Explainable Long- and Short-term Pattern Detection in Projected Sequential Data

Matthias Bittner$^{1[0009-0004-8022-2232]}$,
Andreas Hinterreiter$^{2[0000-0003-4101-5180]}$,
Klaus Eckelt$^{2[0000-0001-6832-9070]}$, and
Marc Streit$^{2[0000-0001-9186-2092]}$

[1] Christian Doppler Laboratory for Embedded Machine Learning,
TU Wien, Gußhausstraße 27-29, 1040 Vienna, Austria,
matthias.bittner@tuwien.ac.at

[2] Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria,
{firstname.lastname}@jku.at

Abstract. Combining explainable artificial intelligence and information visualization holds great potential for users to understand and reason about complex multidimensional sequential data. This work proposes a semi-supervised two-step approach for extracting long- and short-term patterns in low-dimensional representations of sequential data. First, unsupervised sequence clustering is used to identify long-term patterns. Second, these long-term patterns serve as supervisory information for training a self-attention-based sequence classification model. The resulting feature embedding is used to identify short-term patterns. The approach is validated on a self-generated dataset consisting of heart-shaped paths with different sampling rates, rotations, scales, and translations. The results demonstrate the approach's effectiveness for clustering semantically similar paths and/or path sequences. This detection of both global long-term patterns and local short-term patterns facilitates the understanding and reasoning about complex multidimensional sequential data.

Keywords: pattern-detection · projected paths · self-attention

1 Introduction

Data are collected continuously from various sources, ranging from human perception to sensor-based recordings. Medical devices, environmental data, video surveillance, or computational simulations are examples that create multidimensional sequence or time series data [2]. However, the amount of data that can be recorded surpasses our capacity to analyze it manually. To cope with these vast amounts of data, we rely on algorithms and visual representations to process the data and detect meaningful patterns.

One common approach for understanding and visualizing multidimensional sequence data is to represent it with projected paths, two-dimensional projections of the sequences that are generated by applying Dimensionality Reduction (DR)

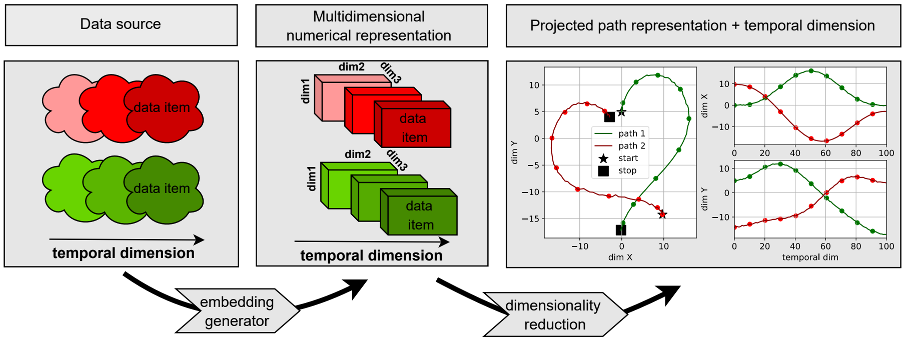


Fig. 1: Processing steps for the visualization of multidimensional sequential data.

techniques [2,3,4]. The low-dimensional representation can come with distortions, which can be quantified by certain quality measures, such as correlation coefficients or trustworthiness/continuity [9]. Despite these possible mismappings, multidimensional projections belong to the most important visualization techniques in this context. Figure 1 depicts a typical processing pipeline. To effectively find patterns within projected paths, Visual Analytics (VA) tools are employed, combining interactive visualization techniques with automatic analysis methods to support users in the analysis process [6,11].

In order to unveil hidden information effectively, these tools must incorporate automatic pattern detection methods. However, existing frameworks often lack this capability and are predominantly tailored for specific applications, limiting their broader applicability. Additionally, we argue that the applied pattern detection methods must be explainable to understand and trust the results [5]. This paper, therefore, proposes an automatic pattern detection pipeline. The main contribution is a semi-supervised two-step pattern extraction pipeline for automatically detecting long- and short-term patterns in projected sequential data:

- In the first step, *long-term* patterns are detected using unsupervised sequence clustering based on Dynamic Time Warping (DTW). The use of DTW ensures that semantically similar sequences form clusters.
- In the second step, a self-attention-based sequence classification Neural Network (NN) identifies *short-term* patterns by clustering attention scores of the self-attention mechanism for 1D convolutions with different kernel sizes.

A joint exploration of the extracted long- and short-term patterns in the low-dimensional representation can help users understand and reason about patterns in the high-dimensional sequence data.

2 Background & Related Work

This section provides preliminary definitions concerning multidimensional sequential projections and presents the state-of-the-art in pattern detection and eXplainable Aritficial Intelligence (XAI) for time series and sequential data.

