Int. J. Advance Soft Compu. Appl, Vol. 7, No. 3, November 2015
ISSN 2074-8523

A Time-weighted Average-based PAA Representation for Time Series Symbolization

Yahyia Benyahmed[1], Azuraliza Abu Bakar[1], Abdul Razak Hamdan[1], and Sharifah Mastura Syed Abdullah[2]

[1]Center for Artificial Intelligence Technology (CAIT),
Faculty of Information Science and Technology,
University Kebangsaan Malaysia, 43600 Bangi, Selangor Darul Ehsan, Malaysia.
e-mail: ybyconiny@gmail.com, azuraliza@ukm.edu.my, arh@ukm.edu.my
[2]Institute of Climate Change, Universiti Kebangsaan Malaysia,
43600 Bangi,Selangor Darul Ehsan, Malaysia.
e-mail: mastura@ukm.my

Abstract

The operation of time series analysis to effectively manage the large amounts of data with high dimensional became an important research problem. Choose effective and scalable algorithms for appropriate representation of data is another challenge. A lot of high-level representations of the time series have been proposed for data extraction, such as spectral transfers, wavelets, piecewise polynomial, symbolic models, etc. One of the methods is Piecewise Aggregate Approximation (PAA) which minimizes dimensionality by the mean values of equal-sized frames, but this focus on mean value takes into consideration only the central tendency and not the dispersion present in each segment, which may lead to some important patterns being missed in some time series data sets. We propose method based on Time-Weighted Average for Symbolic Aggregate approximation method (TWA_SAX) compare its performance with some current methods. TWA_SAX is enables raw data to be specifically compared to the minimized representation and, at the same time, ensures reduced limits to Euclidean distance. It can be utilized to generate quicker, more precise algorithms for similarity searches, which improves the preciseness of time series representation through enabling better tightness of the lower bound.

Keywords: *Time Series, Dimensionality reduction, Piecewise aggregate approximation, Symbolic, Discretize.*

1 Introduction

Time series data are produced and processed within a wide range of application domains in various fields. A time series is a sequence of real values that are 'discovered' and that are generally stamped with time [1, 2]. The main tasks relevant time series include; clustering [3], time series prediction [4-6], and time series segmentation. [7, 8]. One of the significant problems in time series analysis is reduction of data, since time series conclusions are made in sequence, the association between sequential data items in a time series enables data analysts to minimize the dimensions of the data, without significant loss of information [9]. Data reduction handle enormous time series data to present a summation of the data, which proposed for the efficient representation of time series data to minimize a massive amount of data into a feasible brief data structure and, at the same time, mostly conserve the attribute of the data. It is the foundation for rapid analysis and the finding of pertinent information from a massive volume of data [10].

The analysis of such intriguing data might reveal patterns that merit further attention. There are several methods have been proposed that focus on overcoming time series data representation challenges. One more method used in several effective similarity researches on time series rest on is Piecewise Aggregate Approximation (PAA) [11]. The main concept of PAA is to partition every sequence into k segments of similar size and to utilize the average value of every section to represent the latter, which focus on reducing the dimensions, and lower bounding function. The aggregate method can take many formulas including sets, sum, average, max, min, variance and Etc., to measure the similarity between such aggregated data items and a range of dimension. The majority of the representation algorithms need the parameters of alphabet size and word size as inputs. However, it may be very difficult to know the best values for alphabet and word size in advance in real-world applications. The Symbolic Aggregate Approximation (SAX) approach [12] aims to overcome the problems created by time series representations. However, the SAX algorithm using fix parameters of word and alphabet size, the word size depend on domains and 10 alphabet size. The algorithm does not clearly show how to define word and alphabet size using a time series data set. There are other various algorithms [2, [13] that attempt to solve the problem of defining the optimal (minimum) word and alphabet size, but these methods do not result in minimal information loss. Hence, the Harmony search algorithm HSAX [14] used as the initial stage with proposed method to find the optimum word and alphabet size, while reducing loss of information.

The remainder of this paper is organized as follows: Section 2 discusses the related work on comparative techniques. Sections 3 and 4 introduce the concepts of the methods used in this study, namely, Time-weighted Average (TWA) and SAX. Section 5 presents the experimental design and the results and section 6 contains a discussion of our findings. Finally, section 7 concludes our study.

