CBR-fox: a Case-based Explanation Method for Time Series Forecasting Models

Moisés F. Valdez-Ávila[1] [0000−0001−7761−9635],
Carlos Bermejo-Sabbagh[1] [0000−0002−6053−1175],
Belen Diaz-Agudo[2] [0000−0003−2818−027X],
Mauricio G. Orozco-del-Castillo[1] [0000−0001−5793−6449], and
Juan A. Recio-Garcia[2] [0000−0001−8731−6195]

[1] Tecnológico Nacional de México/IT de Mérida,
Departamento de Sistemas y Computación, Mérida, México
{e18081131,carlos.bs,mauricio.orozco}@itmerida.edu.mx
[2] Department of Software Engineering and Artificial Intelligence
Instituto de Tecnologías del Conocimiento
Universidad Complutense de Madrid, Spain
{belend,jareciog}@ucm.es

Abstract. Explainable Artificial Intelligence refers to methods that help human experts understand solutions developed by Artificial Intelligence systems in the form of black-box models, making them transparent and understandable. This paper describes CBR-fox, a post-hoc model-agnostic case-based explanation method for forecasting models. This method generates a case base of explanation examples through a sliding-window technique applied over the time series. Then, these explanation cases can be retrieved using a wide range of well-established metrics for time series comparison. Moreover, we introduce and evaluate a novel similarity metric named Combined Correlation Index. The proposed retrieval approach considers as a signal the similarity series resulting from applying the comparison metrics. This way, the signal can be smoothed using noise removal filters, such as the Hodrick-Prescott and low-pass filters, to avoid maximally similar cases that may overlap or represent a local slice of the source time series. The resulting signal allows then to foster diversity in the retrieved explanation cases presented to the user. The proposed case-based explanation approach is evaluated in the weather forecasting domain using an artificial neural network as the black-box model to be explained.

Keywords: Explainable Artificial Intelligence, Time Series Forecasting, Case-based Explanation, Artificial Intelligence of the Things

1 Introduction

The current rise of the Internet of Things (IoT) technology has produced a wide range of sensing solutions that are progressively being integrated into our daily life devices such as mobile phones or wearables [11]. The combination of such sensing capabilities with Artificial Intelligence (AI) is producing Artificial Intelligence of Things (AIoT) and Internet of Everything (IoE) applications that provide an enhanced user experience [34].

Since IoT sensing devices obtain readings that vary in time, they have been subject to time series analysis and modeling. A time series is a sequence of observed values of a variable at equally spaced timestamps t, represented as a set of discrete values [23]. Time series forecasting is the prediction of future data values based on collected data and has been an area of great interest in science, engineering, and business. Traditional time series forecasting is usually approached by the analysis of its internal structure: autocorrelation, trend, seasonality, etc., to capture the pattern of the long-time behavior of the system [23]. These models predict future values of a target $y_i(t)$ for a given observed value i at a time t. In IoT, these observed values usually represent measurements from sensors. While this applies to univariate forecasting, the extension to multivariate models can be performed without loss of generality [28]. Machine Learning (ML) has, however, positioned as the next generation of time series forecasting models [1] due to increasing data availability and computing power. One of the main techniques in ML is artificial neural networks (ANNs), which have proven to be a reliable tool for time series analysis [3], and numerous ANNs design choices have emerged given the diversity of time series problems across multiple domains. Of these designs, the most commonly used one in time series forecasting is recurrent neural networks (RNNs) due to the natural interpretation of time series data as sequences of inputs and targets [18].

Although ML approaches have demonstrated very good prediction performance, they have significant limitations regarding their explainability. Neural network models are considered as "black boxes" because their internal processes are challenging to interpret with respect to the predictions they produce [10]. EXplainable AI (XAI) methods help human experts understand solutions developed by AI. Solving the black-box problem is a requirement for auditing the reasoning behind incorrect predictions taken by AIoT systems, and foreseeing the data patterns that may lead to a concrete prediction.

This paper describes CBR-fox (**CBR** - **fo**recasting e**x**planation), a post-hoc sliding-window explanation-by-example method that enables the explanation of black-box forecasting models using Case-Based Reasoning (CBR). This method follows the twin surrogate CBR explanation approach that enables making the forecasting process understandable [14]. Here, time series are split into different time-window cases that serve as explanation cases for the outcome of the prediction model. This paper exemplifies and evaluates the benefits of CBR-fox in the weather forecasting domain. The presented use case is based on the readings from an environmental sensor for mobile devices[3] capable of sensing several weather variables [21, 31], with the potential of ANNs to compute climate predictions.

[3] BOSCH Sensortec BME680: https://www.bosch-sensortec.com/products/environmental-sensors/gas-sensors/bme680/

The paper runs as follows. Section 2 presents the background of this work. Then, Section 3 describes our case-based explanation method. Section 4 presents the evaluation results and Section 5 concludes the paper and opens lines of future work.

