ALSAX: Alternative Empirical SAX for Time Series Symbolic Representation

Damien Porcher[1,2] (✉)[0009−0008−9228−9732], Ali Ayadi[1], Ali Ismail-Fawaz[2], Maxime Devanne[2], Jonathan Weber[2], and Cédric Wemmert[1]

[1] ICube, Université de Strasbourg, 300 Bd Sébastien Brant, 67400 Illkirch-Graffenstaden, France {ali.ayadi, wemmert}@unistra.fr
[2] IRIMAS, Université de Haute-Alsace, 12 Rue des Frères Lumière, 68093 Mulhouse, France {damien.porcher, ali-el-hadi.ismail-fawaz, maxime.devanne, jonathan.weber}@uha.fr

Abstract. Time series representation methods are widely adopted in the community to improve our understanding of data distributions. Among them, Symbolic Aggregate approXimation (SAX) is one of the most famous for its interpretability and simplicity. In this paper, we propose a new method called ALSAX to represent the series with symbols derived from the empirical distribution of the data. This method is based on the existing SAX and can be seen as an improvement in versatility due to the possibility of changing the symbolization according to the data. Grounded in new theoretical results, ALSAX enables the integration of prior knowledge to optimize data representation while maintaining differentiability. Moreover, we present the Filter Representation Aggregation (FRA) to replace the Piecewise Aggregate Approximation (PAA), yielding an intermediate representation focused on relevant time series trends such as increases, decreases, or peaks. To enrich and generalize these approaches, we applied an ensembling strategy combining SAX and ALSAX, which demonstrates statistically significant improvements on the UCR Archive. Finally, we propose a fusion of multiple dictionaries using different FRA parameterizations that significantly outperforms individual methods.

Keywords: Symbolic representation · Time series classification · XAI

1 Introduction

The task of Time Series Classification (TSC) has attracted significant attention over the last few decades due to its numerous applications in domains such as healthcare, finance, transportation, and industrial monitoring. Benchmark repositories, such as the UCR archive [2] provide a large collection of heterogeneous datasets and have become standard evaluation frameworks for comparing TSC methods.

Existing TSC approaches include distance-based methods, feature-based approaches, interval methods, dictionary methods, shapelet-based representations, and hybrid-methods, which combine different approaches. Many dictionary-based

methods rely on symbolic representations. Among them, Symbolic Aggregate approXimation (SAX) [9] remains one of the most widely used symbolic representations because of its low computational cost, interpretability, and lower-bounding property with respect to the Euclidean distance. By combining dimensionality reduction through Piecewise Aggregate Approximation (PAA) [8] with symbolic discretization, SAX enables efficient indexing [1] and pattern extraction on large time series datasets.

Despite these advantages, SAX presents several limitations. First, the discretization process relies on breakpoints derived from the Gaussian distribution, implicitly assuming that normalized time series follow a standard normal law. In practice, this assumption rarely holds and may lead to poorly adapted symbolic representations. Second, the PAA reduction step may remove discriminative local temporal structures such as trends, transitions, or peaks, which are often critical for classification tasks. Finally, classical symbolic representations remain difficult to integrate into differentiable learning frameworks.

In this paper, we propose ALSAX, an adaptive symbolic representation framework for time series classification. The proposed method introduces data-dependent symbolic breakpoints estimated from the empirical cumulative distribution function of the dataset instead of fixed Gaussian quantiles. In addition, we propose replacing PAA with a Filter Reduction Aggregation (FRA) strategy designed to preserve local temporal dynamics before symbolic encoding, thus extending the derivative representation to a wider receptive field. The proposed framework also introduces almost everywhere differentiable symbolic mappings, enabling future integration into gradient-based optimization pipelines. The contributions of this work are summarized as follows:

- We introduce a data-adaptive symbolic discretization method based on empirical distribution breakpoints;
- We propose a trend-preserving reduction framework using increasing, decreasing, and peak filters;
- We establish theoretical properties linking derivative representations, filtered representations, and Euclidean distances;
- We experimentally compare ALSAX and SAX on datasets from the UCR archive under several configurations.

Experimental results show that ALSAX and SAX provide complementary symbolic representations, while FRA can improve the encoding of local temporal dynamics in several configurations. These results suggest that symbolic approaches remain relevant for TSC when enriched with data-dependent discretization and interpretable trend-aware reductions. The GitHub repository is available here `https://github.com/MSD-IRIMAS/ALSAX`

2 Related work

TSC has been widely studied through several families of methods, including distance-based approaches such as DTW [4], statistical models, shapelet-based