2 Literature Review

One of the most significant and successful methods has been used in several research studies on time series representation is Piecewise Aggregate Approximation (PAA) [11]. The main concept of PAA is to partition every sequence into k segments of similar size and to utilize the average value of every segment to represent the latter. In [15] an Adaptive Principal Component Analysis (APCA) is proposed, where the pieces may be of diverse lengths and which provides a more efficient compression as opposed to PAA. A novel time series representation, Dispersion-based PAA (DPAA) And Piecewise Linear Aggregate Approximation (PLAA) [16]. In other work [17], two extended versions of the PAA method are proposed: Linear Statistical Feature-based PAA (LSF_PAA) and Square Root Statistical Feature-based PAA (SSF_PAA). Each of these methods employs a combination of two statistical features, the mean and the variance. Recently, [18] have an approval PAA based on the major shape features of time series. Another work presented by [19] the approach design a measure to calculate the distance of trends using the starting points and ending points of each segments which give better classification performance than PAA.

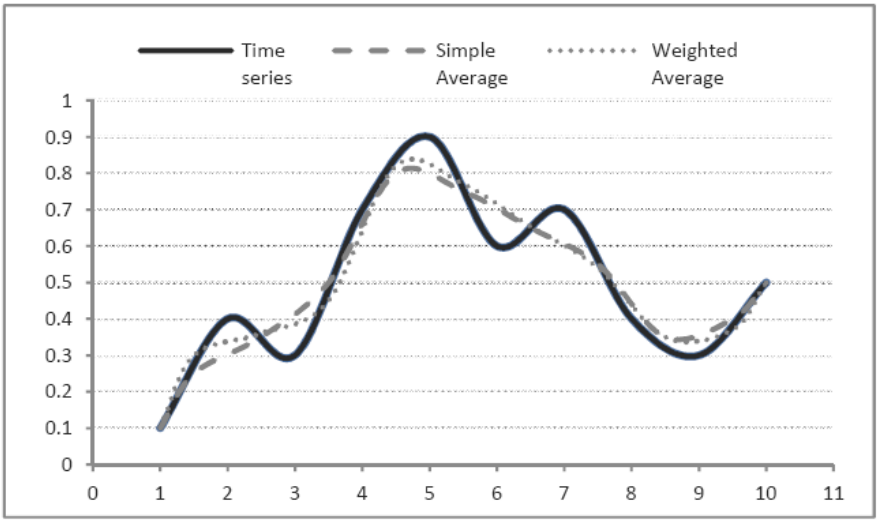

Fig. 1 Process of data loss as a result of a focus on the centre of the time series trend using SAX algorithm compared with TWA_SAX algorithm.

The SAX method [12] creates a symbolic representation of time series based on PAA. It allows distance measurements to be less than those delineated in the symbolic space so that a common representation of time series can be converted into a symbolic representation. The SAX suffers some defects [16]. First, PAA reduces dimensionality by the mean values of equal-sized frames, but this focus on mean value takes into consideration only the central tendency and not the dispersion present in each segment, which may lead to some important patterns being missed

in some time series data sets because some of the Euclidean distances are quite large [20]. Therefore, the SAX representation method may not produce a good tightness of the lower bound. Fig. 1 illustrates how SAX focuses just on the centre of the time series. In addition, the algorithm requires the alphabet and the size of the input word. This is the main drawback of SAX because it is not clear how to define these inputs for certain time series data sets.

There has been surprisingly little work to extend the SAX representation itself. It simply works very well for most problems. According to [21] have proposed an extended method based on SAX, known as Extended SAX (ESAX), depends on the PAA representation for lowering of dimensionality, by the mean values of equally scaled frames including two exclusive novel points, max and min points of each segment, to completely represent time series data. Another method is indexable Symbolic Aggregate approXimation (*i*SAX) presented by [22]. The main structure of *i*SAX index that is, a set of time series represented by an *i*SAX word can be split into two mutually exclusive subsets by increasing the cardinality along one or more dimensions. Recently, Trend-based symbolic approximation (TSX) was introduced by a novel symbolic representation, which may reflect not only the segment average value feature, but also trend feature and information with good resolution. Combining Trend-based and Piecewise Linear Approximation with value-based approximations (SAX) for time series classification can be seen in [16, 23, 24].

To address these issues, we propose a novel method based on PAA called Time Weighted Average for Symbolic Aggregate Approximation (TWA_SAX). One distinctive benefit of TWA_SAX can enable raw data to be specifically compared to the minimized representation and, at the same time, ensures reduced limits to Euclidean distance. Fig. 1 shows the different between SAX using simple average and TWA_SAX using weighted average, the Euclidean distance with original time series have been calculated which given the smallest at TWA_SAX compared with original SAX. Therefore, the TWA_SAX representation produces a good tightness of the lower bound. Hence, it can be utilized to generate quicker, more precise algorithms for similarity searches.

