Ensemble Deep Learning for Forecasting ^{222}Rn Radiation Level at Canfranc Underground Laboratory

Miguel Cárdenas-Montes[1] [0000−0003−4445−8868], Iván Méndez-Jiménez[1]

Centro de Investigaciones Energéticas Medioambientales y Tecnológicas, Madrid, Spain
(miguel.cardenas, ivan.mendez)@ciemat.es

Abstract. Ensemble Deep Learning Architectures have demonstrated to improve the performance in comparison with the individual architectures composing the ensemble. In the current work, an ensemble of variants of Convolutional and Recurrent Neural Networks architectures are applied to the prediction of the ^{222}Rn level at the Canfranc Underground Laboratory (Spain). To predict the low-level periods allows appropriately scheduling the maintenance operations in the experiments hosted in the laboratory. As a consequence of the application of Ensemble Deep Learning, an improvement of the forecasting capacity is stated. Furthermore, the learned lessons from this work can be extrapolated to other underground laboratories around the world.

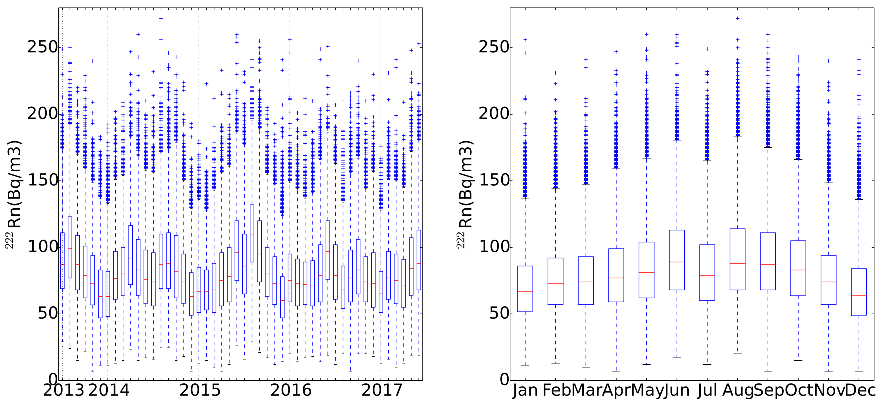
Keywords: Ensemble, Time Series Analysis, Deep Learning, Forecasting, Convolutional Neural Networks, Recurrent Neural Networks, Seasonal and Trend Decomposition using Loess

1 Introduction

Scientific laboratories generate large data volumes with relevant information about the environment in which the experiments are hosted. These data embody information that should be timely processed and provided to the experiments managers.

The modelling and forecasting of ^{222}Rn in underground laboratories are very relevant tasks. ^{222}Rn is a radionuclide produced by the ^{238}U and ^{232}Th decay chains. Being gas at room temperature, it can be emanated by the rocks and concrete of the underground laboratory, diffusing in the experimental hall. This contamination in the air is a potential source of background, both directly and through the long life radioactive daughters produced in the decay chain, which can stick to the experimental surfaces. The ^{222}Rn contamination in air can be reduced by orders of magnitude only in limited closed areas, flushing pure N_2 or "Rn-free" air produced by dedicated structures. In the deep underground laboratories the average activity depends on the local conditions and must be constantly monitored. It typically ranges from tens to hundreds of Bq/m^3, with periodic and non-periodic variations. Seasonal dependence has been observed in some cases [1, 15]. A detailed understanding of the ^{222}Rn periodicity can be fundamental for a precise comprehension of the background of rare-event search experiments. This is particularly true in case of the dark matter direct searches, whose distinctive feature is the annual modulation of the signal foreseen by the hypothesis of a weakly interactive massive particle (WIMP) halo model. At the same time, the prediction of the evolution of the ^{222}Rn concentration in the laboratory is relevant in order to correctly organize the operations foreseeing the exposure of the detector materials to the air, minimizing, in such a way, the deposition of the radionuclide on the surfaces.

