JONAS MACHADO MIGUEL

MACHINE LEARNING-BASED SPATIO-TEMPORAL FORECASTING OF WIND POWER GENERATION

São Paulo
2020

JONAS MACHADO MIGUEL

MACHINE LEARNING-BASED SPATIO-TEMPORAL FORECASTING OF WIND POWER GENERATION

Trabalho apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Engenheiro Mecânico.

São Paulo
2020

JONAS MACHADO MIGUEL

MACHINE LEARNING-BASED SPATIO-TEMPORAL FORECASTING OF WIND POWER GENERATION

Trabalho apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Engenheiro Mecânico.

Orientador:

Prof. Dr. Fábio Gagliardi Cozman

Co-orientador:

Dr. Alexandre Cristovão Maiorano

São Paulo
2020

Catalogação-na-publicação

Miguel, Jonas
Machine Learning-based Spatio-Temporal Forecasting of Wind Power Generation / J. Miguel -- São Paulo, 2020.
80 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecânica.

1.Aprendizado de Máquina 2.Energia Eólica 3.Predição 4.Estatística I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecânica II.t.

To my parents,
Marcionete and Pedro,
for their love and courage.

RESUMO

Predizer o comportamento de sistemas regidos por correlações temporais e espaciais é uma tarefa a que se tem atribuída crescente importância em diversas áreas de aplicação, desde neurociência, epidemiologia e criminologia a logística e transporte. Neste trabalho, delineamos o estado da arte para métodos de predição espaço-temporal e implementamos uma seleção desses métodos para a predição de geração de energia eólica no nível distrital na Alemanha. Na análise, levamos em conta tanto séries temporais com resolução horária entre 2000 e 2015, como também especificações de projeto e de instalação de turbinas eólicas individuais. Os modelos são avaliados em períodos não modelados e comparados com métodos estatísticos de previsão.

Palavras-Chave – Análise de Séries Temporais, Predição Espaço-Temporal, Aprendizagem de Máquina, Redes Neurais, Energias Renováveis, Energia Eólica. Palavra, Palavra.

ABSTRACT

Forecasting the behavior of systems in which both temporal and spatial dependencies play a central role has received increased attention, with applications domains including neuroscience, epidemiology, criminology and transportation. We review the state-of-the-art for spatio-temporal forecasting methods and implement selected approaches for predicting wind power generation at the district-level in Germany. Besides hourly time series for power generation in individual districts in 2000-2015, the analysis considers design and installation specifications for single wind turbines. The models are evaluated on unmodelled periods and locations and benchmarked against conventional statistical time series forecasting methods.

Keywords – Time Series Analysis. Spatio-Temporal Forecasting. Machine Learning, Neural Networks, Wind Power.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

Phenomena presenting high socio-economical relevance which are governed by complex dependencies of both spatial and temporal nature are found in diverse domains such as epidemiology, criminology, transportation, climate science and astrophysics [AKK17]. Indeed, the ability to describe a system's behavior is most valuable on instances downstream in the arrow of time: forecasting [Sco02]. Accurate, scalable and feasible rule-based forecasting modeling, however, remains elusive in many cases. Especially as ubiquitous and continuous monitoring data become available, data-driven approaches emerge as a promising alternative.

Conventional data-driven approaches alone, however, have often shown to add limited value in spatio-temporal forecasting [**Makridakis2018**]. A major reason for this limitation lies on the assumptions they rely upon being typically violated in spatio-temporal settings. Stationarity assumption most of the statistical approaches from time series analysis, while earlier machine learning methods assume data instances are independent and identically distributed (i.i.d.) [AKK17]. Recently, deep learning-based approaches have shown to be able to overcome this essentially by (a) modelling both spatial and temporal dependencies and (b) considering spatial similarities in terms less obvious than geographical proximity alone [Li+17; Gen+17; Wu+19].

In the context of renewables, accurately estimating power generation ahead of time poses a major obstacle in progressing towards carbon neutrality in power generation. Heavily conditioned on weather and climate, harvesting energy from renewable sources is characterized by intermittency. Wind power generation, for instance, depends primarily on local wind speeds, which heavily vary in both time and space. Climate changes further aggravates this character, as wind speeds variability are expected to increase [Moe+18]. Not accurately knowing how much wind power will be harvested in a certain time and region means power providers have to rely on unnecessarily larger safety margins provided by conventional power plants for ensuring sufficient power supply. This ultimately hampers the expansion of wind farms and represents therefore a loss for the society, as part of the paid overall generated power is lost, as well as for the environment, as less environment-friendly power sources have to be relied upon [Del+15].

For countries committed to large-scale initiatives such as the *Energiewende* in Germany, this poses a major hindrance in decreasing overall carbon footprint in a sustainable fashion. Accuracy on wind power generation forecasting hence has significant impact on both socio-economical and environmental aspects, in both short and long terms.

1.1 Problem Statement

In spatio-temporal problems, observations of a variable of interest over neighboring locations present not only temporal but also spatial dependencies. While local time series can be predicted individually using conventional univariate statistical techniques, information contained in their spatial and spatio-temporal correlations represent a potential for improving forecasting accuracies. More sophisticated models that allow capturing these dependencies require however supporting evidence on their potential gains that justify the tipicaly longer development times they entail.

1.2 Our Hypothesis

We hypothesize that, in use cases dominated by spatio-temporal dependencies, significant forecasting performance gains can be achieved by spatio-temporal, multi-variate, Machine Learning-based approaches.

1.3 Our Contribution

First, we delineate the state-of-the-art approaches for temporal and spatio-temporal forecasting in different domains, including statistical, machine learning-based approaches. Second, we apply selected approaches for forecasting weekly regional wind power generation in Germany. By benchmarking against more conventional temporal, univariate, statistical approaches, we investigate to what extent more sophisticated modeling approaches add value in terms of accuracy in the use case of onshore wind power generation.

2 BACKGROUND

2.1 Liberalized Electricity Markets

In a liberalized electricity market, multiple entities are involved in supplying energy to final consumers, as 1 illustrates. In the EU, these parties are electricity generators, transmission system operators (TSO), distribution system operator (DSO), electricity supplier, and regulator [Lar+10]. TSOs are responsible for long-distance transport of energy and for balancing supply and demand in timeframes under quarter-hour. Imbalances of this nature cause deviations from the nominal frequency and shortages in more severe cases. DSOs are responsible for delivering electricity to consumers. Electricity suppliers buy energy from generator parties and resell it to consumers.