3 Preliminaries

3.1 Time-Weighted Average approach (TWA)

We propose the use of a TWA to reduce the high dimensionality of time series data. The TWA method is a smoothing method that uses a weighted moving average of past time series values as a prediction for those instances when a time series contains a large amount of noise, when it may be hard to see any trends. The moving average smoother is applied to split the value of the time series from the noise.

In statistics and time series processing, generally a time series smoother, such as a weighted average, is employed to ensure that it estimates a function, which tries to capture essential patterns of behaviour in the data, particularly in non-stationary

time series. In time series analysis, it is essential to reveal the significance of the data, and the TWA, with its higher coefficient for data, can accomplish the preferred function, and it triggers the portions of the time series so that they are precisely recognized. It is noteworthy that a moving average can also minimize noise, which is a substantial part of real data.

According to [25], the weighted moving average filter considers each data point in the data window to be similarly essential, while computing the average value. However, in active and authentic systems, the most recent values usually better reveal the condition of the process. Therefore, a filter that focuses more on the most current data could be more practical, particularly in time series segmentation. The TWA has been found to have superior effectiveness in segmenting time series. The weighted average employs the average of the most recent weighted data values in the time series to compress data to facilitate computation in terms of space and time. Statistically, a weighted average of order *wj* is as follows:

$$\overline{C}_i = \sum_{i=1}^{N} \frac{\sum_{j=n-N+i}^{i} w_j c_j}{\sum_{j=n-N+i}^{i} w_j}, \tag{1}$$

Where, *Ci* is a new value of the time series in period *i*, *cj* is real values of the time series, The weight *wj* is a time for each value of the time series, *n* is time series length, *N* is reduced value (number of segments). We estimate the time series parameter of the model at time *j* as the average of the last *wj* observations, if *wj* is the moving average period. The moving term is employed because every time a new observation is available for the time series, it replenishes the earliest observation in the formula, and a new average is calculated. Consequently, the average will transform as novel observations become accessible. In the weighted average method, every time series observation in the average computation obtains the same weight (time). However, in a weighted average, the computation includes choosing a variety of times time for every data value and then calculating a TWA of the most current *wj* values (time) as the prediction. In many cases, the most recent observation takes the most time, and the time decreases in the case of older data values.

3.2 Symbolic Aggregate approXimation (SAX)

The SAX method [12] creates a symbolic representation of time series. It allows distance measurements to be less than those delineated in the symbolic space so that a common representation of time series (a sequence of data points interpolated by a line) can be converted into a symbolic representation. (1) The SAX method implements discretization in two phases (see Fig. 2). Initially, a time series is split into equal-sized segments *s*, the values of each section are estimated and then

substituted by a single coordinate. The accumulation of these *s* coordinates forms the standard method of the strings (TWA) representation of time series. Next, to change the TWA coordinates to representations, the breakpoints, which split the distribution space in an area into equal portions, are measured, where a is the size of alphabet described by the user. Basically, the breakpoints are identified to ensure that a segment in any of the areas is roughly the same. If the symbols of the representation are not equally similar, some channels are more probable than others (therefore, probabilistic tendency in the process is included).

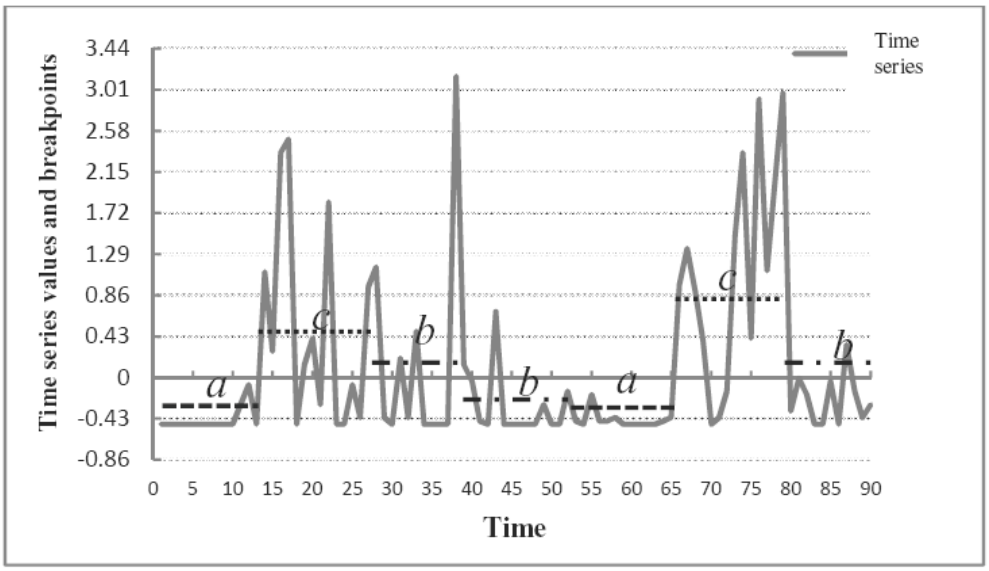

Fig. 2 Time series converted with SAX method, adopted from [12].