2.1 Preliminaries

Definition 1. *A* sequence*, or* time series $\mathbf{X} = (\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_T)$ *is a time-ordered set of real values with* $\mathbf{x}_i \in \mathbb{R}^d$*, where* T *is the length of the sequence and* d *is the dimension of the feature vector for each data item in the sequence. If* $d = 1$ *we specify the sequence* $\mathbf{X}$ *as univariate, if* $d > 1$ *we call it a* multidimensional *or* multivariate *sequence.*

Definition 2. *We define* Dimensionality Reduction (DR) *of a multidimensional sequence as a transformation* $\mathbf{X}_{low} = \Phi(\mathbf{X}_{high})$*, which maps the high-dimensional set of features* $\mathbf{x}_{i,high} \in \mathbb{R}^{d_{high}}$ *to a low-dimensional set of features* $\mathbf{x}_{i,low} \in \mathbb{R}^{d_{low}}$*, under the condition of* $d_{high} > d_{low}$*.*

Definition 3. *A two-dimensional* projected sequence *is defined as a sequence* $\mathbf{P} \mathrel{\widehat{=}} \mathbf{X}_{low}$*, with* $\mathbf{x}_{i,low} \in \mathbb{R}^2$*.*

Definition 4. *We define a* long-term pattern *as a set of projected paths with a persistent and recurrent behavior or structure over the whole sequence. Sequences within a long-term pattern have similar temporal characteristics over their entire length* T*.*

Definition 5. *A* short-term pattern *can be seen as a descriptive segment within a brief period of time for a single projected path. The length of a short-term pattern is defined as* $t_i \ll T$ *and the union over the number of short-term patterns* K $\bigcup_{i=1}^{K} t_i = T$ *corresponds to the sequence length.*

2.2 Pattern Detection for Sequential Data

For multidimensional sequential datasets, it is a common approach to use the combination of dimensionality reduction and information visualization [11]. In such scenarios, engineers or data scientists try to automatically generate interactive representations and visualizations to gain insights into otherwise hidden patterns. Traditionally this follows a typical processing pipeline that starts with forwarding the multidimensional data to a DR algorithm and visualizing the projections in the two-dimensional space. Then, an analyst can interact with the visualization to better understand the data. To fill the missing gap of automatically detecting patterns and providing hints for interesting regions within the projections, automatic detection of patterns needs to be incorporated into these processing steps. Especially XAI methods for time series are suitable for this purpose since their main focus is on finding explanations, descriptions, regularities, and correlations in the data to gain a more detailed understanding.

The problem of automatically detecting patterns in projections can be viewed as a semi-supervised task. Unsupervised methods can reveal the structure and features of the data, e.g., clusters in the two-dimensional space, or clusters in the time series feature representation. Liao summarizes numerous different possibilities for clustering time series data [18]. Once the interesting regions are marked and identified in the underlying data, the human analyst can try to derive patterns, which can then be detected with the help of supervised methods. If the patterns have no temporal relationship, it might be an easy task to train supervised classifiers, e.g., decision trees, support vector machines, and Neural Networks (NNs). However, for temporal patterns, additional concepts, such as pattern matching, might have to be introduced. Mendhurwar et al. provide a qualitative overview of different time series and associated pattern-matching possibilities [8].

Literature focusing on the automatic and unsupervised extraction of sequential patterns in multidimensional projected paths is scarce. Most closely related to our work is an approach by Ali et al., which uses a convolutional autoencoder-based architecture to learn feature representations of the projections [1]. The learned feature embedding is then used for visualization purposes to enable tasks such as pattern and outlier analysis.

2.3 XAI for Time Series Data

According to XAI literature [10,16], several definitions and categorization schemes have been developed for XAI in the time series domain. We distinguish between *post-hoc* and *ante-hoc* approaches to reflect the point in time when explainability is introduced. *Post-hoc* explainability approaches are separated from the model itself and can provide insight into the model's learned behavior without actively changing its structure. They are mainly applied after the training and applied to black-box models like standard NNs. *Ante-hoc* methods, such as models with a built-in attention mechanism, are models that can be considered to be directly interpretable due to their internal structure and design.

Another categorization scheme to discriminate different XAI methods is related to the granularity of the produced explanation. The three main categories are *time point-based explanations*, *subsequence-based explanations*, and *instance-based explanations* [16].

Time point-based explanations focus on specific time points within a time series. They provide insights and assign a relevance score or weight to every time point of a time series [16].

Subsequence-based explanations refer to sub-parts or segments of the time series. They offer explanations that are relevant to specific patterns or trends observed within those subsequences and identify sub-parts of a time series responsible for the classification outcomes [16].

Instance-based explanations rely on the whole time series instance and the overall behavior and characteristics to reason about the decision. These explanations are often related to global explanations which try to identify explanations generalizable for the whole dataset [16].

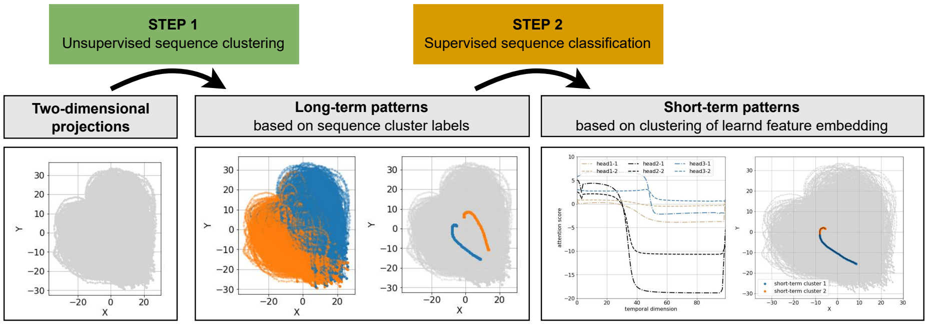

STEP 1
Unsupervised sequence clustering
STEP 2
Supervised sequence classification
Two-dimensional projections
Long-term patterns
based on sequence cluster labels
Short-term patterns
based on clustering of learnd feature embedding
X
Y
head1-1
head1-2
head2-1
head2-2
head3-1
head3-2
attention score
temporal dimension
short-term cluster 1
short-term cluster 2

Fig. 2: Overview of the proposed automatic detection pipeline for long- and short-term patterns.