methods, dictionary-based methods, and more recently deep learning architectures such as FCN, ResNet, InceptionTime [7], and ROCKET [3]. Although deep models often achieve strong predictive performance, their lack of interpretability remains a limitation in sensitive domains. This motivates the study of symbolic representations, which provide compact and more interpretable descriptions of temporal patterns. Among symbolic methods, Symbolic Aggregate approXimation (SAX) [9] is one of the most influential approaches. It combines Piecewise Aggregate Approximation (PAA) with a discretization step based on breakpoints derived from the standard Gaussian distribution. Several extensions and applications of SAX have been proposed, including iSAX [1], SAX-VSM [14], and SAX-CNN [15]. Other works have also explored adaptive or trend-aware symbolic representations, for instance by modifying the discretization strategy, using data-dependent breakpoints, or enriching the representation with local temporal dynamics. ALSAX follows this line of work, but focuses specifically on replacing Gaussian breakpoints with empirical distribution-based breakpoints, making the symbolic encoding more consistent with the observed data distribution. SFA [11] constitutes another major symbolic representation family. Unlike SAX-based approaches, SFA represents subsequences in the frequency domain before discretization. Based on SFA, dictionary-based classifiers such as BOSS [10], WEASEL [12], and WEASEL 2.0 [13] build discriminative bags of symbolic words for classification. These methods primarily target classification performance through dictionary construction and feature selection, whereas ALSAX focuses on the symbolic discretization step itself. Therefore, ALSAX can be seen as complementary to dictionary-based classifiers rather than as a direct replacement. Local temporal structures such as trends, transitions, and peaks are known to be important for TSC. This has motivated the use of shapelets [5], convolutional filters, and multi-scale architectures. The proposed Filter Reduction Aggregation (FRA) is related to this idea but differs from learned convolutional filters: it uses predefined and interpretable filters to replace PAA before symbolic encoding. Its goal is not to learn features end-to-end, but to preserve trend-related information in a symbolic representation framework.

Symbolic Aggregate approXimation method (SAX) In [9], Lin et al. use a transformation of time series called **Piecewise aggregate approximation** (PAA) [8] to create a new type of symbolic representation.

Definition 21. *Let $T, w \in \mathbb{N}^*$ with $w|T$. Let $C \in \mathbb{R}^n$ be a time series. The **piecewise aggregate approximation** of C in a space of dimension w is defined as:*

$$\forall i \in [\![1, w]\!], \overline{c_i} = \frac{w}{T} \sum_{j=\frac{T}{w}(i-1)+1}^{\frac{T}{w}i} c_j.$$

Definition 22. *A sorted list $B = \beta_0, ..., \beta_a$ is a set of **breakpoints** if:*

$$\forall X \sim \mathcal{N}(0,1), \quad \int_{\beta_i}^{\beta_{i+1}} d\mu_X(x) = \frac{1}{a},$$

where $\beta_0 = -\infty$ *and* $\beta_a = +\infty$.

Remark: Here the breakpoints are defined implicitly but, if $F_{\mathcal{N}(0,1)}$ denotes the repartition function of the Gaussian law, then:

$$\forall i \in [\![0, a]\!], \quad \beta_i = F^{-1}_{\mathcal{N}(0,1)}\left(\frac{i}{a}\right). \tag{1}$$

After defining the breakpoints, we can assign a symbol ε_i to each interval $]\beta_i, \beta_{i+1})$ for all $0 \leqslant i \leqslant a-1$. Each subsequence defined by the PAA is then assigned to a symbol. The symbolic representation of the time series is then similar to a word. It can be summarized into a dictionary as follows:

x	*symbol*
$]-\infty, \beta_1]$	ε_0
$]\beta_1, \beta_2]$	ε_1
$\vdots$	$\vdots$
$]\beta_{a-1}, +\infty[$	ε_{a-1}

This method is called Symbolic Aggregate approXimation (SAX). Moreover, we can define a distance on this dictionary that has a fundamental property: it lower bounds the Euclidean norm.

Definition 23. *Let* $(\varepsilon_i)_{i\in[\![0,a-1]\!]}$ *be a family of symbols from a SAX dictionary of breakpoints* $(\beta_i)_{i\in[\![0,a]\!]}$. *We define for all* $(i,j) \in [\![0, a-1]\!]^2$*:*

$$dist(\varepsilon_i, \varepsilon_j) = \begin{cases} 0, & \text{if } |i-j| < 1 \\ \beta_{\max(i,j)-1} - \beta_{\min(i,j)}, & \text{otherwise.} \end{cases}$$

Definition 24. *Let* $T \in \mathbb{N}^*$ *and* $(Q, C) \in \mathbb{R}^T$. *We denote by* $\hat{Q}$ *and* $\hat{C}$ *SAX representations of respectively* Q *and* C. *We define:*

$$MINDIST(\hat{Q}, \hat{C}) = \sqrt{\frac{T}{w}}\sqrt{\sum_{i=1}^{w} dist(\hat{q}_i, \hat{c}_i)^2}$$

Theorem 21.

$$\forall (Q, C) \in \mathbb{R}^T, \quad MINDIST(\hat{Q}, \hat{C}) \leqslant ||Q - C||_2. \tag{2}$$

3 Methods

3.1 How to improve SAX?

SAX biases In reality, it is only relevant to make this density choice if the time series has been normalized using the same law as shown below. In general, this choice should be related to the data, for instance in human motion where

preserving the skeleton shape is primordial. An assumption that the series are all realizations of a Gaussian law random variable does not hold in most cases. We created two improvements to SAX. On the one hand, a method to calculate symbols that permits backpropagation (almost everywhere differentiable) and on the other hand, a method relaxing the gaussian assumption.