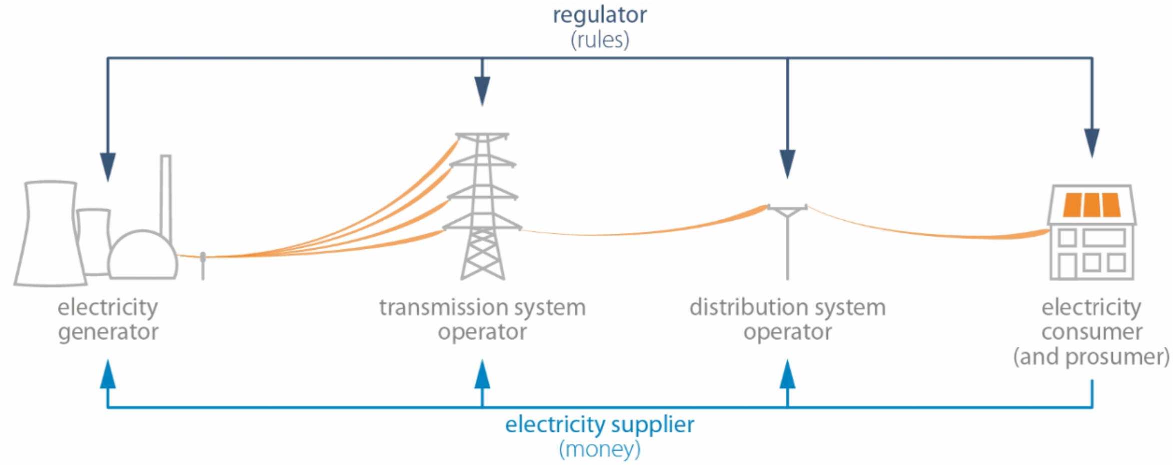

regulator
(rules)
electricity generator
transmission system operator
distribution system operator
electricity consumer (and prosumer)
electricity supplier
(money)

Figure 1: The different stages of electricity supply and the responsible parties in a liberalized market. Adapted from [Lar+10].

2.2 Wind Power Generation

In 1920, Betz [Bet20] modeled a generic wind harvesting system as an open-disc actuator and, by using the energy conservation equation for a stream tube flowing through this disk, he derived an upper limit for the power harvested by a horizontal-axis wind turbine. The *Betz Limit*, as it is known, is a function of rotor diameter D (via the rotor swept area A) and the average free stream wind velocity v at hub height H (2.1).

$$P_{ideal} = \frac{1}{2}\rho \cdot A(D) \cdot v^3 \tag{2.1}$$

Due to losses such as those associated to (1) momentum deficit in lower atmosphere boundary layer, (2) wakes from neighboring turbines, (3) suboptimal yaw angle and (4) blade tip vortices, the power harvested by the turbine rotor is only a fraction C_p (coefficient of power) of this idealized maximum. Further losses (a) of mechanical nature in the interfaces rotor-gearbox and gearbox-generator (2), (b) of electrical nature in the interface generator-converter are modeled by the fractions η_m and η_e, respectively, to yield the actual power generation as measured at the power converter, 2.2 [Alb09].

$$P = C_p\eta_m\eta_e \cdot \frac{1}{2}\rho \cdot A(D) \cdot v^3 \tag{2.2}$$

In this equation, D and H are design variables. The air density ρ may vary during operation due to changes in air temperature, but its effects are often negligible. Finally, C_p, v depend both on design (e.g., hub height H, blade profiles) and operation conditions (e.g., velocity speed and direction).

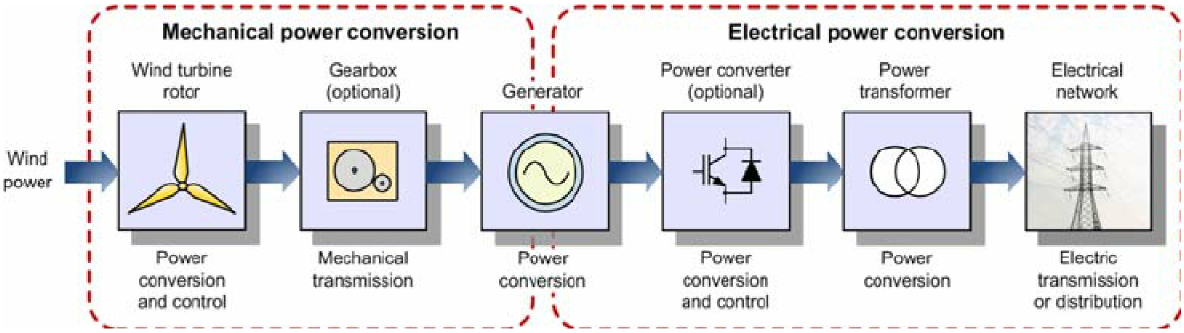

Mechanical power conversion
Electrical power conversion
Wind power
Wind turbine rotor
Gearbox (optional)
Generator
Power converter (optional)
Power transformer
Electrical network
Power conversion and control
Mechanical transmission
Power conversion
Power conversion and control
Power conversion
Electric transmission or distribution

Figure 2: The different stages of the overall wind power conversion process. Adapted from [MM11].

In operation, the dominant source of variability for the generated power is v. Being climate and weather-dependent, it is also the main reason for the intermittency and non-dispatchability of wind power [DeM+07]. This dependence motivates the usage by designers and generation operators of the so-called *wind-to-power curves* (or *power curves*), which are semi-empirical relations that allow one to determine the generated power P by knowing the wind velocity v.

As design, planning, operation, maintenance, and trading of wind power are subject to such high variabilities, forecasting wind power generation (WPG) provides value for the different players in the electricity grid, illustrated in 1. Table 2.2 gives some examples of how various system operation aspects can profit from forecasts at different time scales.

Table 1: Forecasting horizons in wind power generation and main applications.

Forecasting Horizon	Definition	Applications
Very Short	$\sim secs\ - 0.5h$	turbine control, load tracking
Short	$0.5h - 72h$	pre-load sharing
Medium	$72h$ –1 *week*	power system management, energy trading
Long	1 *week* –1 *year*	turbines maintenance scheduling

Power generation from single turbines can also be aggregated at different levels. Market operators, for example, profit the most from regional aggregations, since for energy trading, this resolution is sufficiently high, with higher resolutions across the same space scales of interests often too costly [JB14].

In countries such as Germany, where continental and national renewables-promoting public funding initiatives such as the *Energiewende* resulted in high penetration of wind power in the grid, forecasting wind power generation accurately has a tangible impact both environmentally and economically.

The intermittency of renewables motivated an alternative measure of power generation: the *capacity factor* (CF). CF is defined as the ratio of the actual generated power and the installed capacity. When considering WPG data across long timespans for both analysis and forecasting, it is usual that new commissionings take place, which manifests itself as a step perturbation into the overall generated power. In this case, CF can be useful as it is mostly insensitive to single new commissionings.

Climate and weather-conditioned local wind velocities imply for the power generation not only significant temporal dependencies but also significant spatial dependencies. As air masses influence one another in different scales, wind power generation in neighboring turbines tends to present higher correlations than turbines distant from one another [Eng+17]. Therefore, wind power generation is a phenomenon with dominant spatio-temporal dependencies.

Usual approaches to forecasting wind power generation are physical, statistical, and machine learning-based [JB14]. The physical approach relies on the modeling of the power curve using Computational Fluid Dynamic (CFD) models, taking Numerical Weather Prediction (NWP) as inputs for defining the boundary conditions. The main limitations of this approach are (a) the high costs involved in the development of such models, along with (b) the large uncertainties entailed by the NWP data. The statistical approach uses historical data and statistical time series models to produce forecasts for wind speed, which is then used in the power curve for forecasting the power generation itself. Finally, in machine learning approaches, one uses historical data for wind speed or power generation, eventually combined with historical data of weather conditions to forecast either (a) local wind speeds, with their subsequent transformation into generated power via power-curve or (b) generated power directly.

2.3 Time Series Forecasting

In [BD96], Brockwell & Davis define time series as “a set of observations y_t, each one being recorded at a specific time t.” When observations are recorded at discrete times, they are called a discrete-time time series, on which we focus this work.