The Canfranc Underground Laboratory (LSC) is composed of diverse halls for hosting scientific experiments with requirements of very low-background. The two main halls, Hall A and Hall B —which are contiguous—, have instruments for measuring the level of ^{222}Rn, particularly there is an Alphaguard P30 in each hall recording the radioactivity level every 10 minutes, with an accumulated record from July 2013 to June 2017 (Fig. 1).



(a) (b)

Fig. 1. Monthly box-plots of ^{222}Rn level at Hall A of the LSC, by year (Fig. 1(a)) and gathering the months independently of the year (Fig. 1(b)). Data taking corresponds to the period from July 2013 to June 2017. Hereafter, the monthly medians are the values used for creating the time series, and therefore, for further analyses.

With regard to the measurements, few missing values are in the data set, as well as gaps of several days in some years. The larger gaps appear in July 2014 with 913 missing values, in June 2015 with 1053, and in January 2016 with 585. In the worst case, the gap spans over a week (7.3 days). However, the missing values are not representative in comparison with the total number of observations (more than 200,000), nor the number of observations per month ($\approx$ 4,000).

^{222}Rn time series is very noisy. Only the monthly median exhibits a certain modulation (Fig. 1). Therefore, the monthly medians of the ^{222}Rn level has been selected as the monthly representative value. This critically penalizes the data volume accessible, reducing it to 48 values.

In the past, Convolutional Neural Networks (CNN) has been applied for time series classification [24, 23] and for time series prediction [21, 11, 5], including ensembles

[17]; as well as Recurrent Neural Networks (RNN) [2, 8, 20]. In [9] an review of these deep learning architectures is presented.

In this work, an ensemble of deep learning architectures (EDL) variants of Convolutional and Recurrent Neural Networks architectures are used to improve the forecasting the ^{222}Rn monthly level at LSC. The ensemble is composed of RNN, Bidirectional Recurrent Neural Networks (BRNN), CNN, and a variant of CNN, termed CNN+STL, in which the original observations used as input are replaced by the components generated by Seasonal and Trend decomposition using Loess (STL): trend, seasonal and remainder components [16].

Concerning the previous efforts in the analysis of the ^{222}Rn level at LSC, in [15] the initial efforts for preprocessing and modelling this time series using classical and deep-learning-based approaches are shown. In this paper, the times series is modelled using CNN and Recurrent Neural Networks, being the main focus on the forecasting capacity for scheduling maintenance operations of the experiment hosted at LSC, and the characterization of the annual modelation of observations.

The rest of the paper is organized as follows: a brief description the deep learning algorithms used for building the ensemble are presented in Section 2. The models comparison and the results obtained are presented and analysed in Section 3. Finally, Section 4 contains the conclusions of this work.

2 Methods and Materials

2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNN) [13, 12] are specialized Neural Networks with special emphasis in image processing [9], although nowadays they are also employed in time series analysis [5, 22]. The CNN consists of a sequence of convolutional layers, the output of which is connected only to local regions in the input. These layers alternate convolutional, non-linear and pooling-based layers which allow extracting the relevant features of the class of objects, independently of their placement in the data example. The CNN allows the model to learn filters that are able to recognize specific patterns in the time series, and therefore it can capture a richer information from the series than other methods. It also embodies three features which provide advantages over the Multilayer Perceptrons (MLP): sparse interactions, parameter sharing and equivariance to translation [9].

Although Convolutional Neural Networks are frequently associated to image or audio classification —2D grid examples— or video sequence —3D grid examples—, it can also be applied to time series analysis —1D grid examples—. When processing time series, instead of a set of images, the series has to be divided in overlapping contiguous time windows. These windows constitute the examples, where the CNN aims at finding patterns. At the same time, the application to time series modelling requires the application of 1D convolutional operators, which weights are optimized during the training process.

