



HAL
open science

WindDragon: Enhancing wind power forecasting with Automated Deep Learning

Julie Keisler, Etienne Le Naour

► **To cite this version:**

Julie Keisler, Etienne Le Naour. WindDragon: Enhancing wind power forecasting with Automated Deep Learning. 2024. hal-04456905v1

HAL Id: hal-04456905

<https://hal.science/hal-04456905v1>

Preprint submitted on 21 Feb 2024 (v1), last revised 22 Apr 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

WINDDRAGON: ENHANCING WIND POWER FORECASTING WITH AUTOMATED DEEP LEARNING

Julie Keisler^{* 1,2}, **Etienne Le Naour**^{* 1,3}

¹ EDF R&D, Palaiseau, France

² INRIA Lille Nord Europe

³ Sorbonne Université, CNRS, ISIR, 75005 Paris, France

{*julie.keisler, etienne.le-naour*}@edf.fr

ABSTRACT

Achieving net zero carbon emissions by 2050 requires the integration of increasing amounts of wind power into power grids. This energy source poses a challenge to system operators due to its variability and uncertainty. Therefore, accurate forecasting of wind power is critical for grid operation and system balancing. This paper presents an innovative approach to short-term (1 to 6 hour horizon) wind power forecasting at a national level. The method leverages Automated Deep Learning combined with Numerical Weather Predictions wind speed maps to accurately forecast wind power.

1 INTRODUCTION

To meet the 2050 net zero scenario envisaged by the Paris Agreement ([United Nations Convention on Climate Change, 2015](#)), wind power stands out as a critical energy source for the future. Remarkable progress has been made since 2010, when global electricity generation from wind power was 342 TWh, rising to 2,100 TWh in 2022 ([International Energy Agency \(IEA\), 2023](#)). The IEA targets approximately 7,400 TWh of wind-generated electricity by 2030 to meet the zero-emissions scenario. However, to realize the full potential of this intermittent energy source, accurate forecasts of wind power generation are needed to efficiently integrate it into the power grid.

Research in wind power forecasting has developed a wide range of methods ([Giebel and Kariniotakis, 2017](#); [Tawn and Browell, 2022](#)), including statistical ([Riahy and Abedi, 2008](#)), physical ([Lange and Focken, 2006](#)), hybrid ([Shi et al., 2012](#)), and deep learning (DL) ([Wang et al., 2021](#)) approaches. These methods use a variety of data sources, including historical wind power records, geospatial satellite data, on-site camera imagery, and numerical weather prediction (NWP) forecasts. Among these, typical NWP-based methods primarily focus on using local time series of wind speed forecasts for local wind power prediction ([Piotrowski et al., 2022](#)). However, NWP forecasts produce richer outputs, notably spatial predictions of physical quantities such as wind speed and direction over large scale grids (e.g. national or regional). Predicting aggregated (e.g national or regional) wind power from such fine-grained spatial information appears promising and is largely unexplored in the literature ([Higashiyama et al., 2018](#)). Thus, we propose to explore how wind speed maps combined with suitable machine learning models can capture complex patterns, improving large scale wind power predictions.

In this work, we propose to leverage the spatial information in NWP wind speed maps for national wind power forecasting by exploiting the capabilities of DL models. The overall methodology is illustrated in Figure 1. To fully exploit DL mechanisms potential, we introduce WindDragon, an adaptation of the DRAGON¹ ([Keisler et al., 2023](#)) framework. WindDragon is an Automated Deep Learning (AutoDL) framework for short-term wind power forecasting using NWP wind speed maps. WindDragon’s performances are benchmarked against conventional computer vision models, such

* Equal contribution

¹<https://dragon-tutorial.readthedocs.io/en/latest/index.html>

as Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs), as well as standard baselines in wind power forecasting. The experimental results highlight two findings:

- The use of full NWP wind speed maps coupled with DL regressors significantly outperforms other baselines.
- WindDragon demonstrates superior performance compared to traditional computer vision DL models.