Example 1. *If we normalize a positive TS defined on* $[0, T]$ *with:*

$$normalization(TS)(t) = \frac{1}{||TS||_\infty} TS\left(\frac{t}{T}\right),$$

then choosing values according to a Gaussian law is flawed. Since $F^{-1}_{\mathcal{N}(0,1)}\left(\frac{1}{3}\right) \approx -0,43$ *and by symmetry of the Gaussian density, let us set the dictionary:*

x	*symbol*
$]-\infty, -0.43]$	a
$]-0.43, 0.43[$	b
$[0.43, +\infty[$	c

Then, the normalization implies a higher number of b than c and no a in the symbolic representations SAX. This highlights the importance of the dictionary to be consistent with the data.

Moreover, the distance between symbols in SAX considers series to be identical except for an interval. In fact, it is a lower bound for norm 2, but this poses a problem because we know that if two TSs are close in norm 2, they are close in symbolic norms, but the converse is false: *these two norms are not equivalent.*

Example 2. *Let us take the dictionary from the previous example. Let* $X = (0.4, 0.5)$ *and* $Y = (0.5, 0.4)$ *in* $\mathbb{R}^2$*. Then:*

$$MINDIST(\hat{X}, \hat{Y}) = 0 \quad and \quad ||X - Y||_2 = \frac{\sqrt{2}}{10}.$$

This proves that the two norms are not equivalent.

This is why the tightness of lower bounds is used. We cannot study both norms separately.

Backpropagable symbols First, we need to obtain numbers to have something differentiable. Second, the symbol should be explicitly data dependent. This motivates the following choice.

Remark: Here, we do not use PAA because it is unnecessary to transform a TS into a symbolic representation. PAA simply turns a TS into a TS of a lower dimension. As well, this function is almost everywhere differentiable, but can be smoothed into a differentiable one as usual.

Data distribution based breakpoints The idea is to replace in (Eq. 1) the Gaussian cumulative distribution function by the cumulative distribution function of the dataset. Since we do not have access to the true function, we will replace it by its corresponding empirical function, that is to say:

Definition 31. *Let $T \in \mathbb{N}^*$. Let $\mathcal{D} = \{X_1, ..., X_n\}$ be a set of time series in $\mathbb{R}^T$. We define the **empirical cumulative distribution function** (ECDF), denoted as $\hat{F}_n$, as:*

$$\forall x \in \mathbb{R}, \quad \hat{F}_n(x) = \frac{1}{nT} \sum_{i=1}^{n} \sum_{j=1}^{T} \mathbb{1}_{\{X_i(t_j) \leqslant x\}}(x)$$

Since we need to inverse the ECDF, we can define it as:

Proposition 31.

$$\forall u \in [0, 1], \ \hat{F}_n^{-1}(u) = \min_{x \in \mathcal{D}} \{x \mid u \geqslant \hat{F}_n(x)\}$$

We can now define our breakpoints as:

Definition 32. *Let $a \in \mathbb{N}^*$ be the number of symbols in the dictionary. We define:*

$$\forall i \in [\![1, a-1]\!], \quad \beta_i = {\hat{F}_n}^{-1}\left(\frac{i}{a}\right).$$

This defines breakpoints that are dependent on the distribution of data.

Dimension reduction on trend criteria In this section, we want to replace the PAA by a dimension reduction method that preserves trend information. According to Ismail-Fawaz et al. [6], trend information is of high importance in time series classification. We will use their filters to construct our reduction method.

Definition 33. *Let $t \in \mathbb{N}^*$. We define a **time series** $X = (x_1, ..., x_T) \in \mathbb{R}^T$ as a vector with values in $\mathbb{R}$ such that indices are ordered following the time flow.*

Definition 34. *We define an **increasing trend filter** of length k the vector ω:*

$$\omega_k = \left((-1)^i\right)_{i \in [\![1;k]\!]}.$$

The increasing trend filtered time series X is then given by:

$$\overset{\nearrow}{X} = \left(x_i \mathbb{1}_{\left\{ Y \in \mathbb{R}^T \mid \sum_{l=-\frac{k}{2}}^{\frac{k}{2}} y_{i+l}(-1)^l \geqslant 0 \right\}}(X) \right)_{i \in [\![1,T]\!]}.$$

Definition 35. *We define a **decreasing trend filter** of length k the vector:*

$$w_k = \left((-1)^i\right)_{i \in [\![0;k-1]\!]}.$$

The decreasing trend filtered time series X is then given by:

$$\overset{\searrow}{X} = \left(x_i \mathbb{1}_{\left\{ Y \in \mathbb{R}^T \,|\, \sum_{l=-\frac{k}{2}}^{\frac{k}{2}} y_{i+l}(-1)^{l+1} \geqslant 0 \right\}}(X) \right)_{i \in [\![1,T]\!]}.$$

Definition 36. *We define **peak filter** a discretization of the function of length k+1:*

$$\begin{aligned} f : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto -(4x^2 - 2)e^{-x^2} \end{aligned}$$

where the graph of f is represented in figure 1.

For instance, we choose to discretize it on $[-2; 2]$. Let σ_k denote the regular partition of $[-2; 2]$ into k subintervals. Then:

$$\forall i \in [\![1; k]\!],\ w_k^i = (f(\sigma_k^i)).$$

The peak filtered time series X is then given by:

$$\overset{\updownarrow}{X} = \left(x_i \mathbb{1}_{\{Y \in \mathbb{R}^T | Y * w \geqslant 0\}}(X) \right).$$

Filter Reduction Aggregation (FRA) Let $X \in \mathbb{R}^T$ denote a time series and let $w \in \mathbb{N}^*$ specify the window length. We define a subsequence as $X_{a,b} := (x_a, x_b)$. Let $\overset{\nearrow}{X}$, $\overset{\searrow}{X}$, and $\overset{\updownarrow}{X}$ represent the filtered versions of X resulting from the application of the increasing trend, decreasing trend, and peak trend filters. These filters operate by retaining the original input values where the specific trend condition is met and outputting 0 elsewhere. Finally, all zero elements are removed from the resulting representations.