An important task in time series analysis is time series forecasting, which concerns “the prediction of data at future times using observations collected in the past” [HA18]. Time series forecasting permeates most aspects of modern business, such as business planning from production to distribution, finance and marketing, inventory control, and customer management [Ore+19]. In some business use cases, a single point in forecasting accuracy may represent millions of dollars [Kah03; Jai12].

Time series forecasting tasks can be categorized in terms of (a) inputs, (b) modeling, and (c) outputs. In terms of inputs, one can use exogenous features or not, one or more input time series (univariate *versus* multivariate). In terms of modeling, one must define a resolution (e.g., hourly, weekly), can aggregate data in different levels (hierarchical *versus* non-hierarchical), and can use different schemes for generating models (we distinguish statistical from machine learning-based). Finally, regarding outputs, a forecasting task might involve making predictions in terms of single values or whole distributions (deterministic *versus* probabilistic), point-predictions or prediction intervals, predict values for either a single point or for multiple points in future time (one-step-ahead *versus* multi-step-ahead). In this work, we focus on deterministic, one-step ahead point forecasting, where one is interested in obtaining a function $f : \mathbb{R}^T \rightarrow \mathbb{R}$ (a *forecasting model*) that maps historical observations $\boldsymbol{y}_{1:T} = \{y_1, \ldots, y_T\}$ of a variable y_t to its value in a future time step $T + h$, for a forecasting horizon of interest h.

The main requirement for a forecasting model concerns the accuracy of its forecasts $\hat{y}_{t|T}$. This accuracy is quantified by a *metric*, which summarizes the distribution of the forecast error $e_t = y_t - \hat{y}_{t|T}$ over the different evaluation timesteps t. In the following subsections, we introduce some typical options for (a) schemes for defining the evaluation timesteps t (2.3.1), (b) accuracy metrics (2.3.2), as well as (c) approaches for generating forecasting models (2.3.3.1, 2.3.3.2, 2.3.3.3).

2.3.1 Model Evaluation

Assessing the performance of a model f requires defining the time indexes t for evaluating the forecast errors e_t . In a naive approach, one could use all available data for both model inference and evaluation. This would, however, result in a highly biased estimate of the model generalization performance. Less biased estimations could be attained instead by partitioning the available dataset into a *training dataset* $\boldsymbol{y}_{1:T} = \{y_1, \ldots, y_T\}$, exclusive for model inference, and a *test dataset* $\boldsymbol{y}_{T+1:T'} = \{y_{T+1}, \ldots, y_{T'}\}$, used for model evaluation (3). Once an estimate for the model performance is attained, a separate model inference using both partitions can be carried out, so that the epistemic part of the generalization error, resulting from limited data in model inference, is kept at a minimum.

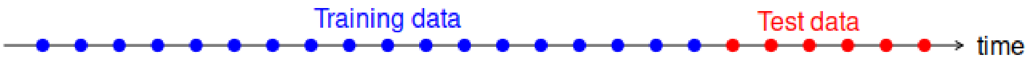


Figure 3: Partitioning the available data in training and test datasets (adapted from [Kri19]).

Furthermore, it is necessary that this partitioning results in two sets of successive observations, in order to preserve the *Markovian dependence* underlying the sequential observations. Even under this constraint, however, the choice on what point to split the data is still arbitrary, implying that assessing model performance on a single arbitrary choice would result in a biased estimate. To minimize this bias, the model performance can be assessed for several different splitting points. The partial results are then aggregated, typically by averaging, into an overall result of model performance. This procedure is known as *out-of-sample cross-validation*.

As the forecast error generally increases for longer forecasting horizons, the out-of-sample estimate might overestimate the generalization error, especially if only one-step forecasts are of interest. For overcoming this, only the first point in the test data is used in evaluating the error. This approach is known as *expanding window cross-validation*, and is illustrated in 4.

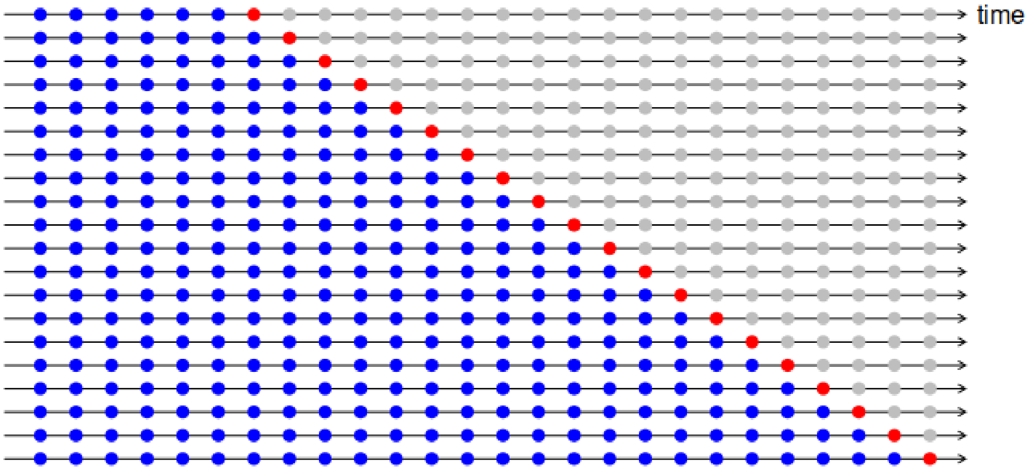


Figure 4: The expanding window cross-validation scheme (adapted from [Kri19]).

2.3.2 Accuracy Metrics

Many different metrics exist, each one summarizing the error distribution in a different way. Some of the most usual definitions are presented from 2.3 to 2.10 (see e.g., [Wu+19; Gen+17; HK06]. In particular, $MASE$ and $MdRAE$ use as denominator the forecast errors of the naïve model, which takes the last known value to forecast the next point. The naïve model can be shown to be optimal for a random walk process [HK06].

$$RMSE = \sqrt{\mathbb{E}(e_t^2)} = \sqrt{\frac{1}{(T' - T - 1)} \sum_{t=T+1}^{T'} e_t^2} \tag{2.3}$$

$$MAE = \mathbb{E}(|e_t|) = \frac{1}{(T' - T - 1)} \sum_{t=T+1}^{T'} |e_t| \tag{2.4}$$

$$MAPE = \mathbb{E}(|e_t/y_t| \cdot 100\%) = \frac{100\%}{(T' - T - 1)} \sum_{t=T+1}^{T'} \left| \frac{e_t}{y_t} \right| \tag{2.5}$$

$$sMAPE = \frac{100\%}{T' - T - 1} \sum_{T=T+1}^{T'} \frac{|e_t|}{(|y_t| + |\hat{y}_t|)/2} \tag{2.6}$$

$$MdAPE = q_{0.5}(|e_t/y_t| \cdot 100\%) \tag{2.7}$$

$$sMdAPE = q_{0.5}\left(200\% \cdot \frac{|e_t|}{y_t + \hat{y}_t}\right) \tag{2.8}$$

$$MASE = \mathbb{E}\left(\left|\frac{e_t}{e_{t,na\ddot{i}ve}}\right|\right) \tag{2.9}$$

$$MdRAE = q_{0.5}\left(\left|\frac{e_t}{e_{t,na\ddot{i}ve}}\right|\right) \tag{2.10}$$

By summarizing the forecast error distribution into a reduced set of values, forecasting metrics are essential in model development as well as in method development. To forecasters (model developers) and forecast users, metrics offer a concise, unambiguous way to communicate accuracy requirements and specifications. For methods developers, it allows comparing different methods across different use cases, forecasting settings, and datasets.