In Section 3 we use this categorization scheme to contextualize the individual explanations created with our approach. For long-term pattern extraction, we cluster time series instances. The similarity measure applied by the clustering algorithm provides *instance-based* explanations. The self-attention-based supervised sequence classification architecture for short-term pattern extraction can be categorized as *ante-hoc* and *time point-based* method.

3 Methodology

The proposed explainable pattern detection approach can be interpreted as a combination of information visualization and pattern detection. The crucial part of extracting patterns is divided into two steps, which are dedicated to extracting long- and short-term patterns (Figure 2).

The first step of extracting long-term patterns from a set of projected paths is performed with the help of unsupervised sequence clustering. This approach uses Dynamic Time Warping (DTW) and K-Means for finding patterns in the entire set of sequences. The second step of extracting short-term patterns uses a self-attention-based supervised sequence classification architecture. The extracted cluster labels from step one serve as supervisory information and the learned attention embeddings for the classification task are used to cluster and visualize short-term patterns.

Hohman et al. propose an interrogative categorization scheme of Deep Learning (DL) based visual analytics concepts [5]. We use these slightly modified questions to categorize our work:

- **Why** do we care about long- and short-term patterns in projected paths?
- **When** can we apply the extraction of long- or short-term patterns?
- **How** do we extract the patterns?
- **What** do we visualize in the different steps?
- **Which** XAI category do the results belong to?

The following two subsections provide detailed answers to these questions for long-term (Section 3.1) and for short-term patterns (Section 3.2).

3.1 Grouping Paths – Long-term Patterns

Long-term patterns describe similar temporal characteristics of sequences over the whole time series. The overall goal of extracting long-term patterns is to identify trends and gain a deeper understanding of the behavior and structure of the underlying data.

Why do we care about long-term patterns in projected paths?

Facing a new and unknown large set of projected paths, without any prior knowledge about groups or clusters, it is of great interest to reveal hidden information. Prominent reasons why revealing long-term patterns in projected paths is important are:

- Facilitated Perception: The capability of grouping similar paths together can resolve the issue of cluttered scenes. Visualizing all paths at once might be of interest for certain tasks (e.g., to understand distributions). However, the resulting visual clutter can make more specific insights hard to obtain. By highlighting/color-coding the extracted clusters, or aggregating them visually, clutter can be avoided.
- Trend analysis: Long-term patterns can help to find dynamic trends in the data such as strong increases or decreases.
- Structure and behavior understanding: Understanding relationships between different long-term patterns can help to better understand the process, structure, and behavior of the underlying data.
- Anomaly detection: Projected sequences that strongly differ from long-term patterns can be identified as unique, abnormal patterns.
- Decision-making support: Identifying long-term patterns can be highly helpful for making decisions in order to improve control strategies or the behavior of the system.

When can we apply the extraction of long-term patterns?

The detection of long-term patterns as proposed in this work is designed to be applied after all multidimensional sequences have been recorded, stored, and projected to the two-dimensional space.

How do we extract the patterns?

We define a long-term pattern as a set of projected paths $\mathbf{P}$ with similar temporal characteristics. For evaluating the similarity within the temporal domain of the projections we utilize DTW [12] as similarity measure and K-Means as an unsupervised clustering approach.

What do we visualize?

The combination of DTW and K-Means allows us to assign a cluster label to each projected path $\mathbf{P}$. The labels are used for color encoding. By highlighting the

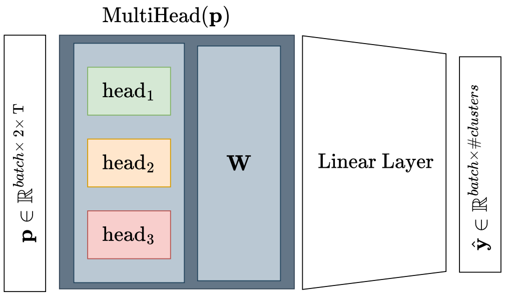

MultiHead(**p**)
$\mathbf{p} \in \mathbb{R}^{batch\times\, 2\times\, \mathrm{T}}$
head$_1$
head$_2$
head$_3$
W
Linear Layer
$\hat{\mathbf{y}} \in \mathbb{R}^{batch\times\#clusters}$

Fig. 3: Network architecture for learning the feature embeddings (head$_1$, head$_2$, head$_3$). Choosing different kernel sizes for the 1D-Convolutions in the attention heads leads to the advantage that each head can extract features at different scales.

cluster centroids, trends within the data can be made more visually prominent. Additionally, highlighting all paths with their corresponding label/color encoding allows for different preservation of the long-term patterns within the projections.