Building upon the aforementioned symbolic representation method, each window is characterized by its corresponding ECDF. Consequently, we map each FRA to its ECDF-derived symbols using a sliding window approach, akin to [10]. Ultimately, the global representation of a time series is formed by sliding a window and concatenating the FRA outputs from each extracted subsequence.

3.2 Theoretical results

Let $(Q, C) \in \left(\mathbb{R}^T\right)^2$ be two time series of coordinates $(q_i)_{i \in [\![1,T]\!]}$, $(c_i)_{i \in [\![1,T]\!]}$. We denote by $Q' = (q_{i+1} - q_i)_{i \in [\![1,T]\!]}$, $C' = (c_{i+1} - c_i)_{i \in [\![1,T]\!]}$ where we impose periodicity conditions such that $q_{T+1} = q_1$ and $c_{T+1} = c_1$. It is relevant since we can extend a time series of finite length with any value.

Proposition 32. $||Q' - C'||_2 \leqslant 2||Q - C||_2$.

Theorem 31. *if:*

$$\sum_{i=1}^{T} q_i - c_i = 0,$$

then:

$$||Q - C||_2 \leqslant 2 \sin\left(\frac{\pi}{T}\right) ||Q' - C'||_2$$

We now propose a theoretical result on the FRA representation for which the previous one does not hold.

Proposition 33. $||\overset{\nearrow}{Q} - \overset{\nearrow}{C}||_2 \leqslant \sqrt{2}\sqrt{||Q||_2^2 + ||C||_2^2}$ *and* $||\overset{\searrow}{Q} - \overset{\searrow}{C}||_2 \leqslant \sqrt{2}\sqrt{||Q||_2^2 + ||C||_2^2}$.

This proposition provides a general norm bound for the increasing and decreasing filtered representations. It ensures that the filtered representations remain bounded by the energy of the original time series.

3.3 Representation comparison

In this section, we compare the different reduction and representation methods.

Filters reduction We wish to compare the data with its reduction in terms of derivatives and the corresponding filter. Thus, we compare in Figure 1 the increasing trend filter with the first-order criterion filter of positive speed.

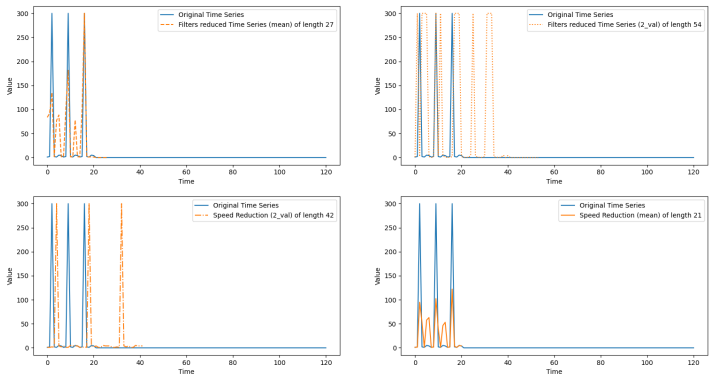


Fig. 1: Synthetic data reduced with the increasing filter for upper figures and reduced with speed for lower figures. On the left, reduction is performed with the mean, while on the right, we take the first and last values where the filter activates.

This illustrates that the trend filters do not exactly correspond to the definition of increasing and decreasing trends in terms of derivatives' sign. However, it allows for taking into account a larger receptive field at the same time, hence more information. We compared both and showed that, on the UCR Archive, trend filters perform better than derivatives (see Figure 7).

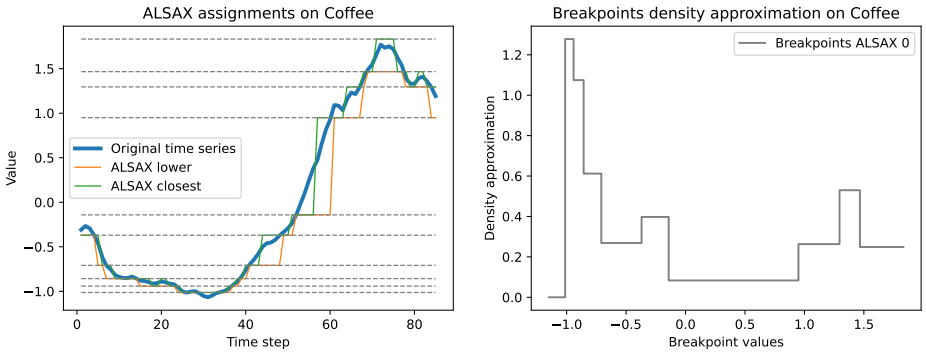


Fig. 2: Alphabet size=10. On the left, the original data from *Coffee* in blue, its representation with the closest breakpoint in Euclidean norm in orange and the lower bounding breakpoint in green.