After the transformation of a database of a TWA series, another transformation is performed to get a discrete representation. There should be discretization technique to symbol the time series data with equal-probability occur [26]. This is easily accomplished because normalized time series have a Gaussian distribution. To show this, we derive 90 sequences with a length of 7 for various time series and draw a regular possibility plot of the data, as illustrated in Fig. 2.

Table 1: Lookup table containing breakpoints, adopted from [12]

$\beta_i \backslash a$	3	4	5	6	7	8	9	10
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
β_4			0.84	0.43	0.18	0	-0.14	-0.25
β_5				0.97	0.57	0.32	0.14	0
β_6					1.07	0.67	0.43	0.25
β_7						1.15	0.76	0.52
β_8							1.22	0.84
β_9								1.28

Given the fact that the time series of Gaussian distribution simply, to identify the breakpoints that occur in areas of equal size Gauss curve. The breakpoints are a sorted list of numbers $B = \beta 1,\ldots, \beta a\text{-}1$ so that the space under a $N(0,1)$ Gaussian curve from βi to $\beta i+1 = 1/a$ ($\beta 0$ and βa are defined as $-\infty$ and ∞, respectively). These breakpoints might be identified by searching in a statistical table. Table 1 provides an example of the breakpoints for the values of *a,* from 3 to 10.

The modified TWA time series data are symbolized making use of discrete representation with the SAX algorithm. Because normalized time series have a Gaussian distribution, we can determine the "breakpoints" that will produce c equal-sized areas under the Gaussian distribution curve. The time series data are then dispersed into a discrete symbolic series based on these breakpoints. All TWA coordinates that are less than the most basic breakpoint are modified to the symbol a, while all coordinates that are higher than or equal to the smallest breakpoint but less than the second smallest breakpoint are converted to the symbol b, Fig. 2 shows how time series is categorized by initially acquiring a TWA approximation and then using pre-specified breakpoints to map the TWA coordinates into SAX symbols. In the example given above, with n = 90, w = 7 and a = 3, the time series is associated with the word '*acbbacb*'.

4 Fifth Generation (5G) Technology

TWA is replaced with PAA and compared with SAX to handle the problem of time series data reduction. The problem of dimensionality reduction can be defined as follows. Assume we have a time series query , , ..., 1 2 *n C c c c i* = and let *N* be the dimensionality of the transformed space we wish to index (1 £ *N* £ *n*). For ease, we have presumed that *N* is a factor of n. A time series *C* of length *n* is represented in *N* space by a vector , , ..., 1 2 *N C c c c i* = , where *Ci* is computed by Eq. 1. The weighted method is combined with SAX to improve the minimization of the data from n dimensions to N dimensions depending on the TWA. We have called this method TWA_SAX. The main idea of the proposed integration of TWA is to split every series into S equally-sized segments and to utilize the weighted moving average value of every segment to represent the alphabet. This representation

minimizes the data from n to *N* dimensions by splitting the time series into *N* arbitrary length segments. The weighted average value of the data belonging to a segment is computed and a vector of these values turns into a data-reduced representation. The weighted average method is combined with SAX to improve the minimization of the data from n dimensions to *N* dimensions depending on the TWA. We have called this method TWA_SAX. The main idea of the proposed integration of TWA is to split every series into S equally-sized segments and to utilize the weighted average value of every segment to represent the alphabet. This representation minimizes the data from n to *N* dimensions by splitting the time series into *N* arbitrary length segments. The weighted average value of the data belonging to a segment is computed and a vector of these values turns into a data-reduced representation. Eq. 1 illustrates how a weighted average is used to reduce the length of the time series with weighting of each segment over the time (*t*). The proposed method using linear interpolation approach [27, 28] to find the cut-offs of each segments. In General, linear interpolation [27, 28] takes two data points, say *x1* and *xn+1* , and the interpolant is given by Eq. 2.

$$y_{i+1} = \sum_{i=1}^{N-1} x_i + \frac{(x_1 + x_{n+1})}{N} \tag{2}$$

Where *yi* = 1 represents the lower bound of the first segment, *yi+1* is cut-off of the segment *i, xi* is the lower and upper bounds for cut-offs *xi = yi,yi-1… yN-i-1* and *i = 1, 2,… N-1,* when *i* = 1 then *xi = y1* is representing the first cut-off *y1* for first segment of time series, while the cut-off *y2, i* = 2 and *xi+1 = yi*-1 that represented by the second segment, when *i* = 3 then *xi+2 = yi-2* which represented by the cut-off of third segment and so on.