One of the most identifiable feature of CNN is parameter sharing. Parameter sharing allows extending and applying the model to examples of different form. Conversely, if sequence-based specialization is used, e.g. Multilayer Perceptrons (MLP), then separated parameters are generated for each value of the time index. This leads to the impossibility to generalize to sequence lengths not seen during the training process, nor share statistical strength across different sequence lengths and across different positions in time.

In this work, Keras [3] has been used for implementing the CNN and RNN architectures. The CNN employed are composed of two convolutional layers of 32 and 64 filters with `relu` as activation function, `MaxPooling1D` with size 2, and an output layer composed of a dense layer with a single neuron with linear activation function, and trained with 10 epochs. In all the algorithms checked, the loop-back parameter is configured to 12 values of the time series, the Mean Squared Error has been selected as the `loss function`, and the weights are optimized by using `Adam optimizer`.

2.2 Seasonal and Trend decomposition using Loess

The intuition behind the time series decomposition is that the time series is the composition of three more elementary series, $Y_t = T_t + S_t + R_t$. On the one hand, a trend (T_t), which is responsible of long-term increase or decrease of data. It does not have to be linear. On the other hand, a seasonal pattern is the second component (S_t). It is influenced by seasonal factors, such as: the month, the day of the week, or the quarter of the year. It has mean null in the seasonal period. Finally, the third component is the remainder or random component (R_t).

Diverse techniques for time series decomposition have been proposed. STL, *Seasonal and Trend decomposition using Loess* [4], was proposed taking into account the limitations of previous classical decomposition methods, for example X-12-ARIMA. In contrast with X-12-ARIMA, STL can handle any type of seasonality, not only monthly or quarterly; and the seasonal component can change over time, being the amount of the allowed change controlled by a parameter of the algorithm. Besides, the smoothness of the trend component can be also controlled by the algorithm.

In CNN+STL strategy, the ^{222}Rn time series is predicted with CNN using as input the three series resulting from the STL decomposition, in contrast with the use as input of the original observations [16]. This approach implies to handle three CNN for independent predictions of each of the three components time series from STL, with final merging for obtaining the prediction.

2.3 Recurrent Neural Networks

Recurrent Neural Networks (RNN) are a set of neural networks which the main purpose is to process sequential data $x^1, \ldots, x^\tau$ [18, 9, 14]. Whereas CNN aims at processing grid of data, such as images, the RNN are specialized networks for processing a sequence of data. Some example of the application of RNN to time series analysis can be found in [2, 8, 20]. Similarly to CNN, the RNN can process sequences of different length without sequence-based specialization, such as MLP.

With regards to parameter sharing, when CNN are used for analysing time series, strong similarities appear with RNN. However, in comparison with RNN, shared parameters in CNN are considered as shallow. In the convolution operation, a network shares parameters across the time among a reduced number of neighbouring members of the input. The concept of parameter sharing appears in the application of the same convolution kernel at each time step. In RNN, each member of the output is a function of the previous members of the output, and it is produced by using the same rule which has been applied to the previous outputs (Eq. 1).

$$
\begin{aligned}
h &= \sigma(W_{hh}\, h_{i-1} + W_{hx}\, x_i + b_h) \\
\hat{y}_i &= W_{yh}\, h_i
\end{aligned}
\tag{1}
$$

where x_i is the input vector, y_i is the output vector, h_i is the hidden state, W_{hx} is input-to-hidden weights, W_{hh} is hidden-to-hidden weights, W_{yh} is hidden-to-output weights, and $\hat{y}$ is the predicted values. In the current work, a single hidden layer is used, with 24 LSTM elements, and trained with 100 epoch. Hyperbolic tangent is used as activation function. The output layer is composed of a single dense layer of a neuron with linear activation function.

Long Short-Term Memory (LSTM) introduces self-loops in the RNN schema, allowing these self-loops be conditioned on the context [10]. This architecture has the same inputs and outputs as an ordinary RNN, although it has more parameters and a system of gating units that controls the flow of information.

