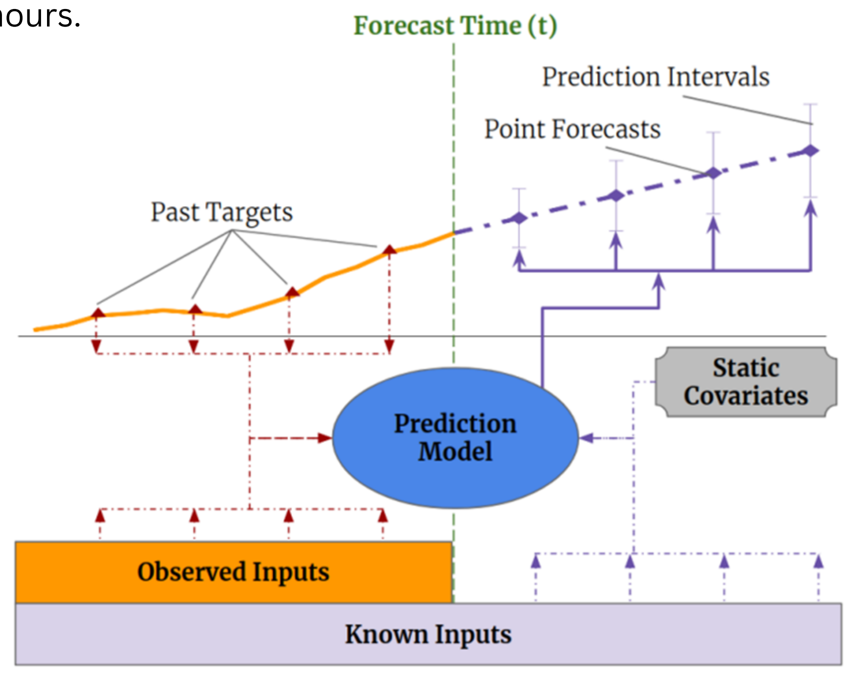
WIND POWER FORECASTING

Tymoteusz Barcinski (s221937), Jan Eberle (s221667), Sumukha Shridhar (s221654)