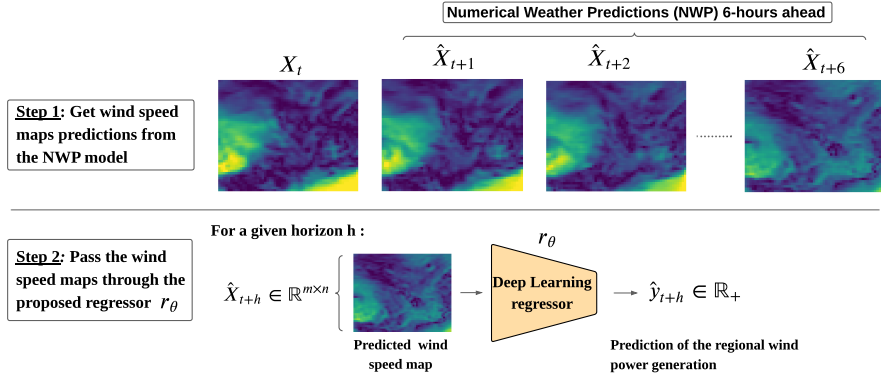


Figure 1: Global scheme for wind power forecasting. Every 6 hours, the NWP model produces hourly forecasts. Each map is processed independently by the regressor which maps the grid to the wind power corresponding to the same timestamp.

2 WINDDRAGON: A FRAMEWORK FOR REGRESSION ON WIND SPEED MAPS

Deep Learning models have the ability to capture complex spatial patterns, which makes them well suited for modeling non-linear relationships between meteorological features and wind energy production. These models are especially useful when wind farms are scattered across the map (see Figure 3) and wind speed has significant variance across locations.

CNNs and ViTs, both prominent in computer vision, might underperform in the context of wind speed map regression for global wind power forecasting. By learning local and spatial patterns, CNNs efficiently map structured inputs to numerical values. However, CNN’s shift-invariant property (Zhang, 2019) can hinder wind power forecasting because identical wind speeds at different map locations do not equate to the same power generation due to the uneven distribution of wind farms. Conversely, ViTs excel at image classification by segmenting images into patches and applying self-attention mechanisms, but the size of the considered datasets (less than 20000 points for the training dataset) might limit their effectiveness. Given these concerns, the use of AutoDL frameworks to automatically identify the most appropriate DL architecture is a promising solution.

The DRAGON framework. DRAGON (Keisler et al., 2023) is an AutoDL framework which automatically generates well-performing deep learning models for a given task. Compare to other AutoDL frameworks (Liu et al., 2019; Hutter et al., 2019; Zimmer et al., 2020; Deng et al., 2022), DRAGON provides a flexible search space, which can be used on any task. It allows the extension of the possibilities in terms of architectures and is adapted when the type of architecture to use is unclear or when high performance is sought by tuning hyperparameters. We used several tools from the generic framework to adapt it for wind power forecasting from wind speed maps.

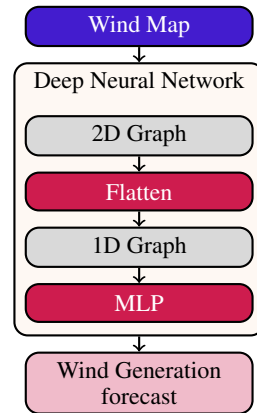


Figure 2: WindDragon’s meta model for wind power forecasting

WindDragon: adapting the DRAGON framework for wind power forecasting. The neural networks in DRAGON are represented as directed acyclic graphs, with nodes representing the layers and edges representing the connections between them. In our case, a value $\hat{y}_t \in \mathbb{R}$ is predicted from a 2D map $X_t \in \mathbb{R}^{m \times n}$. The search space is then restricted to a specific family of constrained architectures, as represented in Figure 2. A first graph processes 2D data and can be composed by convolutions, pooling, normalization, dropout, and attention layers. Then, a flatten layer and a second graph follow. This one is composed by MLPs, self-attention, convolutions and pooling layers. A final MLP layer is added at the end of the model to convert the latent vector to the desired output format. We optimized the solutions from our search space using an evolutionary algorithm, as detailed Appendix A.

3 EXPERIMENTS

Datasets. The wind speed maps used are 100-meter high forecasts at a 9 km resolution provided by the HRES² model from the European Centre for Medium-Range Weather Forecasts (ECMWF). The maps are provided at an hourly time step and there are 4 forecast runs per day (every 6 hours). Only the six more recent forecasts are used here as the forecasting horizon of interest is six hours. The hourly french regional and national wind power generation data came from the french TSO³.