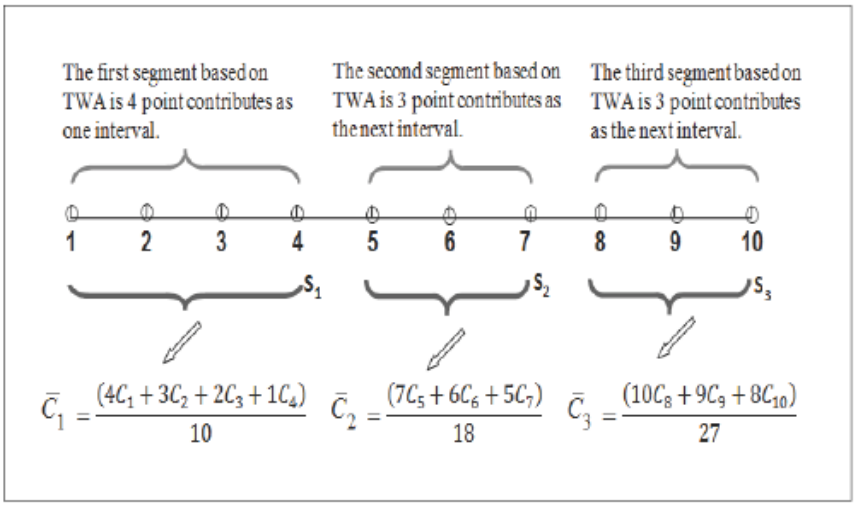

Fig. 3 Time-weighted Average representation for 10 points of time series data

Suppose that there are 10 data points *n* = 10 that need to be reduced to *N* = 3. In order to get 3 segments, the linear interpolation equation Eq. 2 can be applied for

all data points are located between the known first and last points represented by 1 and 10. It results in 3 equal-width periods of time due to get the interval terms based on $N = 3$ (see Fig. 3). In this case, the bounds are $y1 = 1$, $y2 = 4.33$, y3 = 7.66 and $y4 = n+1$. Table 2 shows the obtained segments, their lower and upper bounds, and segment data points in which the numbers of points of the segments are 4, 3 and 3, respectively. Eq. 1 illustrates how a weighted average is used to reduce the length of time series with weighting of each segment over the time.

Table 2: Segmentation bounds obtained from linear interpolation method

Segment	Lower bound	Upper bound	Segment data points
S1	1	4.33	[1,2,3,4]
S2	4.33	7.66	[5,6,7]
S3	7.66	$n+1$	[8,9,10]

The TWA algorithm is employed to minimize the time series length from length n to length *N*, The output of the TWA given a time series data reduction, *Ci* represents segmentations of time series, in which every segment is calculated by weighted (time) average to get a time series reduction based on time. After that, SAX algorithm applied to generate symbols, which can be easily accomplished because normalized time series have a Gaussian distribution [27]. As soon as the breakpoints have been acquired, we can implement a discretization of time series as follows: First, all the TWA coordinates that are equal to the lowest breakpoint are associated with the symbol 'a', while those that are higher than or equal to the lowest breakpoint and less than the second lowest breakpoint are associated with the symbol 'b', etc. A time series is discretized by first acquiring the TWA approximation and then predefined breakpoints are used to convert the TWA coordinates to SAX representations.

5 Experiments

Experiments were conducted in which the TWA was integrated into the SAX time series data representation to generate sufficient reduction of time series data. The performance of TWA_SAX was evaluated by error rate on 15 benchmark time series data sets. The TWA_SAX algorithm was compared to the HSAX [14], GENEBLA [2] and original SAX algorithms [12].

5.1 Time-Weighted Average approach (TWA)

The performance of the TWA_SAX method was validated using 15 data sets that are available on the time series classification and clustering website [18]. These data sets have been used by [12] to test the SAX method. The TWA and SAX were

used in the experiment to evaluate an efficient number of intervals for the alphabet size (a) and the word size (N) is given in [29-31]. The TWA_SAX algorithm is run with these same parameters to enable comparison of the TWA_SAX with the initial Eumonn's 1-NN (1-NN EU), SAX [12], HSAX and GENEBLA [29-31].