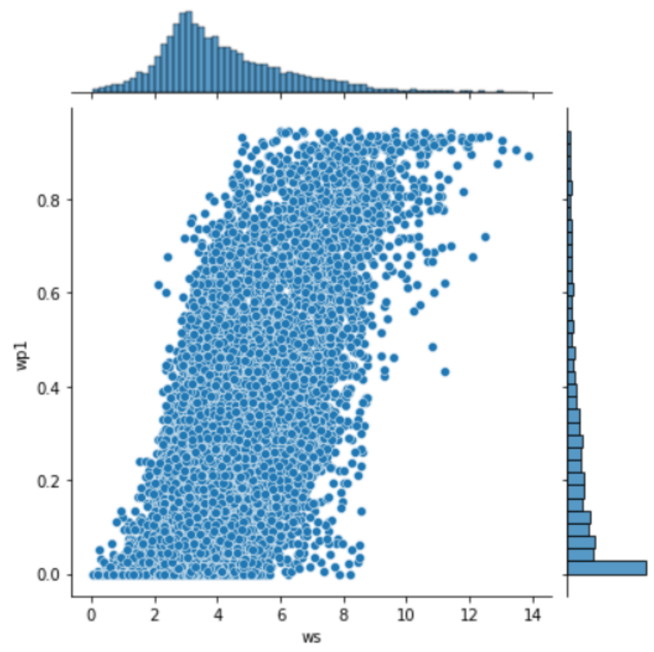
PROBLEM FORMULATION

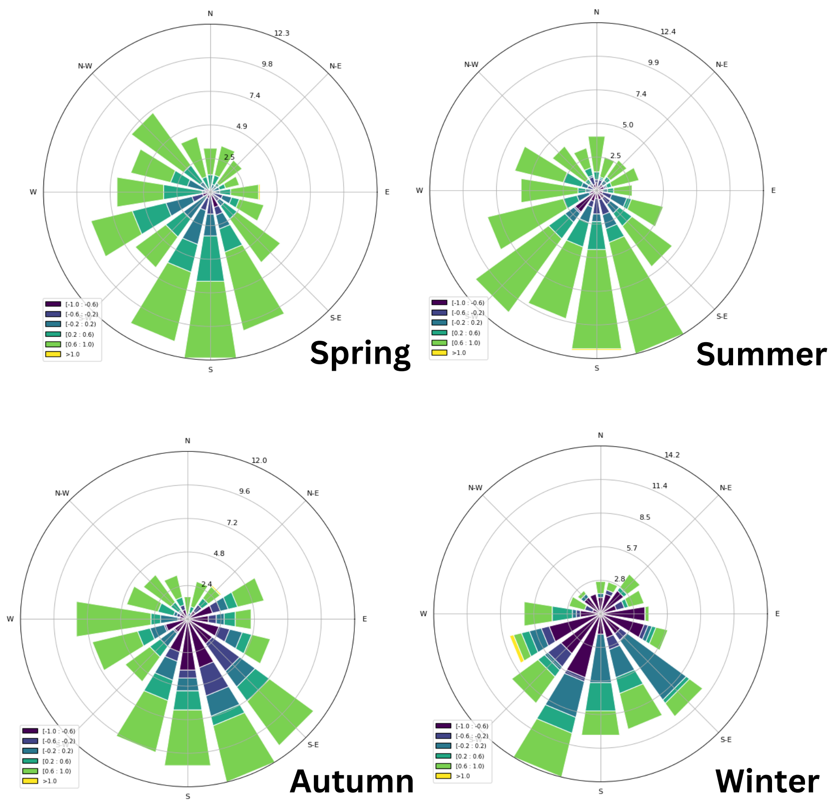
- Windmills are a crucial part of our entire power grid. Wind power forecasting is a difficult non-linear multivariate multi-step time series problem with deep learning approaches entering the field.
- The goal is to predict the wind power 36 hours ahead at noon and midnight given the historical data and weather forecast for the next 48 hours.



DATA & EXPOLORATORY ANALYSIS

- The data comes from a Kaggle competition held in 2012.
- It includes 7 windmills with a 48 forecast every 12 hours for 1,5 years.
- The windspeed is highly corelated to the windpower produced by the plant





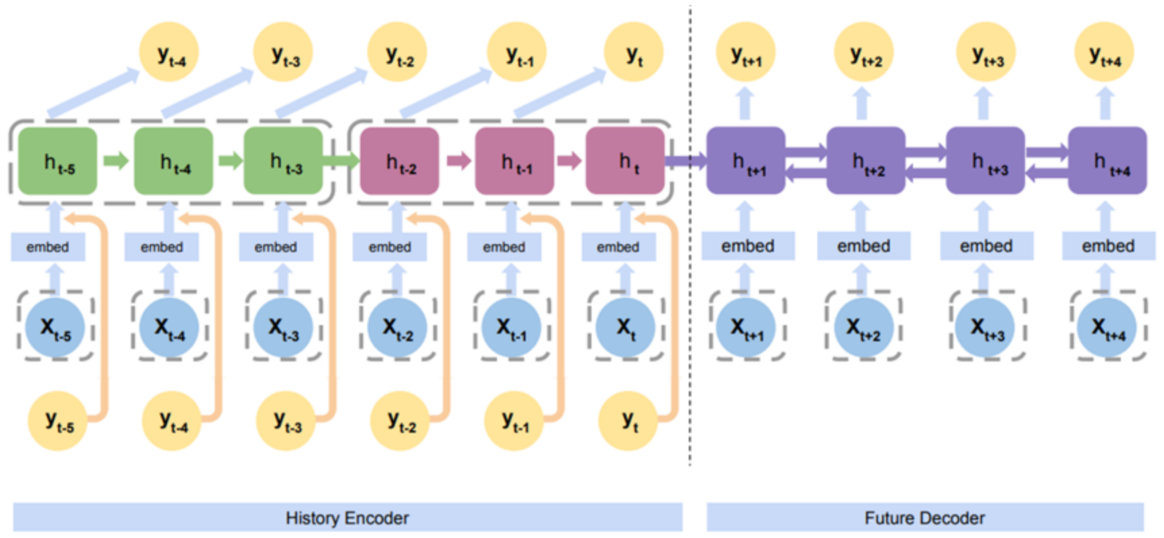
PREPROCESSING

- In order to include the cyclic variables in the model, such as wind direction and months we applied trigonometric transformations.
- We used a sliding window approach over the past observation of the size 60.
- We standardized the data
- we used data augmentation and trained the model to give power prediction not only at noon and midnight but every hour. This approach gave us 12 times more data to train the model.