2 Background

This paper proposes the use of CBR as an explanation method for black-box forecasting models. However, there are other approaches that have used CBR as a forecasting technique itself in various domains, such as finance, energy, and healthcare [9, 17, 25]. The benefits of generating predictions based on past cases are the inherent interpretability of the CBR process. However, its accuracy has been outperformed by other black-box models such as ANNs. This way, CBR is more valuable as a post-hoc explanation method than as a forecasting model. The work by [14] presents a systematic review of "ANN-CBR twins": post-hoc explanation-by-example approaches that rely on the twinning of ANNs with CBR systems. One example is the proposal by Li et al. [16] which combines the strength of deep learning and the interpretability of CBR to make an interpretable deep ANN. This approach modifies the ANN architecture to encode prototypes in the output layer that partially allows tracing the classification path for a new observation. Bebarta et al. [2] also present an intelligent stock trading system utilizing dynamic time windows with case-based reasoning, and a recurrent function link artificial neural network (FLANN).

However, there are few works on case-based explanations for time series forecasting. Some of them have focused on counterfactual explanations [8], but not many have aimed at time series. Corchado and Lees [7] presented a hybrid approach to forecasting the thermal structure of the water ahead of a moving vessel that combines CBR and ANNs. Nevertheless, this approach is not a post-hoc explanation method as it exploits the generalizing ability of the ANN to guide the adaptation stage of the CBR mechanism. The paper by Olsson et al. [22] presents a general method for explaining the predictions made by probabilistic ML algorithms using cases. The method comprises two main parts: 1) a measure of similarity between cases, which is defined with respect to a probability model, and 2) a case-based approach to explaining the probabilistic prediction by estimating the prediction error. The paper demonstrates the use of this method in explaining the predictions of the energy performance of households. Other related case-based explanation methods also use sliding-window approaches to generate cases. For example, in electric load forecasting [24] or speech emotion recognition [26]. However, Lorenzo and Arroyo [19] present an alternative approach where clustering is applied over time series to obtain prototypes that act as cases.

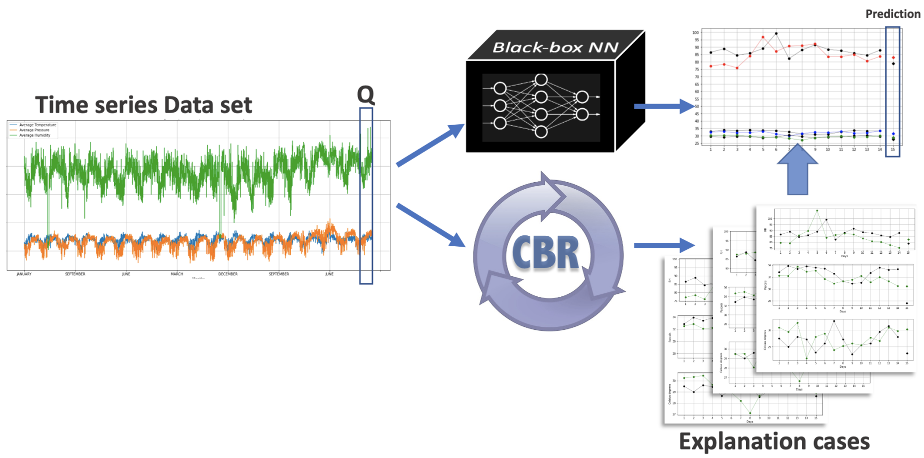

Fig. 1: Global schema of the ANN-CBR twin approach of CBR-fox.

3 Method

The main contribution of this paper is the use of CBR for the generation of explanations associated with the prediction of a certain black-box ANN model. However, it is important to note that the proposed explanation method—CBR-fox—is applicable to other forecasting black-box models. We propose a solution for the explanation of the outcomes of the ANN, where a black-box system is explained by an interpretable twin CBR system [14]. This approach is illustrated in Figure 1, where the same dataset is used as the input of the ANN and to create the explanatory cases provided by the CBR system.

Explanatory cases are generated using a sliding-window method over the whole time series: $C_t = \langle [t-w, t], R_{t+1} \rangle$ for $t \in [w, L-1]$where w is the window size and R_{t+1} is the solution of the case, which corresponds to the following reading, and L is the length of the time series. This way, the case base is generated from the source time series ts and contains $len(ts) - w - 1$ cases that will be used to explain any forthcoming prediction of the ANN. It is important to note that, by using this sliding-window method, consecutive cases C_t and C_{t+1} overlap in $w - 1$ timestamps.

Once the case base is generated, the twin system works as follows. Given a query timestamp t_q, the ANN predicts the following time series values: $Pred(t_q)$. In parallel, the CBR explanation system receives the corresponding time window query $Q = [t_q - 1 - w, t_q - 1]$ and returns the most similar explanatory cases to explain the prediction $Pred(t_q)$. These explanatory cases can be directly presented to the user or combined into a single explanation case.

The critical elements of the CBR-fox explanation method are the similarity metrics, the retrieval and reuse processes, and the visualization of the explanation cases. These steps are explained next.

3.1 Time series similarity

The quality of the retrieved explanation cases highly depends on the similarity metrics used to compare them. Next, we present several metrics that are integrated into our case-based explanation method. Most are established state-of-the-art metrics for this task, although we introduce a novel metric—denoted CCI, Combined Correlation Index—specifically designed to retrieve time series that resembles the original query.