Data preparation. The national forecast of wind power generation is obtained by summing the forecasts of the 12 administrative regions of Metropolitan France. According to our first experiments, this bottom-up technique produced better results than predicting national production directly. The division of a national map into regions is a challenge, as shown in Figure 3 as wind turbines are not evenly distributed across the regions. Therefore, we selected areas around each wind farm in the region and took the convex hull of all the considered points. The result is a seamless map that includes local wind turbines with no gaps to disrupt the models. Installed capacity data for each region - corresponding to the maximum wind power a region can produce - is available and updated every three months. It was collected and used to scale the wind power target. Years from 2018 to 2019 are used to train the models, and data from 2020 is used to evaluate how the models perform.

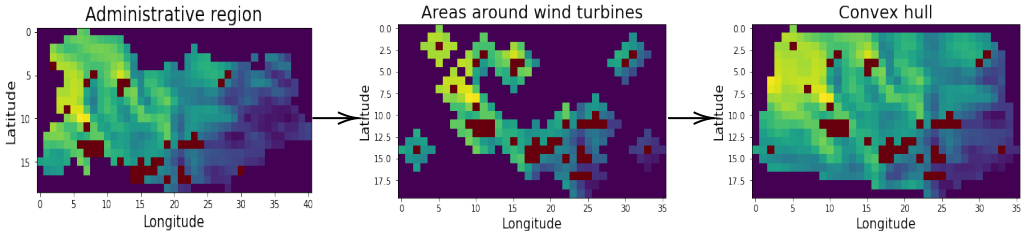


Figure 3: Data preparation for the region Auvergne-Rhône-Alpes. The wind farms are represented in red. The first image shows the distribution of wind farms across the administrative region.

We use the following baselines to compare hourly forecasts for an horizon h ($h \in \{1, \dots, 6\}$):

- **Persistence:** Predicts wind power generation at future time $t + h$ as equal to the observed generation at current time t .
- **XGB on Wind Speed Mean:** Forecasts wind power at $t + h$ using a two-step approach: (i) Compute the mean wind speed for the considered region at $t + h$ using NWP forecasts. (ii) Apply an XGBoost regressor (Chen and Guestrin, 2016) to predict power generation based on the computed mean wind speed.
- **Convolutional Neural Networks (CNNs).** Forecasts wind power at $t + h$ using the NWP predicted wind speed map. CNNs can efficiently regress a structured map on a numerical value by learning local and spatial patterns (LeCun et al., 1995).

²<https://www.ecmwf.int/en/forecasts/datasets/set-i>

³<https://www.rte-france.com/eco2mix>

- **Vision Transformers (ViTs)** Forecasts wind power at $t + h$ using the NWP predicted wind speed map. The map is segmented into patches and a self-attention mechanism is used to capture the dependencies between these patches (Dosovitskiy et al., 2020).

The implementation details of the baselines are described in Appendix B.

We compute two scores: **Mean Absolute Error (MAE)** in Megawatts (MW), showing the absolute difference between ground truth and forecast, and **Normalized Mean Absolute Error (NMAE)**, a percentage obtained by dividing the MAE by the average wind power generation for the test year.

Results. We run experiments for each of the 12 French metropolitan regions and then aggregate the predictions to derive national results. The national prediction results are presented in Table 1, while detailed regional results can be found in Table 2 (Appendix C).

Table 1: National results: sum of the regional forecasts for each models. The best results are highlighted in bold and the best second results are underlined.

	WindDragon		CNN		ViT		XGB on mean		Persistence	
	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE
France	346.7	7.7 %	<u>369.0</u>	<u>8.1 %</u>	385.7	8.5 %	416.7	9.2 %	779.7	17.3 %

The results in Table 1 highlight three key findings:

- (i) **Improved performance with aggregated NWP statistics.** Using the average of NWP-predicted wind speed maps coupled with an XGB regressor significantly outperforms the naive persistence baseline.
- (ii) **Gains from full NWP map utilization.** More complex patterns can be captured by using the full predicted wind speed map, as opposed to just the average, thereby improving forecast accuracy. In this context, both the ViT and CNN regressors applied to full maps yielded gains of 31 MW (7.4%) and 47 MW (11.5%), respectively, over the mean-based XGB.
- (iii) **WindDragon’s superior performances.** WindDragon outperforms all baselines, showing an improvement of 22 MW (6%) over the CNN. On an annual basis, this corresponds to approximately 193 GWh, which is equivalent to the annual consumption of a French town of 32,000 inhabitants⁴. Refer to Appendix A for WindDragon’s architecture example.