Which XAI category do the results belong to?

The cluster labels provide explanations on a global level, whereas the DTW similarity measure between the different sequences is used to discriminate the paths on a per instance level.

3.2 Highlighting Patterns in Projections – Short-term Patterns

Short-term patterns, in the context of projected paths, can be seen as descriptive segments within a brief period of time. In contrast to long-term patterns, short-term patterns are more specific and capture more dynamic properties within the projections.

Why do we care about short-term patterns in projected paths?

Short-term patterns can be used to identify sudden changes or short sequences with interesting behavior, e.g., abrupt changes in the latent space of a neural network during training, specific subsequences in rubrics cube solving or chess games [4,14].

When can we apply the extraction of short-term patterns?

The extraction of short-term patterns in our approach is applicable once we have the cluster labels of our long-term patterns. The labels are necessary because the long-term patterns serve as supervisory information for short-term pattern extraction.

How do we extract the patterns?

In the proposed approach, we utilize the labeled long-term patterns (projected path clusters), as supervisory information to train a sequence classification model (Figure 3).

The proposed NN architecture can be described by

$$\text{MultiHead}(\mathbf{p}) = \text{Concat}(\text{head}_1,\ \text{head}_2,\ \text{head}_3)\ \mathbf{W}, \tag{1}$$

where each head$_i$ is defined by:

$$\begin{aligned} \text{head}_i &= \text{Attention}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) = \text{softmax}\big(\frac{\mathbf{Q}_i\mathbf{K}_i^T}{\sqrt{d}}\big)\mathbf{V}_i \\ &= \text{Attention}(\text{Conv1d}_i^Q(\mathbf{p}), \text{Conv1d}_i^K(\mathbf{p}), \text{Conv1d}_i^V(\mathbf{p})). \end{aligned} \tag{2}$$

The architecture was designed to derive patterns on a local scale and is inspired by the self-attention mechanism [7,17]. Unusual for a time series classification problem, we are not interested in the task itself, but rather we are interested in the learned feature embeddings head$_i \in \mathbb{R}^{2\times T}$. The attention vectors result from the softmax scaled dot product between the three components, query $\mathbf{Q}$, key $\mathbf{K}$, value $\mathbf{V}$, $\in \mathbb{R}^{2\times T}$. All three embeddings are generated with corresponding 1D convolutions. The recent work of Tang et al. [15] provided valuable insights about the importance of using different kernel sizes to extract features at different scales. We used these insights and designed our model to extract features with different kernel sizes, which results in different receptive fields of the corresponding convolutions of each head$_i$. Additionally, we apply padding to the resulting feature maps, in order to preserve the original sequence length. The 1D convolutions are designed to have two input channels and two output channels. This allows for direct visualization of the feature embedding in the two-dimensional space and the temporal dimension. Additionally, this two-dimensional design enables an interpretation of the extracted representations (attention scores head$_{i,1}$, head$_{i,2}$) for each of the two dimensions of the analyzed projected path $\mathbf{P} \in \mathbb{R}^{2\times T}$. After having trained the architecture until convergence for the individual dataset, we analyze the learned feature embedding of each head$_i$ by applying K-Means clustering. The resulting cluster labels are then the label/pattern information and can be used for visualization in the original two-dimensional projections.

What do we visualize?

The cluster labels for the short-term patterns resulting from the process described above are then used for color-encoding sub-sequences of the projected paths.

Which XAI category do the results belong to?

The proposed self-attention-based sequence classification model can be categorized as an *ante-hoc* method. We use the attention matrices of the individual projected paths to derive the short-term patterns. The granularity of the explanations is based on a *point-based* clustering of the resulting attention scores for specific instances.

4 Experimental Results

The following section provides experimental results of the proposed methodology evaluated for a simulated dataset that consists of paths representing drawn heart

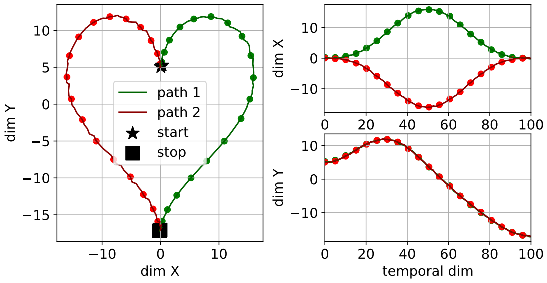


Fig. 4: Two-dimensional path of a drawn heart and their associated temporal representation on the individual coordinate. The temporal evolution of the y coordinate is equal for the two paths.

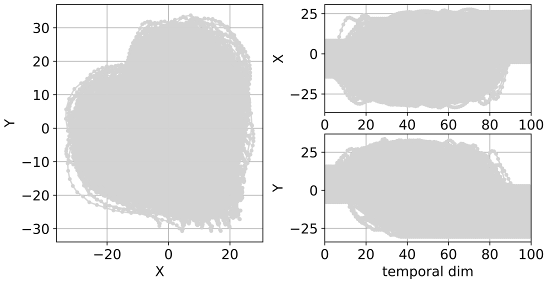


Fig. 5: The set of 1600 random augmented two-dimensional paths of drawn heart halves and their associated temporal representation on the individual coordinate.

halves at different positions, scales, rotations, and speeds. We also provide the code and the data to reproduce the results[1].

