ExTea: An Evolutionary Algorithm-Based Approach for Enhancing Explainability in Time-Series Models

Yiran Huang(✉), Yexu Zhou, Haibin Zhao, Likun Fang, Till Riedel, and Michael Beigl

Telecooperation Office, Karlsruhe Institute of Technology, Karlsruhe, Germany
{yhuang,zhou,hzhao,fang,riedel,michael}@teco.edu

Abstract. In the expanding realm of sensor-based applications, the reliance on time-series data has surged, posing challenges in explaining the decisions of complex black-box time-series models. Existing Explainable Artificial Intelligence (XAI) approaches such as SBXAI, MCXAI and TS-MULE offer insights into these models but face limitations in generating multiple explanations, exploring time-series-specific characteristics, optimizing found cognitive blocks, and setting appropriate hyperparameters. Addressing these challenges, we introduce an EXplainable artificial intelligence method targeting Time-series model based on Evolutionary Algorithm (ExTea). ExTea conceptualizes explanations as evolving individuals and employs an innovative pyramidal structure for optimizing potential explanations, categorized into newborn, tested, and elite stages. This approach incorporates time-series characteristics into the fitness function of individual evaluation, thereby enhancing the overall explanatory power. Extensive experiments on six benchmark datasets with four target models demonstrate that the performance of ExTea significantly exceeds the state-of-the-art time-series XAI algorithms, SBXAI and MCXAI.

Keywords: explainable artificial intelligence · time-series · evolutionary algorithm · Baldwin effect

1 Introduction

The increasing prevalence of sensor-based applications across various domains, such as daily life [1,2] and industrial production [4,6], has amplified the reliance on time-series data. However, the inherent multimodal nature of time-series data [3] often leads to the creation of intricate and opaque models that present trust challenges in practical applications. Recently, several Explainable Artificial Intelligence (XAI) methods for time-series black-box models have been proposed. For instance, SBXAI [14] sheds light on how sequential structures in different cognitive blocks influence decision-making processes, utilizing a Directed Acyclic Graph (DAG). Here, the cognitive block refers to the crucial data segments for model decisions. In a similar vein, MCXAI [13] investigates relationships among

A. Bifet et al. (Eds.): ECML PKDD 2024, LNAI 14950, pp. 429–446, 2024.
https://doi.org/10.1007/978-3-031-70381-2_27

various cognitive blocks using the Monte Carlo tree structure. Another approach, TS-MULE [15], adapts LIME [17] to time-series data, employing multiple distinct segmentation strategies. While these methods make strides in elucidating black-box time-series models, they are not without limitations.

One major challenge is the generation of multiple explanations for a single input. State-of-the-art models often involve an ensemble decision mechanism [8], indicating that identical inputs can be governed by multiple underlying rules. These rules are typically manifested through internal model mechanisms, such as the ensemble approach in a random forest model and the dropout mechanism in a neural network [5]. Prevailing XAI methods, which focus on feature importance and Pertinent Negative counterfactual [18], tend to combine all rules into a singular explanation, leading to potential confusion and inaccuracy. Moreover, most Pertinent Positive counterfactual-based methods [18], which are yet to be tested on time-series data, offer only one optimal explanation, disregarding the range of possible decision rules. Secondly, certain time-series characteristics, like frequency information or sequential interference, are not adequately explored. Thirdly, the optimization of cognitive blocks in MCXAI and SBXAI is under strong constraint, which can impede the discovery of high-quality cognitive blocks. Furthermore, setting hyperparameters in current methodologies, such as the number of segments in SBXAI, remains a formidable challenge.

In summary, we encounter a counterfactual-based XAI task pursuing multiple, distinct, and optimized explanations under a multi-objective (pertaining to different time-series characteristics) constraint. We posit that evolutionary algorithms [23] are well-suited to tackle these challenges. Consequently, we propose ExTea, an EXplainable artificial intelligence method targeting time-series model based on Evolutionary Algorithm (ExTea). In ExTea, each individual represents a potential explanation and is endowed with a self-optimization function. Our method diverges from traditional evolutionary algorithms by employing a pyramidal structure for the individual pool, segmented into layers for newborn, tested, and elite individuals. This structure facilitates differentiated optimization across layers and is tailored to multi-objective tasks with clear rejection criteria. Additionally, we integrate explanatory factors into the fitness function, thereby enhancing the explanatory power of selected individuals.

The key contributions of this paper are twofold: *(i)* the introduction of ExTea, a model-agnostic XAI algorithm for time-series, which advances the search for high-quality cognitive blocks and further investigates the time-series-specific characteristics of these blocks; *(ii)* extensive validation of ExTea's effectiveness through experiments conducted on six benchmark datasets using four target models.

2 Related Work

In the realm of time-series analysis, model-agnostic explanation methods can be broadly classified into three types based on their foundational units of explanation: time-point-based, subsequence-based, and instance-based (or

feature-based) explanations. Each type has distinct approaches and limitations: SoundLime [19] generates new samples by introducing minor perturbations to the original audio data. The importance of each time point is assessed based on the model's predictions for these altered samples. Tsinsight [20] employs an auto-encoder trained on the training dataset and explains the input with the reconstructed data. Salience cam [21] generates a salience map based on the gradients of the model's output with respect to the input data and uses it as an explanation for the decision. While these approach effectively determines the significance of individual time points, it falls short in exploring time-related features in the input, such as frequency and trend. These features often require an analysis that integrates data across multiple time points.