Using 1-NN to assess the efficiency of the proposed TWA_SAX method and used absolute distance to determine classification performance. Raw data (continuous time series) was used in our assessment of classification performance; a low error rate signifies effective classification. It is noteworthy that TWA_SAX utilizes its own similarity measure [12]. These experiments were conducted to determine whether TWA can produce the same or improve on the error rate compared to the previous approaches to the time series problem.

5.2 Results and Discussions

The experimental results show that the proposed TWA_SAX algorithm performs better in terms of the error rates returned for various data sets when compared with the 1-NN EU technique. Also, the TWA_SAX yielded exceedingly low error rates when compared to the original SAX algorithm in most data sets that have larger alphabet and word sizes.

Table 3 shows that compared to SAX, GENEBLA (denoted as GA) and HSAX, TWA_SAX achieved lower error rates in 10 data sets using alphabet and word size, with the advantage that the TWA_SAX uses a high number of alphabet and word sizes from [14], whereas in the SAX, a high error rate was back with a fixed alphabet and word size. In few data sets, TWA_SAX offers better interpretation as compared SAX algorithm. Table 3 shows TWA_SAX method achieved better error rates on most of the data sets when compared to HSAX, GA and SAX. If we compare TWA_SAX method with GA, SAX and HSAX methods, we get a lower error rate on 8 data sets (CBF, Coffee, Control chart, FaceAll, Gun Point, Olive Oil, Swedish Leaf and Trace data sets), whereas GA gets a perfect error rate in 2 data sets (Adiac and Wafer) comparing with other methods. In comparison with SAX and TWA_SAX, SAX gets also the lower error rate in 2 data sets (FaceFour and Yoga data sets). HSAX ties with other methods, where it gets a minimum error rate in 4 data sets (50word, Beef, Two pattern and Yoga, data sets). The lowest error rate based on average has recorded in TWA_SAX method with 0.284, which enhances the preciseness of time series representation through enabling better tightness of the lower bound.

Table 3: TWA_SAX, HSAX, GA and original SAX based on error rate

No	Data set (No. Classes)	TS length (n)		Error rate			
		Original (n)	HSAX (n)	TWA_ SAX	HSAX	SAX	GA
1	50word (50)	270	140	0.336	**0.300**	0.341	0.440
2	Adiac (37)	176	102	0.621	0.502	0.890	**0.490**
3	Beef (5)	470	236	0.533	**0.400**	0.567	0.500
4	CBF (3)	128	66	**0.070**	0.102	0.104	0.100
5	Coffee (2)	286	166	**0.321**	0.398	0.464	0.420
6	ControlChart (6)	60	45	**0.007**	0.100	0.467	0.310
7	FaceAll (14)	131	55	**0.246**	0.299	0.330	0.330
8	FaceFour (4)	350	139	0.352	0.185	**0.170**	0.210
9	Gun Point (2)	150	86	**0.120**	0.290	0.180	0.200
10	Olive Oil (4)	570	189	**0.333**	0.533	0.833	0.380
11	Swedish Leaf (15)	128	62	**0.389**	0.390	0.483	0.400
12	Trace (4)	275	141	**0.210**	0.450	0.460	0.500
13	Two pattern (4)	128	61	0.055	**0.033**	0.081	0.280
14	Wafer (2)	152	74	0.012	0.038	0.020	**0.010**
15	Yoga (2)	426	209	0.242	**0.195**	**0.195**	0.200
Average				**0.284**	0.288	0.378	0.329

The TWA_SAX algorithm shows that generated the optimal word and alphabet size stamped with the entropy for each data set and this was verified by the Relative Frequency (RF) algorithm based on [14].

Table 4 illustrates the results obtained by the proposed algorithm (TWA_SAX) and other algorithms identified in the literature review in terms of alphabet size and compression ratio for the SAX process. The best results are presented in bold. Table 4 displays the alphabet sizes obtained by the integrated algorithm TWA_SAX compared with HSAX, GA and original SAX. The alphabet size of TWA_SAX is compared with HSAX in [14] that gave predefined alphabet sizes, which were chosen as optimal alphabet sizes after the HS algorithm had run 10 times for each data set, and these are denoted in Table 5 as a HSAX. The alphabet size obtained by TWA_SAX was also compared with SAX [12] which used a=10 as fixed alphabet size.