4.1 Simulated Heart Drawings

The purpose of this example is to illustrate the effectiveness of the proposed method, with a more detailed analysis of DTW-based K-Means clustering for finding long-term patterns and the self-attention-based sequence classification model used for finding the short-term patterns.

For generating a single path in the form of a heart in the two-dimensional space we use

$$\mathbf{P}(\boldsymbol{\theta}) = \begin{bmatrix} 16\sin(\boldsymbol{\theta})^3 + \boldsymbol{\epsilon}_1 \\ 13\cos(\boldsymbol{\theta}) - 5\cos(2\boldsymbol{\theta}) - 2\cos(3\boldsymbol{\theta}) - \cos(4\boldsymbol{\theta}) + \boldsymbol{\epsilon}_2 \end{bmatrix}, \tag{3}$$

where we define $\boldsymbol{\theta} = \boldsymbol{\theta}_1$ as a linearly spaced array of 100 values between 0 and π, for the right half of the heart and $\boldsymbol{\theta} = \boldsymbol{\theta}_2$ in a range between 2π and π for the

[1] https://github.com/mbitob/long-and-short-term-pattern-detection

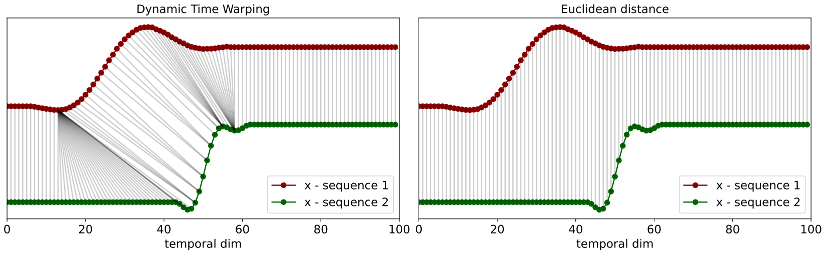


Fig. 6: Difference between the DTW similarity measure and the Euclidean distance for two heart drawing sequences from the same source, but different augmentations in rotation, scale, translation, and resolution. One observes that DTW is capable to resolve the temporal dependencies, while the Euclidean distance is not able to resolve the similarity. For the sake of visualization, sequence 2 is shifted vertically.

left half. The additive random noise $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$ are set to normally distributed random variables with zero mean and a standard deviation of 0.1. The two paths resulting from equation 3 evaluated with $\boldsymbol{\theta}_1$, and $\boldsymbol{\theta}_2$ are visualized in Figure 4.

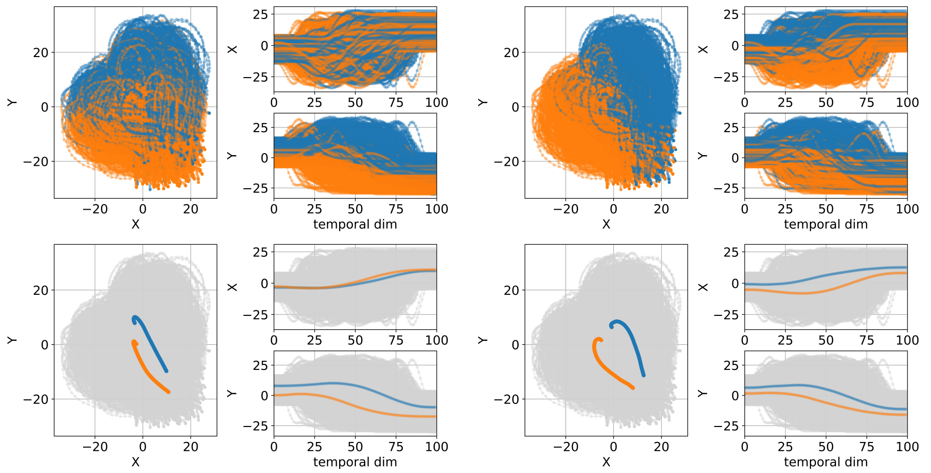
To generate a set of random augmented two-dimensional paths, we use the transformations

$$\mathbf{T}_{\text{rot}} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}, \; \mathbf{T}_{\text{scale}} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}, \text{ and } \mathbf{T}_{\text{trans}} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (4)$$

where ϕ represents a rotation of the path, s_x and s_y are the scaling factors of the individual dimensions, and t_x and t_y are used for controlling translation. For the experiments, we generate 1600 augmented paths, where we randomly sample $\phi \in [1/6\pi, 1/4\pi]$, s_x, $s_y \in [0.5, 1.5]$ and t_x, $t_y \in [-10, 10]$ from a uniform distribution. In order to simulate different drawing speeds we apply random rescaling in the temporal domain, combined with padding. The whole set of generated sequences including the temporal evolution for the x and the y axis are depicted in Figure 5. Without any prior knowledge, or the extraction of clusters and patterns, it would be a hard task for the data analyst to gain any insights and knowledge of this cluttered representation.