- *Dynamic Time Warping (DTW)* computes the distance between two time series by considering the pointwise (usually Euclidean) distance between all elements of the two-time series. It then uses dynamic programming to find the warping path that minimizes the total pointwise distance between realigned series [27].
- *Weighted Dynamic Time Warping (WDTW)* incorporates a multiplicative weight penalty corresponding to the warping distance (time series with lower phase differences have a smaller weight imposed) [12].
- *Derivate Dynamic Time Warping (DDTW)* considers the 'shape' of the time series represented by the first derivative of the sequence in an attempt to improve on DTW [15].
- *Weighted derivative dynamic time warping (WDDTW)* considers not only the shape, but also the phase, of the time series by adding a weight to the derivative [13].
- *Move-Split-Merge (MSM)* uses three fundamental operations: Move, Split, and Merge, which have an associated cost and can be applied in sequence to transform any time series into any other time series. The MSM distance is defined as the cost of the cheapest sequence of operations that transforms the first time series into the second one [30].
- *Edit Distance For Real Penalty (ERP)* proposes the idea of sequences of points that have no matches [4] and attempts to align time series by carefully considering how indexes are carried forward through the cost matrix.
- *Longest Common Subsequence (LCSS)* looks for the longest common sequence between two time series and returns the percentage of the longest common sequence [33].
- *Time Warp Edit (TWE)*, a measure for discrete time series matching with time 'elasticity', is well-suited for the processing of event data for which each data sample is associated with a timestamp [20].
- *Edit Distance for Real Sequences (EDR)* determines the percentage of elements that should be removed from the input signals x and y such that the sum of distances between the remaining elements is below a specified tolerance level [5].
- *Combined Correlation Index (CCI)* is a novel metric presented in this paper specifically designed from the point of view of explainability. It is aimed to optimize the similarity between to time series according to the shape and distance. It provides a way to measure how a given time window case $\mathbf{C}$ is related to a target query window $\mathbf{Q}$:

$$\mathrm{CCI}(\mathbf{C}, \mathbf{Q}) = (\alpha_1 + \rho(\mathbf{C}, \mathbf{Q}) - \alpha_2 \|(\mathbf{C}, \mathbf{Q})\|) \cdot \alpha_3 \tag{1}$$

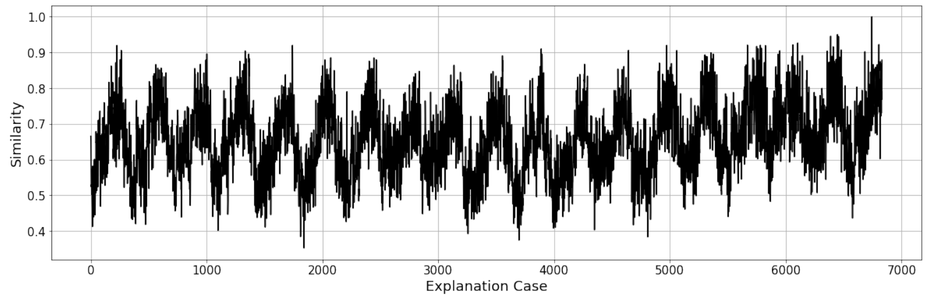

Fig. 2: Example of the similarity series (n=6847). Maximally similarity peaks tend to focus on a local portion of the case base.

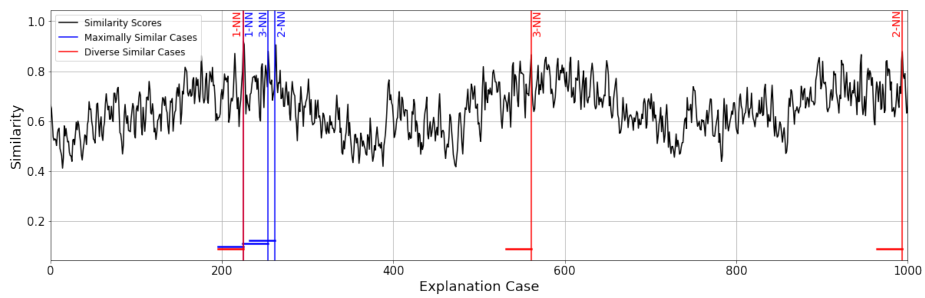

Fig. 3: Example of high-frequency components (maximally similarity peaks) in a subsample of the similarity series ($n = 1000$). Blue lines represent the maximally similar cases (that tend to overlap or focus on a concrete portion of the case base) whereas red lines represent similar but diverse cases.

where ρ is the function that calculates the Pearson correlation coefficient, and the double bars represent the normalized Euclidean distance between those vectors. The correlation component deals with the morphological similarity of the time windows, while the Euclidean distance component deals with the proximity between the time series in the given time windows. Remaining parameters are used to shift the correlation coefficient (α_1), normalize the Euclidean distance (α_2), and reshape the resulting values to the $[0-1]$ range (α_3).

3.2 Retrieval process

The retrieval process first computes the similarity between the query and each explanation case. It is computed using any of the metrics presented in the previous section, producing a *similarity signal* analogous to the one presented in Figure 2, where values correspond to the similarity between the query and the cases generated for every timestamp.

