MSc ESDA Title Page

UCL Candidate Code: LSHM1, MHCR1, PDXT9, PKHJ8, LDYY7

Module Code: BENV0148

Module Title: Advanced Machine Learning for Energy Systems

Coursework Title: Forecasting Day-ahead electricity prices in Norwegian NO2 bidding zone using deep learning

Module Leader: Dr. Amir Gharavi

Date: 28/04/2025

Word Count: 2933

By submitting this document, you are agreeing to the Statement of Authorship:

I/We certify that the attached coursework exercise has been completed by me/us and that each and every quotation, diagram or other piece of exposition which is copied from or based upon the work of other has its source clearly acknowledged in the text at the place where it appears.

I/We certify that all field work and/or laboratory work has been carried out by me/us with no more assistance from the members of the department than has been specified.

I/We certify that all additional assistance which I/we have received is indicated and referenced in the report.

Please note that penalties will be applied to coursework which is submitted late, or which exceeds the maximum word count. Information about penalties can be found in your Course Handbook which is available on Moodle: https://moodle.ucl.ac.uk/mod/book/view.php?id=2234010

- ***Penalties for late submission***
- ***Penalties for going over the word count***

In the case of coursework that is submitted late and is also over length, then the greater of the two penalties shall apply. This includes research projects, dissertations and final reports.

Forecasting Day-ahead electricity prices in Norwegian NO2 bidding zone using deep learning

1. Introduction

The increasing complexity and volatility of electricity markets have amplified the need for accurate price forecasting, particularly in liberalised markets like the Nordic power system. Day-ahead electricity prices are central to operational planning, trading strategies, and market efficiency, yet their prediction remains a challenging task due to the influence of numerous dynamic and nonlinear factors.

This project focuses on forecasting day-ahead electricity prices in the Norwegian NO2 bidding zone, a region characterised by significant hydropower generation, high renewable integration, and strong interconnections with other European markets. The NO2 zone plays a crucial role in regional energy flows, and price volatility here can have broader implications for market participants and grid stability.

To address this forecasting challenge, the project leverages deep learning techniques, which have demonstrated strong performance in capturing complex temporal dependencies and nonlinear patterns in time-series data. Specifically, the study compares the performance of traditional machine learning models with deep learning architectures such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) and Temporal Convolutional Networks (TCNs).

A comprehensive dataset is constructed using publicly available data from ENTSO-E Open-Meteo, load forecasts, weather conditions and transmission constraints. The models are evaluated based on standard error metrics, and their ability to generalise across different time periods is analysed.

By building and comparing advanced forecasting models, this project aims to contribute to the ongoing development of intelligent tools for energy market participants, ultimately supporting a more efficient and sustainable power system in Norway and the broader Nordic region.

2. Literature review

Electricity price forecasting (EPF) is a well-established but continually evolving research area due to the growing volatility of energy markets, increased renewable integration, and liberalization of

power systems. Day-ahead electricity markets, where prices are determined through hourly auctions, are particularly sensitive to fluctuations in demand, renewable supply, weather conditions, and cross-border flows — making accurate forecasting both crucial and challenging.

Historically, electricity prices have been forecast using statistical methods such as Autoregressive Integrated Moving Average (ARIMA) and Generalised Autoregressive Conditional Heteroskedasticity (GARCH). These models are valued for their simplicity and interpretability but often underperform in volatile or nonlinear settings (Weron, 2014). Their limited ability to capture complex market dynamics has led to a shift toward more data-driven approaches.

Machine learning (ML) models, such as Support Vector Machines, Random Forests, and Gradient Boosted Trees, have shown improved performance in EPF by modelling nonlinear patterns and interactions between variables. Lago, De Ridder, and De Schutter (2021) conducted a comprehensive review and empirical comparison of ML approaches, among other things finding that tree-based models often outperform classical statistical models, especially in shorter-term forecasts.

More recently, deep learning methods, particularly Recurrent Neural Networks (RNNs) and their variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models, have gained popularity due to their capacity to model sequential dependencies in time-series data. LSTMs, introduced by Hochreiter and Schmidhuber (1997), have been widely used in electricity markets with success (Lago et al., 2019), although they can be computationally expensive. GRUs offer similar accuracy with fewer parameters and faster training (Chung et al., 2014), making them a viable alternative.

Beyond recurrent architectures, Temporal Convolutional Networks (TCNs) have emerged as an alternative by using dilated causal convolutions to model long-range temporal dependencies without recurrence. Bai, Kolter, and Koltun (2018) demonstrated that TCNs can outperform LSTMs in several sequence modelling tasks while offering parallel computation and greater training efficiency. Hybrid models combining CNNs and RNNs have also been explored, allowing local feature extraction followed by temporal learning (Zhang et al., 2018).

Attention-based models and transformers represent the latest developments in time-series forecasting. Zhou et al. (2021) introduced Informer, a transformer variant designed for long-

sequence forecasting that improves accuracy while reducing computational complexity. Although transformers offer exciting possibilities, they will not be tested in this study.

The Nordic market, and Norway in particular, presents unique challenges due to its hydro-dominated energy mix, weather dependency, and strong cross-border interconnections. Weron (2014) emphasises the importance of contextual data such as water reservoir levels and physical flows in improving price forecasts. However, research applying deep learning specifically to the Norwegian NO2 bidding zone is limited, providing the rationale for this project.

3. Methodology

This study aims to forecast day-ahead electricity prices in the Norwegian NO2 bidding zone by constructing a comprehensive, multivariate time series dataset and applying deep learning models. The methodology involves data collection from multiple sources, preprocessing, and temporal alignment of all variables to an hourly resolution.