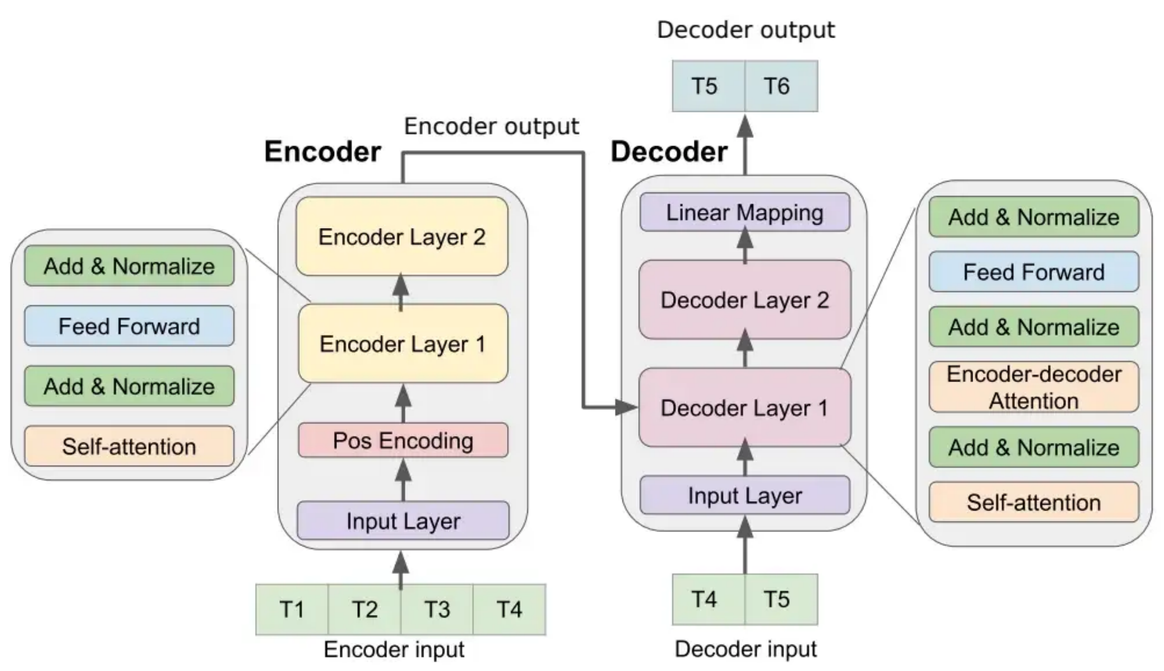
	u	v	ws	wd_sin	wd_cos	month_sin	month_cos	wp1
2009-07-01 01:00:00	0.233662	-0.198982	-0.858144	0.928484	-0.382437	1.224647e-16	-1.0	-0.657462
2009-07-01 02:00:00	0.181121	-0.268072	-0.895117	0.873757	-0.518663	1.224647e-16	-1.0	-0.929667
2009-07-01 03:00:00	0.187689	-0.344071	-0.837016	0.821647	-0.621071	1.224647e-16	-1.0	-0.762156
2009-07-01 04:00:00	0.236946	-0.409707	-0.720814	0.793305	-0.669613	1.224647e-16	-1.0	-0.824973

ARCHITECTURES

Encoder-Decoder LSTM with autoregressive component



Transformer



MODELLING

1. Hyperparameter training with several different parameters
2. Training with most promising parameters for 400 epochs
3. Validate the models
4. *Test the models*

In order to validate the model, we used the time series split cross-validation. To keep the time dependency, after the validation loss was calculated on one 36 hour horizon, the model was retrained. We used the ADAM optimizer and MSE as a loss criteria.

Encoder-Decoder-LSTM:

Parameter:

- learning_rate = [1e-04, 1e-05, 1e-06]
- hidden_size = [300, 320, 340],
- num_layers = [3, 5],
- hidden_size_backward = [64, 128, 258],
- num_layers_backward = [1, 2, 3]

Training:

- 400 Epochs on DTU cluster
- ~8.5 hours
- training loss RMSE: 0.496
- validation loss RMSE: 0.598