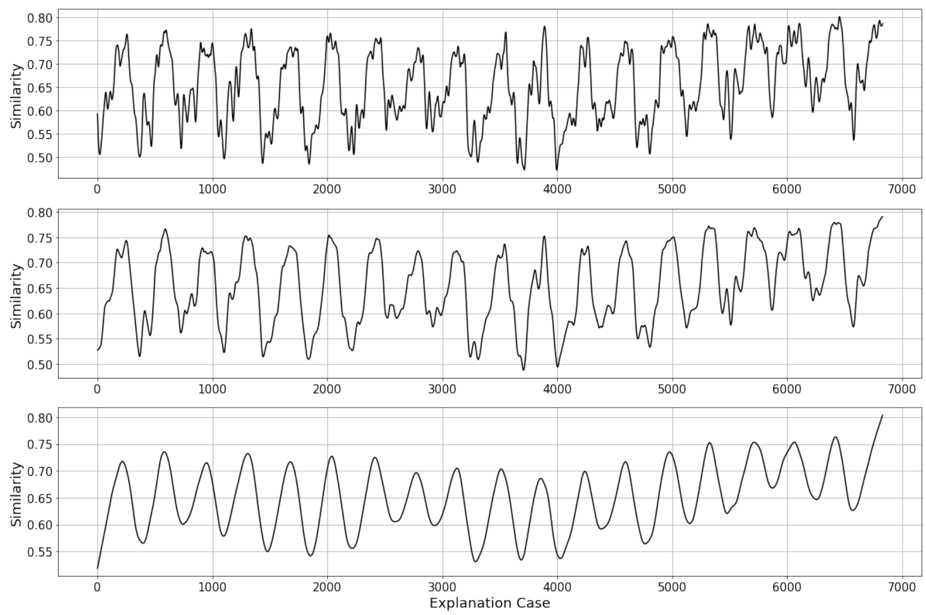

Fig. 4: Top chart: application of the Hodrick-Prescott filter to obtain the tendency series of the similarity signal. The middle and bottom charts represent the iterative application of the LOWESS filter to the resulting signal. After successive iterations, the similarity signal converges (bottom chart), allowing the identification of the most similar but diverse explanation cases.

As we can observe, the similarity values for overlapping consecutive cases yield undesirable high-frequency components. Figure 3 illustrates this problem with a limited portion of the case base. Overlapping time windows tend to obtain a very close similarity value, and therefore, if the retrieval process returns the maximally similar cases directly, they will overlap and focus on a concrete slice of the case. This is a well-known problem in the CBR field related to the retrieved cases' diversity [29]. In this figure, the indexes (timestamp t) of the maximally similar cases are identified by a vertical blue line, whereas the horizontal segment on the bottom represents the time window $[t-w, t]$ to be retrieved as the explanation case. We can clearly observe that these maximally similar cases overlap and only represent a concrete slice of the time series. In contrast, cases represented in red are the top similar but diverse cases, where the locality effect of maximal similarity is avoided by retrieving other suboptimal cases.

To address this problem, we apply the Hodrick–Prescott (HP) filter, a mathematical tool used in time series analysis for removing the cyclic component. The HP filter has become a benchmark for getting rid of cyclic movements in data and is broadly employed for macroeconomics research [32]. This filter returns the tendency component of the signals represented by the top chart in Figure 4. Then, a low-pass filter can be applied over this tendency signal to smooth it progressively. Concretely, CBR-fox applies the Locally Weighted Scatterplot Smoothing (LOWESS) method, a non-parametric technique for fitting a smooth curve to data points [6]. After iteratively applying this filter until the resulting signal no longer changes, we obtain a smoothed similarity signal as presented in the bottom chart of Figure 4.

This smoothed similarity signal allows us to identify the k most similar explanation cases to the query Q. To do so, its numerical derivative is obtained, helping in the identification of peaks and valleys, which will be used to segment the series into groups of convex and concave curves. Convex curves are then explored to find the timestamp with the highest similarity value. Only one timestamp is obtained for each convex curve. Once the maximal values for each convex curve have been collected, they are ranked and returned as the most similar explanation cases presented to the user to explain the prediction given by the black-box model.

3.3 Reuse

Once the k nearest neighbors have been retrieved, CBR-fox allows either to present them directly to the user or combine them to obtain a joint explanation case. In the latter case, we propose two different reuse strategies:

Simple Average. This strategy computes the average for each timestamp of the k nearest neighbors:

$$S(C_1, \ldots, C_k) = [S_0, \ldots, S_{w-1}], \tag{2}$$

where

$$S_t = \frac{1}{k} \sum_{i=1}^{k} C_i[t]$$

Weighted Average. Generates a combined solution through a weighted average according to the similarity of the k nearest neighbors:

$$W_t(C_1, \ldots, C_k) = [W_0, \ldots, W_{w-1}], \tag{3}$$

where

$$W_t = \frac{\sum_{i=1}^{k} C_i[t] \cdot sim(C_i, Q)}{\sum_{i=1}^{k} sim(C_i, Q)}$$

3.4 Visualization

The visualization of the explanation cases to the user is exemplified in Figure 5. CBR-fox allows visualizing either all the k-NNs or the combined explanation case. Notice how it also includes the prediction of the ANN, $Pred(t_q)$, as well as the actual value stored in the solution of the case, R_{t+1}.

