A Patch-Based Transformer Approach to Nonlinear Dynamics Natural Gas Price Forecasting

Muhamad Syukron

***Abstract*—Natural gas prices are a critical economic indicator influencing various sectors of the global economy. Accurate forecasting is essential for effective policy formulation and strategic decision making. However, natural gas price movements often exhibit complex non-linear patterns that traditional statistical time series models fail to capture. Furthermore, many deep learning architectures struggle to effectively model these intricate dynamics. To address this challenge, we employ the Patch-Based Transformer (PatchTST) model for natural gas price forecasting. The comparative results reveal that PatchTST achieves substantially higher predictive accuracy than both statistical and other deep learning models. Its Transformer-based architecture, combined with patching and channel independence, enables the model to effectively capture both temporal dependencies and localized variations. The model achieved mean squared error (MSE) and mean absolute percentage error (MAPE) values of 0.1176 and 7.57%, respectively. These findings demonstrate that PatchTST provides robust and precise forecasts, offering valuable insights for decision-making in the energy market.**

Keywords: Natural Gas, Time Series Forecasting, Transformer Model, Energy Market Prediction.

I. INTRODUCTION

THE global energy landscape is undergoing a transition toward low-carbon sources in response to international climate agreements such as the Paris Accord [1]. Renewable energy has expanded rapidly in recent years, yet fossil fuels continued to supply around 80% of global primary energy in 2018 [2]. Long-term projections further indicate that fossil resources will remain a central component of the energy system for several decades [3], [4]. Within this group, natural gas is expected to play a particularly resilient role, with demand either increasing or declining at a slower pace than coal and oil even under stringent climate policy scenarios [2]. These trends highlight the enduring significance of natural gas, making the study of its price dynamics and forecasting an important research priority.

In the United States, natural gas has strategic significance both domestically and globally. It is the dominant fuel for power generation, a critical input for industries, and an important source of household heating. Price volatility therefore has direct consequences for inflation, household expenditure, and industrial competitiveness. At the same time, the United States has emerged as the world's largest exporter of liquefied natural gas, supplying Europe and Asia in the wake of disruptions caused by the Russia–Ukraine conflict [5], [6]. This dual role as a major consumer and exporter makes natural gas price prediction particularly important for U.S. policymakers. Moreover, as natural gas continues to be positioned as a transitional fuel in U.S. decarbonization strategies, accurate forecasting is vital for balancing climate objectives with energy security and economic stability.

The prediction of natural gas prices has been approached using different models. Econometric methods such as autoregressive (AR) [7], time-varying coefficient stochastic volatility (TVCSV), Markov switching (MS) [8] and hybrid models have been applied to capture market dynamics. However, these models often face limitations when dealing with nonlinear structures and sudden regime shifts [9]. Deep learning approaches such as recurrent neural networks (RNNs) [10] and long short-term memory (LSTM) [11] networks have also been explored. Some studies combine them with decomposition techniques and attention mechanisms to improve accuracy [12]. Despite these advances, deep learning models still struggle with vanishing gradients, limited parallelization and the challenge of modelling long-range dependencies. To overcome these issues, this study adopts a Transformer-based [13] framework especially Patch-Based Transformer (PatchTST) [14] to capture both short-term fluctuations and long-term patterns in natural gas prices.

II. PRELIMINARIES

Recent research has explored a wide range of frameworks for natural gas price forecasting, spanning econometric specifications, hybrid machine learning techniques, and deep learning architectures. Gao, Hou, and Nguyen [9] examined flexible econometric models across the U.S., European, and Japanese markets. Their analysis focused on time-varying coefficient stochastic volatility (TVCSV) models, Markov switching (MS) models, and hybrid extensions. The findings indicated that stochastic volatility and regime-switching structures enhanced predictive accuracy. However, these models remain constrained by their reliance on linear dynamics and fixed regimes, which limit their capacity to capture nonlinear interactions and abrupt structural breaks.