On the one hand, single metrics concisely convey information about the error distribution, which is useful for comparing models and making decisions. On the other hand, a single metric cannot convey all aspects of the error distribution, and often using more than one metric becomes necessary to ensure sufficiency [Sco02]. Therefore, deciding on a group of metrics often involves a trade-off between conciseness and sufficiency.

Metrics differ in interpretability, scale invariance, sensitivity to outliers, symmetric penalization of negative and positive errors, and behavior predictability as $y_t \to 0$ [HK06]. Therefore, it is important that the choice on the metrics set is coherent with the application requirements [Sco02]. For example, while failing to forecast single sudden peaks in local wind speed (wind gusts) might not be important in wind farm planning, it might be a primary requirement for wind turbine operation. Table **??** summarizes sensitivities for the presented metrics.

Table 2: Forecasting accuracy metrics and their sensitivities to scale and outliers.

Alias	Name	Scale Sensitivity	Outliers Sensitivity
RMSE	Root Mean Squared Error	●	●
MAE	Mean Absolute Error	●	●
MASE	Mean Absolute Scaled Error	○	●
MAPE	Mean Absolute Percentual Error	○	●
MdAPE	Median Absolute Percentual Error	○	○
sMAPE	Symmetric Mean Absolute Percentual Error	○	○
sMdAPE	Symmetric Median Absolute Percentual Error	○	○
MdRAE	Median Relative Absolute Error	○	○

Although often the most important one, accuracy is often just one of many requirements in a forecasting model development. In [Sco02], Armstrong reports that value inference time, cost savings resulting from improved decisions, interpretability, usability, ease of implementation, and development costs (human and computational resources) tend to be of comparable importance to researchers, practitioners, and decision-makers.

2.3.3 Forecasting Approaches

In general, forecasting approaches, statistical or machine learning-based alike, attain models by minimizing the forecast errors on the training set. This optimization process, often iterative, uses an optimization algorithm to update the model parameters configuration towards one that either (a) maximizes their likelihood or (b) minimizes a loss function on the training set.

The likelihood is, in essence, the relative number of ways that a configuration of model parameters can reproduce the provided data [Mce19]. In contrast, loss functions summarize the distribution of forecast errors, much like accuracy metrics. Loss functions are subject to an additional requirement, however, which is their suitability as objective function in the convex optimization underlying most of model inferencing schemes. Therefore, although it is important that the objective function guiding the model inference is coherent with the metrics used for evaluating the models, they do not have to coincide. The Mean Squared Error (MSE, 2.11) is a typical choice for a loss function for continuous-type responses, as it accounts for both bias and variance errors, besides exhibiting smoothness amenable to convex optimization [GBC16].

$$MSE = \frac{1}{T}\sum_{t=1}^{T} e_t^2 \tag{2.11}$$

5 provides an overview of the approaches reviewed in this work. We start by presenting simple forecasting approaches[1], which are often used as baselines for other approaches [HA18].

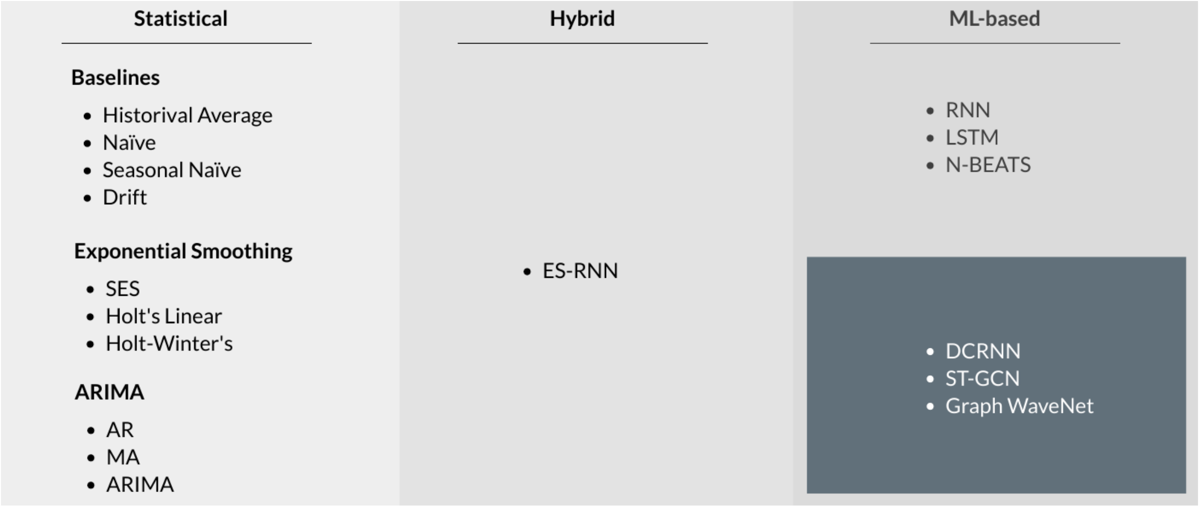


Figure 5: Forecasting approaches presented in this work. Most of these methods only model dependencies of temporal nature and are presented in this section. Exception are DCRNN, ST-GCN, and Graph WaveNet (ML-based), presented in section 2.4. They explicitly approach a more general forecasting setting where capturing both temporal and spatial dependencies is a central concern.

2.3.3.1 Baseline Approaches

Naïve method. Forecast for any point assumes a constant value: the value from the last observation (2.12). As the naïve forecast is the optimal prediction for a random walk process, it is also known as the *random walk* method.

$$\hat{y}_{T+h|T} = y_T \tag{2.12}$$

[1] Analogous to Murphy in [Mur12], we draw distinctions between the concepts of method, model, and model inference algorithm. A method can specify (1) how training data is used to generate a model (training, model inference, i.e., inference of its parameters) and (2) how a generated model uses its parameters and its input to make a prediction (inference). We denote by a model any unique configuration of parameters in a space defined by a method. Equivalently, a model represents a response surface (deterministic model) or the distribution of the response conditional on its inputs (probabilistic model).

Seasonal Naïve method. Time series are modeled as harmonic with period k observations (i.e., perfectly seasonal with seasonal period k), and for a given point in future, suggest the corresponding last observed value from the last season (2.13). For example, all monthly forecasts for any future June assume the value from the last observed June value.

$$\hat{y}_{T+h|T} = y_{T+h-k} \tag{2.13}$$

Drift method. The forecast for any point assumes a constant value rate of change, with values themselves starting from the latest observed value:

$$\hat{y}_{T+h|T} = y_T + h\left(\frac{y_T - y_1}{T-1}\right). \tag{2.14}$$

Historical Average (HA) method. The forecast for any point assumes a constant value: the average of the historical data (2.12).

$$\hat{y}_{T+h|T} = \frac{1}{T}\sum_{t=1}^{T} y_t \tag{2.15}$$

2.3.3.2 Statistical Approaches

Statistical forecasting approaches are characterized by the modeling of the time series as a realization of a stationary stochastic process [RBD90], [BBL13]. The two most widely used families of statistical methods are the Exponential Smoothing (ES) family and the ARIMA family [HA18].