3.1 Data Collection and Processing

We collected publicly available data from the ENTSO-E Transparency Platform using the API. The following datasets were retrieved:

- **Load Forecasts**: Hourly forecasts for NO2 and neighbouring bidding zones where data was available — including NO1, NO5, Denmark (DK), the Netherlands (NL), and Germany-Luxembourg (DE_LU).
- **Net Transfer Capacities (NTC)**: Week-ahead NTC values between NO2 and the aforementioned neighbouring zones.
- **Wind and Solar Forecasts:** Expected renewable energy production, which directly impacts electricity supply and pricing.
- **Hydropower Reservoir Levels**: Weekly reservoir data specific to NO2.

Additionally, we collected weather forecast data using the Open-Meteo API. For each bidding zone, a representative location was selected:

- NO1: Oslo (59.9127, 10.7461)
- NO2: Kristiansand (58.1467, 7.9956)

- NO5: Bergen (60.3930, 5.3242)
- Denmark (DK): Aalborg (57.0480, 9.9187)
- Netherlands (NL): Rotterdam (51.9225, 4.4792)
- Germany-Luxembourg (DE_LU): Kiel (54.3213, 10.1349)

From Open-Meteo, we extracted hourly forecasts for:

- **Temperature** (2 meters above ground)
- **Total Cloud Cover** (used as a proxy for solar generation conditions)
- **Wind Speed** (80 meters above ground, relevant for wind generation)

The final dataset spans from 1 October 2023 to 30 March 2025, with all data aligned to an hourly frequency. For weekly data (i.e., hydro reservoirs), the most recent available value was carried forward across the relevant hours.

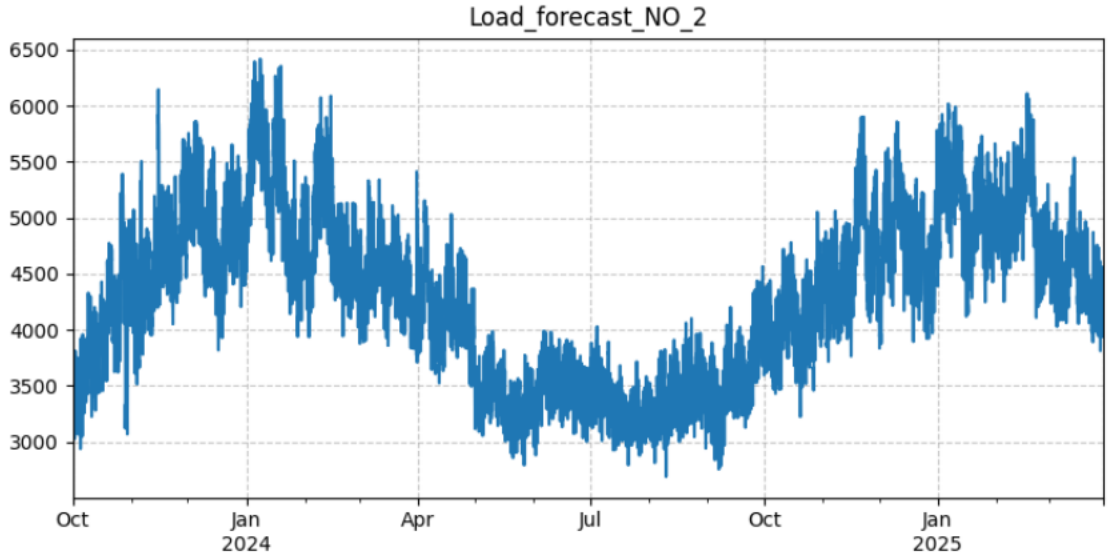


Figure 1: Load forecast in NO2 zone [MW]

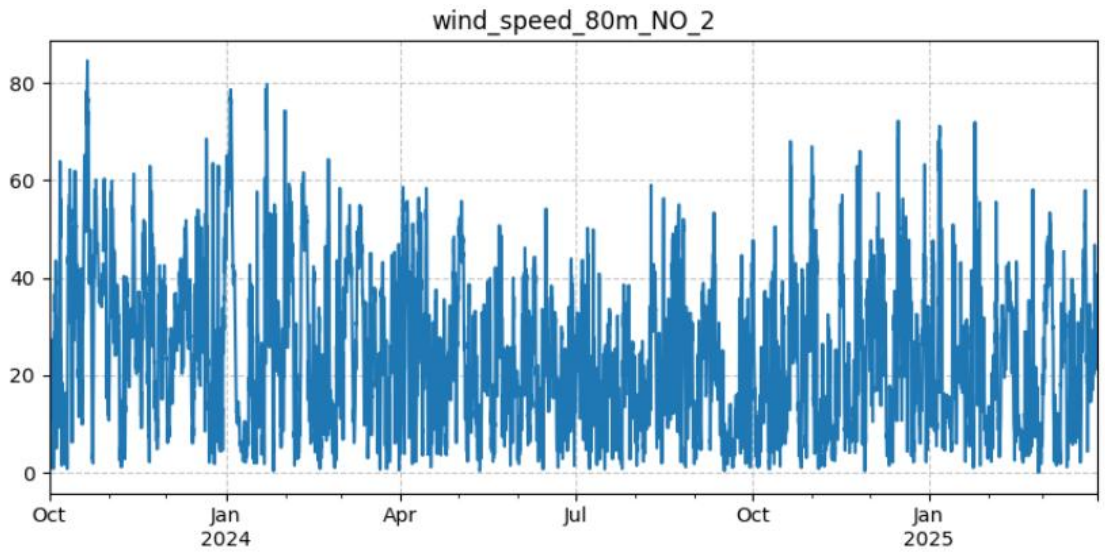


Figure 2: Wind speed in NO2 zone [km/h]

Two representative time series plots are included to illustrate key trends in the dataset. Figure 1 shows the load forecast for NO2, revealing clear seasonal and weekly demand patterns. Energy consumption tends to rise during winter months, with pronounced fluctuations suggesting weekday-weekend cycles. These patterns are consistent with temperature-driven demand and societal activity levels.