4.2 Long-term Patterns

Combining DTW with K-Means for extracting long-term patterns and grouping paths with similar temporal characteristics is effective. But DTW is an iterative process and its computational complexity is higher compared to a distance measure such as the Euclidean distance. To highlight the effects and differences between these two similarity measures, we consider two heart drawing sequences from the same source, but with different rotation, scale, translation, and resolution, as depicted in Figure 6. DTW is an alignment-based similarity measure and tries to align the features which match distinct patterns of the time series [12].



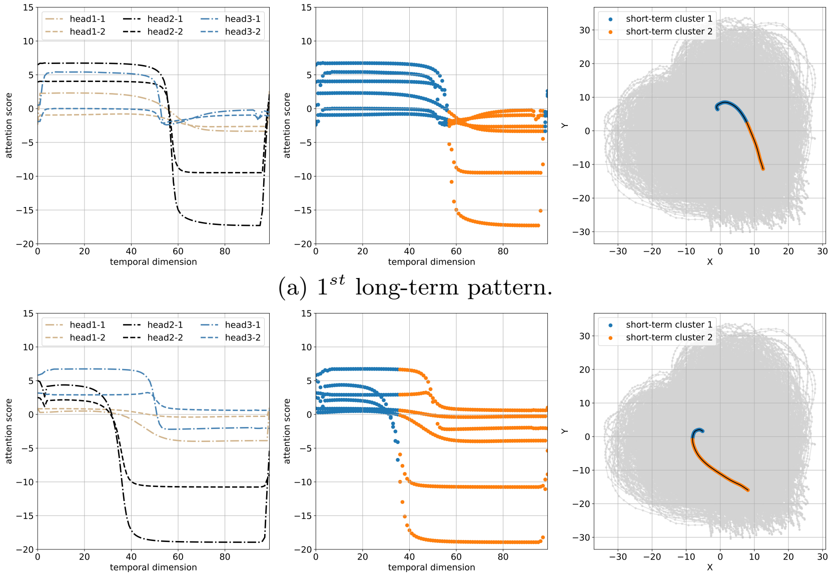
(a) Eucledian distance

(b) DTW

Fig. 7: A comparison between the Euclidian and the DTW similarity measure for the extraction of long-term patterns for the set of random augmented two-dimensional paths of drawn hearts.

The similarity is then evaluated by the distance of the matched points. On the other hand, when using the Euclidean distance, no feature matching takes place, which means that when a prominent pattern is shifted in time, the Euclidean distance results in a higher dissimilarity measure compared to DTW. Exactly this behavior is observable in Figure 6. The similarity measure with DTW for the two investigated sequences is $d_{\mathrm{DTW}} = 81.9$, whereas the Euclidean distance of $d_{\mathrm{euclidian}} = 81.9$ is much higher.

Next, we investigate the results of extracting long-term patterns on the whole set of drawn heart halves and compare the effects of using DTW and the Euclidian distance as a distance measure. Figure 7a shows the extracted long-term patterns with the Euclidean distance as a dissimilarity measure. Knowing that there should be two distinctive directions in this graph—i.e., paths going from top to left, and paths going from top to right—we notice that the clustering algorithm can resolve this pattern to a certain degree. However, due to the various different augmentations within the paths, the distinction is not very clear. This confusion is also confirmed by visualizing only the cluster centroids. In contrast, by applying DTW as a dissimilarity measure for K-Means (Figure 7b), we find a much more accurate result of the clusters. The cluster centroids exactly reflect the fact that there are two classes (left heart halves paths, and right heart halves paths). Despite the higher computational complexity of the DTW based clustering approach, it is worth applying it, especially when patterns within the projected paths are shifted in the temporal domain.



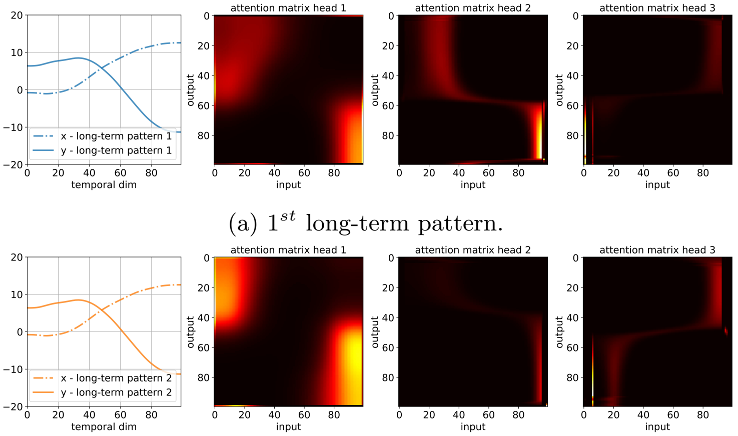
(a) 1^{st} long-term pattern.

(b) 2^{nd} long-term pattern.

Fig. 8: Extracted multi-head attention scores and the resulting short-term patterns for the two long-term patterns of the simulated heart drawing dataset.