In the ES approach, the time series is modeled as combination of interpretable components [RBD90]. In the *classical decomposition* [MH97], these components are trend component m, seasonal component d, and random noise (*white noise*) ε_t, which are linearly combined to reconstruct the time series:

$$y_t = m_t + s_t + a_t. \tag{2.16}$$

We now describe some of the most known methods from the ES family. **SES (Simple Exponential Smoothing method)** predicts for the next period the forecast value for the previous period, adjusting it using the forecast error (2.17). Parameter: $\alpha \in \mathbb{R}_{[0,1]}$.

$$\hat{y}_{t+1} = \hat{y}_t + \alpha(\hat{y}_t - \hat{y}_{t-1}) \tag{2.17}$$

Holt's Linear method features an additive trend component [Hyn+08]. Parameters:

$(\alpha, \beta^*) \in \mathbb{R}^2_{[0,1]}$

$$\begin{aligned} \hat{y}_{t+h|t} &= \ell_t + b_t h, \\ \textit{where} \;\; \ell_t &= \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \qquad (\textit{level}) \\ b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*) b_{t-1} \quad (\textit{growth}) \end{aligned} \tag{2.18}$$

Holt-Winters' method features additive trend and multiplicative seasonality components, for a seasonality length m, and forecasting horizon h. Parameters: $(\alpha, \beta^*, \gamma) \in \mathbb{R}^3_{[0,1]}$ (usual bounds, refer to [Hyn+08] for details).

$$\begin{aligned} \hat{y}_{t+h|t} &= (\ell_t + b_t h) s_{t-m+h_m^+}, \\ \textit{where} \; \ell_t &= \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1}) \qquad (\textit{level}) \\ b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*) b_{t-1} \qquad (\textit{growth}) \\ s_t &= \gamma y_t / (\ell_{t-1} + b_{t-1}) + (1-\gamma) s_{t-m} \qquad (\textit{seasonal}) \end{aligned} \tag{2.19}$$

ARIMA (Autoregressive Integrated Moving Average) methods [BP70] rely on repeatedly applying a difference operator to the observed values until the differenced series resemble a realization of some stationary stochastic process [RBD90]. We denote by $\nabla^k(\cdot)$ the difference operator of order k. For $k = 1$, $\nabla y_t = y_t - y_{t-1}$; for $k = 2$, we have $\nabla^2(y_t) = \nabla(\nabla y_t) = \nabla y_t - \nabla y_{t-1} = y_t - 2y_{t-1} + y_{t-2}$ and so forth. Another operator useful in ARIMA methods is the *backshift operator* $B^k(\cdot)$ with lag k. For $k = 1$, we have $By_t = y_{t-1}$. For $k = 2$, $B^2(y_t) = B(B(y_t)) = y_{t-2}$.

AR (Autoregressive) method. Linear regression with past values of the same variable (lagged values) as predictors. A constant level c and a white noise $\varepsilon_t \sim WN(\mu_\varepsilon, \sigma_\varepsilon^2)$ are considered. Parameters: $\boldsymbol{\phi} = [\phi_1 \; \phi_2 \; \cdots \; \phi_p]^\top$, μ_ε, σ_ε, c. Hyperparameter: p.

$$\hat{y}_t = c + \varepsilon_t + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} \tag{2.20}$$

MA (Moving Average) method. Linear regression with lagged forecast errors $\varepsilon_\tau = \hat{y}_\tau - y_\tau$ as predictors. Parameters:$\boldsymbol{\theta} = [\theta_1 \; \theta_2 \; \cdots \; \theta_q]^\top$, μ_ε, σ_ε, c. Hyperparameter: q.

$$\hat{y}_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q} \tag{2.21}$$

(Non-seasonal) ARIMA method. Linear regression, with lagged *differenced* values y'_τ and lagged errors as predictors. It combines autoregression on the differenced time series with a moving average model, hence the name *Autoregressive Integrated Moving

Average*, with *integration* referring to the reverse operation of differencing, used when reconstructing the original time series from its differenced version. Parameters: $\boldsymbol{\phi} = [\phi_1\ \phi_2\ \cdots\ \phi_p]^\top, \boldsymbol{\theta} = [\theta_1\ \theta_2\ \cdots\ \theta_q]^\top,\ \mu_\varepsilon,\ \sigma_\varepsilon, c$. Hyperparameters: p, d, q.

$$\hat{y}'_t = c + \varepsilon_t + \phi_1 y'_{t-1} + ... + \phi_p y'_{t-p} + \cdots + \theta_1 \varepsilon_{t-1} + ... + \theta_q \varepsilon_{t-q} \tag{2.22}$$

2.3.3.3 Machine Learning Approaches

)

Approaches solely based on Machine Learning struggled until recently to consistently outperform statistical time series forecasting approaches [MSA18]. Despite relying on biased evidence (e.g., models were evaluated across all time series without any sound choice nor search for hyperparameters), Makridakis claimed in [MSA18] that "hybrid approaches and combinations of methods are the way forward for improving the forecasting accuracy and making forecasting more valuable." Oreshkin et al. challenged in [Ore+19] this conclusion, introducing N-BEATS, a pure deep learning method that was shown to outperform statistical and hybrid methods, while also ensuring interpretability of intermediate outputs.

Below we present selected deep learning methods helpful for understanding current state-of-the-art approaches for both wind power generation-specific applications and in general univariate time series forecasting applications.

RNN (Recurrent Neural Network). Use the recurrent layer as building block: a cell that updates its state according to (a) its previous state h_{t-1} and (b) its current input x_t (6). By performing this update at every timestep of a time series, this basic structure allows the RNN to express temporal dependencies in time series. An RNN can be built by serializing several of these self-looping cells between the input layer and the output layer for achieving higher-order mappings and thus capturing more complex temporal dependencies. The major limitation of RNN in its basic design (recurrent layer as in 6) is its inability to capture dependencies that exist across longer periods than a few timesteps. It arises from a phenomenon called *vanishing gradients*: while inferring optimal parameters via gradient descent (learning phase), the gradients calculated via backpropagation through time become too small to guide the optimization.

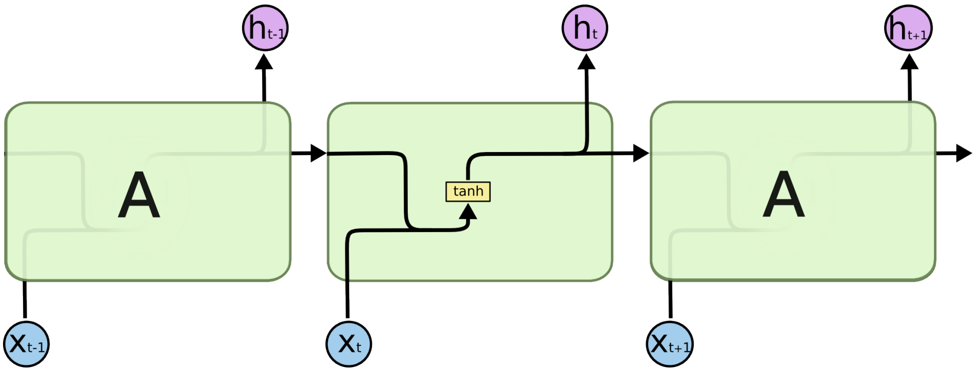


Figure 6: The basic RNN architecture in its unfolded representation. Arrows indicate transfers of input and hidden states (adapted from [Pan]. Every block concatenates the last hidden state with the current input, passing the result to an activation function (tanh in this illustration). The result is carried forward as the updated hidden state.