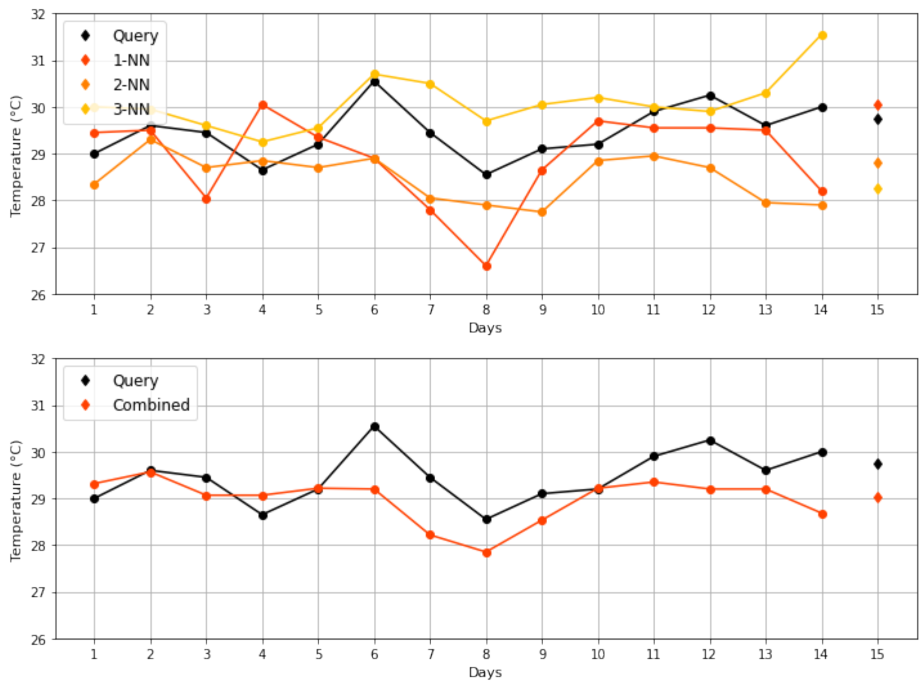

Fig. 5: Visualization of the explanation cases to the user. Best explanation cases (top) and combined explanation case (bottom). Window size: $w = 14$. The predictions from the ANN model, $Pred(t_q)$, and the case solutions, R_{t+1}, are also displayed to the user (day 15).

4 Evaluation

In this section, the evaluation of the CBR-fox system is addressed. First, we explain the dataset used to train the ML model, next, we discuss the validation and evaluation metrics and, finally, the results are presented.

4.1 Dataset and model

The dataset used for the evaluation of CBR-fox consists of meteorological variables recorded by the Mexican National Water Council (*Comisión Nacional del Agua, CONAGUA*) ground station located at the city of Mérida; among the data provided, the following variables were found: temperature (T), vapor pressure (P), and relative humidity (H). Daily records were obtained from January 1, 2000, to September 30, 2018.

A recurrent neural network (RNN) using long short-term memory (LSTM) cells was trained over the dataset of time windows and weather labels. 70% of the dataset was used during training, while the remaining 30% was used for testing purposes. The dataset is then divided into cases (time windows) with $w = 14$, obtaining explanation cases with 14 days of weather evolution.

4.2 Methodology

The evaluation consists of leave-one-out cross-validation for each case (time window) where the k-nearest neighbors (explanation cases) are obtained using any of the similarity metrics presented in Section 3.1.

The quality of each explanation case is obtained by comparing the original query time series with the Euclidean distance. This distance is computed for all the timestamps in $[t - w, t]$ and then averaged to obtain the global distance to the query. This evaluation metric aims to resemble the perception of the user regarding the geometrical difference between the query and the explanation cases. In the case of several explanation cases ($k > 1$) the values for each timestamp within the time window are averaged, and this average is then compared to the query. This process is performed for the three time series (temperature, vapor pressure, and relative humidity) that compose the explanation case. The global quality of the explanation case is then computed as the root sum squared (RSS) of the three distances to the query.

Additionally, the diversity of the explanation cases is evaluated through a dispersion metric that estimates the distance between the timestamps of the retrieved explanation cases. Following the example in Figure 3, diverse explanation cases will be scattered along the time series, whereas maximally similar cases without diversity will correspond to nearby timestamps. The dispersion metric used to measure the diversity is the average of the mutual differences between the timestamps of the k retrieved examples.

4.3 Results

Performance results are displayed in Figure 6. This figure shows the quality of the explanation cases obtained through the similarity metrics available in CBR-fox for different values of k and both reuse strategies. The first conclusion is that there are no remarkable differences between the simple and weighted averages. This indicates that similarity values are very homogeneous without irregular differences between neighbors.

Globally, the best metric is *Edit Distance For Real Penalty (EDRP)* for an only explanation example. However, the most useful metric is the *Combined Correlation Index (CCI)* when presenting two or three explanation examples to the user. From more than five explanation cases, all the metrics perform similarly, except *Longest Common Subsequence (LCS)*, which always obtains unsatisfactory results. Further research is needed to gain insight about the deteriorating behavior of *CCI* as the value k increases.