Figure 2 presents wind speed at 80 meters for NO2, which is critical for estimating wind energy generation potential. Unlike the load forecast, the wind speed series exhibits higher volatility and less obvious seasonality.

3.3 Model Selection

To evaluate the effectiveness of deep learning approaches in forecasting day-ahead electricity prices in the Norwegian NO2 bidding zone, we implemented a baseline traditional machine learning models and four advanced neural network architectures. These models are trained on the same multivariate dataset and evaluated using a consistent set of error metrics.

Baseline Model

- **Extreme Gradient Boosting (XGBoost):** An ensemble of boosted trees that captures nonlinearities and interactions, providing strong baseline performance at low computational cost.

Deep Learning Models

All models use the TensorFlow Keras API and predict electricity prices 24 hours ahead based on historical inputs.

1. **Recurrent Neural Network (RNN):** The basic form of recurrent networks that uses a hidden state to retain sequential information. While effective for short sequences, RNNs suffer from vanishing gradients, limiting their ability to model long-term dependencies.
2. **Long Short-Term Memory (LSTM):** A type of RNN with memory cells and gating mechanisms (input, forget, output gates) that allow the network to retain important information over long time horizons.
3. **Gated Recurrent Unit (GRU):** A simplified variant of LSTM that uses two gates (reset and update) to control information flow. GRUs often achieve comparable performance to LSTMs with fewer parameters, resulting in faster training and reduced overfitting.
4. **Temporal Convolutional Network (TCN):** A 1-dimentional convolutional architecture that uses causal and dilated convolutions to process sequences in parallel. TCNs are capable of modelling long-range dependencies without the need for recurrence, and their parallelism enables faster training on longer sequences.

Each model was trained with early stopping and dropout for regularization. Hyperparameters such as sequence length, number of layers, units per layer, batch size, and learning rate are tuned using time-series cross-validation.

3.4 Evaluation Metrics

To assess model performance, we use a combination of point error metrics and directional accuracy metrics:

- **R-squared**

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Shows how well the model explains the variance in data. Adjusted R-squared was also added which adjusts for the number of predictors, preventing overfitting.

- **Mean Absolute Error (MAE)**

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

A robust and interpretable metric that calculates the average magnitude of the errors.

- **Root Mean Squared Error (RMSE)**

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Penalises larger errors more heavily than MAE, highlighting volatility in predictions.

- **Mean Absolute Percentage Error (MAPE)**

$$\mathrm{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

Expresses the average forecast error as a percentage of actual values. It is useful for interpretation but should be used with caution when actual values approach zero.

- **Directional Accuracy (DA)**

$$\mathrm{DA} = \frac{1}{n-1}\sum_{i=2}^{n}\mathbb{I}\left((y_i - y_{i-1})(\hat{y}_i - \hat{y}_{i-1}) > 0\right)$$

Evaluates whether the model correctly predicts the direction of price movement. This is especially relevant in energy markets where directional decisions impact bidding and trading.

4. Deep Learning Model Design and Implementation

This section presents the design and implementation of deep learning models developed to forecast day-ahead electricity prices in Norway's NO2 region. Multiple architectures were constructed and

evaluated. Prior to training, the dataset was split into training and test sets, with features (X_train, X_test) standardized separately to prevent data leakage.

4.1 XGBoost model

The XGBoost model was trained with optimised hyperparameters, including a learning rate of 0.05, 200 trees, and a maximum depth of 3, using full subsampling of rows and features. The model targeted minimising squared error and showed good predictive performance, capturing trends in the data while maintaining a balance between bias and variance. Figure 3 illustrates a satisfactory fit, although underfitting is evident.

