Explaining Memristive Reservoir Computing Through Evolving Feature Attribution

Xinming Shi, Zilu Wang
Southern University of Science and Technology, China
University of Birmingham, UK
xxs972@student.bham.ac.uk
Southern University of Science and Technology, China
wangzl@sustech.edu.cn

Leandro L. Minku* and Xin Yao*
University of Birmingham, UK
l.l.minku@bham.ac.uk
Southern University of Science and Technology, China
University of Birmingham, UK
xiny@sustech.edu.cn

ABSTRACT

Memristive Reservoir Computing (MRC) is a promising computing architecture for time series tasks, but lacks explainability, leading to unreliable predictions. To address this issue, we propose an evolutionary framework to explain the time series predictions of MRC systems. Our proposed approach attributes the feature importance of the time series via an evolutionary approach to explain the predictions. Our experiments show that our approach successfully identified the most influential factors, demonstrating the effectiveness of our design and its superiority in terms of explanation compared to state-of-the-art methods.

KEYWORDS

Explainability, evolutionary algorithm, reservoir computing, memristor

ACM Reference Format:
Xinming Shi, Zilu Wang and Leandro L. Minku* and Xin Yao. 2023. Explaining Memristive Reservoir Computing Through Evolving Feature Attribution. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3583133.3590619

1 INTRODUCTION

Reservoir computing (RC) is a popular, accurate, and biologically plausible paradigm in Recurrent Neural Network (RNN) modeling. Its development is an active research area, with new architectures improving its performance. RC is not limited to software implementation, with hardware implementations also being explored to improve power consumption [3].

Memristor is resistance-changeable, non-volatile, power-efficient and high-density [7, 8], thus memristor-based RC has attracted a large number of researchers [5]. For instance, some researchers followed the straightforward method of implementing RNN-based reservoirs by using both neuron and synapse circuits, so that memristor-based Echo State Network (ESN) [13] and Liquid State Machine (LSM) [15] could be realized.

However, the lack of model explainability in existing Memristive Reservoir Computing (MRC) systems poses a significant challenge in properly explaining the decision-making, thereby rendering this neuromorphic computing architecture unaccountable, untrustworthy, and hard to understand and verify. The "black box" nature of reservoirs presents a significant challenge for the development and understanding of these computing systems, particularly in the context of physical reservoirs. This lack of transparency in the internal workings of the model hinders the deeper understanding of how the reservoir is able to achieve its learning task, thereby limiting the potential for further advancement and optimization of it.

In this work, we propose a new evolutionary framework for explaining the time series predictions of MRC using feature attribution explanation. Extensive experimental studies have been carried out to verify that our proposed approach can explain the MRC, and obtain superior results compared to state-of-the-art approaches in terms of explainability and circuit performance.

2 PROPOSED FAE FRAMEWORK FOR MRC

2.1 Feature Attribution by Dynamic Mask

We propose a novel approach for dynamically obscuring irrelevant features within input data by utilizing a dynamic masking technique as follows. A mask associated with an input sequence $\mathbf{z} \in \mathbb{R}^{T^* \times n}$ and a MRC system $f : \mathbf{z} \longrightarrow \mathbf{y}^{T^*+1}$ is a matrix $\mathbf{M} \in [0, 1]^{T^* \times n}$ of the same dimension as the input sequence. T^* is the length of the time series, and n is the dimension of the time series. Each element $m_{t,i} \in \mathbf{M}$ represents the importance of feature i at time t for MRC system f to produce the prediction $\mathbf{y} = f(\mathbf{x})$, i.e., the mask coefficients $m_{t,i}$ represent the saliency of the features. We define a dynamic mask as a linear perturbation operator $\Pi_M : \mathbb{R}^{T^* \times n} \longrightarrow \mathbb{R}^{T^* \times n}$ associated with a mask $\mathbf{M} \in [0, 1]^{T^* \times n}$, that acts on the inputs time series:

$$[\Pi_{\mathbf{M}}(\mathbf{z})] = \pi(\mathbf{z}, m_{t,i}; t, i). \tag{1}$$

Taking the advantage of the dynamic nature of the data into account, a dynamic mask is applied to involve the perturbation with neighboring history, which can be described as follows:

$$\pi(\mathbf{z}, m_{i,t}, ; t, i) = m_{t,i} \cdot z_{t,i} + (1 - m_{t,i}) \cdot \mu_{t,i}, \tag{2}$$

where π can be interpreted as a fading operation applied to the moving average perturbation $\mu_{t,i}$ that is based on neighboring history:

$$\mu_{t,i} = \frac{1}{1 + W} \sum_{j=t-W}^{t} \mathcal{B}^j z_{t,i}, \tag{3}$$

[*] Corresponding authors

GECCO '23 Companion, July 15–19, 2023, Lisbon, Portugal
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0120-7/23/07.
https://doi.org/10.1145/3583133.3590619

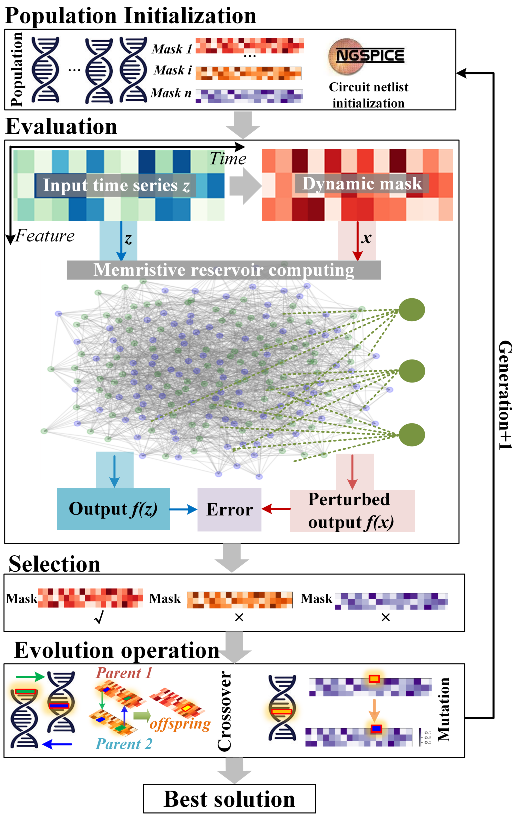

Figure 1: Overall framework of explainable MRC by an evolutionary algorithm.

where W is the window size that controls how the moving window depends on neighboring times. $\mathcal{B}$ is the backshift operator, defined as $\mathcal{B}^j \mathbf{z} = \mathbf{z}_{t-j,i}$ for $j \geq 0$.

2.2 Framework of Explainable MRC Using Evolutionary Algorithm

To assign a saliency score to each component of the input time series $\mathbf{z}$, we design an evolutionary framework of explainable MRC able to evolve $\mathbf{M}$, as shown in Figure 1. The evaluation component perturbs the input time series $\mathbf{z}$ using the mask $\mathbf{M}$ via a perturbation operator Π, generating a perturbed signal $\mathbf{x}$, which is fed into the MRC system to produce a perturbed prediction $f(\mathbf{x})$. The difference between the original prediction $f(\mathbf{z})$ and the perturbed prediction $f(\mathbf{x})$ is used by the evolutionary algorithm to adapt the saliency scores contained in mask M.

2.2.1 Evolutionary Flowchart. To initialise the population with *num_Pop* individuals, each corresponding to a mask $\mathbf{M}$, we adopt a sparse initialization method that ensures the input feature's parsimony. Each element $m_{t,i}$ in the mask is represented by the memristor's conductance in the memristive dynamic mask, with a valid range of $[0, 1]$. Additionally, each individual is associated with a binary matrix $W_{bool} \in \mathbb{R}^{T^* \times n}$, which is initialised as an Erdös–Rényi random graph with binary values. The probability of $W_{bool}^{t,i} = 1$ is given by $p(W_{bool}^{t,i}) = \frac{\varepsilon(T^+ n)}{T^\cdot n}$, where $\varepsilon \in \mathbb{R}^+$ controls the sparsity level. If $W_{bool}^{t,i} = 1$, the corresponding element mt, i in the same position is initialised with a value randomly selected from $[0, 1]$. If $W^{t,i} bool = 0$, the corresponding element mt, i is initialised as zero. The next step is the evaluation. Specifically, the dynamic mask $\mathbf{M}$ is employed to generate a perturbed version of the input signal. This perturbed input sequence is then inputted into the MRC system to produce a perturbed prediction, denoted as $f(\mathbf{x})$. The difference between the original prediction, $f(\mathbf{z})$, and the perturbed prediction, $f(\mathbf{x})$, is utilized to construct the fitness as follows:

$$fitness = \sqrt{\left\langle \| f(\mathbf{z}) - f(\mathbf{x}) \|^2 \right\rangle}. \tag{4}$$

Elitism is applied by saving the best individual for the next generation. Next, parents are selected via *num_Tour* tournament selection and undergo mutation and crossover with probabilities P_m and P_c (see Sections 2.2.2 and 2.2.3). The best individuals among the parent and offspring are selected to survive for the next generation, and this process is repeated for *num_Gen* generations.

2.2.2 Crossover. A parent individual could be regarded as the product of each element $m_{t,i} \in \mathbf{M}$ with the elements in the corresponding positions of matrix $W_{bool}^{t,i}$. Given parents $\mathbf{M}^1$ and $\mathbf{M}^2$, the mask value of offspring's reservoir is determined as follows [1]:

$$m_{t,i} = \begin{cases} \frac{m_{t,i}^1 + m_{t,i}^2}{2} & \text{if } m_{t,i}^1, m_{t,i}^2 \neq 0 \text{ and } random < 0.5, \\ m_{t,i}^1 & \text{if } m_{t,i}^1, m_{t,i}^2 \neq 0 \text{ and } 0.5 \leq random < 0.75, \\ m_{t,i}^2 & \text{if } m_{t,i}^1, m_{t,i}^2 \neq 0 \text{ and } 0.75 \leq random < 1.0, \\ m_{t,i}^1 & \text{if } m_{t,i}^2, = 0 \text{ and } m_{t,i}^1 \neq 0, \\ m_{t,i}^2 & \text{if } m_{t,i}^1, = 0 \text{ and } m_{t,i}^2 \neq 0, \\ 0 & \text{if } m_{t,i}^1, m_{t,i}^2 = 0. \end{cases} \tag{5}$$

A corresponding matrix W_{bool} is created for each offspring based on which of the offspring elements in $\mathbf{M}$ are zero or non-zero.

2.2.3 Mutation. When an individual is to go through mutation given the probability of mutation P_m, one of two mutation operators is chosen to be applied uniformly at random:

Boolean mutation: In terms of the binary value in the Boolean matrix W_{bool}, each of them has the probability P_{mb} to be flipped to its opposite value. If the current value $W_{bool}^{t,i}$ is 1, then it will mutated as 0 and its corresponding $m_{t,i}$ in the mask M will be mutated into 0 also. If the current value $W_{bool}^{t,i}$ is 0, then then it will mutated as 1 and its corresponding $m_{t,i}$ in the mask M will be mutated into a uniformly random value in the range of $[0, 1]$.

Value mutation: For the values in $m_{t,i}$ corresponding to the position where W_{bool} is not zero, there will be the probability P_{mm} to mutate them to a new value taken uniformly at random within the allowable range $[0, 1]$.

3 EXPERIMENT

In this section, the experiment will be introduced. The MRC is simulated in NGSPICE, and the evolutionary approach is implemented in Python, where the detailed implementation could be found in github [1]. To assess the accuracy of features identified as salient by

[1] https://github.com/embeddedsky/ExplainableMRC.git

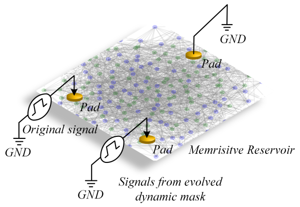

Figure 2: The circuit setting to the black-box MRC system.