The evaluation of the diversity of the retrieved explanation cases is presented in Figure 7. Each plot in the figure corresponds to a similarity metric and compares the average of the mutual differences between the timestamps of the k retrieved examples between the smoothed similarity signal and the original one.

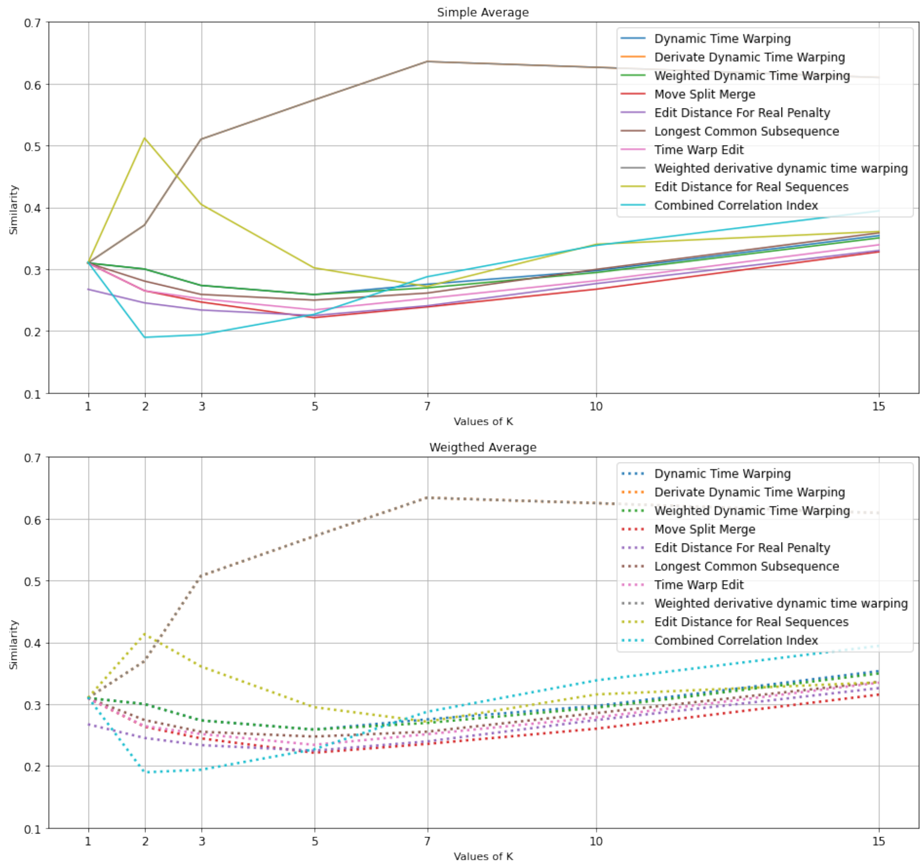

Fig. 6: Line plot showing the quality of the explanation cases obtained through the similarity metrics for different values of k and both reuse strategies: simple average (top), weighted average (bottom).

As expected, diversity rises when the number of explanation cases presented to the user increases.

5 Conclusions

The rise of the IoT and its combination with AI (AIoT and IoE) has led to a wide range of systems based on analyzing time series corresponding to sensors' readings. These systems are primarily based on ML forecasting models such as ANNs due to their higher performance. However, these models lack enough transparency to let users understand the reasons for a given prediction. Here, CBR is a proven solution to twin the ML forecasting model and provide transparency by means of explanation cases.

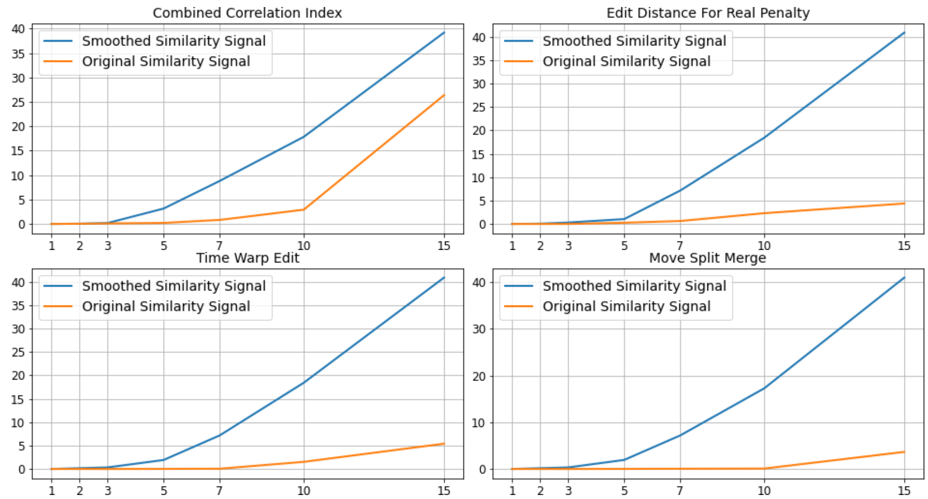

Fig. 7: Line plots comparing the diversity of the original and smoothed similarity signals for the most relevant metrics.