Table 4: Alphabet size obtained by TWA_SAX, HSAX, GA and original SAX

No	Data set	TS length (n)	Comp. Ratio %	Alphabet size (a)			
				TWA_SAX	HSAX	SAX	GA
1	50word	140	52	**29**	20	10	6
2	Adiac	102	58	31	**52**	10	35
3	Beef	236	50	4	**20**	10	17
4	CBF	66	52	**23**	10	10	3
5	Coffee	166	58	**18**	15	10	5
6	Control chart	45	75	**25**	10	10	6
7	FaceAll	55	42	**26**	13	10	7
8	FaceFour	139	40	**38**	23	10	3
9	Gun Point	86	57	19	**32**	10	5
10	Olive Oil	189	33	15	**22**	10	17
11	Swedish Leaf	62	48	**31**	**30**	10	8
12	Trace	141	51	25	**50**	10	13
13	Two pattern	61	48	13	**15**	10	4
14	Wafer	74	49	10	**40**	10	**40**
15	Yoga	209	49	**37**	25	10	8

The TWA_SAX algorithm performs better in 8 data sets in terms of maximizing the information (alphabet size) compared to the alphabet size produced by GA, HSAX and original SAX algorithms, where TWA_SAX ties with HSAX, SAX and GA in 8 data sets, such as alphabet size of 50word data set is 29 and Control chart dataset is 25. In comparison with other algorithms, HSAX gets also better alphabet size in 8 data sets, such as Beef data set is 20, Swedish Leaf data set is 30 and two pattern is 15 alphabet sizes. PAA is the most preferred approaches because it enables effective dimensionality reduction; PAA representation does not generate good stability of the lower bound. In addition, the PAA approach focuses only on the central tendency and is not concerned with the dispersion in each segment. Therefore, the proposed method depends on the PAA algorithm as an improvement for symbolic representation. The TWA_SAX are a combination of a weight-based (time) dimensionality reduction. Both alphabet size and word size achieved from TWA_SAX can be indicative of the amount of data reduction. Therefore, the proposed method results indicates less information loss in terms of representing the number of alphabets and the word size compared to HSAX, SAX and GENEBLA. The TWA can enhance representation precision through a superior tightness of the lower bound in comparison with PAA.

6 Conclusion

The task of representation in time series data mining is critical because the direct handling of continuous data with high dimensionality is extremely challenging to manage efficiently. Since time series data is ubiquitous, a lot of research efforts have been devoted to the problems inherent in time series data mining in recent years. To address particular issue, namely, dimensionality reduction of symbolic time series, which focuses on the proposed method that could reduce the data from n dimensions to N dimensions by dividing the time series into N equal-sized 'frames', without loss high amount of information as much as possible. In this study, we proposed the integration of the TWA approaches with SAX to improve the performance of SAX, a symbolic time series data representation. The main feature of TWA is that it can reduce time series data by partition the time series into N equal-sized as segments, which is aggregated or collected within a segment based on weighted average value of the data falling, and a vector of these values becomes a large amount of data into a reduced representation. The results of our experiments and analyses showed that the proposed TWA_SAX method can potentially generate lower error rates than comparable methods in the literature. When continuous data, such as time series data, needs to be discretized, the aim is to optimize the bin number without losing information. This goal is very important to achieve for instance when analysing weather data, in which patterns from each time series may contribute important information that would otherwise be lost due to the application of less effective methods.

ACKNOWLEDGEMENTS
This work is supported by National University of Malaysia (UKM), Grant no. ERGS/1/2012/STG07/UKM/01/1.

References

[1] Han, J., Dong, G., and Yin, Y. Efficient mining of partial periodic patterns in time series database. in *Data Engineering, Proceedings.99, 15th International Conference on*, 1999, pp. 106-115.

[2] García-López, D.-A. and Acosta-Mesa, H.-G. 2009. Discretization of time series dataset with a genetic search, in *MICAI 2009: Advances in Artificial Intelligence*, ed: Springer, pp. 201-212.

[3] Keogh, E. and Lin, J. 2005. Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and information Systems,* vol. 8, pp. 154-177.

[4] Shasha, D. E. and Zhu, Y. 2004. High performance discovery in time series: techniques and case studies. Springer.

[5] Marroquín, M. G. V., Cervantes, M. C. A., Flores, J. L. M., and Cabrera- Ríos, M. 2009. Time series: Empirical characterization and artificial neural network-based selection of forecasting techniques. *Intelligent Data Analysis,* vol. 13, pp. 969-982.

[6] Park, S.-H., Lee, J.-H., and Lee, H.-C. 2011. Trend forecasting of financial time series using PIPs detection and continuous HMM. *Intelligent Data Analysis,* vol. 15, pp. 779-799.

[7] Korn, F., Jagadish, H. V., and Faloutsos, C. 1997. Efficiently supporting ad hoc queries in large datasets of time sequences. *ACM SIGMOD Record,* vol. 26, pp. 289-300.

