Weather-Independent Forecasting for State-Wide Energy Markets Using Hybrid GPI-DSSM Model

Rishi Malyala*, Kuthsav Thattai, Anam Malik, Jayashri Ravishankar
School of Electrical Engineering and Telecommunications
University of New South Wales, Sydney
Sydney, Australia
ORCID*: https://orcid.org/0009-0002-3502-8487

***Abstract*—This paper introduces a novel hybrid forecasting model for state-wide energy markets, combining a Graph-based Patch Informer (GPI) and Deep State Sequential State Memory (DSSM) architecture to predict electricity prices. Integrating spatial-temporally variable weather data can introduce noise and inconsistencies, as different regions within a state experience varying weather conditions. The GPI-DSSM model avoids these issues by relying solely on demand and price data, focusing on intrinsic and hidden relationships between aggregate demand and regional reference price (RRP). By using aggregate demand as a hidden state within the Gated Recurrent Unit (GRU) based DSSM, the model captures dependencies driving price fluctuations, while the GPI transformer handles long-range temporal patterns. This hybrid architecture provides robust, accurate forecasts, demonstrating that the geo-spatial-temporal limitations of weather data on aggregate demand forecasting can be effectively managed by focusing on core market variables like demand and price.**

Keywords—Regional Reference Price, State-wide energy systems, Electricity Markets, Graph Patch Informer, Deep Sequential State Memory

I. INTRODUCTION

In electricity markets, market participants rely on the state-wide Regional Reference Price (RRP) to make critical decisions regarding energy dispatch, bidding, and operations. RRP, set by market operators, reflects the overall supply-demand ($/MWh) dynamics at the state level, making accurate forecasting of RRP and aggregate demand essential for effective market participation. Notably, the RRP is influenced solely by the aggregate demand requirements across the entire system and does not factor in localized variables like transmission constraints or distributed generation, making it a more stable indicator of state-wide demand. Traditionally, forecasting models have incorporated weather data to predict demand fluctuations. However, for state-wide markets, the spatial variability of weather conditions across many regions introduces noise, complicating the integration of this data into forecasts.

For instance, models like Spatio-Temporal Convolutional Graph Neural Networks (ST-GNN) effectively capture local dependencies but struggle to scale for broader economic price forecasting needs of the market [1]. Furthermore, these models do not capture the hidden relationships between energy demand and RRP. Similarly, Self-Supervised Transformer Variant [2] for Renewable Energy Forecasting focuses on renewable output, heavily relying on weather data, which limits its use for RRP forecasting where aggregate demand is a more dominant driver, and has not been tested for short-term demand interdependency.

Similarly, the transformer-based approach like the Graph Transformer have been proposed for time-series forecasting (TSF) and long-term energy forecasts [3]. Recent work on hybrid CRN-GNN models also combines convolutional recurrent networks (CRN) and graph neural networks (GNN) to offer probabilistic load forecasting [4] with a similar approach. However, they also fall short on addressing demand interdependency.

In contrast, recent studies have explored energy demand forecasting models that do not rely on weather inputs, focusing on internal economic and demand-related variables, which are keenly prompted and followed by market participants. Support vector regression model that forecasts energy demand based on economic factors, effectively bypasses weather-related uncertainties [5]. Similarly, forecasting model for multi-energy systems that optimizes market participation without incorporating weather data has been introduced [6], which provide fairly accurate results for long-range forecasting. While these approaches demonstrate that reliable forecasts can be achieved without external and independent inputs, they also fall short in addressing the interdependencies in short-term everyday forecasts, which is crucial for market participants.

Given these research gaps, a more targeted approach is needed—one that focuses on the direct relationship and interdependencies between aggregate demand and RRP, without the need for spatially-dependent noisy weather variables inputs. This paper introduces a novel hybrid model combining a GPI and DSSM. The GPI-DSSM model uses only historical RRP and demand data to forecast future prices, capturing both long-term and short-term dependencies as illustrated in Fig. 1. The GPI component handles long-term dependencies, while the DSSM effectively manages short-term fluctuations, ensuring the model adapts to real-time price and demand movements. This capability directly addresses the limitations of earlier models and strengthens the practical applicability of RRP and demand forecasting in large-scale energy market participants for everyday scheduled dispatch decisions. This paper is organized as follows: Section II discusses data collection and preprocessing; Section III covers the model implementation; Sections IV explains the forecasting and hyperparameter tuning of the GPI-DSSM model, with the discussion and conclusions covered in Section V.