LSTM (Long-Short Term Memory). A type of RNN, it improves on its basic design most importantly by including an long memory state which is allowed to be transferred across several update steps with only minimal changes (superior horizontal line inside the repeating module in 7. This allows information to persist across many cell updates, thus making it possible to capture long-term dependencies. The extent to which this long memory state is preserved is controlled by forget gate, illustrated in 7) by the leftmost vertical path inside the cell. The other paths represent other gated state transfers, which determine how (a) the previous cell state, (b) the previous long memory state and (c) the current cell inputs are combined and passed to the next cell iteration and as input to deeper layers.

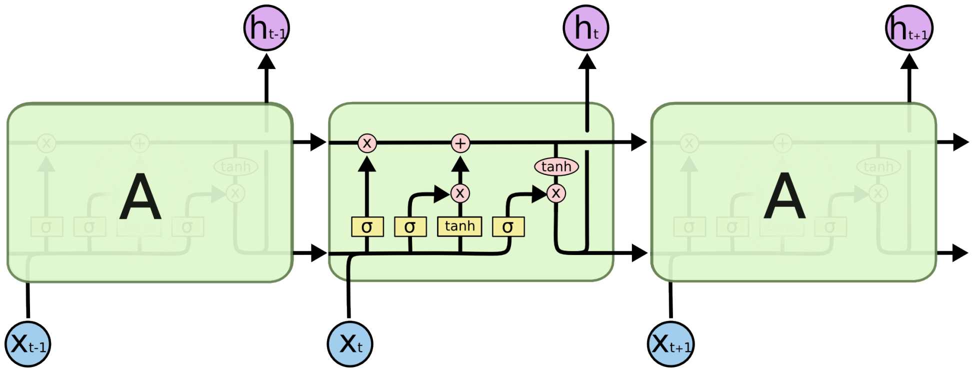


Figure 7: Basic LSTM architecture in its unfolded representation (adapted from [Pan]).

NBEATS. Uses as building block (a) a multi-layer fully connected network with ReLU nonlinearities, which feed (b) basis layers that generate a backcast and a forecast output. Blocks are arranged into stacks, organized to form a model (8. Models resulting from this architecture consistently outperformed state-of-the-art methods for univariate forecasting across different horizons and thousands of time series datasets of different nature, while using a single hyperparameter configuration [Ore+19].

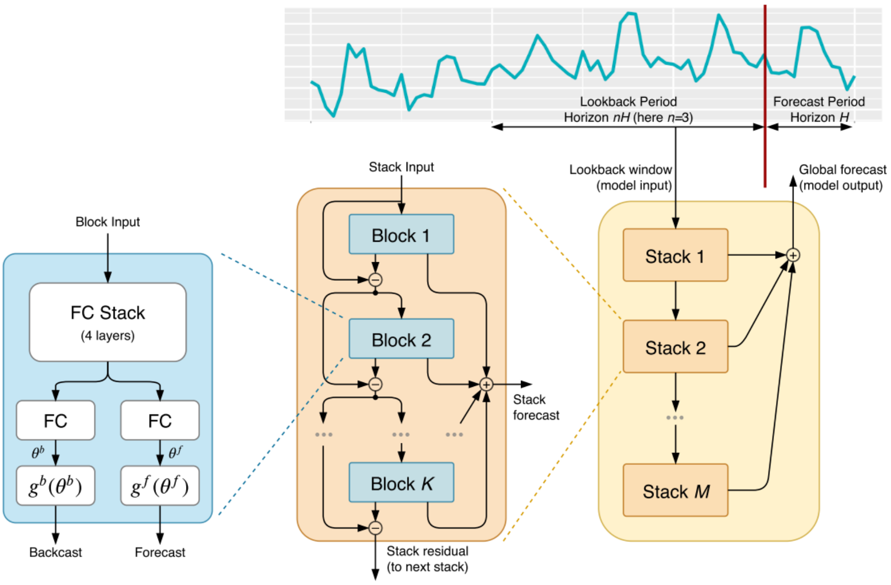


Figure 8: NBEATS architecture (adapted from [Ore+19]).

2.3.3.4 Hybrid Approaches

Hybrid methods combine machine learning and statistical approaches by using the outputs from statistical engines as features [Ore+19]. Below we present ES-RNN, a hybrid method winner of the 2017 M4 forecasting competition.

ES-RNN. It uses Holt-Winters' ES method as statistical engine for capturing the seasonal and level components from the time series into features, which are then used by an LSTM model to exploit non-linear dependencies.

$$\begin{aligned} \hat{y}_{t+h|t} &= LSTM(y_t, \ell_t, s_t) \\ where \;\; \ell_t &= \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)\ell_{t-1} \qquad (level) \\ s_t &= \gamma y_t / \ell_t + (1-\gamma) s_{t-m} \qquad (seasonal) \end{aligned} \tag{2.23}$$

2.3.4 Model Selection

As models have parameters, so do methods have their own, often referred to as *hyperparameters*. They may control the space of model parameters configurations, the model inference process or eventually the loss function [HKV19]). Hyperparameters may have a major influence on the performance of resulting models. Accounting for this effect requires yet another partition in order to attain a minimally unbiased estimate of the resulting generalization error. When working with three partitions, one for model inference, another for assessing its generalization error given a hyperparameters configuration and another one for assessing it across different hyperparameters configurations, authors often refer to them as training, validation and test set, respectively.

2.4 Spatio-Temporal Forecasting

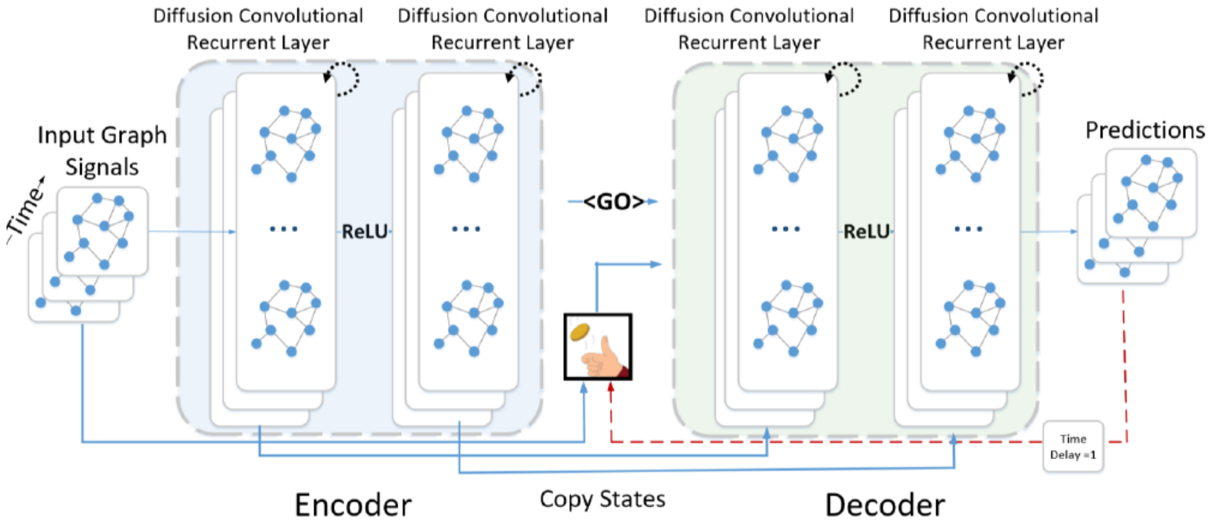
In the spatio-temporal (ST) version of this problem, one aims to attain a function $f : \mathbb{R}^{|V| \times T} \rightarrow \mathbb{R}^{|V|}$ that maps historical observations of a quantity across different regions $v \in V$, $\boldsymbol{y}_t = [y_{1,t} \;\; y_{2,t} \;\; \cdots \;\; y_{|V|,t}]^\top$, to its value $\boldsymbol{y}_{t+1}$ in the next timestep (2.24).