TS-MULE [15] assesses the importance of each cognitive block by constructing local linear models. The generation of cognitive blocks from the sequences is done using different methods including Symbolic Aggregate approXimation (SAX). In SAX-VSM [22], time-series data is segmented using SAX with overlapping windows, and a bag-of-words model is trained based on these segments. The input is explained with the generated 'word'. Both MCXAI [13] and SBXAI [14] provide insights into the relationships between cognitive blocks, with MCXAI focusing on spatial relationships through a tree structure and SBXAI on temporal relationships via a DAG. However, these methods do not sufficiently explore the temporal features of the data. Additionally, the explanations they offer may merge multiple rules, which complicates understanding.

Instance-based methods extract features using statistical techniques. The explanation of these methods largely depends on the interpretability of the features themselves. This necessitates that the model must rely exclusively on interpretable features for decision-making. However, the process of feature extraction inherently leads to a loss of information, which can significantly limit the model's performance.

In summary, while each of these model-agnostic explanation methods offers valuable insights in the context of time-series analysis, they also have inherent limitations. Time points-based methods may neglect broader temporal patterns, subsequence-based methods might not fully capture temporal dynamics, and instance-based methods could suffer from information loss due to feature extraction.

3 Method

3.1 Problem Definition and Individual Coding

Given a block-box model B and an input $o = [o_1, \cdots, o_l]$, where l is the length of the input signal, the objective of local model-agnostic time-series explanation method is to identify a set of masks $\mathcal{M}^o = [m_1, \cdots, m_i, \cdots]$ with $m_i \in \mathcal{M}$ and $\mathcal{M} = \langle 0, 1 \rangle^l$. These masks should highlight the most critical data points that influence the model prediction. The identified mask set $\mathcal{M}^o$ must satisfy the following conditions:

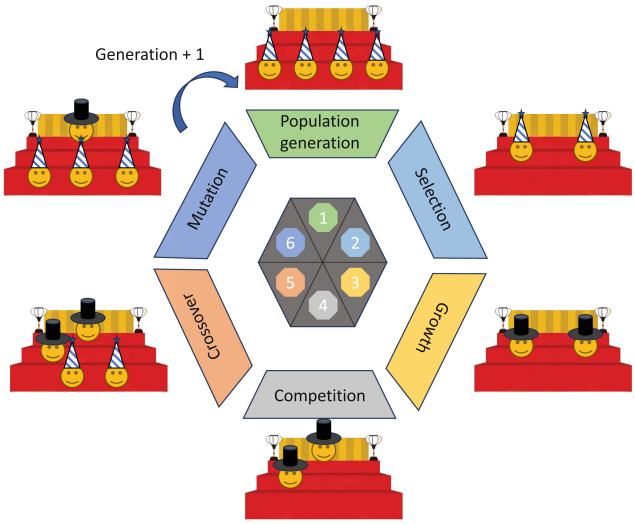

Fig. 1. The pipeline of the proposed ExTea algorithm.

– **Prediction Consistency**:

$$\forall_{m \in \mathcal{M}^o},\ B(m(o)) = B(o), \tag{1}$$

where function $m(o)$ involves assigning 0 to all data in the input o, except at positions marked with 1 by the mask m. The function $B(\cdot)$ yields the prediction of the black-box model.

– **Non-subset relation**:

$$\forall_{m_i, m_j \in \mathcal{M}^o,\ i \neq j},\ m_i \not\subseteq m_j..$$

We define $m_i \subseteq m_j$ if and only if $m_i \wedge m_j = m_i$. This criterion ensures that no mask in $\mathcal{M}^o$ is completely encompassed by another, thus guaranteeing unique contributions from each individual mask.

– **Minimalism**:

$$\nexists_{m \in \mathcal{M} \setminus \mathcal{M}^o} \left[B(m(o)) = B(o) \text{ and } \exists_{m_i \in \mathcal{M}^o}\ m \subseteq m_i \right]. \tag{2}$$

This condition ensures the identified masks are the simplest possible.

As mentioned in Sect. 2, time point-based explanation methods fail to capture the nuanced character of time-series data. To this end, in ExTea, instead of using a boolean array, we represent the mask as a list with $2n$ numerical values, where n is the number of blocks in the mask where all values are set to 1. The list should satisfy the following criteria: *(i)* Each number must be a unique integer, and its value should be less than the length l of the input sequence. *(ii)* The numbers must be sorted in ascending order. Each adjacent pair $\{2i, 2i+1\}$ in the list denotes the i-th block in the mask.

Since a numerical list and a mask can be converted to each other, for simplicity we denote the list also by m. In ExTea, each individual is represented with m and, $m(o)$ signifies the cognitive blocks that explain the model decision for input o. In this way, we force the explanation to be formed of subsequences. This approach facilitates the optimization and exploration of individuals.

For data with multiple channels, signals from different channels are concatenated to form a single-dimensional signal. In this scenario, the length l corresponds to the length of this concatenated signal, simplifying the representation and analysis of multi-dimensional data.

3.2 Population Generation

The proposed algorithm involves a hierarchical structure with three distinct layers for managing the population pool: L_1, L_2, and L_3. Each layer serves a specific purpose in the selection and evolution of individuals, ensuring an efficient and structured progression of candidates through the system. Layer L_3 forms the gateway for all newly created individuals, acting as the initial staging ground for new candidates. Individuals from L_3 are promoted to L_2 upon satisfying the explicit rejection condition, detailed in Eq. 1 (Prediction Consistency). This layer acts as a filter, ensuring only candidates that meet basic criteria advance further. The transition from L_2 to the elite layer L_1 is competition-based. This layer is reserved for the most promising solutions, fostering a focused development of superior candidates.

