ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2.
Available from http://www.i6doc.com/en/.

# Handling missing data in recurrent neural networks for air quality forecasting

Michel Tokic[1], Anja von Beuningen[1], Christoph Tietz[1]
and Hans-Georg Zimmermann[2]

[1] Siemens AG - Otto-Hahn-Ring 6, 81379 München - Germany
[2] Fraunhofer IIS - Nordostpark 93, 90411 Nürnberg - Germany

**Abstract**. Practical applications of air quality forecasting, which typically provide predictions over a horizon of hours and days, often require the handling of missing data due to unobserved relevant variables, sensor defects or communication outages. In this paper we discuss two aspects being important when building air quality forecasting models for essential air pollution parameters such as particular matter and nitrogen dioxides. Using a specialized architecture of a recurrent neural network, we can build models even if (1) unobserved variables or (2) missing data are present.

## 1 Introduction

In the past years, most urban areas experienced periods of increased air pollution mainly due to emissions from motorized vehicles, industry, agriculture and household firings. Air pollutants affect the environment and human health by causing cardio-vascular problems, lung diseases or cancer [1]. Especially, particulate matter with diameters less than $10\,\mu m$ ($PM_{10}$) as well as nitrogen dioxide ($NO_2$) are associated with damages of community health and environment. In the WHO Ambient Air Pollution database the annual mean of $PM_{10}$ of 1600 cities in 91 countries have been reported for the period 2008 to 2013. The average of $PM_{10}$ concentration is $71\,\mu g/m^3$, while a value of $20\,\mu g/m^3$ is recommended. For air pollution control the realtime hourly prediction of air pollutant concentrations is necessary. In this paper we focus on the hourly prediction of $PM_{10}$ and $NO_2$ for the next 48 hours. However, the proposed methods can easily be adapted to other air pollutants and timeframes.

Several methods for forecasting air pollution have been proposed in literature [1–6]. Because physical simulation approaches require detailed knowledge about chemical and physical processes in our atmosphere, data driven methods, which are based on measurements only, are more popular in air quality forecasting.

All following publications train and evaluate their models on measurements in urban areas. The training and test data include air pollutant concentration measurements, meteorological conditions and forecasts. Some models also use calendar and/or geospatial information as input.

Since air pollution depends on a non-linear system including the air mass transport and meteorological models, linear methods usually are not able to predict air pollution accurately. A more accurate and widely used technique for the forecasting of air pollutants utilizes artificial neural networks (ANN). They are able to model even complex, non-linear systems without detailed knowledge about the underlying physical system. The authors of [2] implemented several different neural network models for the prediction of $PM_{10}$ and $NO_2$ concentrations and compared those with a linear and a deterministic model, where the ANNs outperformed the other approaches in terms of accuracy.

Catalano et al. [1] predicted the hourly mean concentration of $NO_2$ using three- and four-layer MLPs and a seasonal ARIMA model. While the ARIMA model required theoretical assumptions, the MLP models were not able to predict peaks accurately. Hooyberghs et al. [3] implemented an ANN for forecasting the daily average of $PM_{10}$ one day ahead. Their study about the importance of input variables showed that meteorological conditions significantly influence the $PM_{10}$ concentration levels.

Freeman et al. [4] showed that an RNN including a long short-term memory (LSTM) module performs slightly better than MLP or SVM models. Similarly, the authors of [5], who used an Elman model to forecast the daily maximum concentration of different air pollutants, showed that RNNs outperform MLP and linear regression models.

In real world model applications such as the *Siemens City Air Management* (CyAM) tool[1] we have to cope with unreliable or missing sensor measurements. Only a few studies consider the problem of missing data. Some studies discard missing data of more than a predefined time interval [6] or interpolate missing data linearly [4]. In this paper we present an RNN model for air quality forecasting, which explicitly handles missing data using internal forecasts.

## 2 Methods for modeling air quality

Building forecasting models for the prediction of air quality requires an identification of the underlying dynamical system of the observed pollution data. In general, recurrent neural networks (RNNs) are a proven choice to identify and describe dynamical systems.

### 2.1 Historically consistent neural networks with architectural teacher forcing

Conventional RNNs for forecasting use external drivers as input neurons in the past part (e.g. meteorological or traffic conditions) and assume a negligible environmental influence in the future.