4.3 Short-term Patterns

Here we provide experimental results for the detection of short-term patterns within the long-term patterns extracted with DTW as a dissimilarity measure (Figure 8). The color encoding for the short-term patterns is based on the attention scores (feature embeddings head_i) of the proposed sequence classification model (Section 3.2). For this specific experiment, we train the proposed multihead-attention classifier with kernel sizes $R_i \in \{3, 9, 15\}$ for the three different attention heads head_i. We use Adam as an optimizer and train for just 10 epochs with a learning rate of 10^{-3}. The cluster labels/long-term patterns serve as supervisory information for the training. The small amount of epochs is sufficient to reach a training accuracy of over 92 %. The attention scores and the resulting short-term patterns for the two extracted long-term sequences are illustrated in Figure 8. Comparing the attention scores of Figure 8a and 8b, the curvature has similar characteristics for both scenarios. The difference mostly lies in the slope, start, and end values of the scores, which correlate with the underlying original two-dimensional and temporal characteristics of the heart paths. Taking a closer look at the resulting short-term patterns, the cluster labels of the attention scores result in splitting the sequence in path snippets with a high curvature and a low curvature in the two-dimensional paths. Based on the cluster labels of the short-term patterns in the temporal domain of the multi-head



(a) 1^{st} long-term pattern.

(b) 2^{nd} long-term pattern.

Fig. 9: 2D input sequences and attention matrixes for the two long-term patterns of the simulated heart halves dataset.

attention scores, we observe that the cluster transition is located near the steep slope of head$_2$.

Interestingly, despite the *point-based explanation* design, the produced explanations are typically of a more long-ranged nature. The network learns a rich and structured feature embedding which results in *subsequence-based* explanations after the clustering.

Finally, we review the self-attention mechanism and the multi-head attention block defined by

$$\text{head}_i = \text{Attention}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) = \mathbf{A}_i \mathbf{V}_i = \text{softmax}\Big(\frac{\mathbf{Q}_i \mathbf{K}_i^T}{\sqrt{d}}\Big)\mathbf{V}_i, \tag{5}$$

where $\mathbf{A}_i \in \mathbb{R}_i^{T \times T}$ denotes the inherent attention matrix. With its quadratic form in the size of the sequence length, the attention matrix is independent of the number of dimensions of the input sequence. To get a better understanding and especially to highlight the effect of the different kernel sizes, we analyze the inherent attention matrices $\mathbf{A}_i$ for the sequence classification network trained on the two long-term patterns of the heart halves dataset. Figure 9 depicts the resulting attention matrices for the two long-term pattern cluster centroids. Each attention matrix can be interpreted as a mapping from the input to the output, where the pixel intensity is an indicator of the importance. This allows for analyzing which segments of the input sequence were relevant to the output. For example, a high importance in the lower-left corner of the attention matrix indicates a strong importance of the first elements of the input for the last elements of the output. Likewise, the lower-right corner of the matrix indicates the importance of the last elements of the input sequence to the last elements of the output sequence.

When comparing the different attention matrices in Figures 9a and 9b we notice that, in general, the attention matrices for the three different heads result in a similar characteristic for the two long-term patterns. Of course, there is a slight difference in the strength of the importance, but the network in general focuses on similar regions. Taking a closer look at the attention matrix $\mathbf{A}_1$ of head$_1$, we can reason that the small kernel size $R = 3$ focuses on shorter details, due to its smaller receptive field. This results in a strong focus on the beginning and the end of the sequence. The network somehow tries to extract and use the information of where in the two-dimensional space the sequence started and where it ended.

On the other hand, when analyzing the attention matrices $\mathbf{A}_2$ and $\mathbf{A}_3$, we see that due to the larger kernel sizes $R_2 = 9$ and $R_3 = 15$ the attention tries to extract dependencies on longer scales. For example, $\mathbf{A}_2$ can be interpreted as answering the question of which aspects from the first few elements in the sequence are important for the first few elements of the output. In contrast, $\mathbf{A}_3$ seems to focus on the inverse question, namely which aspects of the beginning of the sequence are important for the end of the output sequence (and vice versa).

This way, a visualization of the attention matrix can yield additional valuable insights into what is important for the classification of sequential data and which parts the network is focusing on. In the end, the attention head tries to find descriptive element segments that are common for both classes and segments which are unique and distinguishable.

5 Conclusion

The descriptive power of eXplainable Aritficial Intelligence (XAI) has great potential to reason about complex multidimensional sequential data. Reasoning in this context is of great importance for humans who have to develop, interact or analyze processes that are described by such data. Visual Analytics (VA) as proposed by Keim et al. [6] combines approaches, such as data science, deep learning, and information visualization, in order to automate the process of knowledge generation and reasoning from complex and large datasets. However, such frameworks, especially in the domain of sequential data, still lack approaches that perform semi-supervised automatic extraction of patterns on a global and local scale.

With this work, we fill a gap of automatically finding patterns in two-dimensional projections of multidimensional sequential paths. Our approach is able to extract long- and short-term patterns without the need for strong supervision. We only need to specify the number of patterns to find. The proposed approach is easy to incorporate into existing VA frameworks, as the resulting pattern information can be readily used to enrich existing layouts (e.g., through a simple color encoding). The computational efficiency of finding long-term patterns is mostly limited by the used (dis)similarity measure. Replacing the traditional Dynamic Time Warping (DTW) measure with more efficient approximations such as FastDTW [13] can speed up the long-term pattern extraction. The short-term pattern extraction is limited by the training time of the proposed self-attention sequence classification model. However, the slim design of the network with less than 2000 parameters, and the low amount of epochs needed for training, make it feasible to train the network on a simple desktop CPU, without the need for utilizing a cost and energy-intensive GPU. Therefore, our proposed pattern analysis pipeline remains accessible to a wide range of users. Future work will focus on evaluating the methodology on real-world data.