The generation of new individuals within this system is initiated through a random sampling process. This process begins with the generation of a random number, dictating the number of blocks in the individual. Subsequently, a set of unique random integers, double the number of blocks, is selected from the range $[0, l]$. These values are then organized in ascending order to form a numerical list, representing an individual.

We replenish individuals in layer L_3 in accordance with its capacity s_3. This ensures a consistent influx of new candidates into the system, maintaining the diversity and dynamism of the population pool.

3.3 Fitness Function Design

ExTea, adapted to the hierarchical structure of the individual pool, divides the selection process of the general evolutionary algorithm into two distinct processes: selection and competition. Each process targets different layers within the system and uses specific criteria for evaluating individuals.

Selection Process. This process is dedicated to individuals in Layer L_3. The fitness function for selection is defined as follows:

$$f_{sel} = \mathbb{1}_{\{B(m(o))=B(o)\}} \times 2 - 1,$$

where $\mathbb{1}$ denotes the indicator function, B the black-box model, o the target of analysis, and m the mask of the individual. Only those individuals, which

masked information (cognitive blocks) holds the same prediction as the original data are selected to Layer L_2 for further optimization. Individuals failing this process pose evaluation challenges due to relative performance ambiguity and are therefore immediately eliminated, not advancing to subsequent processes.

Competition Process. The competition process in ExTea is designed for individuals in L_2 and L_1. It aims to select superior individuals for promotion to L_1, demoting those in L_1 that fail to meet the competition standards. Evaluation in this process is based on two main criteria:

Cognitive Block Length: The algorithm hypothesizes an inverse correlation between an individual's importance and the length of its cognitive blocks. Shorter blocks imply higher significance and facilitate easier comprehension.

Purity of Influencing Factors: The ideal individual should be influenced by as few factors as possible, enhancing the purity of its explanation. We investigate the impact of various elements like time dependence, location, and frequency information on the model's decision-making process. A purer, less influenced explanation is deemed superior for clarity and understanding.

In ExTea, several explorations based on basic time-series characteristics are conducted, each evaluated separately:

Sequential Relationship: This exploration assesses the impact of the sequential relationship between cognitive blocks on the model's decision-making by altering block positions and observing changes in model predictions. The scoring function f_1 is defined as the proportion of block pairs influencing the decision:

$$f_1 = \frac{2}{n(n-1)} \sum_{\substack{i,j \in \{0,\ldots,n\} \\ i \neq j}} \mathbb{1}_{\{B(c_i^j(m(o))) \neq B(o)\}},$$

where n is the number of cognitive blocks and $c_i^j(o)$ denotes the swapping operation of the i-th and j-th blocks.

Low-Frequency Information: The importance of low-frequency information in cognitive blocks is evaluated by applying a Butterworth high-pass filter to isolate the low-frequency information in the cognitive blocks. The scoring function f_2 of this exploration is defined as whether the cognitive blocks after the filtering retain the original prediction:

$$f_2 = \mathbb{1}_{\{B(b_h(m(o))) \neq B(o)\}},$$

where $b_h(\cdot)$ represents the Butterworth high-pass filtering operation.

High-Frequency Information: Similarly, a Butterworth low-pass filter is used to assess the role of high-frequency information, with the scoring function f_3 formulated as:

$$f_3 = \mathbb{1}_{\{B(b_l(m(o)))\neq B(o)\}},$$

where $b_l(\cdot)$ signifies the Butterworth low-pass filtering operation.

Numerical Trends: By mirroring values within cognitive blocks, we evaluate the influence of numerical trends on model decisions. The scoring function f_4 of this exploration is defined as:

$$f_4 = \frac{1}{n} \sum_{i\in\{0,\ldots,n\}} \mathbb{1}_{\{B(v_i(m(o)))\neq B(o)\}},$$

where function $v_i(\cdot)$ indicates mirroring the data in the i-th cognitive block and n signifies the total number of cognitive blocks.

Blocks Relative Position: We explore the effect of changing block positions on the model prediction through shifting each block forward and backward separately. The scoring function f_5 of this exploration is defined as:

$$f_5 = \frac{1}{2n} \sum_{\substack{j\in\{-d,d\}\\ i\in\{0,\ldots,n\}}} \mathbb{1}_{\left\{B(s_i^j(o,m))\neq B(o)\right\}},$$

where n is the number of segments in the individual, d is a variable that signifies the distance to the neighbor blocks or border of the time-series, and the function $s_i^j(\cdot)$ means shift the i-th block with j distance.

Block Position: We explore the effect of changing all the positions of cognitive blocks synchronously both forward and backward until any of the block reaching the series boundary. If the prediction holds, we set the score f_6 to zero, otherwise one.

Decision Intervals: We examine the extent of numerical adjustment permissible at each point in the cognitive block without altering the model's prediction with the Reinforce method as described in [24]. Due to the high time consumption, this exploration is only executed before the algorithm returns the final results.