Symbolic representation We now compare ALSAX with the original SAX. Both methods are intended to be applied to reduced data. However, we will show results on raw data to highlight the sheer impact of representing data in both cases. Figures 2 and 3 show examples of time series represented with ALSAX with the choice of an alphabet size of 10 and 100, illustrating the convergence of the representation method while increasing the alphabet size. We present two curves. In choosing values among the breakpoints to represent a timestamp, we have multiple choices. The green curve in figure 2 shows the representation with the closest under-bounding breakpoint, while the orange one is obtained by taking the closest breakpoint.

While the latter is the best at approximating the data, the first is interesting in that each illustrates a different point of view: the first considers that patterns can be described by tendencies, while the second takes into account values more precisely. This differentiates data representation and data reduction.

We compare the SAX and ALSAX representations in Figure 4. We highlight that ALSAX allocates symbols differently, capturing patterns that differ significantly from SAX.

4 Results

4.1 Theoretical case study

We design a theoretical scenario to illustrate where ALSAX outperforms SAX. A classic example involves distinguishing a steady-state from a transient regime. If the data points are highly concentrated in the discriminative region of the dataset, ALSAX will allocate more breakpoints than SAX in this critical area, making classification possible with a lower alphabet size. Conversely, if we need to classify an area with sparse data representation, SAX might prove superior since it distributes breakpoints evenly across the entire data range.

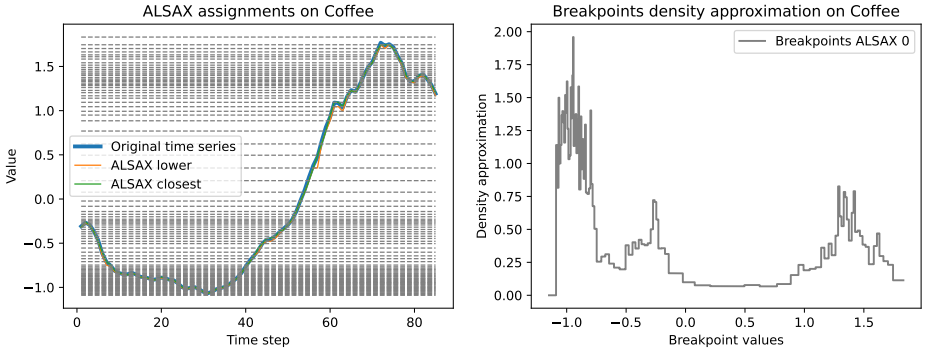


Fig. 3: Alphabet size = 100. On the left, the original data from *Coffee* in blue, its representation with the closest breakpoint in Euclidean norm in orange, and the lower-bounding breakpoint in green.

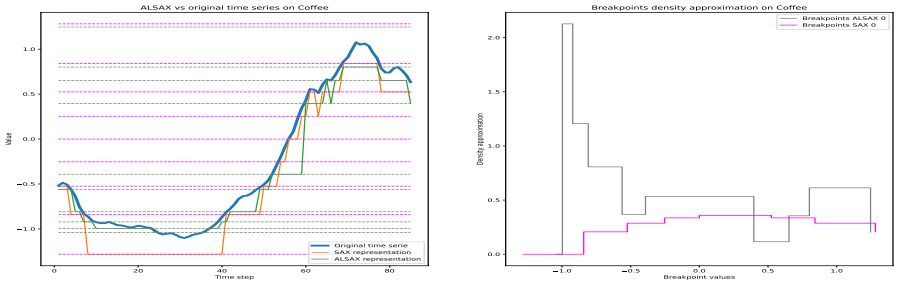


Fig. 4: SAX and ALSAX representations on the first time series of the Coffee dataset. Breakpoints are those matching the first window density for ALSAX. We replaced SAX symbols by the lower bounding breakpoint in the same way as ALSAX.

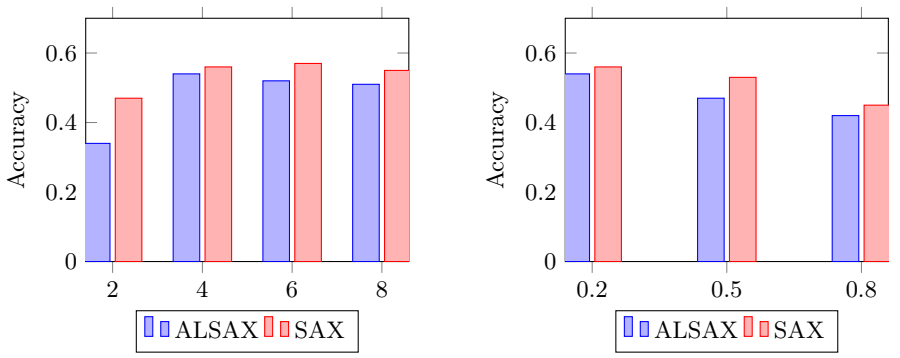
More generally, SAX allocates more breakpoints in the center of the normalized time series and fewer at the edges. ALSAX represents a broader paradigm. Modifying the distribution to place breakpoints in relevant areas is a highly effective way to improve classification. This approach can easily be extended to a distribution that assigns less weight to densely populated regions. For example, on the PigArtPressure dataset, ALSAX underperforms compared to SAX (an 83% difference in accuracy) because the data exhibits numerous oscillations, making frequency detection critical. Conversely, on ECG5000, the two classes present distinct extreme values; this enables ALSAX to classify them better than SAX (an 81% difference in accuracy), as SAX allocates fewer breakpoints to extreme values.