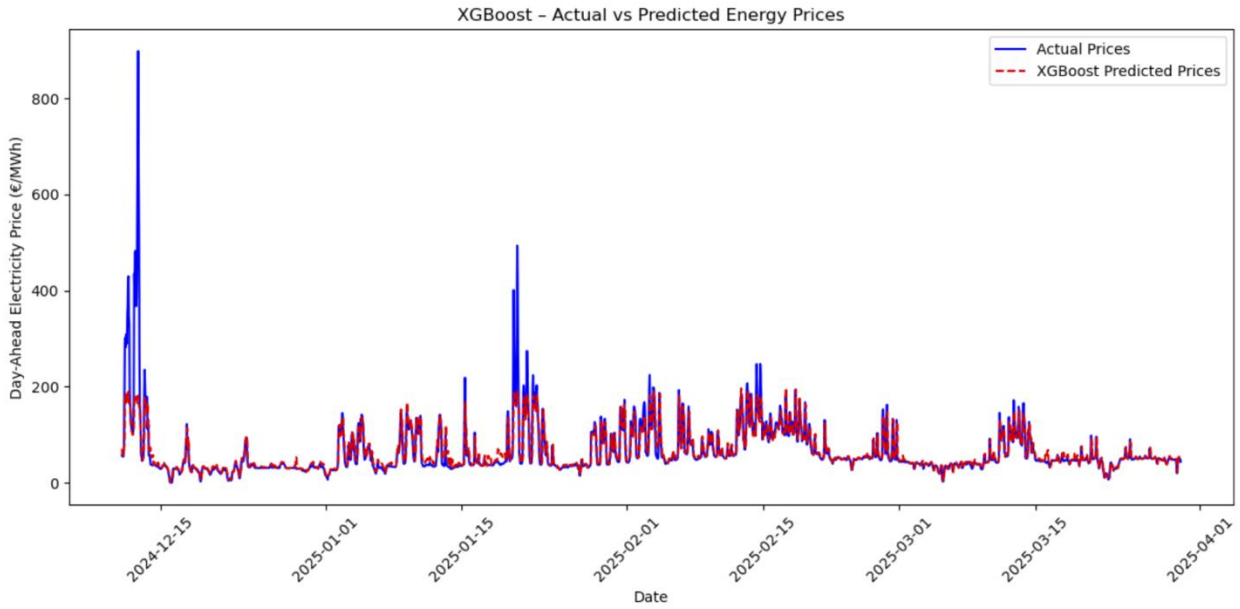


Figure 3: Actual vs predicted price of XGBoost model

4.2 RNN model

The vanilla RNN uses a single-layer SimpleRNN architecture with 64 neurons and uses the ReLU activation function to introduce non-linearity. To improve generalisation and reduce overfitting, an L2 regularisation term ($\alpha = 0.0001$) was applied to the recurrent layer, along with a Dropout layer set at 10%. The model was optimised using the Adam optimiser with a learning rate of 0.001 and trained over 50 epochs with a batch size of 32. Early performance evaluations showed stable learning without significant signs of overfitting. Figure 4 demonstrates a strong overall fit, with improved spike prediction compared to XGBoost, although some discrepancies remain.

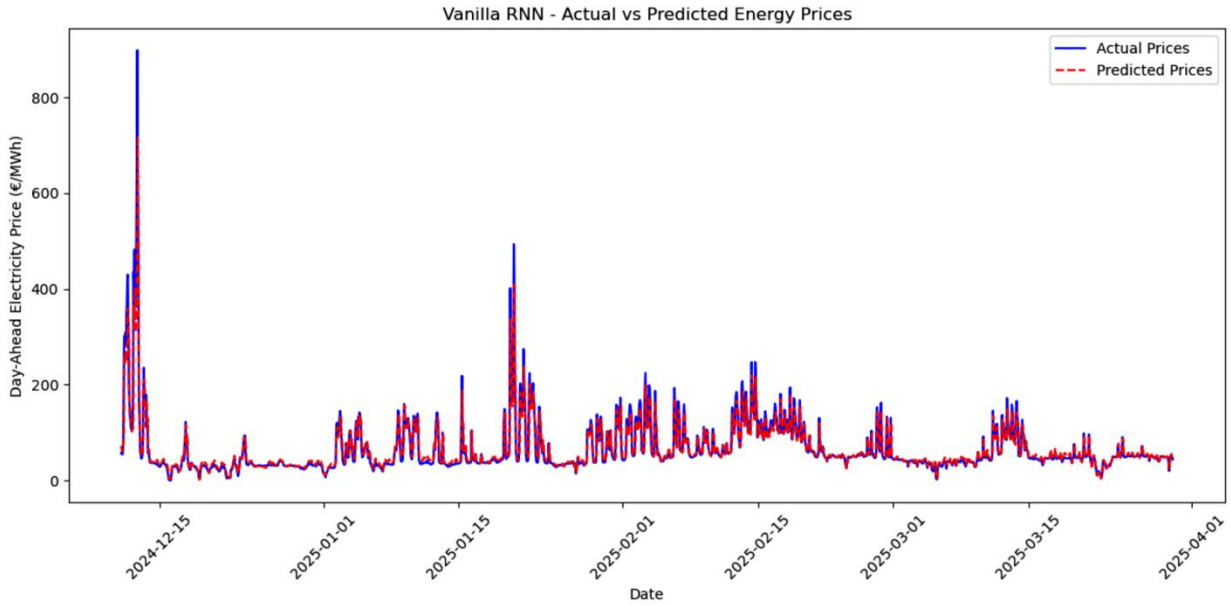


Figure 4: Actual vs predicted price of RNN model

4.3 LSTM model

The LSTM model consists of a single-layer architecture with 64 neurons and ReLU activation to introduce non-linearity. To mitigate overfitting, a 10% Dropout layer and L2 regularisation (α = 0.001) were applied to both the input and recurrent connections. The model was optimised using the Adam optimiser with an early stopping strategy to enhance training efficiency. Hyperparameters were tuned through grid search, with the best performance achieved using a learning rate of 0.001 and a batch size of 64. Figure 5 displays a good fit, although the predicted prices are consistently slightly higher than the actual values.