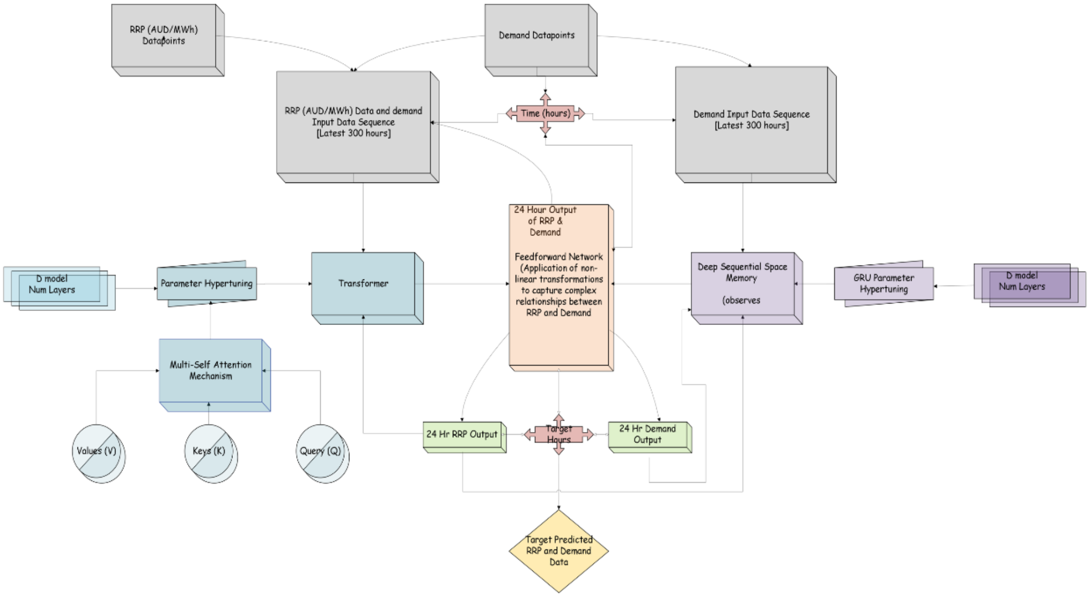

Fig. 1. Flowchart of the Hybrid GPI-DSSM Forecast Model

II. Data Collection and Pre-Processing

The dataset used for this research comprises state-wide electricity market data, spanning Regional Reference Price and Total Demand from 2009 to September, 2024. The data, sourced from Australian Energy Market Operator (AEMO) [7], represents hourly market dynamics and plays a crucial role in modelling energy dispatch decisions with large-scale market participants. Unlike other models explored in previous studies [1], [2], this study focuses purely on RRP and demand relationships.

A. Data representation and scaling

The data set used in this study is represented as $\mathbb{X} \in \boldsymbol{R}^{\boldsymbol{T} \times \mathbf{2}}$, as shown in (1), where $\boldsymbol{T}$ denotes the number of time points, and 2 feature pairs consisting of **RRP** r_t and total demand d_t at each time step $\boldsymbol{t}$.

$$\mathbb{X} = \begin{bmatrix} r_1 & d_1 \\ \vdots & \vdots \\ r_t & d_t \end{bmatrix} \tag{1}$$

The wide variability of r_t and d_t, particularly the extreme spikes in RRP during market fluctuations, necessitates Min-Max scaling to standardize the dataset. Each feature is rescaled into the interval $[-1, 1]$ using the transformation shown in (2).

$$x'_t = \frac{2(x_t - x_{min})}{(x_{max} - x_{min})} - 1 \tag{2}$$

In (2) x_{min} and x_{max} represent the minimum and maximum values each feature. This scaling ensures that both the RRP and Total Demand are normalized to the same range preventing any bias due to different scales. This approach is more robust and computationally efficient compared to models such as ST-GNN [1] and hybrid CRN-GNN models [4], which handle highly volatile external factors like weather data, and suffer from noisy inputs.