In Figure 4, we present the aggregated national wind power forecasts using both WindDragon and the CNN baseline during a given week. While both models deliver highly accurate forecasts, it’s important to highlight that DRAGON demonstrates superior accuracy, particularly in predicting high peak values. See Appendix C.2 for visual comparisons of all baselines performances.

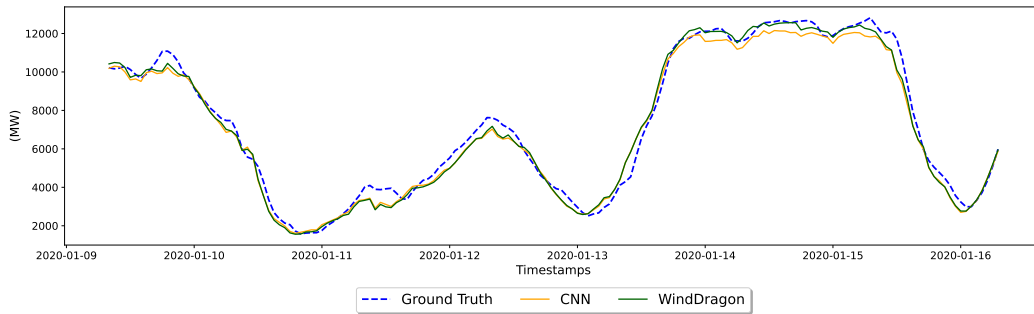


Figure 4: Wind power forecasts for a week in January 2020. The figure displays the ground truth as dotted lines, and the forecasts from the two top-performing models, WindDragon and the CNN.

⁴based on the average European per capita consumption (Statista Research Department, 2022)

4 CONCLUSION AND IMPACT STATEMENT

In this paper, we have presented two key findings that show great promise. First, using NWP wind speed forecasts as a map significantly improves forecast accuracy compared to using only aggregated values. Second, our framework, WindDragon, shows superior performance to all other baseline models. The significant improvement provided by WindDragon is particularly critical in light of the increasing reliance on wind energy, driven by the pursuit of the net-zero scenario.

Future work could adapt our methodology for photovoltaic (PV) systems, applying it to solar radiation maps generated by NWP models. While current deep learning research in PV primarily focuses on short-term nowcasting (Le Guen, 2022), our method holds promise for extending the forecasting horizon, potentially improving the efficiency and reliability of solar power predictions.

REFERENCES

- L. Beyer, X. Zhai, and A. Kolesnikov. Better plain vit baselines for imagenet-1k. *arXiv preprint arXiv:2205.01580*, 2022.
- T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.
- D. Deng, F. Karl, F. Hutter, B. Bischl, and M. Lindauer. Efficient automated deep learning for time series forecasting, 2022.
- A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- G. Giebel and G. Kariniotakis. Wind power forecasting—a review of the state of the art. *Renewable energy forecasting*, pages 59–109, 2017.
- K. Higashiyama, Y. Fujimoto, and Y. Hayashi. Feature extraction of nwp data for wind power forecasting using 3d-convolutional neural networks. *Energy Procedia*, 155:350–358, 2018.
- F. Hutter, L. Kotthoff, and J. Vanschoren. *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.
- International Energy Agency (IEA). Wind power generation, 2023. URL <https://www.iea.org/energy-system/renewables/wind>. IEA, Paris.
- J. Keisler, E.-G. Talbi, S. Claudel, and G. Cabriel. An algorithmic framework for the optimization of deep neural networks architectures and hyperparameters. *arXiv preprint arXiv:2303.12797*, 2023.
- M. Lange and U. Focken. *Physical approach to short-term wind power prediction*, volume 208. Springer, 2006.
- V. Le Guen. Deep learning for spatio-temporal forecasting—application to solar energy. *arXiv e-prints*, pages arXiv–2205, 2022.
- Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.
- H. Liu, K. Simonyan, O. Vinyals, C. Fernando, and K. Kavukcuoglu. Hierarchical representations for efficient architecture search, 2018.
- H. Liu, K. Simonyan, and Y. Yang. Darts: Differentiable architecture search, 2019.
- P. Piotrowski, D. Baczyński, M. Kopyt, and T. Gulczyński. Advanced ensemble methods using machine learning and deep learning for one-day-ahead forecasts of electric energy production in wind farms. *Energies*, 15(4):1252, 2022.
- G. Riahy and M. Abedi. Short term wind speed forecasting for wind turbine applications using linear prediction method. *Renewable energy*, 33(1):35–41, 2008.

