An Ensemble of Attentive Recurrent Networks with Randomized Dilations for Forecasting

Paweł Pełka[1 [0000−0002−2609−811X]], Grzegorz Dudek[1 [0000−0002−2285−0327]], and Slawek Smyl[2 [0000−0003−2548−6695]]

[1] Electrical Engineering Faculty, Czestochowa University of Technology, Poland
{pawel.pelka,grzegorz.dudek}@pcz.pl
[2] slawek.smyl@gmail.com

Abstract. In this work, we propose an ensemble of recurrent neural networks (RNNs) equipped with attention and dilation mechanisms for forecasting time series (TS) with multiple seasonality. Recurrent cells implement a dilation to better capture complex long-term and seasonal dependencies and an attention for dynamic weighting of input vector components. We stack multiple dilated recurrent layers with hierarchical dilations in order to learn temporal dependencies of different scales at different layers. The novelty of this approach is that additional diversity among learners is generated by randomly delayed connections. The model produces both point forecasts and predictive intervals (PIs).

Keywords: Multiple seasonality · Recurrent NNs · TS forecasting.

1 Introduction

TS forecasting is a challenging problem when a TS expresses multiple seasonality, nonlinear trend and varying variance. NNs can flexibly model complex nonlinear relationships in TS and reflect process variability in uncertain dynamic environments [1]. RNNs, which were designed for sequential data, are extremely useful for TS forecasting [2]. They are able to capture both short- and long-term dynamics thanks to their internal memory and gating mechanism.

In this work, we explore a new RNN equipped with dilation and attention mechanisms, which was proposed in [4]. It was developed for TS with multiple seasonality. To improve the accuracy and generalization property of the model, we propose ensembling with additional diversity among learners generated by randomly delayed connections.

2 Forecasting Model

Forecasting problem. We formulate the forecasting problem given a length n forecast horizon and a length M observed TS, $\{z_\tau\}_{\tau=1}^{M}$. To make the considerations more concrete, we focus on the problem of short-term electrical load forecasting (STLF) expressing triple seasonality: yearly, weekly and daily (see

[5] for details). Our goal is to forecast the daily profile (24 hours) for the next day based on historical loads.

Data preprocessing. As the main input information, we introduce a weekly profile, which precedes the forecasted day. This profile is standardized as follows [4]: $\mathbf{x}_t = (\mathbf{z}_t^w - \overline{z}_t^w)/std(z_t^w)$, where $\mathbf{z}_t^w \in \mathbb{R}^{168}$ is the original sequence of the t-th week, $\mathbf{x}_t \in \mathbb{R}^{168}$ is its standardized pattern, and $\overline{z}_t^w$ and $std(z_t^w)$ are its mean and standard deviation, respectively. To enrich the input information, input vector $\mathbf{x}'_t$ is composed of five elements: $\mathbf{x}_t$, $\log_{10}(\overline{z}_t^w)$, which informs about the level of the TS, $\mathbf{d}_t^w \in \{0,1\}^7, \mathbf{d}_t^m \in \{0,1\}^{31}$ and $\mathbf{d}_t^y \in \{0,1\}^{52}$, which are binary one-hot vectors encoding day of the week, day of the month and week of the year.

An output pattern encodes the forecasted daily sequence as follows: $\mathbf{y}_t = (\mathbf{z}_t^d - \overline{z}_t^w)/std(z_t^w)$, where $\mathbf{y}_t \in \mathbb{R}^{24}$ is the t-th daily pattern and $\mathbf{z}_t^d \in \mathbb{R}^{24}$ is the forecasted sequence. The equation for decoding is: $\mathbf{z}_t^d = \mathbf{y}_t std(z_t^w) + \overline{z}_t^w$.

Gated recurrent cell. The proposed RNN employs the dilated RNN cell with attention (adRNNCell) introduced in [3]. It is shown in Fig. 1a) and the corresponding equations are shown in Fig. 1b). adRNNCell was designed for TS with complex seasonality. Its distinguishing features are as follows. First, it combines two dRNNCells [5] to obtain a more efficient cell, which is able to preprocess dynamically the input data. It introduces an attention for weighting the input information. The bottom dRNNCells in Fig. 1a) produces attention vector $\mathbf{m}_t$, whose components weight the inputs to the upper dRNNCell. Vector $\mathbf{m}_t$ has a dynamical character. It is adjusted to the current inputs at time t. Second, both dRNNCells are fed by two cell states $\mathbf{c}$ and two controlling states $\mathbf{h}$, i.e. recent states, $\mathbf{c}_{t-1}$, $\mathbf{h}_{t-1}$, and delayed states, $\mathbf{c}_{t-d}$, $\mathbf{h}_{t-d}$, $d > 1$. Third, the outputs of dRNNCells are split into "real output", $\mathbf{m}_t$ or $\mathbf{y}_t$, and a controlling output $\mathbf{h}_t$, which is an input for the gating mechanism in the following time steps.