B. Resampling and Time Aggregation

The dataset is resampled to hourly intervals to ensure consistency in the time series representation. The resampled dataset, denoted by $\mathbb{X}_h$ is computed by averaging data points within each hour.

$$\mathbb{X}_h = \frac{1}{N_h} \sum_{i=1}^{N_h} \mathbb{X}_i \tag{3}$$

In (3), $\boldsymbol{N_h}$ represents the number of data points within each hour. This resampling method is grounded in its ability to reduce high-frequency noise, which can destabilize model training and lead to overfitting, particularly in large-scale systems where frequent fluctuations are common. By smoothing out these short-term anomalies, resampling enables the model to focus on more stable demand and price trends, enhancing both robustness and accuracy in large-scale market forecasting. Compared to models like Transformer-based Graph Models [3], which often integrate high-frequency, multi-dimensional inputs such as weather data, this approach focuses on simplifying the input while retaining essential market dynamics like price and demand. This approach ensures that the model remains scalable and efficient while capturing core market behaviors without the noise introduced by volatile external factors.

C. Sequence Creation for Time-Series Modelling

To capture the temporal dependencies, the dataset is divided into sequences of 300 hours (inputs) to forecast the next 24 hours (outputs). The input and output sequences S_i and Y_i respectively, for each sample i is formulated as shown in (4)

$$S_i = \begin{bmatrix} x_i \\ x_{i+1} \\ \cdots \\ x_{i+299} \end{bmatrix}, \quad Y_i = \begin{bmatrix} x_{i+300} \\ x_{i+301} \\ \cdots \\ x_{i+323} \end{bmatrix} \tag{4}$$

As shown in (4), the total number of sequences N and total length of the dataset T is determined by (5):

$$N = T - 300 - 24 + 1 \tag{5}$$

Each sample i provides a matrix of input sequences $S_i \in \boldsymbol{R}^{300 \times 2}$ and target sequences $Y_i \in \boldsymbol{R}^{24 \times 2}$. This approach ensures the model captures both long-term (300-hour sequences) and short-term (24-hour predictions) dependencies, providing a structured approach to time-series forecasting. Unlike ST-GNN, which relies on complex spatial-temporal data [1], this model operates with clean and concise RRP and demand sequences, reducing model complexity and improving scalability for state-wide markets.

D. Mini-Batch Processing and DataLoader

For efficient model training, the input sequence S and target sequences Y are grouped into mini-batches, which allows for parallel processing and faster convergence during training. Each mini-batch B_j is structured as shown in (6):

$$B_j = [S_j \quad Y_j] \in \boldsymbol{R}^{8 \times 300 \times 2} \tag{6}$$

By using mini-batch processing, the model can efficiently learn from the large dataset while maintaining scalability, a key factor for state-wide aggregate RRP and demand forecasting. Other models, such as CRN-CNN [4], introduce additional overhead by incorporating external factors, making them less suited for real-time or large-scale forecasting tasks.

III. Model Implementation

Given the input sequence $S_i \in \boldsymbol{R}^{300 \times 2}$, where each element represents RRP and total demand, the GPI creates context-aware embeddings. These embeddings are then processed by the DSSM, which uses GRU to evolve the hidden states over time, adjusting the forecast based on immediate demand-price relationships.

A. Graph Patch Informer (GPI) Architecture

The GPI is based on a Transformer architecture that uses a multi-head self-attention mechanism to model temporal relationships across the input sequence. Multi-head attention and feedforward layer normalization is examined for integration into the GPI-DSSM hybrid model.