$$[\boldsymbol{y}_{t-T+1}, \;\; \cdots \;\; , \boldsymbol{y}_t] \xrightarrow{f(\cdot)} \boldsymbol{y}_{t+1} \tag{2.24}$$

For some forecasting problems such as for the weather-conditioned wind power generation, the spatial dependency might play an important along with the temporal dependencies themselves [Eng+17]). In this work, we consider three different approaches to the ST forecasting problem. In a naïve approach, time series for different locations are modeled independently, thus neglecting spatial dependencies. In a second approach, the time series are modeled jointly via a multivariate forecasting approach, where for generating a single model one relies on historical observations not from a single but from several input variables, which can be expressed by a sequence of input vectors $\boldsymbol{X}_{1:T} = \{\boldsymbol{X}_1, ..., \boldsymbol{X}_T\}$. Finally, we consider the explicit modeling of both spatial and temporal dependencies via dynamic graphs. The latter approach is represented by the methods presented below.

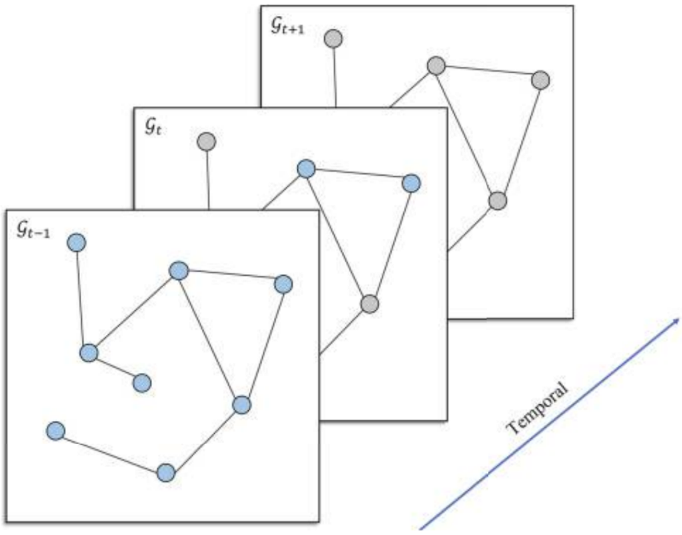
DCRNN (Diffusion Convolutional RNN). RNN is leveraged by replacing the matrix multiplication by a diffusion convolution [Bra+]). Motivated by the traffic forecasting problem, where spatial dependencies are directional (non-Euclidean), Li et al. [Li+17]) recast the spatio-temporal evolution of a variable as a diffusion process on a directed graph, where every node corresponds to a sensor. Learning is performed via (1) diffusion convolution, further integrated with a (2) seq-to-seq learning framework, and a (3) scheduled sampling for modeling long-term dependencies (9).

Figure 9: The DCRNN architecture (adapted from [Li+17]).


Diffusion Convolutional Recurrent Layer
Diffusion Convolutional Recurrent Layer
Diffusion Convolutional Recurrent Layer
Diffusion Convolutional Recurrent Layer
Input Graph Signals
Time
ReLU
<GO>
ReLU
Predictions
Time Delay =1
Encoder
Copy States
Decoder

ST-GCN. A spatial-temporal graph is generated by stacking graph frames from all timesteps, each frame representing the graph state at a specific time (10). The spatial-temporal graph is partitioned, and to each of its nodes is assigned a weight vector. Finally, a graph convolution is performed on the weighted spatial-temporal graph.

Figure 10: ST-GCN underlying principle (adapted from [Gen+17]).



Graph WaveNet. Uses as building blocks a Temporal Convolution Network (TCN) and a Graph Convolution Network (GCN) for capturing spatio-temporal dependencies in every module. A core idea is the usage of a learnable self-adaptative adjacency matrix, which allows node dependencies to change over time and not necessarily be determined by their distances [Wu+19; Bra+].

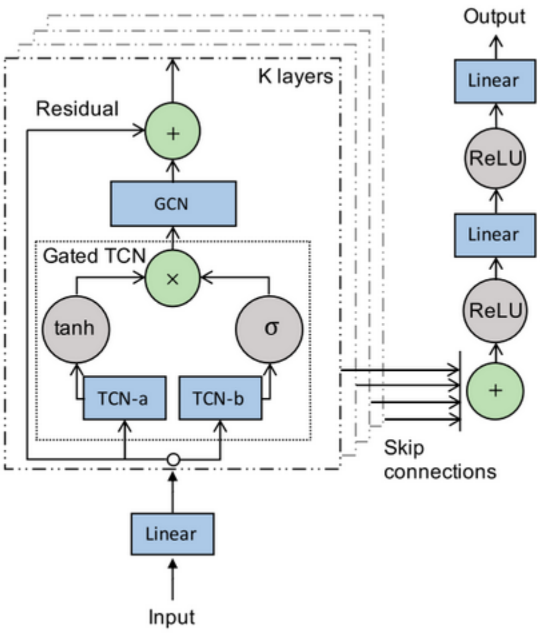


Figure 11: The Graph Wavenet architecture (adapted from [Wu+19]).

2.4.1 Accuracy Metrics

The usual accuracy metrics in spatio-temporal forecasting are similar to their counterparts in the temporal setting, the main difference concerning the aggregation over the $v \in |V|$ regions. We present the most popular ones in 2.4, 2.5, 2.3.

$$MAE = \frac{1}{|V|(T' - T - 1)} \Sigma_{v=1}^{|V|} \Sigma_{t=T+1}^{T'} |\hat{\boldsymbol{y}}_t^{(v)} - \boldsymbol{y}_t^{(v)}| \tag{2.25}$$

$$MAPE = \frac{100\%}{|V|(T' - T - 1)} \Sigma_{v=1}^{|V|} \Sigma_{t=T+1}^{T'} \frac{|\hat{\boldsymbol{y}}_t^{(v)} - \boldsymbol{y}_t^{(v)}|}{|\boldsymbol{y}_t^{(v)}|} \tag{2.26}$$

$$RMSE = \sqrt{\frac{1}{|V|(T' - T - 1)} \Sigma_{v=1}^{|V|} \Sigma_{t=T+1}^{T'} (\hat{\boldsymbol{y}}_t^{(v)} - \boldsymbol{y}_t^{(v)})^2} \tag{2.27}$$

3 USE CASE

In this chapter, we present the use case in terms of its requirements (3.1) and the available data (3.2).

3.1 Requirements

We focus on fulfilling typical balance responsible parties, who are financially responsible for ensuring the balance between power supply and demand cross-regionally on weekly basis [Lar+10]. This sets the resolution and scales of interest: forecasts of interest are taken as regional, week-ahead point forecasting predictions of wind power generation.