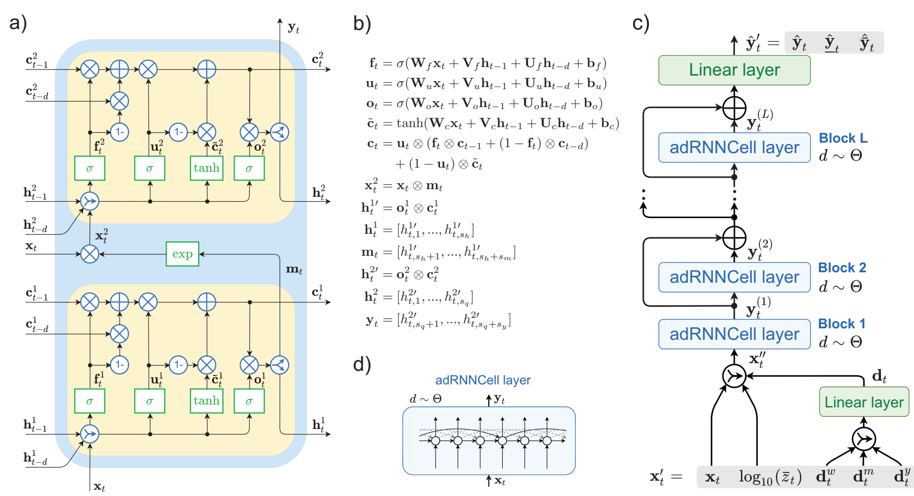

Fig. 1. Proposed RNN architecture and details.

RNN architecture. The proposed RNN extends the architecture proposed in [3] as follows. First, the number of single-layer blocks is not limited to 3. The depth of RNN is controlled by hyperparameter L. Second, the dilation of each block is selected randomly. The distribution of dilations is defined as $\Theta = \{(d_i, p_i)\}_{i=1}^{m}$, where $d_i \in \{2, 3, ...\}$ denotes a dilation, $p_i > 0$ denotes a probability ($\sum_{i=1}^{m} p_i = 1$), and m is an assumed number of possible dilations. We use an ensemble of K RNNs with dilations selected independently. Random dilations generate additional diversity among learners (random initialization is the second source of diversity).

The RNN architecture is depicted in Fig. 1c) and the adRNNCell layer is shown in Fig. 1d). The input linear layer reduces the dimensionality of the calendar variables by embedding them in r-dimensional continuous vector $\mathbf{d}_t$. The output linear layer at the top of the stacked recurrent layers, produces the point forecasts, $\hat{\mathbf{y}}_t$, and two vectors of quantiles, a lower one, $\underline{\hat{\mathbf{y}}}_t \in \mathbb{R}^{24}$, and an upper one, $\hat{\overline{\mathbf{y}}}_t \in \mathbb{R}^{24}$. These quantiles of assumed orders, $\underline{q}$ and $\overline{q}$, define the PI. RNN uses ResNet-style shortcuts between blocks to improve the learning process.

Loss function. To enable RNN to learn both point forecasts and PI quantiles, we employ the following loss function [5]:

$$L = \rho(y, \hat{y}_{q^*}) + \gamma(\rho(y, \hat{y}_{\underline{q}}) + \rho(y, \hat{y}_{\overline{q}})) \tag{1}$$

where $\rho(y, \hat{y}_q) = (y - \hat{y}_q)(q - \mathbf{1}_{(y<\hat{y}_q)})$ is a pinball loss, $q \in (0, 1)$ is a quantile order, y is an actual value (standardized), $\hat{y}_q$ is a forecasted value of q-th quantile of y, $q^* = 0.5$ corresponds to the median, $\underline{q} \in (0, q^*)$ and $\overline{q} \in (q^*, 1)$ correspond to the lower and upper bound of PI, respectively, and $\gamma \geq 0$ is a control parameter.