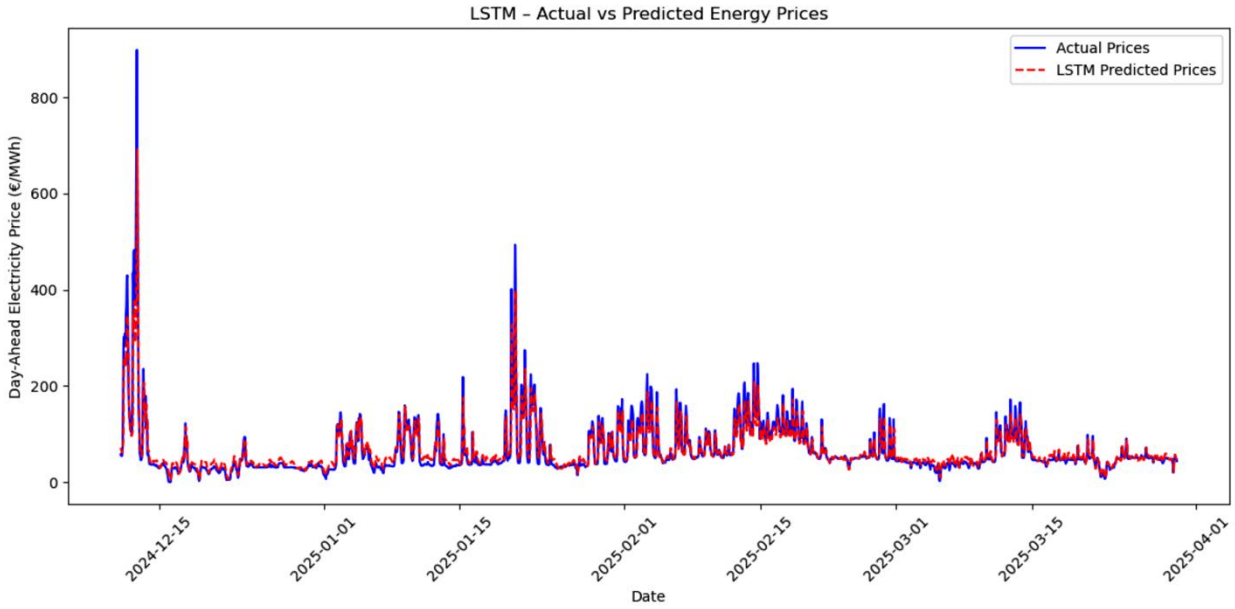


Figure 5: Actual vs predicted price of LSTM model

4.4 GRU model

The GRU model consisted of a single-layer Gated Recurrent Unit architecture with 32 neurons, using the ReLU activation function to introduce non-linearity. To enhance the model's robustness and mitigate overfitting, an L2 regularisation term (α = 0.001) was applied, together with a Dropout layer of 10%. The model was optimised using the Adam optimiser with a learning rate of 0.001 and trained over 50 epochs with a batch size of 64. The final model evaluation showed stable training behaviour. Figure 6 shows a generally good fit.

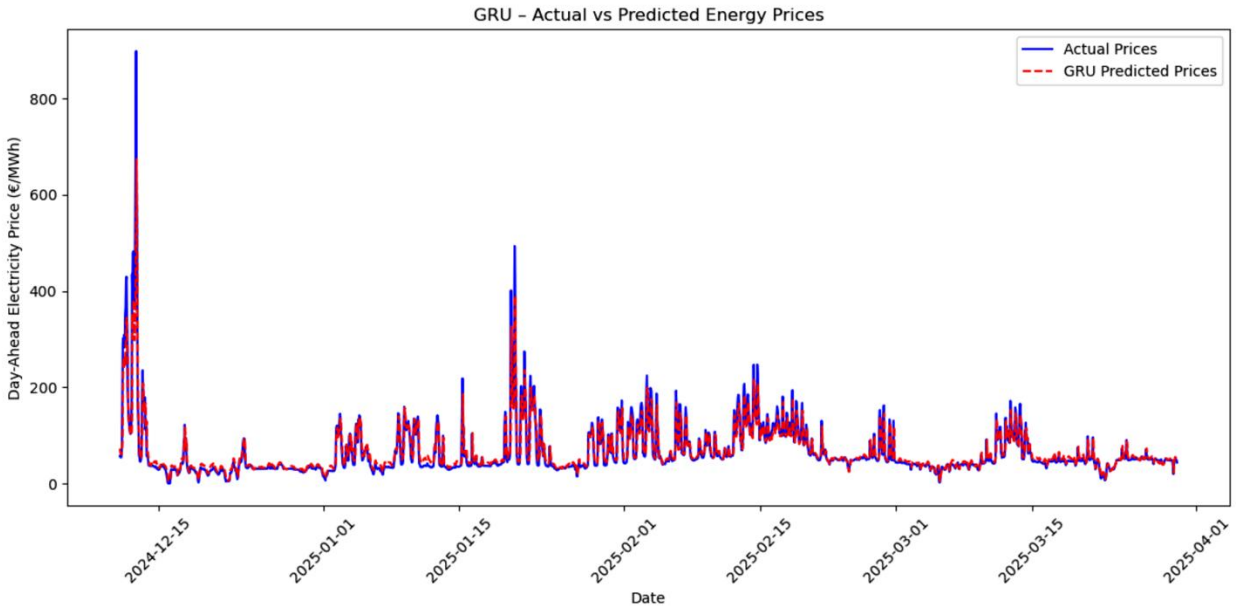


Figure 6: Actual vs predicted price of GRU model

4.5 TCN model

The TCN model consisted of a one-dimensional convolutional layer followed by a Dense layer, both using ReLU activation. Overfitting was mitigated through Early Stopping and a 10% Dropout layer within the TCN. The convolutional layer included 32 filters with a kernel size of 3, and used dilation rates of 1, 2, and 4 to expand the receptive field. A systematic grid search was conducted to optimise hyperparameters. The model was trained using the Adam optimiser (learning rate = 0.0001) over 50 epochs with a batch size of 128. Figure 7 shows a slight improvement in spike prediction, though the forecasts still tend to hover above actual prices.