To address these non-linearities, Lin et al. [12] proposed a hybrid framework that integrates improved complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN) [15], stacked long short-term memory (STLSTM) networks, temporal convolutional networks (TCN) [16], and convolutional block attention modules (CBAM) [17]. In this framework, ICEEMDAN mitigates noise through adaptive decomposition, while STLSTM enhances memory representation for long-term dependencies. The TCN component captures

M. Syukron is with the Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia muhamad.syukron@its.ac.id

Manuscript received October 12, 2025; accepted December 10, 2025.

multi-scale temporal dynamics, and CBAM focuses attention on the most informative features during extraction. Although this approach achieved strong predictive performance across various sliding windows, its architectural complexity required extensive hyperparameter tuning and substantial computational resources, limiting its practicality for real-time forecasting.

Another study by Zheng et al. [18] studied how geopolitical shocks affect natural gas prices. They proposed a hybrid model called FSGA SVR to forecast Henry Hub prices during the Russia and Ukraine conflict. The model used feature selection to identify the most important predictors. A genetic algorithm was then applied to optimize the SVR [19] parameters. This combination improved both accuracy and stability under conflict-driven volatility. However, dependence on SVR reduced adaptability compared with deep learning methods. The regional focus also limited the model's ability to generalize to other markets with different pricing patterns.

Taken together, the literature underscores the trade-offs across different modeling approaches. Econometric models provide interpretability but lack the flexibility to represent nonlinear dynamics. Traditional machine learning methods such as SVR improve feature selection and maintain stability under certain conditions but underperform relative to deep learning models. Deep learning methods such as TCN and LSTM achieve higher predictive accuracy and robustness but are hindered by high computational costs and limited scalability. Motivated by these challenges, this study investigates a patch-based Transformer architecture for natural gas price forecasting that leverages parallelization for efficiency. In addition, we benchmark its performance against the statistical time-series models such as ARIMA [7] and state-of-the-art alternative deep learning models including AutoFormer [20], Informer [21], and DLinear [22].

III. RESEARCH METHODS

A. Transformer Model

The Transformer is a deep learning architecture that has demonstrated outstanding performance across various domains. It was first introduced for natural language processing tasks, as seen in models such as BERT [23], and was later adapted for image recognition in architectures like the Vision Transformer [24]. The key strength of the Transformer lies in its attention mechanism, which enables the model to learn long-range dependencies and focus on the most relevant parts of the input sequence. This capability allows it to outperform traditional models that rely on fixed-size context windows. Although it is widely used in image and text processing, this study explores its potential for numerical time series forecasting, where understanding temporal dynamics and nonlinear dependencies is crucial.

The Transformer architecture consists of two main components: the encoder and the decoder (See Fig. 1). The encoder is responsible for processing input sequences and generating meaningful feature representations, while the decoder transforms these representations into output sequences. However, for forecasting tasks, sometimes the decoder is not necessary. Therefore, the PatchTST model employs only the encoder component to learn temporal representations directly from the input data.

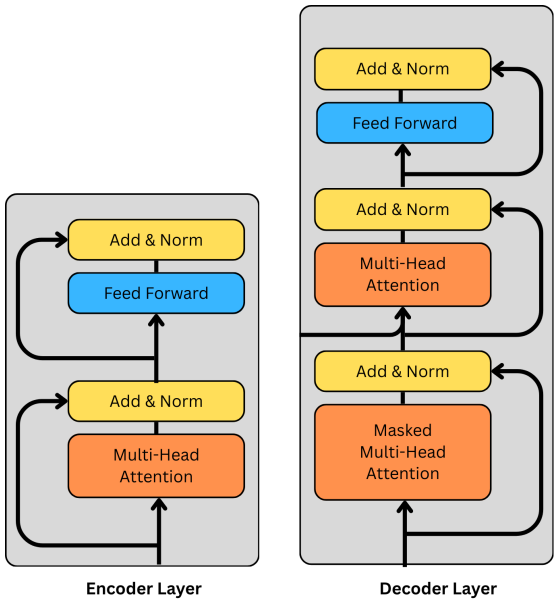

Fig. 1. Transformer Architecture [13]

B. Patch-Based Transformers

PatchTST is a Transformer based model designed for long term time series forecasting [14]. It supports univariate settings through its channel independent architecture. The training and forecasting pipeline for PatchTST is illustrated in Fig. 2. First, the past series X_{past} is divided into a sequence of patches, where each patch contains a fixed number of consecutive values. These patches are created using a predefined patch size and stride, which control how the window moves along the input sequence. After patching, the model treats each patch as a token and processes the sequence of tokens using a standard Transformer encoder.