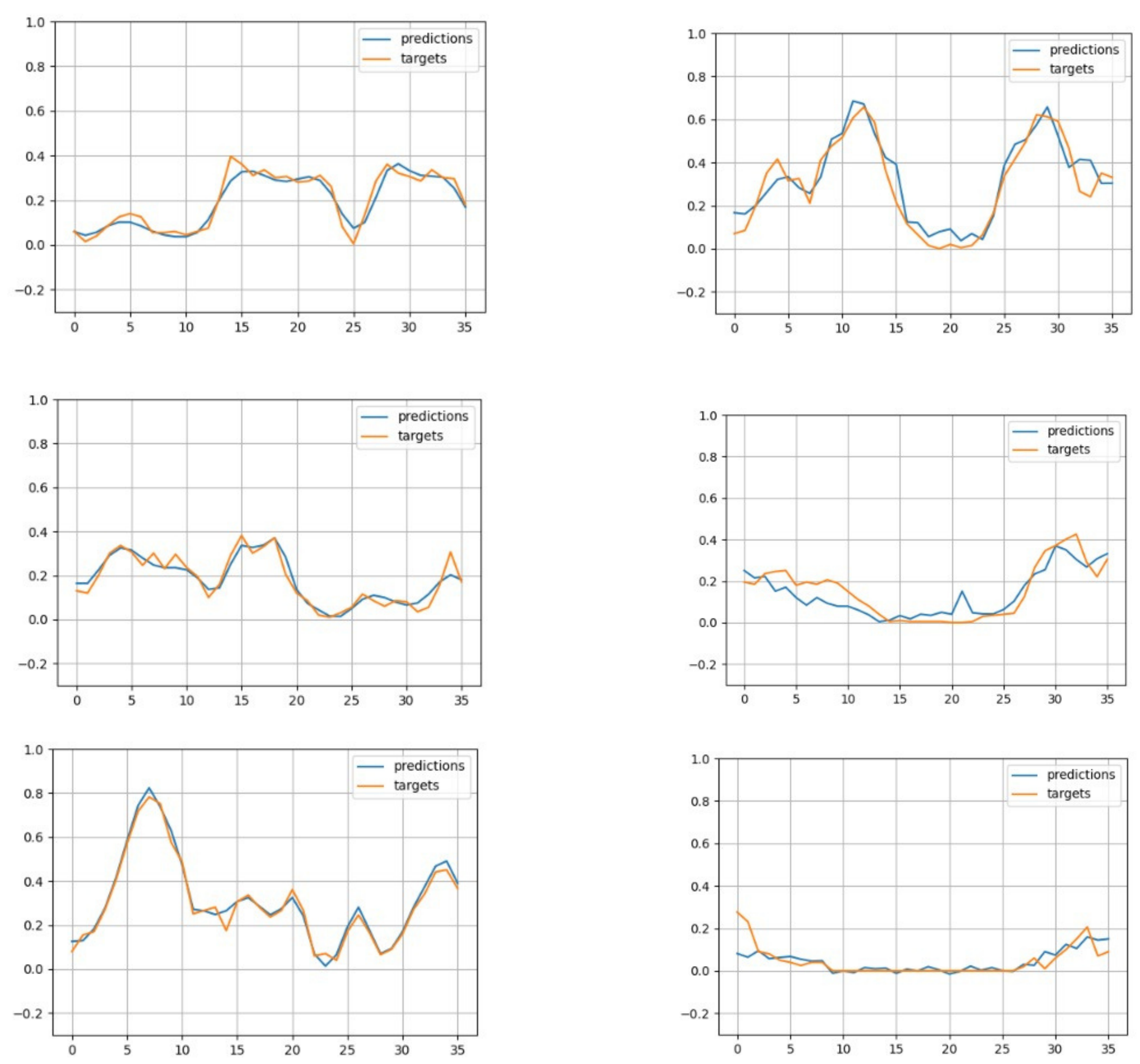
Transformer:

Parameter:

- learning_rate = [1e-04, 1e-05, 1e-06]
- dimension_val = [520, 560, 640]
- n_heads = [5, 7, 8]
- n_decoder_layers = [4, 5, 6]
- n_encoder_layers = [4, 5, 6]

Training:

- 400 Epochs on DTU cluster
- ~3.5 hours
- training loss RMSE: 0.208
- validation loss RMSE: 0.729



REFERENCES

- Short-term wind power forecasting approach based on Seq2Seq model using NWP data (2020), Yu Zhang , Yanting Li , Guangyao Zhang
- Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case (2020), Neo Wu, Bradley Green, Xue Ben, Shawn O'Banion
- Short-term Wind Power Prediction Model Based on Encoder-Decoder LSTM (2018) Kuan LU1, Wen Xue SUN2 , Xin Wang and others

CONCLUSION

- The transformer adapts pretty well to the training data but has a huge problem with overfitting. The LSTM model fits slowly to the training data and will need longer training show good results.

Future work:

- investigate further the hyperparameters of the models especially generalization
- use the CNN network to extract the features from the sliding window and input it to the encoder
- create a model ensemble to account for 7 wind plants and predict the power production simultaneously