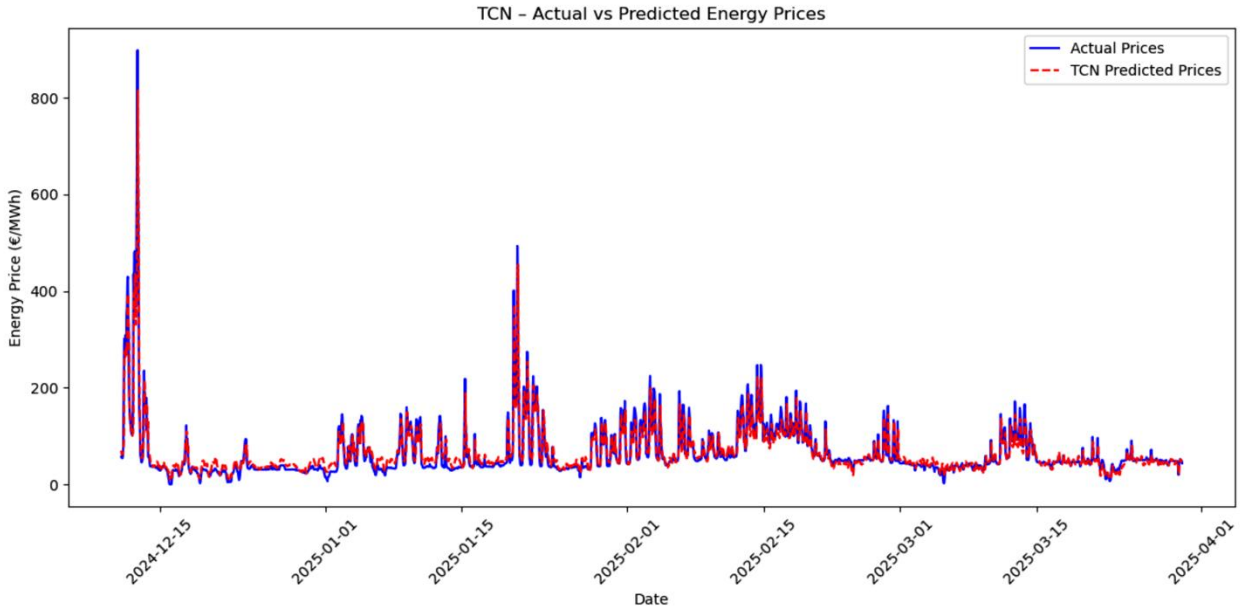


Figure 7: Actual vs predicted price of TCN model

5. Model Performance Evaluation

The performance of all models is summarised in Table 1 and Figure 8 below. All input features and target variables were standardised prior to training, which impacts scale-dependent metrics like MSE, RMSE, and MAE - resulting in smaller, less interpretable absolute values. Therefore, evaluation focuses primarily on the relative comparison of standardised error metrics across models. To better demonstrate overall performance, we also introduce scale-invariant metrics such as R^2, Adjusted R^2, MAPE, and Directional Accuracy (DA).

Model	MAE	RMSE	MSE	R^2	Adjusted R^2	MAPE	DA
XGBoost	10.7951	34.1500	1166.225	0.6559	0.6514	14.83%	67.44%
Vanilla RNN	9.7303	19.8426	393.7283	0.8841	0.8823	18.01%	70.05%
LSTM	10.7866	20.4413	453.6775	0.8661	0.8751	18.66%	68.32%
GRU	9.6948	20.6486	423.8527	0.8749	0.8725	18.21%	69.51%
TCN	12.4202	20.9645	439.5087	0.8703	0.8686	25.44%	67.52%