The channel independent design allows the model to treat each variable separately in the multivariate case. This design avoids direct interactions between variables and helps the model capture temporal dependencies more effectively. In univariate setting, this structure becomes even simpler since the data contains only one channel.

The encoded patch representations are then passed through a linear projection layer to produce the final forecast. This step maps the learned temporal features into the prediction horizon without relying on any auxiliary variables. As a result, PatchTST can forecast future values directly from the historical univariate series.

C. Evaluation Metrics

To evaluate forecasting performance on the test set, we employ two common error metrics, namely the Mean Squared Error (MSE) and the Mean Absolute Percentage Error (MAPE).

MSE is used to measure the average squared deviation between the predicted and actual values, giving greater weight

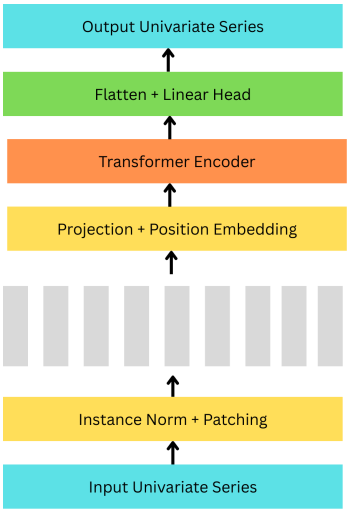

Fig. 2. PatchTST Architecture [14]

to large errors and providing a clear indication of overall prediction accuracy. However, because MSE is scale dependent, it cannot be directly interpreted in relative terms.

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

To complement this, we also use the Mean Absolute Percentage Error (MAPE), which expresses forecasting errors as a percentage and enables scale-independent comparison across different models and datasets.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{\max(|y_i|, \varepsilon)}\right| \quad (2)$$

Here, ε is a small positive constant introduced to prevent numerical instability when y_i is close to zero. In this study, ε is set to 10^{-6}.

IV. RESULT AND DISCUSSION

A. Dataset

In this study, the dataset used is historical data of close price for natural gas commodity futures. This dataset consists of daily data over a five-year period, from June 24, 2020, to June 24, 2025, obtained from the Investing.com platform [25]. The data only includes exchange working days, excluding weekends and market holidays. Table 1 shows the statistical description of the data set.

TABLE I
DESCRIPTIVE STATISTICS ON CLOSE PRICE OF NATURAL GAS FUTURES IN 2020-2025

Variable	N	Mean	St Dev	Min	Max
Close Price	1,314	3.78	1.76	1.54	9.65

Based on Table I, the dataset comprises 1,314 observations of natural gas futures from 2020 to 2025, with an average closing price of 3.78 USD per Million British Thermal Units (MMBtu) and a standard deviation of 1.76. The highest observed price was 9.65 USD on August 22, 2022, while the lowest was 1.54 USD on June 26, 2020.

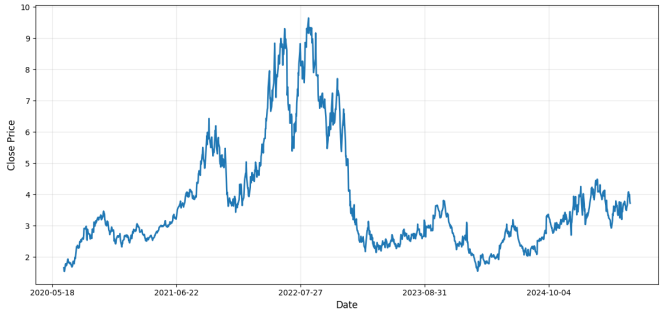

Fig. 3. Time Series Plot for Close Price of Natural Gas Futures in 2020-2025

Fig. 3 illustrates the highly dynamic and volatile movement of natural gas prices over the past five years. There was a significant increase followed by a sharp decline, then a period of consolidation, and finally a gradual rise again.