Bidirectional Recurrent Neural Networks (BRNN) combines two RNN, a first one that moves forward in the temporal sequence, and a second one that moves back [19]. This permits to compute the prediction based on past and future observations. RNN and BRNN have been also included in the Ensemble.

2.4 Statistics

In order to ascertain if the proposed forecasting methods applied to the test set improve the prediction, two different types of tests can be applied: parametric and non-parametric. The difference between both relies on the assumption that data is normally distributed for parametric tests, whereas non explicit conditions are assumed in non-parametric tests. For this reason, the latter is recommended when the statistical model of data is unknown [7, 6]. Statistical inference is used in this work to infer which model produces better results, and if the differences are significant or not.

The Kruskal-Wallis test is a non-parametric test used to compare three or more groups of sample data. For this test, the null hypothesis assumes that the samples are from identical populations.

3 Experimental Results and Models Comparison

The collected data are divided into two sets. The training set –including the three firsts years, from July 2013 to June 2016—, and the testing data set, which includes the last twelve months, from July 2016 to June 2017.

In Fig. 2 and Table 1, the Mean Squared Error (MSE) and the Mean Absolute Error (MAE) after 25 independent runs for the proposed architectures are shown. As can be appreciated, the EDL produces, for both metrics, the best prediction for the test set, and therefore, the lowest error.

The application of the Kruskal-Wallis test to the MSE and the MAE indicates that the differences between the medians are significant for a confidence level of 95% (p-value under 0.05), which means that the differences are unlikely to have occurred by chance with a probability of 95%.

Table 1. Mean Squared Error (MSE) and Mean Absolute Error (MAE) for the deep architectures evaluated for 25 independent runs.

	MSE	MAE
RNN	65 ± 6	6.3 ± 0.2
CNN	60 ± 5	6.2 ± 0.4
CNN+STL	54 ± 5	5.8 ± 1.2
BRNN	54 ± 5	6.1 ± 0.3
EDL	45 ± 4	5.3 ± 0.4

In Fig. 3(a), the real and predicted values of the test set are shown. As can be observed the methods based on deep architectures and EDL reproduce appreciably well the test set. If comparisons with other time series forecasting methods, not involving deep architectures (Fig 3(b)), they are critically outperformed as it is appreciated [15].

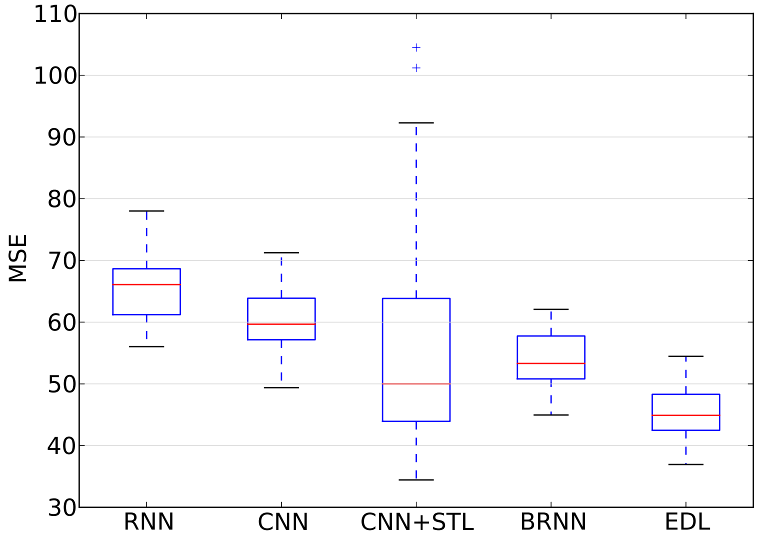
A key point is the month of August 2016. This month behaves differently that in the previous years. In general, this month has high levels of ^{222}Rn, except for the year 2016, where the value is much lower. Deep learning architectures, including EDL, are able to capture enough information from the previous values, and predict a closer value to the real observation of this month.