-
- J. Shi, J. Guo, and S. Zheng. Evaluation of hybrid forecasting approaches for wind speed and power generation time series. *Renewable and Sustainable Energy Reviews*, 16(5):3471–3480, 2012.
- Statista Research Department. Europe: Electricity demand per capita 2022. <https://www.statista.com/statistics/1262471/per-capita-electricity-consumption-europe/>, 2022.
- R. Tawn and J. Browell. A review of very short-term wind and solar power forecasting. *Renewable and Sustainable Energy Reviews*, 153:111758, 2022. ISSN 1364-0321. doi: <https://doi.org/10.1016/j.rser.2021.111758>. URL <https://www.sciencedirect.com/science/article/pii/S1364032121010285>.
- United Nations Convention on Climate Change. Paris Agreement. Climate Change Conference (COP21), Dec. 2015. URL https://unfccc.int/sites/default/files/english_paris_agreement.pdf.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- Y. Wang, R. Zou, F. Liu, L. Zhang, and Q. Liu. A review of wind speed and wind power forecasting with deep neural networks. *Applied Energy*, 304:117766, 2021. ISSN 0306-2619. doi: <https://doi.org/10.1016/j.apenergy.2021.117766>. URL <https://www.sciencedirect.com/science/article/pii/S0306261921011053>.
- R. Zhang. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pages 7324–7334. PMLR, 2019.
- L. Zimmer, M. Lindauer, and F. Hutter. Auto-pytorch tabular: Multi-fidelity metalearning for efficient and robust autodl. *CoRR*, abs/2006.13799, 2020. URL <https://arxiv.org/abs/2006.13799>.

A WINDDRAGON

Search algorithm. The DRAGON framework contains operators, namely mutation and crossover, which are commonly used in meta-heuristics such as the evolutionary algorithm and the simulated annealing, to optimize graphs. The mutation operators are used to add, remove, or modify nodes and connections in the graph, as well as to modify the operations and their hyperparameters within the nodes. Crossover involves exchanging parts of two graphs. The mutation and crossover operators were utilised to construct a steady-state (asynchronous) evolutionary algorithm. Compare to the original algorithm, this version enhances efficiency on HPC by producing two offsprings from the population as soon as a free process is available, rather than waiting for the entire population to be evaluated (Liu et al., 2018).

With the division by region, we slightly modified the generic evolutionary algorithm in WindDragon to avoid having to run an optimisation by region, which would be very costly. In this context, a deep neural network f from our search space Ω is parametrized by its architecture α and its hyperparameters λ . Once α and λ have been settled, the model is trained on the data to optimize the weights θ . We assumed that the architecture α and the hyperparameters λ would be broadly similar across regions. Therefore, we modified our evolutionary algorithm to process all regions at the same time. We create and evolve α and λ independently of the region, and, to optimize the weights θ , we randomly select the region on which the model would be train and evaluate. In order not to penalize models that have been evaluated on regions that are difficult to predict, we use a global loss function, which consists in dividing the loss obtained on the region ℓ_{region} by the loss of our baseline CNN model on that region, $\mathcal{L}_{\text{region}}$. During the optimisation, for each region, we progressively save the best model evaluated on it. The pseudo code for our steady-state evolutionary algorithm can be found Algorithm 1.