B. Implementation Details

The dataset was divided into training, validation, and testing sets with proportions of 70%, 20%, and 10%, resulting in 919 samples for training, 262 for validation, and 133 for testing. For ARIMA and SARIMA, a fixed origin evaluation strategy was adopted to estimate the parameters p,d,q and P,D,Q. This approach avoids repeatedly retraining the models under a rolling origin scheme, which would significantly increase computational cost. In addition, SARIMA requires a sufficiently long historical series to reliably capture seasonal patterns extending beyond one month.

The same fixed origin strategy was also applied to Holt Winters, TBATS [26] and Prophet [27]. These models rely on long historical patterns to estimate level, trend, and one or multiple seasonal components. Using a fixed origin ensures that seasonal structures, such as weekly and semiannual cycles, are learned from the full available training and validation data, leading to more stable and consistent parameter estimation.

For the deep learning models, which are AutoFormer, Informer, and DLinear, rolling origin (sliding window) evaluation was employed with a window size of 60 working days. PatchTST also used a 60 working day rolling window. However, its input sequence was further segmented into patches of size 20 with a stride of 20, resulting in an input shape of 3×20 at each forecasting step.

All models performed multi-step forecasting with a prediction horizon of 20 working days. For statistical models, forecasts were generated iteratively in 20-day windows using a fixed-origin training scheme. For the deep learning model, iterative predictions were made for 20-day horizons using the previous 60 days as input. In both cases, this rolling procedure was repeated until a total of 140 forecasts were produced. Only the first 133 predictions were used to compute MSE and MAPE, as ground truth values beyond this range are unavailable. When multiple predictions were generated for the same day due to overlapping windows, the forecasts were averaged to obtain a single value per day for evaluation. No

data leakage occurred, as all models were trained exclusively on the training dataset.

C. Model Configuration

For AutoFormer and Informer, the hidden dimension is set to 64 and the feedforward dimension to 128 in both the encoder and decoder blocks. Each model employs four attention heads, two encoder layers, and one decoder layer. The GELU activation function is adopted.

DLinear does not follow an encoder–decoder architecture. Instead, it decomposes the input time series into trend and seasonal components using moving average filtering. Independent linear layers are applied to each component, and their outputs are summed to generate the final forecast.

For PatchTST, the hidden dimension is set to 128 and the feedforward dimension to 256. The model employs four attention heads and three encoder layers. Patch tokenization is applied with a patch length of 20 and a stride of 20.

To ensure a fair comparison, all models are trained using the same optimization strategy. The Adam optimizer is used with a batch size of 32 for 100 epochs. The learning rate is fixed at 0.0001, and Mean Squared Error is employed as the loss function.

For the ARIMA model, the orders (p,d,q) were selected based on the results of the Augmented Dickey Fuller test and the analysis of the ACF and PACF, as discussed in the Results and Discussion section. For SARIMA, Holt Winters, TBATS, and Prophet, the seasonal parameters were determined through an inspection of the historical time series plot.

For the Holt Winters model, the smoothing parameters were fixed to $\alpha = 0.2$, $\beta = 0.1$, and $\gamma = 0.3$ to ensure stable learning of the level, trend, and seasonal components. The choice of $\alpha = 0.2$ implies that 20% of new information is incorporated at each update, while 80% of the previous level is retained. The small value of $\beta = 0.1$ reflects the weak and inconsistent trend observed in the data. The seasonal smoothing parameter $\gamma = 0.3$ was selected to allow moderate adaptation to the half yearly seasonal pattern, which is present but not strongly pronounced.

All experiments, including both statistical and deep learning models, were conducted on Google Colab using Python 3.12.12 with approximately 12 GB of RAM, an AMD CPU, and an NVIDIA T4 GPU.

D. Model Evaluation and Comparison

The initial comparison employed the statistical model ARIMA. To assess the stationarity of the time series, we performed the Augmented Dickey-Fuller (ADF) test [28]. The test returned a statistic of -1.58 with a corresponding p-value of 0.48, indicating that the series is non-stationary at the 5% significance level. Consequently, differencing was applied to achieve stationarity. After differencing, the ADF test yielded a statistic of -13.43 and a p-value of 3.87×10^{-25}. This result is well below the 5% significance level, leading to the rejection of the null hypothesis and confirming that the differenced series is stationary.