To achieve the desired level of explanation purity, ExTea calculates the mean of the first six exploration scores. The smaller the mean, the fewer the factors affecting the explanation and the clearer and more concise the corresponding explanation. Moreover, ExTea gives preference to explanations with smaller

block sizes aligning with the assumption that simpler explanations are often more effective. The competition score f_{comp}, is formulated to reflect these priorities:

$$f_{comp} = -\left[\frac{\sum m}{|m|} + \lambda \times \frac{f_1 + f_2 + f_3 + f_4 + f_5 + f_6}{6}\right], \tag{3}$$

where λ denotes the balance weight, the function $\text{sum}(m)$ calculates the sum of the mask m and the function $\text{len}(m)$ returns the length of the mask.

During each generation, individuals in Layers L_1 and L_2 undergo evaluation using Eq. 3. The top-scoring individuals, up to the capacity s_1 of Layer L_1, are then promoted to this layer. This strategy ensures that only the most refined and suitable candidates ascend to the elite layer, maintaining a high standard of quality within the population pool.

It is important to note that after the competition process, a thorough validation of minimalism 2 in the L_1 layer is executed, removing those duplicated individuals. This procedure guarantees the diversity of individuals in L_1, ensuring a distinct representation within that level.

3.4 Growth

The growth stage, targeting individuals at layers L_1 and L_2, sequentially follows the Selection process and precedes the Competition process in each generational cycle. This stage addresses the challenge that randomly generated individuals often harbor superfluous information in their cognitive blocks. The primary objective of the growth is to refine these individuals, ensuring maximal succinctness. This is achieved by systematically eliminating non-essential information, thereby narrowing down each cognitive block in the individual. It is imperative to recognize that pinpointing the minimal requisite set of explanations within the original dataset constitutes an NP-hard problem. Restricting the optimization to the boundaries of cognitive blocks, the search space remains vast. For an individual with n blocks, each of length h, the total number of potential reductions can be approximately quantified as $2nh^2$. Given the time-intensive nature of exhaustively exploring these possibilities we introduced the growth function to streamline this process. It traverses the mask sequentially, and when the boundary of a block is recognized, the growth kernel size u values on the boundary are set to 0 according to a given growth probability α, thus shrinking the corresponding block. The growth kernel size u signifies the unit to narrow the blocks, and growth rate α signifies the possibility. After the growth, the reduced mask is validated using black-box B. If the prediction remains consistent, the alteration is retained; otherwise, it is revoked. In each generation, this process is repeated predefined times.

3.5 Crossover and Mutation

ExTea implements a two-layer crossover mechanism, comprising inner-layer and inter-layer crossovers. Initially, all individuals in the L_1 layer undergo pairing among themselves (inner-layer crossover). Subsequently, L_1 individuals are

paired with those in the L_2 layer, facilitating the inter-layer crossover. It is important to note that not every pair experiences the crossover process; it occurs with a probability dictated by the crossover ratio β. This crossover process employs the half-swap strategy, a prevalent technique in evolutionary algorithms. This strategy involves the exchange of half of the genetic material between two individuals, thereby inducing diversity within the population.

Individuals in the L_1 layer undergo the mutation process, characterized by a mutation ratio γ in ExTea. This process is notable for its distinctive approach of splitting original cognitive blocks. This splitting plays a crucial role because the growth stage, by its inherent design, only removes irrelevant information at the boundary of each block. As a result, when superfluous information is embedded centrally within a block, the growth process alone proves inadequate for its extraction. To overcome this limitation, we integrate the concept of block splitting, a new form of mutation. This method facilitates the removal of superfluous information from any section of the block, increasing the algorithm's capacity to optimize individuals effectively.

3.6 Explanation

In this subsection, we present a demonstrative example of the ExTea algorithm. This demonstration aims to address two primary questions: *(i)* What insights can be gleaned from proposed method *(ii)* How might these insights be applied?

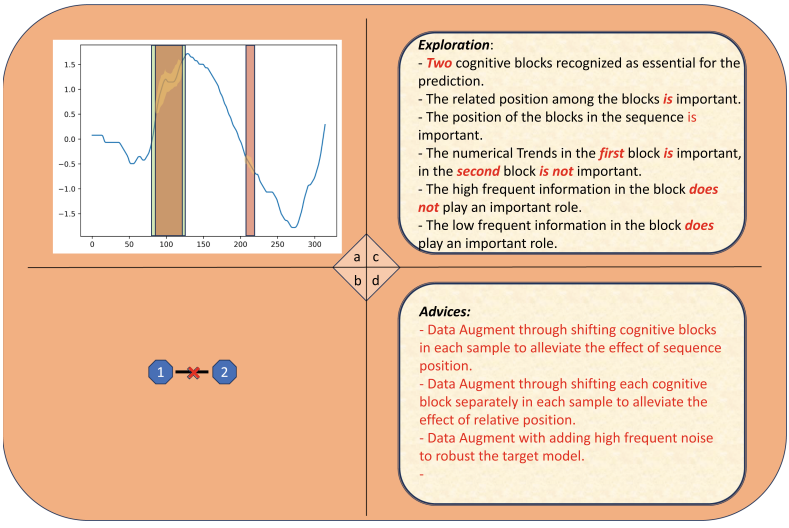

Fig. 2. An example of the ExTea explanation.