4 Conclusions

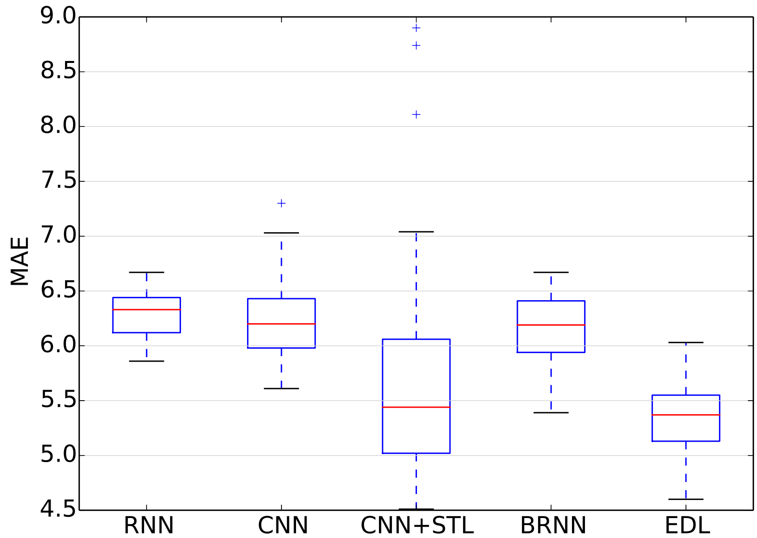
In this paper, an Ensemble Deep Learning approach is proposed to improve the prediction capacity of the ^{222}Rn level at Canfranc Underground Laboratory. The Ensemble is composed of CNN and RNN variants. They include Recurrent Neural Networks, Bidirectional Recurrent Neural Networks, Convolutional Neural Networks and STL Convlutional Neural Networks, which uses as input the series generated by the STL decomposition instead of the original observations.

The results and the statistical analysis state that the proposed Ensemble Deep Learning significantly improves the accuracy of the prediction in comparison with classical techniques used in previous works, such as ARIMA, Holt-Winters Exponential Smoothing and Seasonal and Trend Decomposition using Loess –based, but also in comparison with deep learning-based, such as Convolutional Neural Networks, Recurrent-, and Bidirectional Recurrent Neural Networks.

Finally, the analysis of the performance of alternative and more elaborated deep architectures over five years data, and their collective performance when integrating ensembles, are proposed as Future Work.