Furthermore, since we normalize time series, all empirical densities have bounded support. Hence, the density $\nu = 1 - \mu$ is bounded, and its ECDF provides quantiles that permit the identification of low-density areas for μ.

4.2 Experimental results

In this section, we evaluate the empirical performance of the aforementioned approaches. We calculate the ECDF on the training set of each dataset. We conduct our comparative analysis using the UCR Archive benchmark [2], which necessitates coupling our representation method with a downstream classifier. Following the foundational literature [14], we adopt the Vector Space Model (VSM) as our classifier, setting the word length to 4. It is worth noting that this classifier can easily be substituted with more advanced models to improve results, highlighting the critical synergy between representation and classification algorithms. Ultimately, our objective is to demonstrate that our proposed method achieves competitive baseline results while providing the crucial flexibility to tailor the breakpoints distribution to domain-specific priors.

We first compared SAX-PAA with ALSAX-PAA. There are a few parameters that are studied. The α parameter represents the size of the alphabet for both methods SAX and ALSAX. The win ratio parameter represents the ratio between the size of the sliding window and the length T of the TS. For the FRA method, the filter ratio represents the ratio between the size of the filter and the size of the input, which is the size of the sliding window here.



(a) Effect of alphabet size α (wr $= 0.2$). (b) Effect of window ratio wr ($\alpha = 4$).

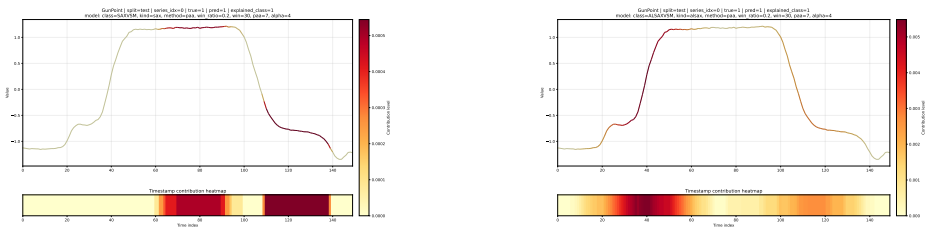
Fig. 5: Results for different PAA configurations. The mean is calculated on the UCR Archive 128 datasets benchmark.

Figure 5b highlights two primary trends. First, a low value of alpha (e.g., alpha=2) leads to a severe drop in the accuracy of ALSAX. Furthermore, increasing the alphabet size consistently degrades performance, irrespective of the

windowing technique or representation method. Second, an increased win ratio also results in deteriorated performance across both scenarios.

Since increasing the size of the alphabet does not improve performance, this verifies the assumption that SAX does not benefit from an increase in alphabet size in describing relevant areas, for instance, steady-states. However, there are datasets on which the value alpha=8 outperforms tremendously the model with alpha=4. On ECG5000, the accuracy reaches 10.89% for alpha=4 and 79.29% for alpha=8. This shows that, even if we found alpha=4 as the best alphabet size on the UCR, it is highly dependent on the task. However, going back to the dictionary phase, we can also assume that discriminative patterns are not weighted correctly, and using more modern methods such as deep learning could bring significant improvements [10].

To investigate whether both models capture the same information, we visualize the contribution of various patterns to the classification task by plotting the VSM weights in Figure 6. Our analysis reveals that SAX-VSM and ALSAX-VSM leverage completely distinct patterns, confirming the fundamental differences in feature representation between these two otherwise similar approaches.



(a) SAX-VSM (b) ALSAX-VSM

Fig. 6: Comparison of the heatmaps generated by SAX-VSM and ALSAX-VSM for the first test instance of the GunPoint dataset.

Next, we compare ALSAX-FRA with SAX-FRA. We evaluate fixed filter sizes, following the approach in [7]. The objective is to test the derivative representation (filter size of 2), for which we establish theoretical results, against larger filters' sizes.

Here we can see that ALSAX is always better than SAX when the filter size is the same across datasets. Moreover, ALSAX is always better than SAX when the filter size is the same across datasets. We further investigate by testing for filter ratios sizes from the windows' sizes instead of fixed sizes. The next diagram presents the results.

Different parameters' impact on the classification with FRA are tested. First, the alphabet size and the window ratios appear to prevail in performance since there are variations of over 10% in mean over the UCR. The alphabet size of 4 is the best for ALSAX and close to being the best for SAX among the different alphabet sizes tested. However, since UCR datasets are quite simple for most

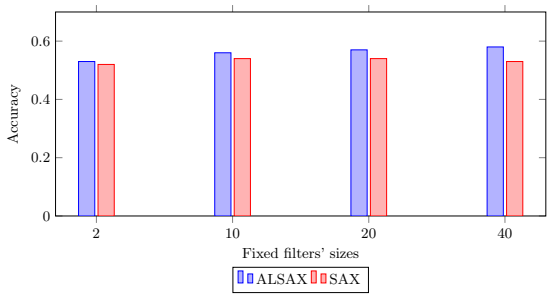
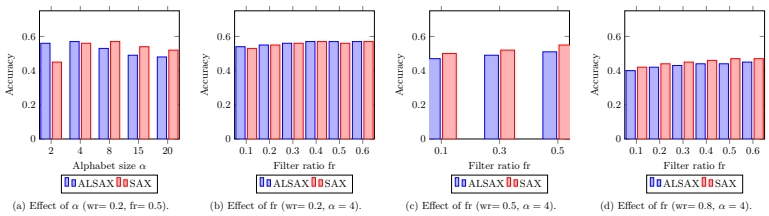


Fig. 7: Comparison between derivatives' representations and extended derivatives' representations.