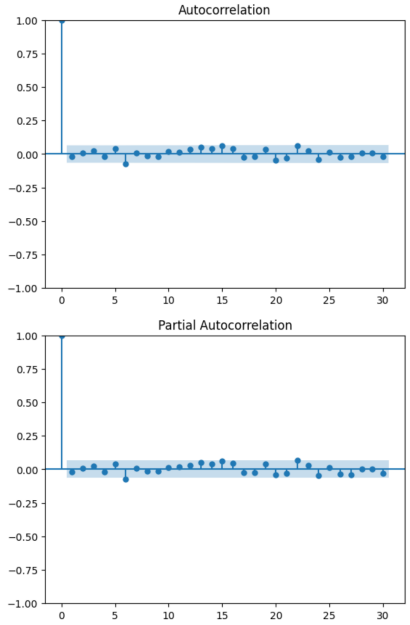

Fig. 4. ACF and PACF Plot

TABLE II
TOP 5 ARIMA MODELS BASED ON LOWEST AIC VALUE

Rank	ARIMA (p, d, q)	AIC	BIC
1	(2, 1, 2)	-250.82	-226.74
2	(3, 1, 2)	-249.65	-220.75
3	(2, 1, 3)	-249.64	-220.74
4	(3, 1, 3)	-249.48	-215.77
5	(4, 1, 3)	-249.45	-210.92

Since first-order differencing was required, the differencing parameter d in the ARIMA model was set to 1. To determine the autoregressive (p) and moving average (q) orders, we examined the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots, shown in Fig. 4. The plots indicate that lags 0 and 1 are significant. Based on this observation, initial candidate values for p and q were 0 and 1. To ensure a broader search, we considered p and q values in the range 0 to 5. The optimal model was selected based on the Akaike Information Criterion (AIC) [29] and Bayesian Information Criterion (BIC) [30], prioritizing the lowest AIC value.

After performing grid-search, the five models with the smallest AIC values are presented in Table II. The best-fitting model was found to have parameters $p = 2$, $d = 1$, and $q = 2$. The estimated coefficients were AR lag 1 $= 0.2398$, AR lag 2 $= -0.8789$, MA lag 1 $= -0.3192$, MA lag 2 $= 0.9015$, with residual standard deviation $\sigma = 0.0440$. We performed the Ljung-Box test [31] to assess the independence of residuals and obtained a test statistic of 8.6437 with a p-value of 0.56. This indicates that the residuals are uncorrelated, confirming that the model adequately captures the temporal dependence in the series.

After obtaining the best ARIMA model, the forecast was generated using the optimal parameters, ARIMA(2, 1, 2), as shown in Fig. 5. The predicted values have been transformed

back to the original scale and are no longer in the differenced form. However, the figure shows that the red line representing the model's predictions is nearly flat. This indicates that the ARIMA model fails to capture the dynamic trend of the series and instead produces approximately constant forecasts.

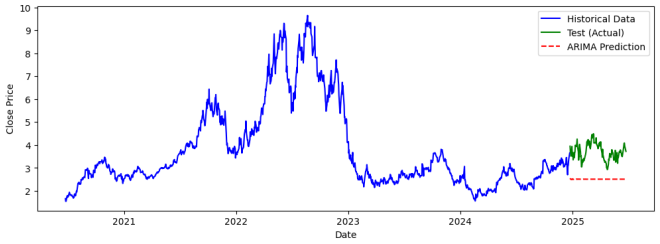

Fig. 5. Prediction Plot using Best ARIMA Model

Since ARIMA cannot capture volatility, we applied a seasonal model, namely SARIMA. To determine the seasonal period, we plotted the series as shown in Fig. 6, marking the start of each month, quarter, and six-month period. The plot indicates that a six-month seasonality is most appropriate, as natural gas prices generally increase and then decrease within each six-month cycle.