Figure 2 illustrates an exemplary ExTea explanation applied to the UWaveGestureLibraryX Dataset [12]. The target black-box model is a random forest

model with default parameter settings from the scikit-learn package [16]. The explanation includes two images and two textual descriptions. From Fig. 2a, the following insights are discerned: *(i)* Cognitive Blocks Influencing Decision-Making: The areas highlighted in red denote the cognitive blocks that significantly influence model decisions. *(ii)* Impact of Cognitive Blocks' Relative Position: The cyan regions surrounding the cognitive blocks represent their permissible movement range. Movement within these zones does not modify the model's decisions. Importantly, this analysis evaluates each cognitive block independently, resulting in a unique cyan area for each element. *(iii)* Permissible Variability within Cognitive Blocks: Within each cognitive block, the blue line depicts the original data value, and the surrounding orange zone indicates the allowable fluctuation range.

Figure 2b explores the effect of interchanging these cognitive blocks on the decision-making process. Each node represents a cognitive block. The number in the node indicates the position from left to right of the cognitive block in Fig. 2a. An edge connecting two cognitive blocks implies that swapping their positions does not alter the model's predictions. Conversely, a cross symbol above an edge indicates that reordering these blocks impacts the decision outcome.

In instances where visual representation of exploratory findings is impractical, we adopt a rule-based methodology to produce descriptive text, as exemplified in Fig. 2c. This technique integrates static text (black) with dynamic text (red) within the text panel, with the dynamic component varying in response to the exploration result. Additionally, this approach reinforces findings that are initially presented visually in images, thereby augmenting user understanding. ExTea encompasses seven distinct exploratory analyses, each aligned with a specific rule. Furthermore, based on the exploratory outcomes, we offer recommendations for improving model performance in Fig. 2d. These suggestions are triggered when the exploration results meet specific requirements. Specific rules and code are available on https://github.com/HuangYiran/extea.

The original signal data depicted in Fig. 2a are captured by the accelerometer during a clockwise circle drawing. According to the cognitive blocks found, ExTea explains this action as an increase in acceleration in the positive direction, followed by a decrease in the negative direction, closely mirroring human cognition. Remarkably, inverting the sequence order of the acceleration information alters the categorization from clockwise to counterclockwise circle drawing, underscoring the significance of sequential order in cognitive processing, as corroborated by the visualized results.

Contrary to our intuitive understanding, however, the model's decision-making is influenced by variations in the acceleration trend. This discrepancy may stem from the homogeneity in acceleration changes in the collected data. Ideally, frequent acceleration variations during circle drawing should not alter the model's inference. To address this, Data Augmentation (DA) can be employed to create samples with diverse acceleration patterns, thereby enhancing the model's resilience to such variations.

Furthermore, the analysis indicates that minor positional shifts between cognitive blocks can also unexpectedly influence model predictions, contradicting conventional knowledge. This challenge can also be mitigated through the strategic use of DA, further refining the accuracy and robustness of the model.

4 Experiment

To conduct a thorough assessment of our proposed methodology, we structured three comprehensive experiments aimed at addressing the pivotal questions: *(i)* Fidelity of the Explanation to the Original Model: This aspect evaluates the extent to which the explanations produced by our approach are representative of the intrinsic mechanisms of the original model. *(ii)* Role of Algorithm Components: This investigates the specific contributions of different components of the proposed algorithm to its overall effectiveness. *(iii)* The influence of the algorithm parameters.

4.1 Experiment Setup

Table 1. Summary of the datasets used in the experiments.

Dataset	Type	Train size	Test size	Sequence length	Number classes
EthanolLevel	Spectro	504	500	1751	4
ECG5000	ECG	500	4500	140	5
ElectricDevices	Device	8926	7711	96	7
InsectWingBeatSound	Audio	25000	25000	256	10
EOGVerticalSignal	EOG	362	362	1250	12
UWaveGestureLibraryX	HAR	896	3582	315	8

Benchmark Dataset. We meticulously chose six datasets from varied domains to showcase the adaptability and broad applicability of our proposed method: *(i)* Ethanol Level Dataset [7]: Originating from the Scotch Whisky Research Institute, this dataset is instrumental in the non-invasive detection of counterfeit spirits. *(ii)* ECG5000 Dataset [9]: As a segment of the BIDMC Congestive Heart Failure Database, this dataset presents Elektrokardiographie (ECG) recordings from a 48-year-old patient diagnosed with severe congestive heart failure. *(iii)* ElectricDevices Dataset: This dataset chronicles the electricity consumption patterns of 251 households. Data was recorded bi-minutely over a one-month period, with each sequence encapsulating a full day's electricity usage. *(iv)* InsectWingBeatSound Dataset [10]: Comprising sound recordings from 5,000 individual insects, this dataset is unique in its acoustic approach. Each recording

is tagged with the corresponding insect species, captured via specialized sensors. *(v)* EOGVerticalSignal Dataset [11]: Utilizing the BlueGain biomedical amplifier, this dataset collects Electrooculography (EOG) signals, which measure the potential difference between electrodes situated near the eye. It features signals from 12 participants, each representing 12 different Japanese katakana characters through eye movements. *(vi)* UWaveGestureLibraryX Dataset [12]: Designed for Human Activity Recognition (HAR) tasks, this dataset aggregates eight distinct gesture patterns from eight users, compiled over a month. A comprehensive summary of these datasets is provided in Table 1.

Target Models. To rigorously evaluate our proposed methodology, experiments were conducted using four distinct models, encompassing both transparent ('white-box') and opaque ('black-box') methodologies. This diverse model selection was instrumental in assessing the robustness and adaptability of our approach across various computational frameworks. The models employed include: *(i)* interpretable 'white-box' models such as Decision Tree (DT) and Support Vector Machine (SVM); and *(ii)* 'black-box' models, namely Random Forest (RF) and Neural Network (NN).