1) Multi-Head Attention and Output

The self-attention mechanism computes attention scores between all pairs of time steps in the input sequence. As shown in (7), for each time step, a query $\mathbb{Q}$, key $\mathbb{K}$, and value $\mathbb{V}$ are computed as linear transformations of sequence S_i:

$$\mathbb{Q} = W_q S_i,\ \mathbb{K} = W_k S_i,\ \mathbb{V} = W_u S_i \tag{7}$$

Where W_q, W_k, W_v $\in \boldsymbol{R}^{\mathbf{d_{model}} \times 2}$ are learnable weight mathematics, and $\mathbf{d_{model}}$ is the hidden dimension. The result of attention weights that determines the contribution in each time step to every other time step with the transformer's multi-head attention mechanism, allows the model to capture relationships across different temporal scales. This implementation aligns with similar methodologies outlined in published literature [8], where the multi-head attention mechanism effectively captures non-linear patterns in time-series data, allowing the model to attend to subtle market fluctuations without being biased by external factors. This flexibility is valuable for RRP forecasting, ensuring that price shifts are driven by demand and are accurately reflected in the model's output.

2) Feed-Forward and Layer Normalization

After the attention mechanism, the output is passed through a feed-forward network (FFN), which introduces non-linearity. The FFN consists of two linear transformations with a rectified linear unit (ReLU) activation that are represented by (8).

$$\mathrm{FFN}(x) = \max(0,\ \ W_1 x + b_1) W_2 + b_2 \tag{8}$$

Where W_1 and W_2 $\in \boldsymbol{R}^{\mathbf{d_{model}} \times \mathbf{d_{ff}}}$ are learnable weight matrices and $\mathrm{d_{ff}}$ is the dimension of feed-forward layer. The result is then passed through Layer Normalization to ensure stability during training as given in (9).

$$\mathbb{H}_i = \mathrm{LayerNorm}(x + \mathrm{FFN}(x)) \tag{9}$$

This output $\mathbb{H}_i \in \boldsymbol{R}^{300 \times \mathbf{d_{model}}}$ represents the enriched temporal embeddings for the input sequence produced by the GPI. The FFN configuration is a common strategy for stabilizing non-linear temporal relationships in time-series data and identifies long-range dependencies.

B. Deep Sequential State Memory (DSSM) Architecture

The DSSM component models short-term dependencies and captures the influence of aggregate demand on RRP using a Gated Recurrent Unit (GRU). The GRU updates its hidden state over time, learning to model the impact of demand fluctuations on price changes.

For each time step t, the GRU updates its hidden state h_t based on the input x_t (RRP and demand) and previous hidden state h_{t-1}. The update gate z_i determines how much of the previous hidden state is retained by using sigmoid function to quash the output between 0 and 1, which is best represented by (10).

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{10}$$

The reset gate r_t controls how much of the past information is forgotten, and the candidate hidden gate $\tilde{h}_t$ is computed based on both the reset hidden state and the current input equation, as shown in (11, 12).

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{11}$$

$$\tilde{h}_t = \tanh(W_h x_t + r_t(U_h h_{t-1}) + b_h) \tag{12}$$

The final hidden state is a combination of the past hidden state and the candidate hidden state weight by the update gate. The relationship is best represented as shown in (13).

$$h_t = (1 - z_t) h_{t-1} + z_t \tilde{h}_t \tag{13}$$

The GRU-based architecture is effective in learning both demand and price relationships, especially when dealing with multi-variate time-series data. Similar approaches, such as those found in other advanced optimization driven GRU and Long-Short Term Memory (LSTM) frameworks [9], have demonstrated success in enhancing model accuracy through advanced parameter tuning. In a similar approach, this hybrid GRU optimizes parameters to capture critical demand-price relationships without reliance on additional external features.

C. GPI-DSSM Hybrid Output and Feed-Forward Prediction

The final step of the GPI-DSSM model combines the outputs from the GPI and DSSM components. The last 24 time steps from the GPI output $\mathbb{H}_i$ are concatenated with the GRPU output h_t, creating a combined output C_i as shown in (14).