The historically consistent neural network (HCNN) [7] overcomes this conceptual weakness, by not only modeling the output of interest, i.e. the air pollutant concentrations, but also the external drivers such as weather and traffic data:

$$\begin{aligned} \text{state transition} \quad & s_{t+1} = A\tanh(s_t) \\ \text{output equation} \quad & y_t = [\ \underbrace{Id}_{\text{outputs of interest}}\ ,\ \underbrace{0}_{\text{unobservable external drivers}}\ ]s_t \end{aligned} \tag{1}$$

[1] https://new.siemens.com/global/en/company/topic-areas/intelligent-infrastructure/city-performance-tool.html

The air pollutant concentrations of the past ($y_t \in \mathbb{R}^N$, $t < 0$) are represented in the past states $s_t \in \mathbb{R}^K, K > N$, while the air pollutant concentrations of the future ($y_t \in \mathbb{R}^N$, $t > 0$) are non-observable variables represented in the future states $s_t \in \mathbb{R}^K$. The fixed matrix $[Id, 0]$ extracts the $N$ observables from the state $s_t \in \mathbb{R}^K$. E.g. only the first $N$ elements of the state vector $s_t$ are observable, while the other $K - N$ elements are hidden variables.

Missing input neurons and the unfolding across the complete time horizon make HCNNs difficult and time-consuming to train. Therefore, architectural teacher forcing (ATF) for HCNNs as shown in Fig. 1 has been introduced in [8]. Note, that the values of two incoming arrows into a single node in Fig. 1 are summed up. The output layers of the standard HCNN are modified to represent a fixed target value of zero (illustrated as $tar = 0$ in Fig. 1). Up to the present time $t = 0$ the expected values $y_t$ of the air pollutant concentrations in the state vector are replaced with the actual observed air pollutant concentrations $y_t^d$. Since there are no observed air pollutant concentrations for future time steps $t > 0$, the HCNN architecture for the future time steps cannot use ATF and is kept unchanged.

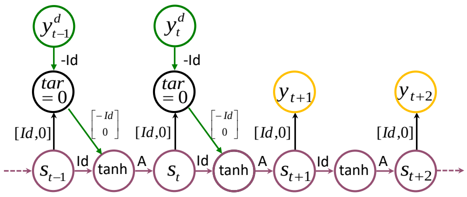

Fig. 1: The Historically Consistent Neural Network (HCNN) with architectural teacher forcing (ATF).

## 2.2 Handling of unobserved variables in HCNNs

For the prediction of air quality and in contrast to [8] we use *Teacher Forcing* not only for the outputs but also for the external drivers, which are available also in the future part of the neural network. In more detail, our neural network model consists of three parts:

The observed **Air pollutant concentrations** as model outputs $y_t^d \in \mathbb{R}^N$ are only available in the past part ($T_p \leq t \leq 0$). We focus on particulate matter with a diameter less than $10\mu m$ ($\mathrm{PM}_{10}$) as well as nitrogen dioxide ($\mathrm{NO}_2$), but the model can easily be adapted to other air pollutants.

The **External drivers** as model inputs $u_t^d \in \mathbb{R}^M$ are available in the past and future part ($T_p \leq t \leq T_f$). Here we focus on calendar information (such as hour, weekday, month), the presence of holidays and special events (i.e. christmas), which may help to explain different traffic situations, and the weather data including temperature, humidity, solar radiation, and cloud cover. The weather data is obtained from sensor measurements for the past ($t \leq 0$) and from a commercial weather forecast for the future ($t > 0$).

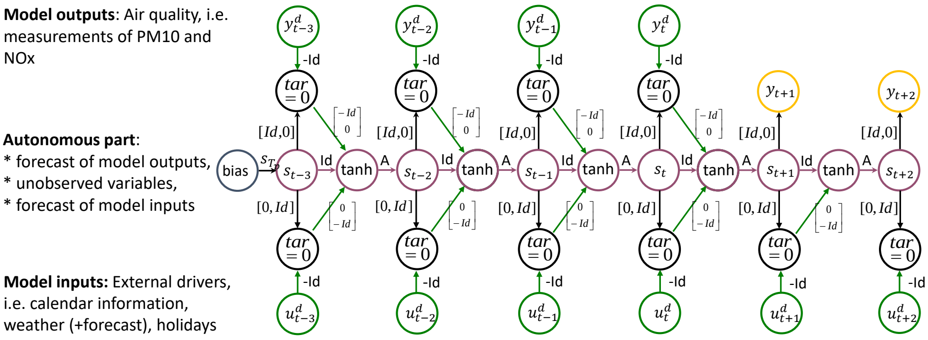
Model outputs: Air quality, i.e. measurements of PM10 and NOx
Autonomous part:
* forecast of model outputs,
* unobserved variables,
* forecast of model inputs
Model inputs: External drivers, i.e. calendar information, weather (+forecast), holidays
$y^d_{t-3}$
$y^d_{t-2}$
$y^d_{t-1}$
$y^d_t$
$y_{t+1}$
$y_{t+2}$
-Id
$tar = 0$
$[Id,0]$
$[0,Id]$
$\begin{bmatrix}-Id\\0\end{bmatrix}$
$\begin{bmatrix}0\\-Id\end{bmatrix}$
bias
$s_{T_p}$
$s_{t-3}$
$s_{t-2}$
$s_{t-1}$
$s_t$
$s_{t+1}$
$s_{t+2}$
Id
A
tanh
$u^d_{t-3}$
$u^d_{t-2}$
$u^d_{t-1}$
$u^d_t$
$u^d_{t+1}$
$u^d_{t+2}$