Benchmark Algorithms. To evaluate the effectiveness, we conducted a comparative analysis with two state-of-the-art (SOTA) XAI methods: MCXAI [13] and SBXAI [14] and one popular XAI methods: LIME [17].

Experiments Design. Three distinct experiments were designed. The first aims to establish the fidelity of our proposed method to the target model, measuring the precision and efficiency of various XAI methods in identifying critical features relied upon by the target model for predictions. This experiment comprises three stages: *(i)* Model Training: For each dataset, the target models are trained using the training set. *(ii)* Sampling and Interpretation: 100 samples are randomly selected from the test set. Each sample is explained using the XAI method to determine the importance of each data point. *(iii)* Reconstruction and Validation: A blank sample is generated and incrementally populated with data points from the original sample, prioritized by their determined importance. The objective is to mirror the target model's prediction with minimal data points. The efficacy of an XAI method is thus judged by the fewest data points required for accurate decision replication, indicating a higher interpretative fidelity.

The second experiment, an ablation study, investigates the impact and effectiveness of the growth function in our proposed method. It contrasts the performance of the method with and without the growth function, monitoring the most proficient individual identified in each generation.

The last experiment was designed to investigate the effect of the algorithm's parameters by comparing the performance of the algorithm under different parameter settings. The experiment also provides guidelines for the algorithm usage.

Parameter Setting. In the first two experiments and the base group in the third experiment, the number of blocks n is set to 5, the growth kernel size u is set to 2. The growth rate is set to 0.5. In each generation, the growth process is repeated 5 times. All the ratios α, β, γ are set to 0.1. The balance parameter λ is set to 0.1. Layer size s_1 is set to 3, s_2 is set to 10 and s_3 is set to 50. The initial population size was the same as that of L_3, i.e., 50. The maximal number of generation is set to 20.

4.2 Evaluation

The results of the first experiment are summarized in Table 2. The values therein represent the ratio of the number of data points required for accurate model prediction to the total sequence length. A lower ratio is indicative of

Table 2. Ratio of information needed to support the model decision, the smaller the better. The bold numbers denote the smallest ratio in the corresponding groups.

Dataset	Target Model	LIME	MCXAI	SBXAI	Proposed
EthanoLevel	DT	0.26	**0.02**	0.27	**0.02**
	SVC	0.49	**0.02**	0.49	**0.02**
	RF	0.21	0.04	0.40	**0.03**
	NN	0.44	**0.04**	0.22	**0.04**
ECG5000	DT	0.23	0.06	0.23	**0.01**
	SVC	0.25	0.03	0.21	**0.01**
	RF	0.26	0.05	0.27	**0.03**
	NN	0.41	0.06	0.39	**0.05**
ElectricDevices	DT	0.54	0.11	0.24	**0.10**
	SVC	0.39	0.10	0.39	**0.08**
	RF	0.50	0.27	0.29	**0.14**
	NN	0.51	**0.13**	0.45	**0.13**
InsectWingBeatSound	DT	0.50	0.06	0.51	**0.02**
	SVC	0.40	0.05	0.44	**0.02**
	RF	0.49	0.23	0.58	**0.03**
	NN	0.38	0.13	0.34	**0.02**
EOGVerticalSignal	DT	0.54	0.05	0.44	**0.02**
	SVC	0.39	0.05	0.65	**0.01**
	RF	0.50	0.06	0.45	**0.05**
	NN	0.51	**0.04**	0.57	0.06
UWaveGestureLibraryX	DT	0.43	0.06	0.46	**0.05**
	SVC	0.51	**0.15**	0.54	0.18
	RF	0.49	0.19	0.54	**0.12**
	NN	0.55	0.13	0.45	**0.10**

a more efficient identification of crucial data points. The results demonstrate that our proposed method significantly surpasses other methods across various datasets and models. This superiority not only attests to the efficacy of our method but also suggests that it yields interpretations closely aligned with the model's inner workings, effectively acting as a localized surrogate for the original model's explanations.

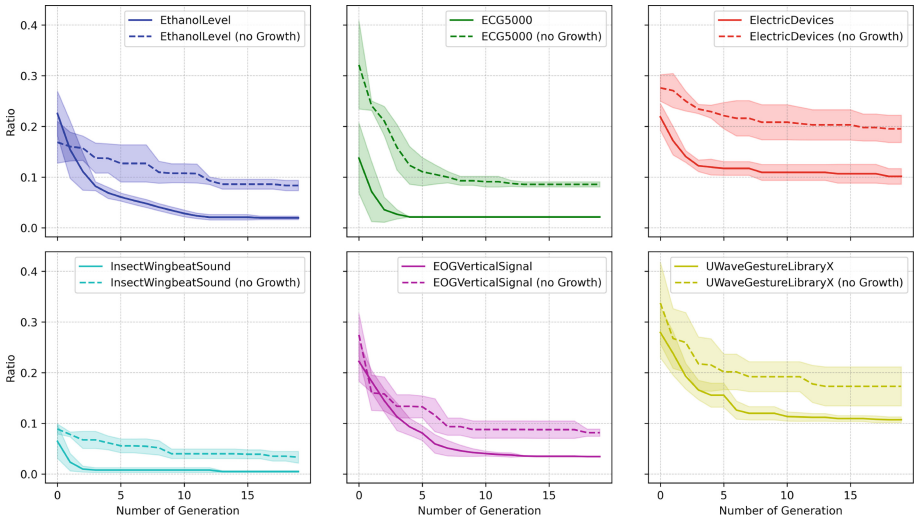

Fig. 3. Comparison of ExTea performance with and without growth processes. The x-axis represents the number of generation (epoch) and the y-axis indicates the ratio between the final cognitive block length and the input sequence length.