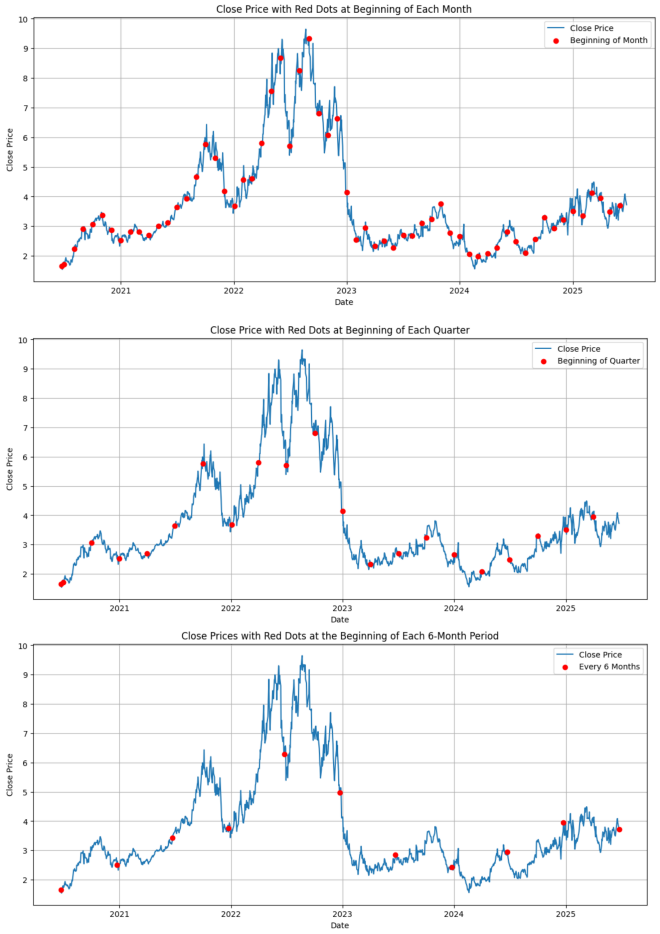

Fig. 6. Seasonality Detection Using Historical Plots

Given the identified six-month seasonal pattern, we modeled SARIMA with parameters $(p,d,q)(P,D,Q,120)$, where 120 working days approximate six months. The non-seasonal differencing d was set to 1 based on ARIMA results, indicating non-stationarity in the original data. Seasonal differencing D was also set to 1, reflecting the observed seasonal pattern. Values of p and q, as well as P and Q, were selected using a grid search over $0,1,2$. After testing 70 parameter combinations, the best model was SARIMA$(1,1,0)(0,1,1,120)$, with the lowest AIC of -44.82.

Training the selected model on the training data yielded test set MSE of 0.4555 and MAPE of 15.57%. The forecasting results are shown in Fig. 7, where predicted values closely follow the actual test data, although some underprediction is observed.

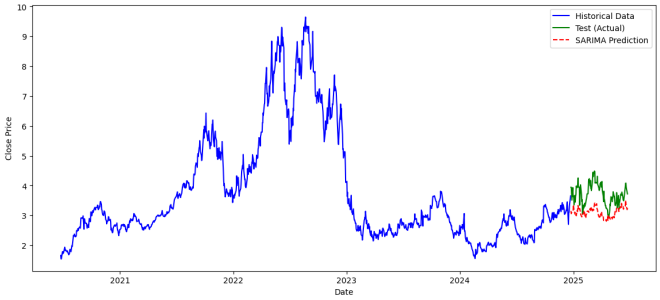

Fig. 7. Prediction Plot using Best SARIMA Model

Next, we evaluated another statistical forecasting model, Holt-Winters [32], which combines Simple Exponential Smoothing (SES) [33] and the Holt method [34], allowing it to capture both trend and seasonality. We set the seasonal period to 120 days, consistent with SARIMA, and used additive seasonality. The resulting forecasts are shown in Fig. 8. Similar to SARIMA, Holt–Winters captures the start and end of the test period well but tends to slightly underpredict in the middle. Nevertheless, it achieves low test set errors, with MSE of 0.1245 and MAPE of 8.98%.

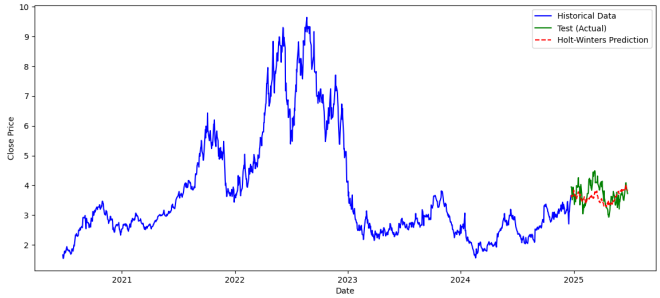

Fig. 8. Prediction Plot using Holt-Winters Model

We then applied two modern models, TBATS [26] and Prophet [27]. For both models, we tested several seasonal configurations, including weekly, monthly, quarterly, and six-monthly cycles. The lowest errors were obtained using a combination of weekly and six-monthly seasonality. Specifically, TBATS achieved MSE of 0.3689 and MAPE of 15.44%, while Prophet achieved MSE of 0.3128 and MAPE of 11.77%.