Fig. 2: The extended HCNN architecture comprising *external drivers* as input $u$, the *autonomous part* as hidden state $s$ and *air pollutant concentrations* as output $y$ of the neural network.

Furthermore, the **Autonomous part** as hidden state $s_t \in \mathbb{R}^K, K > N + M$ is available in the past and future part ($T_p \leq t \leq T_f$) and contains the internal forecast of model input $u_t$, reconstructed unobserved variables (e.g. time-dependent emissions of traffic) and the internal forecast of the model output $y_t$. Note that the activation function *tanh* is also applied to the errors between observed and predicted inputs/outputs within the *autonomous part* of the neural network. Using this approach, we constantly correct the forecast within the state-estimation (past) part of the neural network. Technically, when computing the forecast of air quality in the future part of the neural network, at $t > 0$, the error for $y_t$, fed into the non-linear activation function ($tanh$), is set to zero, which means that the neural network fully trusts its own forecast based on the estimated state $s_{t-1}$.

Note that the initial state $s_{T_p}$ and the connector matrix $A \in \mathbb{R}^{K \times K}$ are the only free parameters to be optimized.

### 2.3 Handling of missing data in HCNNs

In the context of air quality forecasting we often face the problem of missing data, for example, due to sensor outages or sensory misbehavior, i.e. we receive data from an unrealistic range. Trivial heuristics for handling missing data often involve assuming a constant behavior of the variable or using linear interpolation methods [4]. When using HCNN models, the handling of missing data is possible from within the neural network architecture without the need to apply trivial heuristics at the application layer [9].

The HCNN architecture with missing data handling is depicted in Fig. 3 and called HCNN-MD in the following. The vector $\text{mask}^y_t \in \mathbb{R}^N$ indicates the validity of the observations $y^d_t \in \mathbb{R}^N$. Compared to Fig. 2, the architectural teacher forcing mechanism is extended such that the error between the observable in $y^d_t$ and its forecast values in $y_t$ is masked if the observable is missing or has an implausible value.

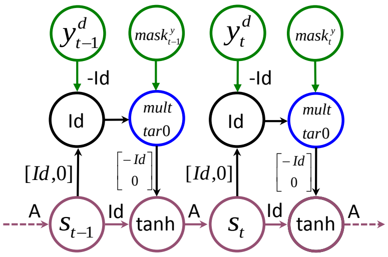

Fig. 3: HCNN with missing data handling (HCNN-MD)

Through the described mechanism of setting the error of a sensor value in $y^d_t$ to zero, the neural network forecast of $s_{t+1}$ is only based on the internal dynamics of state $s_t$ at time $t$. This mechanism is applicable to each teacher forcing node. Missing data in the external drivers $u_t$, e.g. due to a defect sensor for measuring wind speed or temperature, are handled analogously.

## 3 Experiments

We evaluate the presented methods from Sec. 2 on historical air quality data of a large city in Germany. The complete dataset comprises hourly data from the years 2008–2018. The data of years ranging from 2008–2016 are taken as training and validation datasets and the data of the remaining years ranging from 2017–2018 are used as the test dataset. The neural network architecture unfolds $T_p = 120$ hours (5 days) into the past and produces air quality forecasts for $T_f = 48$ hours (2 days) into the future on an hourly grid.

For learning we use error backpropagation through time (EBTT) with learning rate $\eta = 5 \cdot 10^{-4}$ for a maximum of 9000 epochs in combination with early stopping if the test error does not further drop within 1000 epochs. Our results are based on 10 independent neural network trainings for both architectures (HCNN and HCNN-MD), where the hidden state $s$ consists of $K = 50$ neurons.