The result of the first experiment is to be expected, given its more flexible approach in identifying cognitive blocks. MCXAI is limited by its use of a splitting method to optimize cognitive blocks, while SBXAI's optimization is indirect, relying on simulating the original model. LIME is potential for optimal results due to its focus on data points. However, its relatively simplistic local modeling approach may struggle with complex time-series data, which could be the reason for its under-performance.

Figure 3 delineates the influence of the growth process on experimental results across varied datasets, using the Random-Forest model as the target black-box model. The inclusion of the growth process distinctly enhances both the convergence speed and quality. The ExTea algorithm, with the growth process, converges more rapidly and effectively than its counterpart without it. Notably, ExTea converges by the 3rd generation in three datasets, the 7th generation in two, and the 11th in one, indicating dataset-specific generational requirements, with most converging within ten generations. The results of ExTea in the first generation are on par with those of SBXAI and LIME.

Figure 4 illustrates the impact of different parameter settings on algorithm performance. Numerical values next to the straight lines indicate the range of parameter values explored during the experiments, while the violin plots are generated based on the number of times each parameter value achieved the best performance. Our observations reveal that larger quantities of generations and L_1 sizes tend to yield better outcomes. Specifically, the former offers more opportunities for optimization, whereas the latter retains a greater number of high-quality individuals within each generation. Conversely, smaller kernel sizes are preferable, as they provide higher optimization granularity. However, increasing the number of generations and L_1 size, along with decreasing the kernel size, results in longer algorithm runtime. The specific settings should be adjusted based on the application context. Notably, a higher optimization per round is not always beneficial, which may be attributed to its propensity to eliminate individuals that perform well initially but falter in later stages. This aspect warrants further investigation. The choice of the ratio largely depends on the specific dataset, but best performance generally ranges between 0.2 and 0.3.

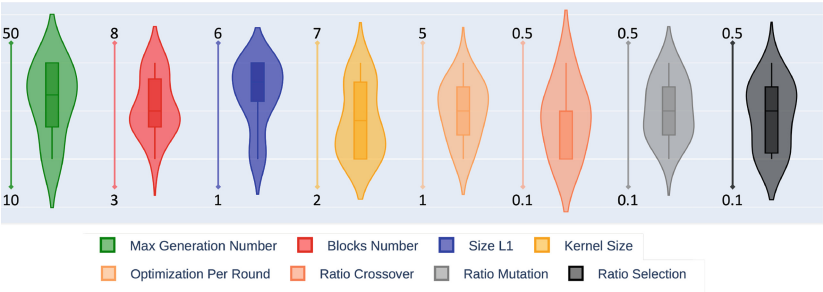

Fig. 4. Comparison of ExTea performance with different parameter setting. The line next to the violin describes the value range of the corresponding parameter. The violin describes the kernel density estimate of achieving the best result.

The execution time of the ExTea algorithm is influenced by several factors, including algorithmic parameters, data sample shape, the complexity of the target black-box model and hardware specifications. For instance, using the ECG5000 dataset and the Random-Forest model, the average duration for one generation, in the absence of parallelization, is roughly five seconds. In contrast, MCXAI's execution time is heavily influenced by whether it delves into the structural relationships among cognitive blocks post-identification. Without this additional step, MCXAI achieves results in an average time of five seconds. SBXAI, meanwhile, requires about 0.5 s to evaluate a single candidate subgroup and approximately 26 s for 50 evaluations to generate explanations, which can be attributed to the intensive parameter settings of the Bayesian optimization's objective function, particularly its default high number of cross-validations.

5 Conclusion

This paper introduces a novel, model-agnostic algorithm, ExTea, designed for elucidating time-series black-box models. ExTea innovatively employs an evolutionary algorithm, conceptualizing explanations as evolving individuals within a growth-oriented optimization process. Central to this method is a bespoke fitness function, tailored to the nuances of time-series data, which directs the algorithm's search space navigation. A distinctive feature of ExTea is its hierarchical pool of individuals, a structure that stratifies individuals across different layers, fostering more efficient model exploration. The algorithm addresses several challenges in the realm of time-series model explanation. These include decomposing complex, hybrid interpretations, ensuring adaptability to varied time-series characteristics, navigating temporal domain features, and managing intricate parameter configurations.

By tackling these challenges, ExTea advances the field of time-series model explanation. The empirical experiments on six datasets from different domains demonstrate that, ExTea offers a more effective and efficient framework for understanding the predictions of time-series black-box models, which is particularly valuable in domains where explainability and transparency are crucial.

Acknowledgment. This work was partially funded by the German Ministry for Research and Education as part of SDI-S (Grant 01IS22095A) as well as by the Ministry of Science, Research and the Arts Baden-Wuerttemberg as part of the SDSC-BW project and by the Carl-Zeiss-Foundation as part of the JuBot project.

