dimensionality reduction for fast similarity search in large time series databases," *Knowledge and Information Systems*, vol. 3, no. 3, pp. 263–286.
[25] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theor.*, vol. 13, no. 1, pp. 21–27, Sep. 2006.
[26] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, Mar 2013.
[27] J. Lin, R. Khade, and Y. Li, "Rotation-invariant similarity in time series using bag-of-patterns representation," *J. Intell. Inf. Syst.*, vol. 39, no. 2, pp. 287–315, Oct. 2012.