Table 1: Summary of performance metrics across all models

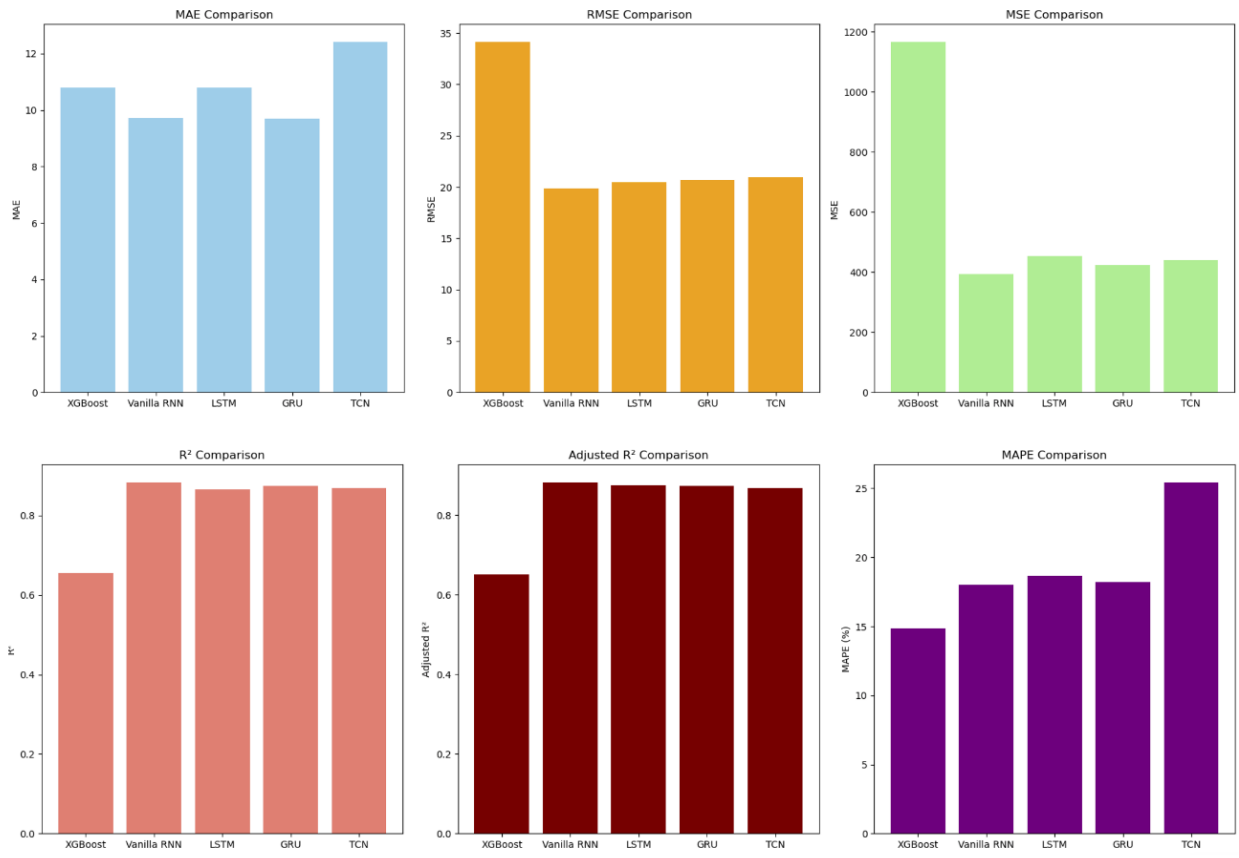


Figure 8: Bar charts comparing performance metrics across all models.

XGBoost serves as the baseline model, achieving moderate performance with an MAE of 10.7951, an RMSE of 34.1500, and an R^2 of 0.6559. Its performance clearly lags behind the deep learning models across all the relevant metrics.

The Vanilla RNN demonstrates the best overall performance among all models. It achieves the lowest RMSE (19.8426) and MSE (393.7283), along with the highest R^2 (0.8841) and Adjusted R^2 (0.8823). The MAE of 9.7303 is also lower than that of XGBoost, and the model achieves a strong DA of 70.05%, indicating excellent trend prediction. The model's low standardised errors and strong explanatory power suggest effective generalisation without signs of overfitting.

The LSTM model underperforms compared to Vanilla RNN and GRU. It has higher error metrics, with an MSE of 453.6775 - the highest among the deep learning models - and a lower R^2 of 0.8661. Although it captures general trends, it struggles with extreme price fluctuations, leading to weaker fit and larger deviations in certain intervals.

The GRU model offers stable and competitive performance. While its RMSE (20.6486) and MSE (423.8527) are slightly higher than those of Vanilla RNN, they are lower than LSTM's. GRU achieves an R^2 of 0.8749 and Adjusted R^2 of 0.8725, demonstrating good explanatory capability. It also records a strong DA of 69.51%, indicating reliable trend prediction across different volatility periods, though still marginally behind the Vanilla RNN.

Among the deep learning models, TCN shows the weakest performance. It records the highest MAE (12.4202) and MAPE (25.44%), with relatively high RMSE (20.9645) and MSE (439.5087). Although it captures major price spikes, the model tends to systematically overestimate prices, as reflected in its lower DA (67.52%) and R^2 (0.8703). The higher error metrics suggest that TCN struggles with model stability and precision in capturing general price movements.

In summary, compared to the baseline XGBoost model, all deep learning models significantly improve forecasting performance. The Vanilla RNN stands out as the best-performing model overall, offering a strong balance between low error, high explanatory power, and accurate trend prediction. GRU model reliable performance, while LSTM struggles more with extreme volatility. TCN demonstrates the highest error rates and the weakest stability across the evaluation period.

6. Residuals Analysis

Residual analysis was performed to evaluate error patterns, shown in Figure 10. Across all four diagnostic plots, residuals are centred around zero and relatively concentrated, indicating generally small errors, though occasional large deviations occur, particularly for the TCN model, which shows a wider spread. Residuals versus predicted values display random distribution around zero, but with increasing dispersion at higher predicted prices, suggesting heteroskedasticity. Residuals over time are mostly stable, although all models exhibit noticeable spikes during winter price surges, with TCN showing the greatest volatility. QQ plots reveal good normality in the central range but significant deviations in the tails, highlighting challenges in modelling extreme price events.