the proposed and existing XAI approaches, we employ the area under the precision curve (AUP) as a metric, with higher values indicating superior performance. To evaluate the proportion of salient features that have been correctly identified, we utilize the area under the recall curve (AUR), with higher values indicating better performance. Let $\mathbf{Q} = (q_{t,i}) \in [0,1]^{T^* \times n}$ be a matrix representing the ground truth significance of the inputs contained in $\mathbf{z} \in \mathbb{R}^{T^* \times n}$, where $q_{t,i} = 1$ indicates that the feature $z_{t,i}$ is deemed salient and $q_{t,i} = 0$ otherwise. The mask $M = (m_{t,i}) \in [0,1]^{T^* \times n}$ is obtained through our explanation method. We consider that there is a detection threshold $\tau \in (0,1)$ to determine the salience of the feature $z_{t,i}$ based on the corresponding value of $m_{t,i}$. This allows us to convert the mask into an estimator $\hat{\mathbf{Q}}(\tau) = (\hat{q}_{t,i}(\tau))$ for $\mathbf{Q}$ through the following equation [2]:

$$\hat{q}_{t,i} = \begin{cases} 1 & \text{if } m_{t,i} \geq \tau \\ 0 & \text{else} \end{cases} \tag{6}$$

Let A denote the sets of truly salient indices and $\hat{A}$ the set of indexes selected by the saliency method, described as follows:

$$A = \{(t,i) \in [1:T^*] \times [1:n] | q_{t,i} = 1\} \tag{7}$$

$$\hat{A}(\tau) = \{(t,i) \in [1:T^*] \times [1:n] | \hat{q}_{t,i} = 1\}. \tag{8}$$

Based on these, the precision and recall curves are defined as:

$$P : (0,1) \to [0,1] : \tau \mapsto \frac{A \cap \hat{A}(\tau)}{\hat{A}(\tau)} \tag{9}$$

$$R : (0,1) \to [0,1] : \tau \mapsto \frac{A \cap \hat{A}(\tau)}{A(\tau)} \tag{10}$$

The AUP and AUR scores are the area under these curves [2]:

$$AUP = \int_0^1 P(\tau) d\tau \tag{11}$$

$$AUR = \int_0^1 R(\tau) d\tau \tag{12}$$

The black-box MRC system was evaluated with either the signals from the memristive dynamic mask or the original signals, as shown in Figure 2. Our method's parameter setting was as follows: $num\_Gen = 200$, $num\_Pop = 40$, $num\_Tour = 3$, $P_m = 0.8$, $P_c = 0.5$, $P_{mm} = 0.8$, $P_{mb} = 0.8$. The dynamic mask had a sparsity of $\varepsilon = 0.2$ and a window size of $W = 8$. We set the hyperparameters of existing XAI approaches to the same values as in their corresponding papers [6, 9, 12] since we used the same dataset. Table 1 lists the key hyperparameters and experimental results for all methods due to space constraints. To verify our proposed method, the following research questions will be investigated.

3.1 How well can the proposed approach explain the results compared with existing state-of-the-art explanation approaches?

We compared our proposed method with several popular XAI approaches, including FIT [12], IG [10], FO [11], DeepLIFT [9], and LIME [6], by assessing the importance of each feature at each time step using these methods. Since the existing XAI benchmark approaches are not compatible with MRC systems, therefore, in order ensure a fair comparison, all the benchmarks and our proposed approach are applied on a compatible and same "black box" model, which is a RNN that is also based on the recurrent connections. The setting of black box RNN is the same as work [14].

Experiments on a state dataset with known ground truth feature saliency are used for the comparisons. A 2-state hidden Markov model (HMM) was used to generate the data with $T^* = 200$ time steps, which is the same as [2, 6, 12]. 1000 time series data points were used, out of which 800 were utilized for training and 200 for testing. Table 1 shows the average experimental results on the state dataset based on 10 runs. We can see that our method outperformed the other benchmarks according to AUP, while obtaining the second best AUR. IG got the highest AUR, which indicates that it identifies more salient features. However, this was at the cost of a much lower AUP, showing that this method considers too many non-salient features as being salient. Even though our method obtained an AUR that is 0.09 smaller than IG, it obtained an AUP that is 0.18 larger. We also visualize the feature importance identified by various methods in Figure 3. We can clearly see that our proposed method Figure 3 (b) is capable of generating a feature importance map that more accurately reflects the true saliency of the inputs. Our method particularly stands out due to its ability to effectively distinguish salient inputs from others through the utilization of high contrast.