In this paper, we present CBR-fox, a post-hoc sliding-window explanation-by-example method that enables the explanation of black-box forecasting models using CBR. The major novelty of this method is to consider the similarity values between the query and each explanation case as a signal. This signal results from applying well-established similarity metrics to a case base generated by a sliding-window method that obtains a sequence of partially overlapping methods.

This way, the similarity signal can be processed to foster diversity in the explanation cases presented to the user. The application of noise removal filters, such as the HP and low-pass filters, avoids maximally similar cases that may overlap or represent a local slice of the source time series.

Additionally, we have presented a novel time series similarity metric –the Combined Correlation Index– designed explicitly for retrieving explanation examples as it is based on the comparison of shape and distance. This metric is experimentally compared to a wide range of established similarity metrics, achieving very high performance. This evaluation in the weather forecasting domain also highlights the impact of the proposed method regarding the diversity of the retrieved explanation examples.

This research opens many lines of future work. First, we will analyze the behavior of CBR-fox when reducing the overlapping of the generated cases by adding a step in the sliding window process. Additional evaluation metrics are also required to ensure the appropriateness of each similarity metric, which should be evaluated in further domains. This may lead to classifying the similarity metric according to their appropriateness to a concrete IoT domain.

The source code of CBR-fox and the dataset used for the evaluation presented in this paper are available at https://github.com/aaaimx/CBR-fox.

Acknowledgements This research is a result of the Horizon 2020 Future and Emerging Technologies (FET) programme of the European Union through the iSee project (CHIST-ERA-19-XAI-008, PCI2020-120720-2).
Supported by the PERXAI project PID2020-114596RB-C21, funded by the Ministry of Science and Innovation of Spain (MCIN/AEI/10.13039/501100011033) and the BOSCH-UCM Honorary Chair on Artificial Intelligence applied to Internet of Things.
It is also part of projects 10428.21-P and 13933.22-P of the Tecnológico Nacional de México/IT de Mérida.

References

[1] Ahmed, N.K., Atiya, A.F., El Gayar, N., El-Shishiny, H.: An empirical comparison of machine learning models for time series forecasting. Econometric Reviews **29**(5), 594–621 (2010), ISSN 07474938, https://doi.org/10.1080/07474938.2010.481556
[2] Bebarta, D.K., Das, T.K., Chowdhary, C.L., Gao, X.Z.: An intelligent hybrid system for forecasting stock and forex trading signals using optimized recurrent flann and case-based reasoning. International Journal of Computational Intelligence Systems **14**, 1763–1772 (2021), ISSN 1875-6883, https://doi.org/10.2991/ijcis.d.210601.001
[3] Orozco-del Castillo, M.G., Ortiz-Alemán, J., Couder-Castañeda, C., Hernández-Gómez, J., Solís-Santomé, A.: High solar activity predictions through an artificial neural network. International Journal of Modern Physics C **28**(6) (2017), ISSN 01291831, https://doi.org/10.1142/S0129183117500759
[4] Chen, L., Ng, R.: On the marriage of lp-norms and edit distance. In: Proceedings 2004 VLDB Conference, pp. 792–803, Morgan Kaufmann (2004), ISBN 978-0-12-088469-8, https://doi.org/10.1016/B978-012088469-8.50070-X
[5] Chen, L., Özsu, M.T., Oria, V.: Robust and fast similarity search for moving object trajectories. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, p. 491–502, Association for Computing Machinery (2005), ISBN 1595930604, https://doi.org/10.1145/1066157.1066213
[6] Cleveland, W.: Lowess: A program for smoothing scatterplots by robust locally weighted regression. The American Statistician, **35**, 54 (1981)
[7] Corchado, J.M., Lees, B.: A hybrid case-based model for forecasting. Applied Artificial Intelligence **15**(2), 105–127 (2001), https://doi.org/10.1080/088395101750065723
[8] Delaney, E., Greene, D., Keane, M.T.: Instance-based counterfactual explanations for time series classification. In: 29th International Conference on Case-Based Reasoning, ICCBR, p. 32–47, Springer-Verlag (2021), ISBN 978-3-030-86956-4, https://doi.org/10.1007/978-3-030-86957-1_3
[9] Gogineni, V.R., Kondrakunta, S., Brown, D., Molineaux, M., Cox, M.T.: Probabilistic selection of case-based explanations in an underwater mine

clearance domain. In: 27th International Conference on Case-Based Reasoning, ICCBR, Lecture Notes in Computer Science, vol. 11680, pp. 110–124, Springer (2019), https://doi.org/10.1007/978-3-030-29249-2_8

[10] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM computing surveys (CSUR) **51**(5), 1–42 (2018)

[11] Jara, A.J.: Wearable internet: Powering personal devices with the internet of things capabilities. In: 2014 International Conference on Identification, Information and Knowledge in the Internet of Things, pp. 7–7 (2014), https://doi.org/10.1109/IIKI.2014.9

[12] Jeong, Y.S., Jeong, M.K., Omitaomu, O.A.: Weighted dynamic time warping for time series classification. Pattern Recogn. **44**(9), 2231–2240 (sep 2011), ISSN 0031-3203, https://doi.org/10.1016/j.patcog.2010.09.022