After training the statistical models, we evaluated several state-of-the-art deep learning architectures. Fig. 9 presents the

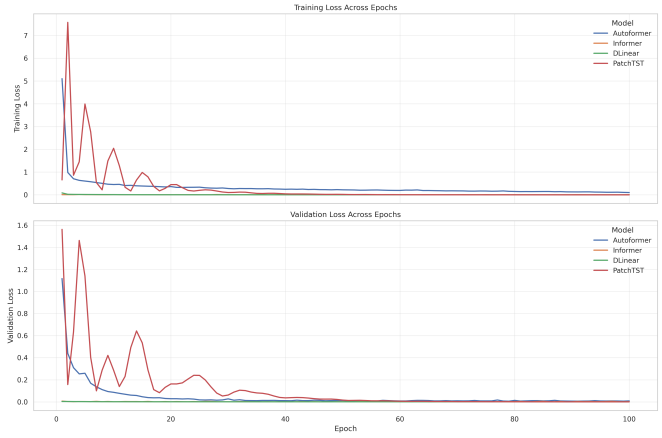

Fig. 9. Loss Training and Validation Dataset

training and validation loss curves of AutoFormer, Informer, DLinear, and PatchTST over 100 epochs. The results show that all models experience a sharp reduction in loss during the early epochs, followed by convergence to stable values. PatchTST exhibits higher volatility in the initial training phase compared to the other models, yet it consistently stabilizes as training progresses. Despite these early fluctuations, PatchTST achieves one of the lowest final losses among all evaluated models, highlighting its strong capability in capturing complex temporal dependencies in natural gas price forecasting.

After training all models, the MSE and MAPE results are summarized in Table III. All of the values are calculated using the original units. The results indicate that some deep learning models, such as AutoFormer and Informer, perform worse than the statistical baseline, particularly the Holt-Winters method. In contrast, DLinear and PatchTST demonstrate superior performance by achieving substantially lower MSE and MAPE values. Notably, PatchTST outperforms DLinear, the second-best model, by a clear margin. Specifically, PatchTST attains an MSE of 0.1176 compared to 0.1693 for DLinear, and a MAPE of 7.57% compared to 8.67%.

TABLE III
COMPARISON MODEL BY EVALUATION METRICS SCORE

Model	MSE	MAPE
ARIMA	1.5547	31.68%
SARIMA	0.4555	15.57%
Holt-Winters	0.1245	8.98%
TBATS	0.3689	15.44%
Prophet	0.3128	11.77%
AutoFormer	0.6073	16.88%
Informer	0.3848	15.28%
DLinear	0.1693	8.67%
PatchTST	0.1176	7.57%

Fig. 10 shows the time series plot of the forecast result for the test dataset using PatchTST as a best model. The predicted values show fluctuations and correctly follow the trend patterns of the latest data. This is indicating that PatchTST is capable of adapting to current dynamics and providing realistic short-term projections. These results further support the quantitative findings, confirming that PatchTST not only achieves low error

Fig. 10. Prediction Plot using PatchTST

metrics but also generates forecasts that are consistent with the observed temporal trends.

V. CONCLUSION

This study evaluates both statistical forecasting and advanced deep learning models for predicting natural gas prices. The results demonstrate that PatchTST achieves the strongest overall performance, with an MSE of 0.1176 and an MAPE of 7.57%. Its Transformer-based architecture, combined with patch-level tokenization and channel-independent processing, enables the model to effectively capture both short-term fluctuations and long-term temporal dependencies while maintaining computational efficiency.