3.2 How well can the proposed approach explain the predictions of MRC system for time series prediction and recognition tasks?

Our proposed approach is applied to seven time series tasks, including Narma10/20, Santa Fe Laser data, Tree Ring, DJI, ARFIMA series, and Dynamic Gesture Recognition [7]. Due to the lack of ground truth salience, AUP and AUR cannot be used. Instead, we qualitatively analyze salience based on knowledge of the long- or short-term memory of each time series. Our proposed FAE method is the only one that can explain MRC systems. Figure 4 visualizes the dynamic masks for each task, with blue indicating the most influential feature and yellow indicating less importance. Narma 10 and 20 show input feature influence on recent time steps, while DJI, ARF, and Tree exhibit influence on more distant steps. Santa A displays a periodic pattern, while Dynamic Gesture Recognition shows salience in both distant and recent steps.

Based on the autocorrelation analysis in [14], our datasets can be categorized as short-term memory (Narma 10, Narma 20, Santa A) or long-term memory (DJI, ARF, Tree, DGR). Figure 4 confirms this memory behavior, with Narma 10 relying on the most recent 10 steps and Narma 20 extending to 20 steps. Santa A exhibits periodic feature importance, aligning with the physical properties of the laser it represents [4]. DJI, ARF, and Tree display long-term

Table 1: Parameter setting and results for RQ1.

	AUP	AUR
IG [10]	0.58 ±0.01	0.77 ± 0.01
FIT [12]	0.43± 0.01	0.53± 0.02
DEEPLIFT [9]	0.64± 0.00	0.41±0.00
LIME [6]	0.46± 0.01	0.52 ± 0.01
FO [11]	0.63 ± 0.01	0.46± 0.02
Ours	0.76 ±0.01	0.68 ± 0.00

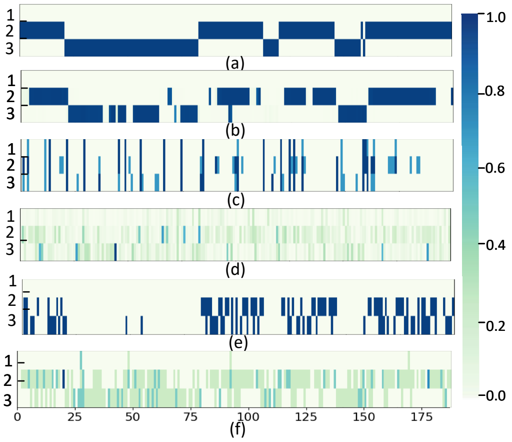

Figure 3: The feature importance masks produced by various methods. (a) True salient feature; (b) Our proposed method; (c) FIT; (d) DEEPLIFT; (e) LIME; (f) FO.

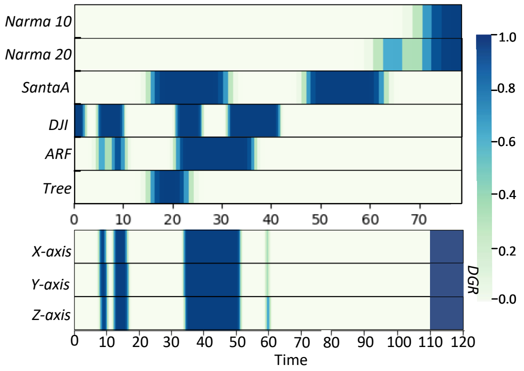

Figure 4: The visualization of the evolved dynamic mask for different time series tasks.

memory, while DGR shows a mix of long-term and recent time step influence, indicating distinguishable gestures in recent data.