References

1. Zhou, Y., et al.: AutoAugHAR: automated data augmentation for sensor-based human activity recognition. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. **8**(2), 1–27 (2024)
2. Huang, Y., Zhou, Y., Zhao, H., Riedel, T., Beigl, M.: A survey on wearable human activity recognition: innovative pipeline development for enhanced research and practice. In: 2024 IEEE International Joint Conference on Neural Networks (IJCNN 2024), Yokohama, 30th June-5th July 2024 (2024)
3. Zhou, Y., Zhao, H., Huang, Y., Riedel, T., Hefenbrock, M., Beigl, M.: TinyHAR: a lightweight deep learning model designed for human activity recognition. In: Proceedings of the ACM International Symposium on Wearable Computers, pp. 89–93 (2022)
4. Zhao, H., Pal, P., Hefenbrock, M., Beigl, M., Tahoori, M.B.: Towards temporal information processing–printed neuromorphic circuits with learnable filters. In: Proceedings of the 18th ACM International Symposium on Nanoscale Architectures, pp. 1–6 (2023)
5. Zhou, Y., et al.: Enhancing efficiency in HAR models: NAS meets pruning. In: 2024 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), pp. 33–38. IEEE (2024)

6. Zhou, Y., Hefenbrock, M., Huang, Y., Riedel, T., Beigl, M.: Automatic remaining useful life estimation framework with embedded convolutional LSTM as the backbone. In: Dong, Y., Mladenić, D., Saunders, C. (eds.) ECML PKDD 2020. LNCS (LNAI), vol. 12460, pp. 461–477. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-67667-4_28
7. Lines, J., Taylor, S., Bagnall, A.: HIVE-COTE: the hierarchical vote collective of transformation-based ensembles for time series classification. In: 2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 1041–1046. IEEE (2016)
8. Huang, Y., Zhou, Y., Riedel, T., Fang, L., Beigl, M.: randomHAR: Improving Ensemble Deep Learners for Human Activity Recognition with Sensor Selection and Reinforcement Learning (2023). arXiv preprint arXiv:2307.07770
9. Chen, Y., Hao, Y., Rakthanmanon, T., Zakaria, J., Hu, B., Keogh, E.: A general framework for never-ending learning from time series streams. Data Min. Knowl. Discov. **29**(6), 1622–1664 (2014). https://doi.org/10.1007/s10618-014-0388-4
10. Chen, Y., Why, A., Batista, G., Mafra-Neto, A., Keogh, E.: Flying insect classification with inexpensive sensors. J. Insect Behav. **27**(5), 657–677 (2014). https://doi.org/10.1007/s10905-014-9454-4
11. Fang, F., Shinozaki, T.: Electrooculography-based continuous eye-writing recognition system for efficient assistive communication systems. PloS One **13**(2), e0192684 (2018). Public Library of Science San Francisco, CA USA
12. Liu, J., Zhong, L., Wickramasuriya, J., Vasudevan, V.: uWave: accelerometer-based personalized gesture recognition and its applications. Pervasive Mob. Comput. **5**(6), 657–675 (2009). Elsevier
13. Huang, Y., Schaal, N., Hefenbrock, M., Zhou, Y., Riedel, T., Beigl, M.: McXai: local model-agnostic explanation as two games. In: 2023 International Joint Conference on Neural Networks (IJCNN), pp. 01–08. IEEE (2023)
14. Huang, Y., Li, C., Lu, H., Riedel, T., Beigl, M.: State graph based explanation approach for black-box time series model. In: Longo, L. (eds.) Explainable Artificial Intelligence. xAI 2023. Communications in Computer and Information Science, vol. 1903. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-44070-0_8
15. Schlegel, U., Vo, D.L., Keim, D.A., Seebacher, D.: TS-MULE: local interpretable model-agnostic explanations for time series forecast models. In: Kamp, M., et al. Machine Learning and Principles and Practice of Knowledge Discovery in Databases. ECML PKDD 2021. Communications in Computer and Information Science, vol. 1524. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-93736-2_1
16. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
17. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you? Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016)
18. Verma, S., Boonsanong, V., Hoang, M., Hines, K.E., Dickerson, J.P., Shah, C.: Counterfactual explanations and algorithmic recourses for machine learning: a review (2020). arXiv preprint arXiv:2010.10596
19. Mishra, S., Benetos, E., Sturm, B.L.T., Dixon, S.: Reliable local explanations for machine listening. In: 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2020)
20. Siddiqui, S.A., Mercier, D., Dengel, A., Ahmed, S.: TSInsight: a local-global attribution framework for interpretability in time series data. Sensors **21**(21), 7373 (2021). MDPI

21. Zhou, L., Ma, C., Shi, X., Zhang, D., Li, W., Wu, L.: Salience-CAM: visual explanations from convolutional neural networks via salience score. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2021)
22. Senin, P., Malinchik, S.: SAX-VSM: interpretable time series classification using sax and vector space model. In: International Conference on Data Mining, pp. 1175–1180. IEEE (2013)
23. Eiben, A.E., Smith, J.E.: What is an evolutionary Algorithm?. In: Introduction to Evolutionary Computing. Natural Computing Series. Springer, Berlin, Heidelberg (2003). https://doi.org/10.1007/978-3-662-05094-1_2
24. Huang, Y., Zhou, Y., Hefenbrock, M., Riedel, T., Fang, L., Beigl, M.: Universal distributional decision-based black-box adversarial attack with reinforcement learning. In: Tanveer, M., Agarwal, S., Ozawa, S., Ekbal, A., Jatowt, A. (eds.) Neural Information Processing. ICONIP 2022. LNCS, vol. 13625. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-30111-7_18