3.2 Datasets

Two datasets are available, and are provided by [BT17]. Wind turbine operators publicly report hourly power generation in spatial resolution of single wind farms. This data is taken from measurements performed in power control units in individual turbines, then aggregated in hourly basis. Missing data was handled via machine learning-based, validated imputation [BT17].

The first dataset comprises design and installation specifications from individual wind turbines: rated power, hub height, diameter, district NUTS3 ID, latitude, longitude and commissioning date.

The second dataset comprises hourly wind power generation aggregated by districts across all Germany, from January 2000 to December 2015, totalling 140228 observations for each of 299 wind power producing districts.

BIBLIOGRAPHY

[AKK17] Gowtham Atluri, Anuj Karpatne, and Vipin Kumar. "Spatio-Temporal Data Mining: A Survey of Problems and Methods". In: *ACM Computing Surveys* 51.4 (Nov. 2017). ISSN: 15577341. DOI: 10.1145/3161602. URL: http://arxiv.org/abs/1711.04710.

[Alb09] M H Albadi. "Wind Turbines Capacity Factor Modeling—A Novel Approach". In: 24.3 (2009), pp. 1637–1638.

[BBL13] Gianluca Bontempi, Souhaib Ben Taieb, and Yann Aël Le Borgne. "Machine learning strategies for time series forecasting". In: *Lecture Notes in Business Information Processing* 138 LNBIP (2013), pp. 62–77. ISSN: 18651348. DOI: 10.1007/978-3-642-36318-4{_}3.

[BD96] Brockwell and Davis. *Introduction to Time Series and Forecasting.* Vol. 68. 22. 1996, pp. 3180–3182. ISBN: 9783319298528. DOI: 10.1063/1.115817.

[Bet20] Albert Betz. "Das Maximum der theoretisch möglichen Ausnutzung des Windes durch Windmotoren". In: *Zeitschrift fur das gesamte Turbinenwesen 20 (1920).* (1920).

[BP70] George E P Box and David A Pierce. "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models". In: *Journal of the American statistical Association* 65.332 (1970), pp. 1509–1526.

[Bra+] Ronald J Brachman et al. "Introduction to GNNs". In: ().

[BT17] Raik Becker and Daniela Thrän. "Completion of wind turbine data sets for wind integration studies applying random forests and k-nearest neighbors". In: *Applied Energy* 208.September (2017), pp. 252–262. ISSN: 03062619. DOI: 10.1016/j.apenergy.2017.10.044. URL: https://doi.org/10.1016/j.apenergy.2017.10.044.

[Del+15] Erik Delarue et al. "Renewables Intermittency: Operational Limits and Implications for Long-Term Energy System Models". In: *MIT Joint Program on the Science and Policy of Global Change* 277 (2015).

[DeM+07] Edgar A. DeMeo et al. "Accomodating wind's natural behavior". In: *IEEE Power and Energy Magazine* 5.6 (2007), pp. 59–67. ISSN: 15407977. DOI: 10.1109/MPE.2007.906562.

[Eng+17] Kolbjørn Engeland et al. "Space-time variability of climate variables and intermittent renewable electricity production – A review". In: *Renewable and Sustainable Energy Reviews* 79.May 2016 (2017), pp. 600–617. ISSN: 18790690. DOI: 10.1016/j.rser.2017.05.046. URL: http://dx.doi.org/10.1016/j.rser.2017.05.046.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[Gen+17] Xu Geng et al. *Spatiotemporal Multi-Graph Convolution Network for Ride-Hailing Demand Forecasting*. Tech. rep. 2017, p. 19. URL: www.aaai.org.

[HA18] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[HK06] Rob J. Hyndman and Anne B. Koehler. "Another look at measures of forecast accuracy". In: *International Journal of Forecasting* 22.4 (2006), pp. 679–688. ISSN: 01692070. DOI: 10.1016/j.ijforecast.2006.03.001.

[HKV19] Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren. *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.

[Hyn+08] Rob J Hyndman et al. *Forecasting with Exponential Smoothing*. 2008, p. 350. ISBN: 978-3-540-71918-2. DOI: 10.1007/978-3-540-71918-2.

[Jai12] Chaman L Jain. "Answers to your forecasting questions". In: *The Journal of Business Forecasting* 31.2 (2012), p. 3.

[JB14] Jaesung Jung and Robert P. Broadwater. "Current status and future advances for wind speed and power forecasting". In: *Renewable and Sustainable Energy Reviews* 31 (2014), pp. 762–777. ISSN: 13640321. DOI: 10.1016/j.rser.2013.12.054. URL: http://dx.doi.org/10.1016/j.rser.2013.12.054.

[Kah03] Kenneth B Kahn. "How to measure the impact of a forecast error on an enterprise?" In: *The Journal of Business Forecasting* 22.1 (2003), p. 21.

[Kri19] Rami. Krispin. *Hands-On Time Series Analysis with R : Perform Time Series Analysis and Forecasting Using R*. 2019, p. 438. ISBN: 9781788629157.

[Lar+10] Kristian Larsen et al. "Patient-reported outcome after fast-track hip arthroplasty: a prospective cohort study". In: *Health and Quality of Life Outcomes* 8.1 (2010), p. 144. ISSN: 1477-7525. DOI: 10.1186/1477-7525-8-144. URL: http://www.europarl.europa.eu/RegData/etudes/BRIE/2016/593519/EPRS_BRI(2016)593519_EN.pdf%20http://hqlo.biomedcentral.com/articles/10.1186/1477-7525-8-144.

[Li+17] Yaguang Li et al. "Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting". In: (July 2017). URL: http://arxiv.org/abs/1707.01926.

[Mce19] Mcelreath. *Statistical Rethinking*. Vol. 53. 9. 2019, pp. 1689–1699. ISBN: 9788578110796. DOI: 10.1017/CBO9781107415324.004.

[MH97] Spyros Makridakis and Michele Hibon. "ARMA models and the Box–Jenkins methodology". In: *Journal of Forecasting* 16.3 (1997), pp. 147–163.

[MM11] Marcelo Gustavo Molina and Pedro Enrique Mercado. "Modelling and control design of pitch-controlled variable speed wind turbines". In: *Wind turbines*. In Tech, 2011.

[Moe+18] Julia Moemken et al. "Future Changes of Wind Speed and Wind Energy Potentials in EURO-CORDEX Ensemble Simulations". In: *Journal of Geophysical Research: Atmospheres* 123.12 (2018), pp. 6373–6389. ISSN: 21698996. DOI: 10.1029/2018JD028473.

[MSA18] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. "The M4 Competition: Results, findings, conclusion and way forward". In: *International Journal of Forecasting* 34.4 (2018), pp. 802–808.

[Mur12] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[Ore+19] Boris N. Oreshkin et al. "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting". In: (2019), pp. 1–31. URL: http://arxiv.org/abs/1905.10437.

[Pan] Enish Paneru. *Understanding LSTM Networks.*

[RBD90] W. D. Ray, P. J. Brockwell, and R. A. Davis. *Time Series: Theory and Methods*. Vol. 153. 3. 1990, p. 400. ISBN: 9781441903198.

[Sco02] J Scott Armstrong. *PRINCIPLES OF FORECASTING: A Handbook for Researchers and Practitioners*. Tech. rep. 2002.

[Wu+19] Zonghan Wu et al. "Graph WaveNet for Deep Spatial-Temporal Graph Modeling". In: (May 2019). URL: `http://arxiv.org/abs/1906.00121`.