The findings also show that conventional statistical models such as Holt–Winters are able to capture the overall trend of natural gas prices but remain limited in modeling high-frequency volatility and abrupt price movements. In contrast, deep learning models are not guaranteed to outperform statistical approaches in all cases, as demonstrated by the weaker performance of AutoFormer and Informer compared to Holt–Winters. This highlights the importance of model selection and task suitability rather than assuming universal superiority of deep learning methods.

Overall, PatchTST provides a practical and effective approach for forecasting in dynamic energy markets. Future work may investigate its application to multivariate energy datasets, incorporate external influencing factors such as geopolitical events and macroeconomic indicators.

REFERENCES

[1] United Nations, “Paris agreement,” 2015, available online.
[2] International Energy Agency, *World Energy Outlook 2019*. OECD, 2019.
[3] S. Paltsev, “Projecting energy and climate for the 21st century,” *Economics of Energy & Environmental Policy*, vol. 9, no. 1, Jan. 2020.
[4] BP, “Bp energy outlook 2019,” 2019.
[5] M. Boeck and T. O. Zörner, “Natural gas prices, inflation expectations, and the pass-through to euro area inflation,” *Energy Economics*, vol. 141, p. 108061, Jan. 2025.
[6] W. Cui, D. Wu, Q. Huang, and S. Yang, “The dynamic evolution mechanism of liquefied natural gas trade dependency networks: International implications of the russia-ukraine conflict,” *Sustainable Futures*, vol. 9, p. 100730, Jun. 2025.
[7] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco: Holden-Day, 1970.
[8] J. D. Hamilton, “A new approach to the economic analysis of nonstationary time series and the business cycle,” *Econometrica*, pp. 357–384.
[9] S. Gao, C. Hou, and B. H. Nguyen, “Forecasting natural gas prices using highly flexible time-varying parameter models,” *Economic Modelling*, vol. 105, p. 105652, Dec. 2021.

[10] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211.
[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780.
[12] Y. Lin *et al.*, "Forecasting natural gas prices using a novel hybrid model: Comparative study of different sliding windows," *Energy*, vol. 329, p. 136607, Aug. 2025.
[13] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30.
[14] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," Mar. 2023.
[15] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: a noise-assisted data analysis method," *Advances in Adaptive Data Analysis*, vol. 1, no. 1, pp. 1–41.
[16] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," Apr. 2018.
[17] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," pp. 3–19.
[18] Y. Zheng, J. Luo, J. Chen, Z. Chen, and P. Shang, "Natural gas spot price prediction research under the background of russia-ukraine conflict — based on fs-ga-svr hybrid model," *Journal of Environmental Management*, vol. 344, p. 118446, Oct. 2023.
[19] V. N. Vapnik, *Statistical Learning Theory*. Wiley, 1998.
[20] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 22 419–22 430.
[21] H. Zhou *et al.*, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artificial Intelligence*, vol. 35, pp. 11 106–11 115.
[22] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" Aug. 2022.
[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: https://arxiv.org/abs/1810.04805
[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. [Online]. Available: https://arxiv.org/abs/2010.11929
[25] Investing.com, "Natural gas futures price today," https://www.investing.com/commodities/natural-gas, [Accessed: 09-Oct-2025].
[26] A. M. De Livera, R. J. Hyndman, and R. D. Snyder, "Forecasting time series with complex seasonal patterns using exponential smoothing," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1513–1527, 2011.
[27] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.
[28] D. A. Dickey and W. A. Fuller, "Distribution of the estimators for autoregressive time series with a unit root," *Journal of the American statistical association*, vol. 74, no. 366, pp. 427–431, 1979.
[29] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Proceedings of the Second International Symposium on Information Theory*. Budapest: Akademiai Kiado, 1973, pp. 267–281.
[30] G. E. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.
[31] G. M. Ljung and G. E. P. Box, "On a measure of lack of fit in time series models," *Biometrika*, vol. 65, no. 2, pp. 297–303, 1978.
[32] P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Management Science*, vol. 6, no. 3, pp. 324–342, 1960.
[33] R. G. Brown, "Exponential smoothing for predicting demand," Arthur D. Little Inc., Cambridge, MA, Tech. Rep., 1956.
[34] C. C. Holt, "Forecasting trends and seasonals by exponentially weighted moving averages," Carnegie Institute of Technology, Pittsburgh, PA, Tech. Rep., 1957.