[13] Jeong, Y.S., Jeong, M.K., Omitaomu, O.A.: Weighted dynamic time warping for time series classification. Pattern Recognition **44**(9), 2231–2240 (2011), ISSN 0031-3203, https://doi.org/https://doi.org/10.1016/j.patcog.2010.09.022

[14] Keane, M.T., Kenny, E.M.: How Case-Based Reasoning Explains Neural Networks: A Theoretical Analysis of XAI Using Post-Hoc Explanation-by-Example from a Survey of ANN-CBR Twin-Systems. In: 27th International Conference on Case-Based Reasoning, ICCBR, pp. 155–171, Springer (2019), ISBN 978-3-030-29249-2

[15] Keogh, E.J., Pazzani, M.J.: Derivative Dynamic Time Warping, pp. 1–11. SIAM (2001), https://doi.org/10.1137/1.9781611972719.1

[16] Li, O., Liu, H., Chen, C., Rudin, C.: Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI'18, AAAI Press (2018), ISBN 978-1-57735-800-8

[17] Li, W., Paraschiv, F., Sermpinis, G.: A data-driven explainable case-based reasoning approach for financial risk detection. Quantitative Finance **22**(12), 2257–2274 (2022)

[18] Lim, B., Zohren, S., Roberts, S.: Recurrent Neural Filters: Learning Independent Bayesian Filtering Steps for Time Series Prediction. Proceedings of the International Joint Conference on Neural Networks (2020), https://doi.org/10.1109/IJCNN48605.2020.9206906

[19] Lorenzo, L., Arroyo, J.: Analysis of the cryptocurrency market using different prototype-based clustering techniques. Financial Innovation **8**(1), 7 (2022), https://doi.org/10.1186/s40854-021-00310-9

[20] Marteau, P.F.: Time warp edit distance with stiffness adjustment for time series matching. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(2), 306–318 (2009), https://doi.org/10.1109/TPAMI.2008.76

[21] Newton, T., Meech, J.T., Stanley-Marbell, P.: Machine learning for sensor transducer conversion routines. IEEE Embedded Systems Letters **14**(2), 75–78 (2022), https://doi.org/10.1109/LES.2021.3129892

[22] Olsson, T., Gillblad, D., Funk, P., Xiong, N.: Case-based reasoning for explaining probabilistic machine learning. International Jour-

nal of Computer Science and Information Technology **6**, 87–101 (2014), https://doi.org/10.5121/ijcsit.2014.6206
[23] Palit, A.K., Popovic, D.: Computational Intelligence in Time Series Forecasting. Advances in Industrial Control, Springer London, London, 1 edn. (2005), ISBN 1-85233-948-9, https://doi.org/10.1007/1-84628-184-9
[24] Park, S., Jung, S., Jung, S., Rho, S., Hwang, E.: Sliding window-based lightgbm model for electric load forecasting using anomaly repair. The Journal of Supercomputing **77**, 12857–12878 (2021)
[25] Patel, A., Swathika, O.G.: Forecasting off-grid solar power generation using case-based reasoning algorithm for a small-scale system. In: Artificial Intelligence and Machine Learning in Smart City Planning, pp. 257–266, Elsevier (2023)
[26] Pham, N.T., Nguyen, S.D., Nguyen, V.S.T., Pham, B.N.H., Dang, D.N.M.: Speech emotion recognition using overlapping sliding window and shapley additive explainable deep neural network. Journal of Information and Telecommunication pp. 1–19 (2023)
[27] 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
[28] Sen, R., Yu, H.F., Dhillon, I.: Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting. Advances in Neural Information Processing Systems **32**(NeurIPS), 1–10 (2019), ISSN 10495258
[29] Smyth, B., McClave, P.: Similarity vs. diversity. In: 4th International Conference on Case-Based Reasoning, ICCBR, pp. 347–361, Springer (2001), ISBN 978-3-540-44593-7
[30] Stefan, A., Athitsos, V., Das, G.: The move-split-merge metric for time series. IEEE Transactions on Knowledge and Data Engineering **25**(6), 1425–1438 (2013), https://doi.org/10.1109/TKDE.2012.88
[31] Vishnubhatla, A.: Iot based air pollution monitoring through telit bravo kit. In: 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), pp. 1751–1755 (2022), https://doi.org/10.1109/ICAAIC53929.2022.9793252
[32] Vishwas, B.V., Patel, A.: Time-Series Characteristics, pp. 1–21. Apress, Berkeley, CA (2020), ISBN 978-1-4842-5992-4, https://doi.org/10.1007/978-1-4842-5992-4_1
[33] Vlachos, M., Kollios, G., Gunopulos, D.: Discovering similar multidimensional trajectories. In: Proceedings 18th International Conference on Data Engineering, pp. 673–684 (2002), https://doi.org/10.1109/ICDE.2002.994784
[34] Zhang, J., Tao, D.: Empowering things with intelligence: A survey of the progress, challenges, and opportunities in artificial intelligence of things. IEEE Internet of Things Journal **8**(10), 7789–7817 (2021), https://doi.org/10.1109/JIOT.2020.3039359