Acknowledgments

This work was partly supported by the Austrian Federal Ministry for Digital and Economic Affairs, the Austrian Science Fund (FWF DFH 23–N), the National Foundation for Research, Technology and Development, and the Christian Doppler Research Association.

References

1. Ali, M., Jones, M.W., Xie, X., Williams, M.: Timecluster: dimension reduction applied to temporal data for visual analytics. Visual Computer **35**, 1013–1026 (2019). https://doi.org/10.1007/s00371-019-01673-y
2. Bach, B., Shi, C., Heulot, N., Madhyastha, T., Grabowski, T., Dragicevic, P.: Time curves: Folding time to visualize patterns of temporal evolution in data. IEEE Transactions on Visualization and Computer Graphics **22**(1), 559–568 (2016). https://doi.org/10.1109/TVCG.2015.2467851
3. Eckelt, K., Hinterreiter, A., Adelberger, P., Walchshofer, C., Dhanoa, V., Humer, C., Heckmann, M., Steinparz, C.A., Streit, M.: Visual exploration of relationships and structure in low-dimensional embeddings. IEEE Transactions on Visualization and Computer Graphics **29**(7), 3312–3326 (2023). https://doi.org/10.1109/TVCG.2022.3156760
4. Hinterreiter, A., Steinparz, C.A., Heckmann, M., Stitz, H., Streit, M.: Projection path explorer: Exploring visual patterns in projected decision-making paths. ACM Transactions on Interactive Intelligent Systems **11**(3–4), Article 22 (2021). https://doi.org/10.1145/3387165
5. Hohman, F., Kahng, M., Pienta, R., Chau, D.H.: Visual analytics in deep learning: An interrogative survey for the next frontiers. IEEE Transactions on Visualization and Computer Graphics **25**, 2674–2693 (2019). https://doi.org/10.1109/TVCG.2018.2843369
6. Keim, D., Andrienko, G., Fekete, J.D., Görg, C., Kohlhammer, J., Melançon, G.: Visual Analytics: Definition, Process, and Challenges, pp. 154–175. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70956-57
7. Lin, H., Ye, Y., Leung, K.C., Zhang, B.: A multivariate time series classification method based on self-attention. In: Pan, J.S., Lin, J.C.W., Liang, Y., Chu, S.C. (eds.) Genetic and Evolutionary Computing. pp. 491–499. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-3308-2_54
8. Mendhurwar, K., Gu, Q., Mudur, S., Popa, T.: The discriminative power of shape an empirical study in time series matching. IEEE Transactions on Visualization and Computer Graphics **24**(5), 1799–1813 (2018). https://doi.org/10.1109/TVCG.2017.2691322

9. Nonato, L.G., Aupetit, M.: Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. IEEE Transactions on Visualization and Computer Graphics **25**(8), 2650–2673 (2019). https://doi.org/10.1109/TVCG.2018.2846735
10. Rojat, T., Puget, R., Filliat, D., Ser, J.D., Gelin, R., Rodríguez, N.D.: Explainable artificial intelligence (XAI) on timeseries data: A survey. CoRR **abs/2104.00950** (2021), `https://arxiv.org/abs/2104.00950`
11. Sacha, D., Zhang, L., Sedlmair, M., Lee, J.A., Peltonen, J., Weiskopf, D., North, S.C., Keim, D.A.: Visual interaction with dimensionality reduction: A structured literature analysis. IEEE Transactions on Visualization and Computer Graphics **23**(1), 241–250 (2017). https://doi.org/10.1109/TVCG.2016.2598495
12. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing **26**(1), 43–49 (1978). https://doi.org/10.1109/TASSP.1978.1163055
13. Salvador, S., Chan, P.: Toward accurate dynamic time warping in linear time and space. Intelligent Data Analysis **11**(5), 561–580 (2007). https://doi.org/10.3233/IDA-2007-11508
14. Steinparz, C.A., Hinterreiter, A., Stitz, H., Streit, M.: Visualization of rubik's cube solution algorithms. In: Landesberger, T.v., Turkay, C. (eds.) EuroVis Workshop on Visual Analytics. The Eurographics Association (2019). https://doi.org/10.2312/eurova.20191119
15. Tang, W., Long, G., Liu, L., Zhou, T., Blumenstein, M., Jiang, J.: Omni-scale cnns: a simple and effective kernel size configuration for time series classification. In: International Conference on Learning Representations (2022), `https://iclr.cc/virtual/2022/poster/7148`
16. Theissler, A., Spinnato, F., Schlegel, U., Guidotti, R.: Explainable AI for time series classification: A review, taxonomy and research directions. IEEE Access **10**, 100700–100724 (2022). https://doi.org/10.1109/ACCESS.2022.3207765
17. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017), `https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`
18. Warren Liao, T.: Clustering of time series data—a survey. Pattern Recognition **38**(11), 1857–1874 (2005). https://doi.org/10.1016/j.patcog.2005.01.025