Algorithm 1: Steady-state evolutionary algorithm for multi-regions wind power forecasting

Inputs:

- Ω search space
- $[\mathcal{L}_{\text{region}_1}, \dots, \mathcal{L}_{\text{region}_R}]$ CNN losses for each region
- K population size
- T number of iteration

Initialization

Sample K untrained models $f^{\alpha_1, \lambda_1}, \dots, f^{\alpha_K, \lambda_K}$ from Ω

For $k = 1, 2, \dots, K$

Select a region r to train the model

Train f^{α_k, λ_k} to get the model weights θ_k^r on the region r

Get the loss ℓ_r^k on this region, and set the model loss to $\ell_k = \ell_r^k / \mathcal{L}_r$

If ℓ_r^k is the best loss so far on r , save $f_{\theta_k^r}^{\alpha_k, \lambda_k}$

For $t = K + 1, K + 3, K + 5, \dots, T$

Select two parents $f^{\alpha_{k1}, \lambda_{k1}}$ and $f^{\alpha_{k2}, \lambda_{k2}}$ from the population based on their loss ℓ_k

Mutate and evolve $f^{\alpha_{k1}, \lambda_{k1}}$ and $f^{\alpha_{k2}}$ to $f^{\alpha_{K+t}, \lambda_{K+t}}$ and $f^{\alpha_{K+t+1}, \lambda_{K+t+1}}$

Select two regions r_A and r_B

Train respectively $f^{\alpha_{K+t}, \lambda_{K+t}}$ and $f^{\alpha_{K+t+1}, \lambda_{K+t+1}}$ on r_A and r_B to optimize $\theta_{K+t}^{r_A}$ and

$\theta_{K+t+1}^{r_B}$

Get the losses $\ell_{r_A}^{K+t}$ and $\ell_{r_B}^{K+t+1}$, and set the models losses to $\ell_{K+t} = \ell_{r_A}^{K+t} / \mathcal{L}_{r_A}$ and

$\ell_{K+t+1} = \ell_{r_B}^{K+t+1} / \mathcal{L}_{r_B}$

If $\ell_{r_A}^{K+t}$ or $\ell_{r_B}^{K+t+1}$ are respectively the best losses so far on r_A or r_B save $f_{\theta_{r_A}^{\alpha_{K+t}, \lambda_{K+t}}}$ or

$f_{\theta_{r_B}^{\alpha_{K+t+1}, \lambda_{K+t+1}}}$

If ℓ_{K+t} or ℓ_{K+t+1} are lower than the maximum population loss, we replace the worst model with the new one

Output:

The best saved model by region

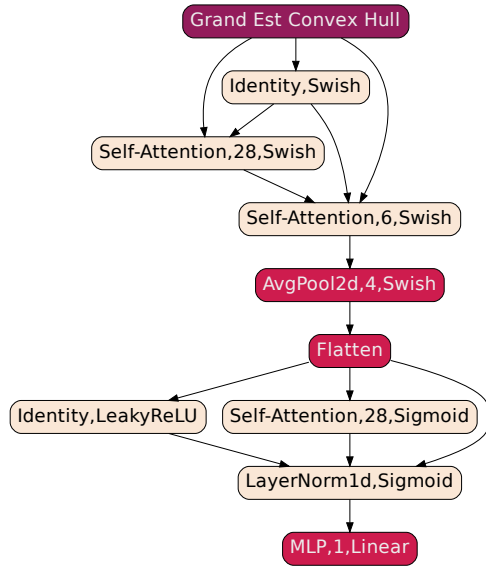


Figure 5: Dragon automatically found architecture applied on the Grand Est region.

Results. The outputs of WindDragon would be by region the best model found during the optimisation and the prediction of this model. The found architectures vary a bit from a region to another. An example of the best model for the region Grand Est can be found Figure 5. This architecture uses self-attention just like in the Transformer (Vaswani et al., 2017), but without the patches that can be found in the ViT architecture. The model is also a lot smaller than a Transformer, which can explain why it outperforms the other baselines on this region

B BASELINES DETAILS

The baselines used in Section 3 are explained in more detail below.