3 Experimental Study

In this section, we use ENTSO-E electricity demand dataset (www.entsoe.eu/data/power-stats) to verify the effectiveness of the proposed model. The dataset details the hourly loads in the period 2006-18 for 35 European countries. The RNN was optimized on data from the period 2006-17 and tested on data from 2018.

We use similar a training and optimization setup as in [4]. Based on experimentation, we assumed the dilations: $d = 2, 3, 5$ and 7 with the corresponding probabilities: $p = 0.5882, 0.1765, 0.0588$ and 0.1765. Number of blocks was $L = 3$, number of ensemble members was $K = 5$, and embedding size was $r = 6$.

Table 1 compares the forecasting quality metrics of the proposed model (ardRNNe) with the baseline models [3]. The results clearly show that ardRNNe outperforms its competitors in terms of accuracy. This was confirmed by a Giacomini-White test for conditional predictive ability. IQR(APE) shows that ardRNNe also produces the least dispersed forecasts compared to its competitors.

Fig. 2 shows the ranking of the examined models based on average MAPE for each country. Note the highest position of ardRNNe, which achieved the lowest error for 21 out of 35 countries and the second lowest error for 13 countries.

Table 1. Forecasting quality metrics.

Model	MAPE	Median(APE)	IQR(APE)	RMSE
Naive	5.08	4.84	3.32	704.34
ARIMA	3.30	3.01	3.00	475.09
ES	3.11	2.88	2.73	439.26
Prophet	4.53	4.32	3.03	619.39
FNM	2.50	2.30	2.29	334.08
GRNN	2.48	2.28	2.27	332.91
MLP	3.05	2.78	2.94	419.01
SVM	2.55	2.29	2.52	357.24
LSTM	2.76	2.57	2.52	381.76
ANFIS	3.65	3.17	3.66	507.08
MTGNN	2.99	2.74	2.69	405.18
ES-adRNNe	2.14	1.94	2.09	292.16
ardRNNe	**2.11**	**1.90**	**2.06**	**287.07**

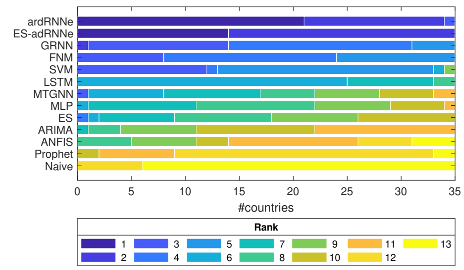

Fig. 2. MAPE ranking.

4 Conclusion

The experimental study performed on a challenging STLF problem with multiple seasonality showed that our proposed model outperforms, in terms of accuracy, its competitors including statistical, ML and hybrid models. Its superior performance is due to its hierarchical architecture, which learns temporal dependencies of different scales at different layers as well as ensembling with diversity generated by randomly dilayed connections. The model is equipped with many mechanisms and procedures designed to increase forecasting efficiency. They include: (i) new recurrent cell with dilation and attention mechanisms, which help in modeling seasonal dependencies as well as selecting input information, (ii) cross-learning, which enables RNN to capture the shared features of the individual series and prevents over-fitting, (iii) a composed asymmetrical loss function based on quantiles, which enables RNN to produce both point forecasts and PI, and also to reduce the forecast bias, and (iv) encoding the output sequences using variables determined from recent history, which helps to capture the current dynamics of the process.

References

1. Benidis, K. et al.: Neural forecasting: Introduction and literature overview. arXiv:2004.10240 (2020)
2. Hewamalage, H., Bergmeir, C., Bandara, K.: Recurrent neural networks for time series forecasting: Current status and future directions. International Journal of Forecasting. **37**, 388–427 (2021)
3. Smyl, S., Dudek, G., Pełka, P.: ES-dRNN with dynamic attention for short-term load forecasting. arXiv:2203.00937 (2022)
4. Dudek, G., Smyl, S., Pełka, P.: Recurrent neural networks for forecasting time series with multiple seasonality: A comparative study. In: 8th International Conference on Time Series and Forecasting (submitted)
5. Smyl, S., Dudek, G., Pełka, P.: ES-dRNN: A hybrid exponential smoothing and dilated recurrent neural network model for short-term load forecasting. arXiv:2112.02663 (2021)