(a) Effect of α (wr= 0.2, fr= 0.5). (b) Effect of fr (wr= 0.2, α = 4). (c) Effect of fr (wr= 0.5, α = 4). (d) Effect of fr (wr= 0.8, α = 4).

Fig. 8: Results for different FRA configurations. The mean is calculated on the UCR Archive 128 datasets benchmark.

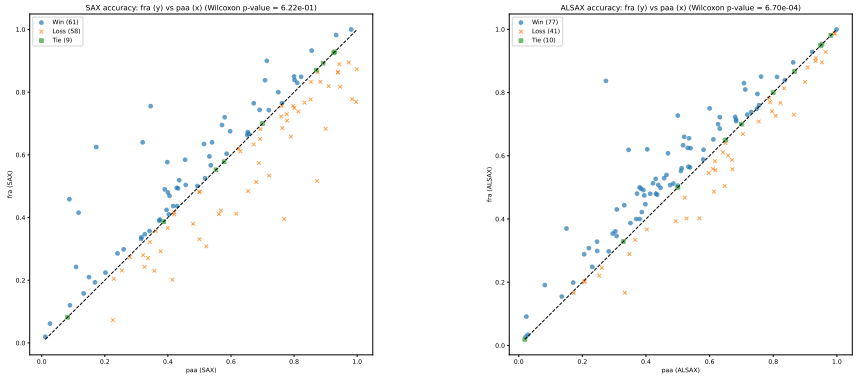
of them, it is difficult to establish that there should be a limit on the alphabet size. The alphabet size should be correlated with the complexity of the task. We confirm this by extracting the highest difference between alpha=4 and alpha=20 for ALSAX. We find that on ProximalPhalanxTW, the difference reaches 33% in favor of the largest alphabet model. The filter ratio appears to be increasing with performance.

To compare ALSAX-FRA and ALSAX-PAA, but also SAX-FRA and SAX-PAA, we present the Win/Tie/Loss diagram in accuracy on Figure 9. Statistical analysis (p-values) demonstrates that coupling FRA with ALSAX yields a significant performance improvement over PAA. Conversely, when applied to standard SAX, the performance difference between FRA and PAA is not statistically significant.

Since SAX and ALSAX work differently by design and give statistically similar results, studying their ensemble is natural. The ensemble is built by taking the output probabilities from both layers and the mean over each coordinate. Figure 10 shows the results of this experiment.

The ensembling between SAX and ALSAX significantly improves performance of at least 5%. This raises the idea that SAX and ALSAX extract diverse information and are complementary. It confirms observations from figure 6.

Finally, in [7], the Inception architecture includes filters of different sizes and incorporates them inside the same network. As its efficiency has been proven, we



(a) SAX with wr=0.2, fr=0.5, α=4. (b) ALSAX with wr=0.2, fr=0.5, α=4.

Fig. 9: Comparison of classification accuracies using FRA and PAA representations paired with SAX and ALSAX across the 128 UCR datasets. Statistical analysis demonstrates that ALSAX achieves significantly better performance when coupled with FRA.

mimic this strategy by fusing the dictionaries obtained from FRA with multiple filter ratios. We fused 6 dictionaries with filter ratios $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$.

As illustrated in Figure 11, the fusion model outperforms its individual constituent models in terms of both accuracy and F1-score. Because varying filter sizes capture distinct temporal features, their aggregation leads to statistically superior performance.

Moreover, trying to find optimal strategies on a subset of UCRs' datasets and testing them on the rest could bring improvements.

5 Conclusion

In this paper, we introduced ALSAX, a novel framework that adapts the SAX symbolic representation in a data-driven manner. This advancement facilitates the generation of a continuous spectrum of differentiable, data-dependent symbolic representations. Empirical evaluations confirm that ALSAX yields performance comparable to standard SAX, while introducing crucial flexibility and enabling the incorporation of domain knowledge via the selected breakpoint distribution.

While our comparative analysis relied on the Vector Space Model (VSM) for classification, coupling this representation with deep learning architectures offers significant potential. Crucially, the differentiability of ALSAX allows the symbolic representation layer to be optimized end-to-end within a broader learning

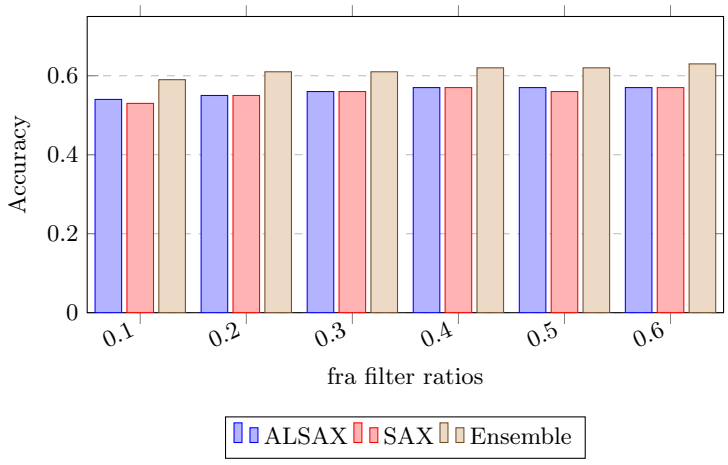


Fig. 10: Results for different FRA ensembling configurations (wr=0.2 and $\alpha = 4$). The mean is calculated on the UCR Archive 128 datasets benchmark.