Convolutional Neural Network (CNN). Figure 6 shows the architecture of the CNN baseline that we implemented. We used a simple grid search to optimize the hyperparameters (e.g. the number of layers, the kernel sizes, the activation functions)

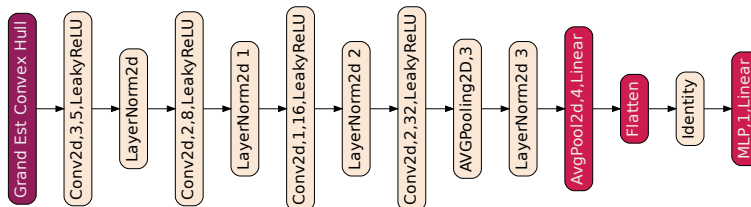


Figure 6: CNN architecture applied on the Grand Est region.

Vision Transformer (ViT). The Vision Transformer used in this paper is based on SimpleViT’s (Beyer et al., 2022) architecture. We reused the implementation from lucidrains package⁵.

⁵<https://github.com/lucidrains/vit-pytorch>

XGboost on the mean of the NWP wind speed map. Figure 7 shows the two-steps procedure of the XGboost baseline.

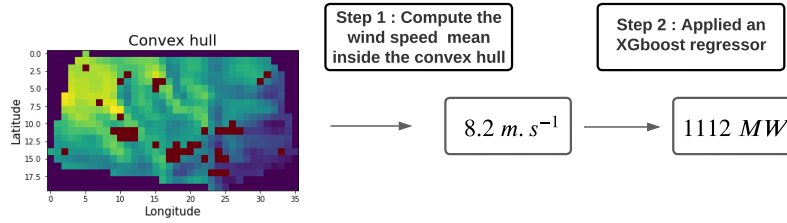


Figure 7: Visual illustration of the XGB two-steps approach on the Auvergne-Rhône-Alpes region.

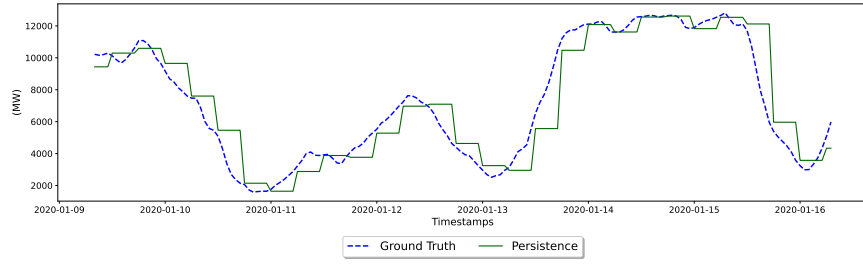
C ADDITIONAL EXPERIMENTAL RESULTS

C.1 REGIONAL RESULTS

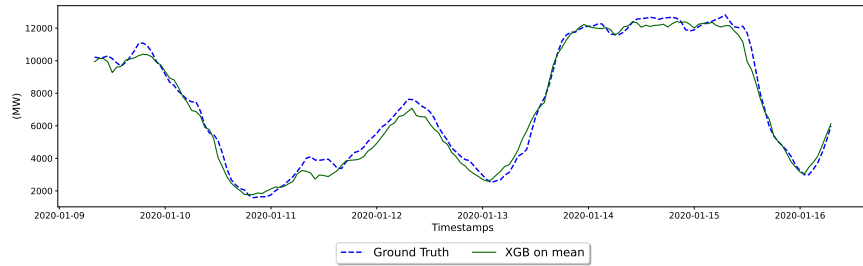
Table 2: Regional results. The best results are highlighted in bold and the best second results are underlined.