[8] Lonardi, J. L. E. K. S. and Patel, P. Finding motifs in time series. in *Proc. of the 2nd Workshop on Temporal Data Mining*, 2002, pp. 53-68.

[9] Zhu, Y., 2004, High performance data mining in time series: Techniques and case studies. New York University,

[10]Montañés, E., Quevedo, J. R., and Prieto, M. M. 2011. A greedy algorithm for dimensionality reduction in polynomial regression to forecast the performance of a power plant condenser. *Intelligent Data Analysis,* vol. 15, pp. 733-748.

[11]Keogh, E., Chakrabarti, K., Pazzani, M., and Mehrotra, S. 2001. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and information Systems,* vol. 3, pp. 263-286.

[12]Lin, J., Keogh, E., Lonardi, S., and Chiu, B. A symbolic representation of time series, with implications for streaming algorithms. in *Proceedings of the 8th*

ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, 2003, pp. 2-11.

[13]López, D. A. G., Mesa, H. G. A., Ramırez, N. C., and Montes, E. M. 2007. Algoritmo de Discretización de Series de Tiempo Basado en Entropıa y su Aplicación en Datos Colposcópicos.

[14]Ahmed, A. M., Bakar, A. A., and Hamdan, A. R. Harmony Search algorithm for optimal word size in symbolic time series representation. in *Data Mining and Optimization (DMO), 2011 3rd Conference on*, 2011, pp. 57-62.

[15]Chakrabarti, K., Keogh, E., Mehrotra, S., and Pazzani, M. 2002. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Transactions on Database Systems (TODS),* vol. 27, pp. 188-228.

[16]Hung, N. Q. V. and Anh, D. T. 2008. An improvement of PAA for dimensionality reduction in large time series databases, in *PRICAI 2008: Trends in Artificial Intelligence*, ed: Springer, pp. 698-707.

[17]Guo, C., Li, H., and Pan, D. 2010. An improved piecewise aggregate approximation based on statistical features for time series mining, in *Knowledge Science, Engineering and Management*, ed: Springer, pp. 234- 244.

[18]Li, H. and Guo, C. 2013. Piecewise aggregate approximation based on the major shape features of time series. *Information Sciences,*

[19]SUN, Y., WANG, R., JIN, Z., and ZHANG, J. 2014. A Trend Based Lower Bounding Distance Measure for Time Series⋆. *Journal of Computational Information Systems,* vol. 10, pp. 5907-5914.

[20]Karamitopoulos, L. and Evangelidis, G. A dispersion-based paa representation for time series. in *Computer Science and Information Engineering, 2009 WRI World Congress on*, 2009, pp. 490-494.

[21]Lkhagva, B., Suzuki, Y., and Kawagoe, K. 2006. Extended SAX: Extension of symbolic aggregate approximation for financial time series data representation. *DEWS2006 4A-i8,* vol. 7,

[22]Shieh, J. and Keogh, E. i SAX: indexing and mining terabyte sized time series. in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 623-631.

[23]Li, G., Zhang, L., and Yang, L. 2012. TSX: A novel symbolic representation for financial time series, in *PRICAI 2012: Trends in Artificial Intelligence*, ed: Springer, pp. 262-273.

[24]Hung, N. Q. V. and Anh, D. T. Combining SAX and piecewise linear approximation to improve similarity search on financial time series. in *Information Technology Convergence, 2007. ISITC 2007. International Symposium on*, 2007, pp. 58-62.

[25]Azami, H., Bozorgtabar, B., and Shiroie, M. 2011. Automatic signal segmentation using the fractal dimension and weighted moving average filter. *Journal of Electrical & Computer science,* vol. 11, pp. 8-15.

[26]Apostolico, A., Bock, M. E., and Lonardi, S. 2002. Monotony of surprise and large-scale quest for unusual words. presented at the Proceedings of the sixth annual international conference on Computational biology, Washington, DC, USA.

[27]Marx, M. L. and Larsen, R. J. 2006. Introduction to mathematical statistics and its applications. Pearson/Prentice Hall.

[28]Meijering, E. 2002. A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proceedings of the IEEE,* vol. 90, pp. 319-342.

[29]Bakar, A. A., Ahmed, A. M., and Hamdan, A. R. 2010. Discretization of time series dataset using relative frequency and K-nearest neighbor approach, in *Advanced Data Mining and Applications*, ed: Springer, pp. 193-201.

[30]Faloutsos, C., Ranganathan, M., and Manolopoulos, Y. 1994. Fast subsequence matching in time-series databases. vol. 23: ACM.

[31]Lonardi, S., 2001, Global detectors of unusual words: design, implementation, and applications to pattern discovery in biosequences. Purdue University,