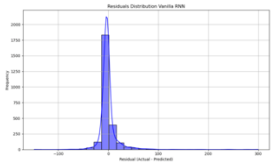
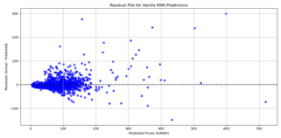
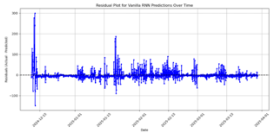
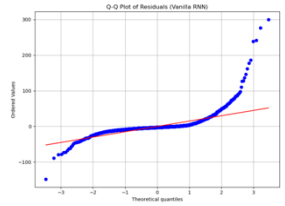
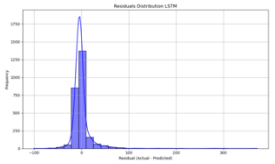
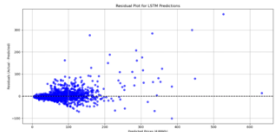
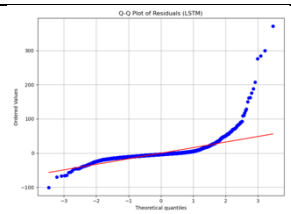
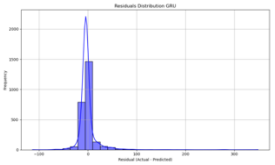
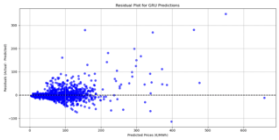
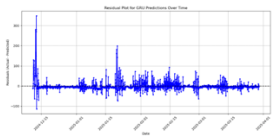
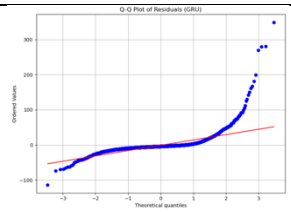
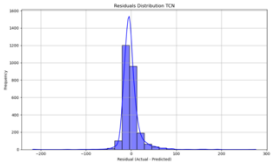
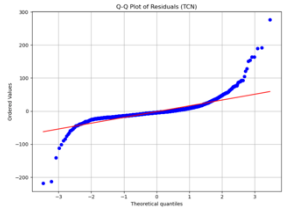
Model	Residual Distribution	Residuals vs Predicted	Residual over Time	QQ Plot
Vanilla RNN				
LSTM				
GRU				
TCN				

Figure 10: Residual plots for each deep learning model

7. Analysis of Results

The evaluation results confirm that deep learning models significantly outperform the traditional XGBoost baseline across all error metrics, consistent with findings from Lago et al. (2021) on the advantages of deep learning for complex, nonlinear time series.

Among deep learning models, the Vanilla RNN achieved the best performance, demonstrating the lowest MSE, RMSE, and MAE, and the highest R^2 and directional accuracy. Although simpler than LSTM and GRU architectures, the Vanilla RNN's lower complexity may have enabled better generalization for short-sequence forecasting.

The GRU model closely followed, balancing performance and model simplicity, aligning with findings from Chung et al. (2014) that GRUs can outperform LSTMs. The LSTM model, despite its ability to model long-term dependencies, showed weaker performance, possibly due to

overfitting risks with a relatively small dataset. TCNs, while effective at capturing some temporal patterns, struggled with stability during extreme volatility.

Residual analysis revealed mild heteroskedasticity across models, with residuals dispersing at higher predicted prices. Vanilla RNN and GRU exhibited the most stable residuals, while TCN showed the greatest variance during price spikes.

Overall, the results reinforce that simpler, well-regularised architectures like Vanilla RNNs and GRUs are highly effective for day-ahead electricity price forecasting in volatile energy markets.

8. Conclusion and Future Work

This study demonstrates that deep learning models, particularly the Vanilla RNN, substantially improve the accuracy and stability of day-ahead electricity price forecasts in Norway's NO2 bidding zone. Vanilla RNN achieved the best performance across error metrics, confirming the value of simple, robust architectures in volatile markets.

While GRU models also performed strongly, LSTM and TCN models exhibited greater instability under extreme price fluctuations. Compared to XGBoost, all deep learning models offered superior handling of complex sequential dependencies.

Future work could incorporate additional market variables (e.g., fuel prices, futures contracts), expand the forecast horizon beyond 24 hours and explore Transformer-based architectures. Developing probabilistic forecasting models, hybrid model approaches, and techniques for improving resilience during extreme price events could further enhance forecasting accuracy and practical relevance.

References

Bai, S., Kolter, J.Z. and Koltun, V., 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.

Chung, J., Gulcehre, C., Cho, K. and Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. *Neural computation*, *9*(8), pp.1735-1780.

Lago, J., Marcjasz, G., De Schutter, B. and Weron, R., 2021. Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. *Applied Energy*, *293*, p.116983.

Weron, R., 2014. Electricity price forecasting: A review of the state-of-the-art with a look into the future. *International journal of forecasting*, *30*(4), pp.1030-1081.

Zhang, G., Patuwo, B.E. and Hu, M.Y., 1998. Forecasting with artificial neural networks:: The state of the art. *International journal of forecasting*, *14*(1), pp.35-62.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H. and Zhang, W., 2021. *Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting*. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 12, pp. 11106-11115).

Contribution table:

Contribution table					
UCL Candidate code	Role in the project	Major contributions	Challenges faced & overcome	Hours contributed	Additional notes
LSHM1	Project manager & Copyeditor	Managing workflow and project structure. Wrote introduction, literature review, methodology, and conclusion.	Gaining and understanding of the relevant literature and how our results fit into it	20	
MHCR1	Data engineer	Collected and preprocessed data from multiple sources including ENTSO-E and Open-Meteo APIs. Merged datasets, handled missing timestamps, and performed exploratory data analysis through time series visualisations.	Handling data inconsistencies and missing timestamps across different datasets Resolved by reindexing and standardising time formats	20	
PDXT9	Model Analyst	Interpreting model architecture, analysing training results, and systematically evaluating model performance	Introduce multi-metric evaluation to handle model results to ensure fair comparison Summarised model architectures by extracting key components	20	
PKHJ8	Research Assistant	Supported the project by researching model architectures and analysing results	Aligning theoretical analysis with practical outputs	20	
LDYY7	Model developer	Handled outliers and built and hyper-tuned the models, ultimately creating different residual plots and finding the metrics for each model	Data leakage issues through StandardScaler Split the data properly and scaled the X data properly to ensure there is no leakage between train and split	20	