Region	WindDragon		CNN		ViT		XGB on mean		Persistence	
	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE	MAE (MW)	NMAE
Auvergne-Rhône-Alpes	19.5	14.9 %	<u>19.6</u>	<u>15.0 %</u>	21.6	16.5 %	29.2	22.4 %	28.7	22.0 %
Bourgogne-Franche-Comté	32.9	14.8 %	<u>34.1</u>	<u>15.4 %</u>	37.2	16.8 %	42.3	19.1 %	58.7	26.6 %
Bretagne	36.1	14.1 %	<u>38.0</u>	<u>14.9 %</u>	39.9	15.6 %	47.1	18.4 %	67.2	26.3 %
Centre-Val de Loire	53.3	15.0 %	<u>57.3</u>	<u>16.1 %</u>	59.0	16.6 %	61.9	17.5 %	96.7	27.3 %
Grand Est	125.6	12.5 %	<u>130.5</u>	<u>13.1 %</u>	161.0	16.1 %	148.8	14.9 %	251.2	25.1 %
Hauts-de-France	159.7	12.1 %	<u>167.6</u>	<u>12.7 %</u>	177.0	13.4 %	178.8	13.5 %	320.1	24.2 %
Île-de-France	6.8	22.6 %	<u>7.17</u>	<u>23.7 %</u>	7.4	24.3 %	7.5	24.9 %	9.5	31.5 %
Normandie	29.6	12.7 %	<u>30.8</u>	<u>13.2 %</u>	31.2	13.4 %	36.8	15.8 %	55.9	24.0 %
Nouvelle-Aquitaine	43.1	15.7 %	<u>44.0</u>	<u>16.4 %</u>	48.4	17.6 %	53.7	19.6 %	77.9	28.4 %
Occitanie	51.2	12.3 %	<u>55.8</u>	<u>13.5 %</u>	64.1	15.5 %	91.6	22.1 %	96.3	23.2 %
PACA	3.5	32.4 %	<u>3.5</u>	<u>32.4 %</u>	4.0	37.2 %	4.5	41.4 %	4.3	39.5 %
Pays de la Loire	37.1	13.6 %	<u>39.0</u>	<u>14.3 %</u>	39.9	14.7 %	41.9	15.4 %	74.9	27.5 %

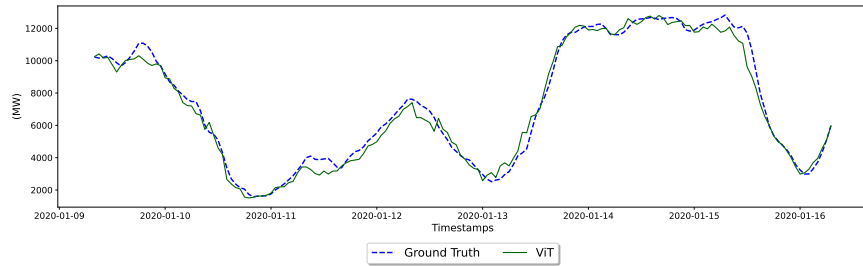
C.2 WEEKLY COMPARATIVE VISUALS OF ALL BASELINE RESULTS



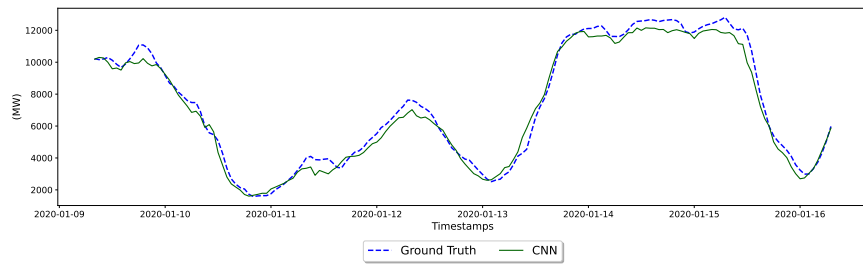
(a) Persistence forecast



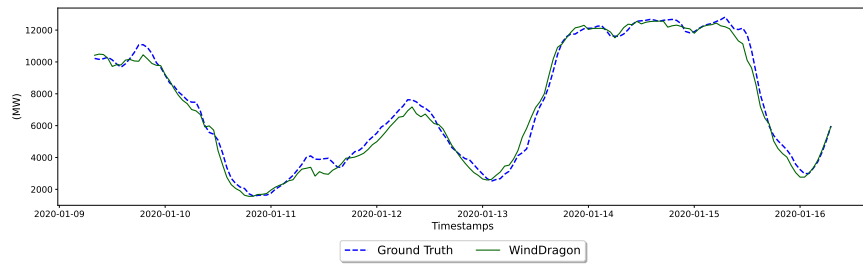
(b) XGB on mean forecast



(c) Vision Transformer forecast



(d) Convolutional Neural Network forecast



(e) WindDragon forecast

Figure 8: Weekly comparative visuals