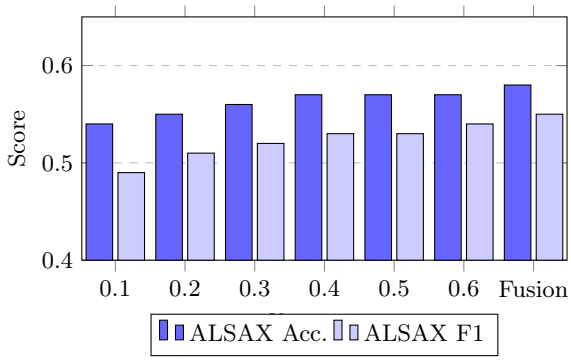


Fig. 11: ALSAX Accuracy and F1 for each FRA configuration (wr= 0.2, $\alpha = 4$) compared to the model where we fused all their dictionaries. The mean is calculated on the UCR Archive 128 datasets benchmark.

framework. Consequently, replacing traditional classifiers like VSM with deep neural networks represents a highly promising avenue for future research.

Acknowledgments. This work was supported by the project TURFU-NET (ANR-24-IAS1-0001) of the French Agence Nationale de la Recherche. The authors would like to acknowledge the High Performance Computing Center of the University of Strasbourg for supporting this work by providing scientific support and access to computing resources. Part of the computing resources were funded by the Equipex Equip@Meso project (Programme Investissements d'Avenir) and the CPER Alsacalcul/Big Data.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Camerra, A., Palpanas, T., Shieh, J., Keogh, E.: isax 2.0: Indexing and mining one billion time series. In: 2010 IEEE international conference on data mining. pp. 58–67. IEEE (2010)
2. Dau, H.A., Bagnall, A., Kamgar, K., Yeh, C.C.M., Zhu, Y., Gharghabi, S., Ratanamahatana, C.A., Keogh, E.: The ucr time series archive. IEEE/CAA Journal of Automatica Sinica **6**(6), 1293–1305 (2019). https://doi.org/10.1109/JAS.2019.1911747
3. Dempster, A., Petitjean, F., Webb, G.I.: Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. arXiv preprint arXiv:1910.13051 (2019)
4. Giorgino, T.: Computing and visualizing dynamic time warping alignments in r: the dtw package. Journal of statistical Software **31**, 1–24 (2009)
5. Huang, B., Jin, M., Liang, Y., Barthelemy, J., Cheng, D., Wen, Q., Liu, C., Pan, S.: Shapex: Shapelet-driven post hoc explanations for time series classification models. Advances in Neural Information Processing Systems **38**, 26639–26677 (2026)
6. Ismail-Fawaz, A., Devanne, M., Weber, J., Forestier, G.: Deep learning for time series classification using new hand-crafted convolution filters. In: 2022 IEEE International Conference on Big Data (Big Data). pp. 972–981 (2022)
7. Ismail Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D.F., Weber, J., Webb, G.I., Idoumghar, L., Muller, P.A., Petitjean, F.: Inceptiontime: Finding alexnet for time series classification. Data mining and knowledge discovery **34**(6), 1936–1962 (2020)
8. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Dimensionality reduction for fast similarity search in large time series databases. Knowledge and information Systems **3**(3), 263–286 (2001)
9. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing sax: a novel symbolic representation of time series. Data Mining and Knowledge Discovery **15**(2), 107–144 (2007). https://doi.org/10.1007/s10618-007-0064-z
10. Schäfer, P.: The boss is concerned with time series classification in the presence of noise. Data Mining and Knowledge Discovery **29**(6), 1505–1530 (Nov 2015). https://doi.org/10.1007/s10618-014-0377-7
11. Schäfer, P., Högqvist, M.: Sfa: a symbolic fourier approximation and index for similarity search in high dimensional datasets. In: Proceedings of the 15th international conference on extending database technology. pp. 516–527 (2012)
12. Schäfer, P., Leser, U.: Fast and accurate time series classification with weasel. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. pp. 637–646 (2017)
13. Schäfer, P., Leser, U.: Weasel 2.0: a random dilated dictionary transform for fast, accurate and memory constrained time series classification. Machine Learning **112**(12), 4763–4788 (2023)
14. Senin, P., Malinchik, S.: Sax-vsm: Interpretable time series classification using sax and vector space model. In: 2013 IEEE 13th international conference on data mining. pp. 1175–1180. IEEE (2013)
15. Wrzesień, M., Wrzesień, M., Korniak, J.: Classification of sax-like time series bitmaps with siamese convolutional neural networks (2025)