4 CONCLUSION

In this work, we proposed an evolutionary framework to explain MRC system. Our approach attributes the feature importance of the time series through an evolutionary process, resulting in a more reliable decision-making process for the MRC system. Our experimental results demonstrate the superiority of our approach in terms of explanation, outperforming state-of-the-art methods. In the future, we will further implement the explainable MRC into hardware. Additionally, we plan to explore the explainability of other memristive computing models, beyond MRC systems, to further advance our understanding of these systems.

ACKNOWLEDGMENTS

This work was supported by NSFC (Grant No. 62250710682), the Research Institute of Trustworthy Autonomous Systems (RITAS), the Guangdong Provincial Key Laboratory (Grant No. 2020B121201001), the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), the Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531).

REFERENCES

[1] Kyriakos C Chatzidimitriou and Pericles A Mitkas. 2013. Adaptive reservoir computing through evolution and learning. *Neurocomputing* 103 (2013), 198–209.
[2] Jonathan Crabbé and Mihaela Van Der Schaar. 2021. Explaining time series predictions with dynamic masks. In *International Conference on Machine Learning*. PMLR, 2166–2177.
[3] Matthew Dale, Julian F Miller, Susan Stepney, and Martin A Trefzer. 2016. Evolving carbon nanotube reservoir computers. In *Proc. Int. Conf. Unconv. Comput. Nat. Comput. (UCNC)*. Springer, 49–61.
[4] Uwe Hübner, Nimmi B Abraham, and Carlos O Weiss. 1989. Dimensions and entropies of chaotic intensity pulsations in a single-mode far-infrared NH 3 laser. *Phys. Rev. A* 40, 11 (1989), 6354.
[5] Suhas Kumar, Xinxin Wang, John Paul Strachan, Yuchao Yang, and Wei D Lu. 2022. Dynamical memristors for higher-complexity neuromorphic computing. *Nat. Rev. Mater.* 7, 7 (2022), 575–591.
[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " 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*. 1135–1144.
[7] Xinming Shi, Leandro L Minku, and Xin Yao. 2022. Adaptive Memory-enhanced Time Delay Reservoir and Its Memristive Implementation. *IEEE Trans. Comput.* (2022).
[8] Xinming Shi, Zhigang Zeng, Le Yang, and Yi Huang. 2018. Memristor-based circuit design for neuron with homeostatic plasticity. *IEEE Trans. Emerg. Topics Comput. Intell.* 2, 5 (2018), 359–370.
[9] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In *International conference on machine learning*. PMLR, 3145–3153.
[10] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*. PMLR, 3319–3328.
[11] Harini Suresh, Nathan Hunt, Alistair Johnson, Leo Anthony Celi, Peter Szolovits, and Marzyeh Ghassemi. 2017. Clinical intervention prediction and understanding using deep networks. *arXiv preprint arXiv:1705.08498* (2017).
[12] Sana Tonekaboni, Shalmali Joshi, Kieran Campbell, David K Duvenaud, and Anna Goldenberg. 2020. What went wrong and when? Instance-wise feature importance for time-series black-box models. *Advances in Neural Information Processing Systems* 33 (2020), 799–809.
[13] Shiping Wen, Rui Hu, Yin Yang, Tingwen Huang, Zhigang Zeng, and Yong-Duan Song. 2018. Memristor-based echo state network with online least mean square. *IEEE Trans. Syst., Man, Cybern., Syst.* 49, 9 (2018), 1787–1796.
[14] Jingyu Zhao, Feiqing Huang, Jia Lv, Yanjie Duan, Zhen Qin, Guodong Li, and Guangjian Tian. Virtual Event, 2020. Do rnn and lstm have long memory?. In *Proc. 37th Int. Conf. Mach. Learn.* 11365–11375.
[15] Zhenyu Zhao, Lianhua Qu, Lei Wang, Quan Deng, Nan Li, Ziyang Kang, Shasha Guo, and Weixia Xu. 2020. A memristor-based spiking neural network with high scalability and learning efficiency. *IEEE Trans. Circuits and Syst. II, Exp. Briefs* 67, 5 (2020), 931–935.