Missing test data is created artificially through masking out teacher forcing inputs of emissions ($\text{mask}^y_i = 0$) and external drivers ($\text{mask}^x_j = 0$) for consecutive blocks of 1, 2, 5 and 10 hours, which refers to 36%, 43%, 58% and 70% of the data. Fig. 4 shows the evaluation results of the emissions forecast for $PM_{10}$ and $NO_2$ at hour $t = +5$ on the generalization dataset for years 2017 to 2018. Compared to a standard HCNN approach, where the missing sensor values are kept constant, our new model (HCNN-MD) significantly reduces the mean absolute error. Using an HCNN-MD, which predicts the missing data within its architecture, the forecast based on the internal dynamics of hidden state $s$ is stabilized when the rate of missing data is increased — even for ten consecutive hours of missing data.

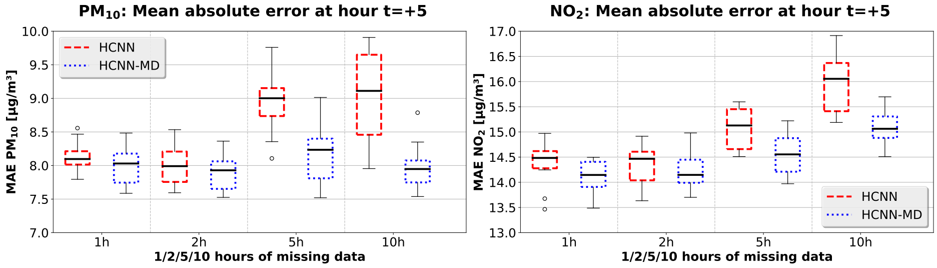

Fig. 4: Evaluation results of the emissions forecast at hour $t = +5$.

## 4 Conclusion

In this paper we showed how recurrent neural networks can be used in air quality forecasting that can inherently cope with unobserved variables and different sources of missing data, i.e. due to sensor failures or outages. In contrast to state-of-the-art approaches using linear interpolation techniques, we utilize an RNN with a special architecture being able to significantly reduce the prediction error even for long intervals of missing data.

## References

[1] M. Catalano, F. Galatioto, M. Bell, A. Namdeo, and A. S. Bergantino. Improving the prediction of air pollution peak episodes generated by urban transport networks. *Environmental Science & Policy*, 60:69 – 83, 2016.

[2] J. Kukkonen, L. Partanen, A. Karppinen, J. Ruuskanen, H. Junninen, M. Kolehmainen, H. Niska, S. Dorling, T. Chatterton, R. Foxall, and G. Cawley. Extensive evaluation of neural network models for the prediction of NO2 and PM10 concentrations, compared with a deterministic modeling system and measurements in central Helsinki. *Atmospheric Environment*, 37:4539–4550, 10 2003.

[3] J. Hooyberghs, C. Mensink, G. Dumont, F. Fierens, and O. Brasseur. A neural network forecast for daily average PM10 concentrations in Belgium. *Atmospheric Environment*, 39(18):3279 – 3289, 2005.

[4] B. Freeman, G. Taylor, B. Gharabaghi, and J. Thé. Forecasting air quality time series using deep learning. *Journal of the Air & Waste Management Association*, 68, 11 2017.

[5] F. Biancofiore, M. Busilacchio, M. Verdecchia, B. Tomassetti, E. Aruffo, S. Bianco, S. Di Tommaso, C. Colangeli, G. Rosatelli, and P. Di Carlo. Recursive neural network model for analysis and forecast of PM10 and PM2.5. *Atmospheric Pollution Research*, 8(4):652 – 659, 2017.

[6] X. Feng, Q. Li, Y. Zhu, J. Hou, L. Jin, and J. Wang. Artificial neural networks forecasting of PM2.5 pollution using air mass trajectory based geographic model and wavelet transformation. *Atmospheric Environment*, 107:118 – 128, 2015.

[7] A. M. Schäfer and H.-G. Zimmermann. Recurrent neural networks are universal approximators. In S. D. Kollias, A. Stafylopatis, W. Duch, and E. Oja, editors, *Artificial Neural Networks – ICANN 2006*, pages 632–640, Berlin, Heidelberg, 2006. Springer.

[8] H.-G. Zimmermann, C. Tietz, and R. Grothmann. Forecasting with recurrent neural networks: 12 tricks. In G. Montavon, G. B. Orr, and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade: Second Edition*, pages 687–707. Springer, Berlin, Heidelberg, 2012.

[9] K. Heesche, S. Vogl, and H.-G. Zimmermann. Modellbasierte ermittlung eines systemzustandes mittels eines dynamischen systems. Patent WO2017207317A1, 2017.