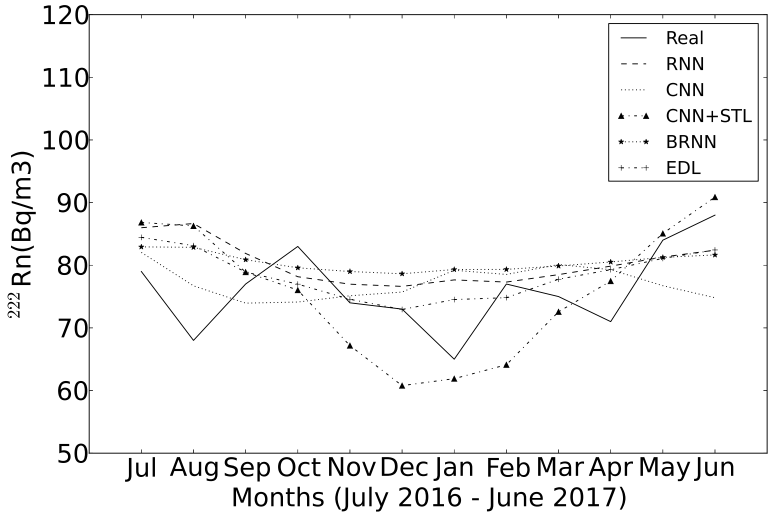


(a) MSE

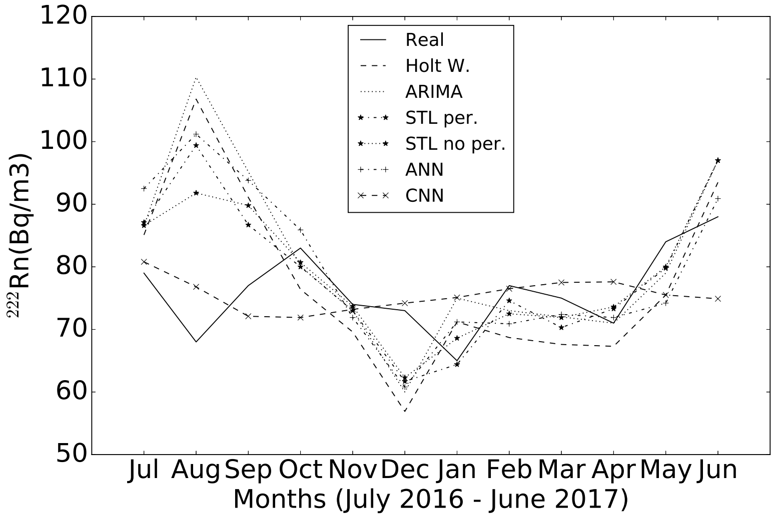


(b) MAE

Fig. 2. Mean Squared Error (MSE) and Mean Absolute Error (MAE) for the deep architectures evaluated for 25 independent runs.



(a) Forecast from SOCO'19



(b) Forecast from HAIS'18 in [15]

Fig. 3. Real values and forecasting for the test set —the fourth year, from July 2016 to June 2017— for the methods used in this study (Fig. 3(a)), and for the methods used in [15] (Fig. 3(b)).

Acknowledgment

The research leading to these results has received funding by the Spanish Ministry of Economy and Competitiveness (MINECO) for funding support through the grant FPA2016-80994-C2-1-R, and "Unidad de Excelencia María de Maeztu": CIEMAT - FÍSICA DE PARTÍCULAS through the grant MDM-2015-0509.

IMJ is co-funded in a 91.89 percent by the European Social Fund within the Youth Employment Operating Program, for the programming period 2014–2020, as well as Youth Employment Initiative (IEJ). IMJ is also co-funded through the Grants for the Promotion of Youth Employment and Implantation of Youth Guarantee in Research and Development and Innovation (I+D+i) from the MINECO.

The authors would like to thank Roberto Santorelli, Pablo García Abia and Vicente Pesudo for useful comments regarding the Physics related aspects of this work, and the Underground Laboratory of Canfranc by providing valuable feedback.

References

1. Bettini, A.: New underground laboratories: Europe, Asia and the Americas. Physics of the Dark Universe **4**(Supplement C), 36 – 40 (2014). https://doi.org/https://doi.org/10.1016/j.dark.2014.05.006, dARK TAUP2013
2. Chniti, G., Bakir, H., Zaher, H.: E-commerce time series forecasting using lstm neural network and support vector regression. In: Proceedings of the International Conference on Big Data and Internet of Thing. pp. 80–84. BDIOT2017, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3175684.3175695
3. Chollet, F., et al.: Keras. https://github.com/fchollet/keras (2015)
4. Cleveland, R.B., Cleveland, W.S., McRae, J., Terpenning, I.: STL: A seasonal-trend decomposition procedure based on loess. Journal of Official Statistics pp. 3–73 (1990)
5. Gamboa, J.C.B.: Deep learning for time-series analysis. CoRR **abs/1701.01887** (2017), http://arxiv.org/abs/1701.01887
6. García, S., Fernández, A., Luengo, J., Herrera, F.: A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. Soft Comput. **13**(10), 959–977 (2009)
7. García, S., Molina, D., Lozano, M., Herrera, F.: A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the cec'2005 special session on real parameter optimization. J. Heuristics **15**(6), 617–644 (2009)
8. Garcia-Pedrero, A., Gomez-Gil, P.: Time series forecasting using recurrent neural networks and wavelet reconstructed signals. In: 2010 20th International Conference on Electronics Communications and Computers (CONIELECOMP). pp. 169–173 (Feb 2010). https://doi.org/10.1109/CONIELECOMP.2010.5440775
9. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735
11. Lago, J., Ridder, F.D., Schutter, B.D.: Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms. Applied Energy **221**, 386 – 405 (2018). https://doi.org/https://doi.org/10.1016/j.apenergy.2018.02.069, http://www.sciencedirect.com/science/article/pii/S030626191830196X
12. LeCun, Y.: Generalization and network design strategies. Tech. rep., University of Toronto (1989)

13. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (Nov 1998). https://doi.org/10.1109/5.726791
14. Lipton, Z.C.: A critical review of recurrent neural networks for sequence learning. CoRR **abs/1506.00019** (2015), http://arxiv.org/abs/1506.00019
15. Méndez-Jiménez, I., Cárdenas-Montes, M.: Modelling and forecasting of the ^{222}Rn radiation level time series at the Canfranc Underground Laboratory. In: Hybrid Artificial Intelligent Systems - 13th International Conference, HAIS 2018, Oviedo, Spain, June 20-22, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10870, pp. 158–170. Springer (2018)
16. Méndez-Jiménez, I., Cárdenas-Montes, M.: Time series decomposition for improving the forecasting performance of convolutional neural networks. In: Advances in Artificial Intelligence - 18th Conference of the Spanish Association for Artificial Intelligence, CAEPIA 2018, Granada, Spain, October 23-26, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11160, pp. 87–97. Springer (2018). https://doi.org/10.1007/978-3-030-00374-6_9
17. Qiu, X., Zhang, L., Ren, Y., Suganthan, P.N., Amaratunga, G.A.J.: Ensemble deep learning for regression and time series forecasting. In: 2014 IEEE Symposium on Computational Intelligence in Ensemble Learning, CIEL 2014, Orlando, FL, USA, December 9-12, 2014. pp. 21–26 (2014). https://doi.org/10.1109/CIEL.2014.7015739, https://doi.org/10.1109/CIEL.2014.7015739
18. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (Oct 1986). https://doi.org/10.1038/323533a0
19. Schuster, M., Paliwal, K.: Bidirectional recurrent neural networks. Trans. Sig. Proc. **45**(11), 2673–2681 (Nov 1997). https://doi.org/10.1109/78.650093
20. Walid, Alamsyah: Recurrent neural network for forecasting time series with long memory pattern. Journal of Physics: Conference Series **824**(1), 012038 (2017), http://stacks.iop.org/1742-6596/824/i=1/a=012038
21. Wang, H.Z., Li, G.Q., Wang, G.B., Peng, J.C., Jiang, H., Liu, Y.T.: Deep learning based ensemble approach for probabilistic wind power forecasting. Applied Energy **188**, 56 – 70 (2017). https://doi.org/https://doi.org/10.1016/j.apenergy.2016.11.111, http://www.sciencedirect.com/science/article/pii/S0306261916317421
22. Wang, Z., Yan, W., Oates, T.: Time series classification from scratch with deep neural networks: A strong baseline. CoRR **abs/1611.06455** (2016), http://arxiv.org/abs/1611.06455
23. Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L.: Time series classification using multi-channels deep convolutional neural networks. In: Li, F., Li, G., Hwang, S.w., Yao, B., Zhang, Z. (eds.) Web-Age Information Management. pp. 298–310. Springer International Publishing, Cham (2014)
24. Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L.: Exploiting multi-channels deep convolutional neural networks for multivariate time series classification. Frontiers Comput. Sci. **10**(1), 96–112 (2016). https://doi.org/10.1007/s11704-015-4478-2