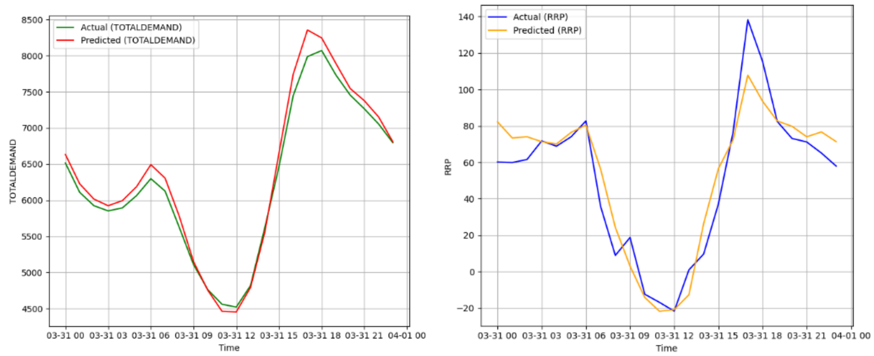

Fig. 2. Comparison of Original and Feed-Forward Output for RRP and Total Demand

$$\mathrm{C}_i = \mathrm{Concat}(\mathbb{H}_{i,276:300}, \quad h_t) \tag{14}$$

This combined output is passed through a final fully connected layer FFN to generate the final 24-hour predictions for both RRP and Total Demand, best represented in (15).

$$\tilde{y}_i = W_f C_i + b_f \tag{15}$$

Where $W_f \in \boldsymbol{R}^{(d_{model}+d_{gru})\times 2}$ and $b_f \in \boldsymbol{R^2}$ are learnable parameters. The output $\tilde{y}_i \in \boldsymbol{R}^{24\times 2}$ represents the 24-hour forecast of RRP and Total Demand. This hybrid model is adaptable to handle different time scales by adjusting the sequence length parameters, allowing it to perform both short-term (hourly) and long-term (daily) forecasts as needed for various market participants. This flexibility makes the GPI-DSSM model particularly suitable for dynamic and large-scale market participants.

D. *Preliminary Model Analysis*

The FFN used in the final prediction layer allows the model to map the combined GPI-DSSM embeddings to the target outputs. Fig. 2. shows the comparison between the original values of RRP and Total Demand versus the predicted outputs generated by the feed-forward process for a one-day period. This close alignment between the predicted and actual values emphasizes the model's proficiency in capturing market behavior, particularly in state-wide aggregated data scenarios, where spatial-temporal variability is not as easily incorporated, as noted in previous works [5]. The performance of the model further aligns with studies [2], [10], where prediction accuracy was maintained even with simplified inputs, demonstrating how the advanced architecture of GPI-DSSM can deliver high accuracy in energy forecasting.

E. *Loss Function and Optimization*

The model is optimized using Smooth L1 Loss (Huber Loss), which is robust to outliers in the RRP data as illustrated in (16):

$$L(y,\hat{y}) = \begin{cases} 0.5(y-\hat{y})^2 & if\ |y-\hat{y}| < \delta \\ \delta(|y-\hat{y}| - 0.5\delta) & otherwise \end{cases} \tag{16}$$

Where y is the true value and $\hat{y}$ is the model's prediction, and δ is the threshold for switching between Mean Squared Error (MSE) for small error and Mean Absolute Error (MAE) for larger errors. Smooth L1 Loss is chosen for its balanced approach for minimizing both minor and major errors, which is particularly beneficial in context of RRP where both small fluctuations and larger outliers impact forecasting accuracy. This makes it more robust for large-scale market data, providing improved tolerance compared to loss functions such as MSE, which over-penalizes outliers, or MAE, which lacks smoothness. The Adam optimizer with a learning rate of $\alpha = 0.001$ is used to train the model, ensuring fast convergence.

IV. FORECASTING AND HYPERPARAMETER TUNING

Hyperparameter tuning plays a crucial role in ensuring the model's ability to generalize effectively and capture the intricate patterns in RRP and Total Demand. Properly tuning parameters like d_{model}, $dim_{hidden}$and num_{layers} is essential for balancing model complexity and performance, and it directly affects the predictive accuracy of the hybrid GPI-DSSM model. The impacts of transformer and DSSM parameters are discussed further with relevant simulations taken over any 24-hour period. TABLE I details the approach used in studying the effects of different parameters being increased while ensuring independency on each other.

TABLE I. STUDY OF DIFFERENT SCENARIOS AND PARAMETERS USED

Scenarios	***Parameters Being Increased***	***Constant Parameters***
A	d_{model} (8, 12, 16, 20, 24, 28, 32)	num_{layers} (2) dim_{hidden} (8)
B	num_{layers} (2, 4, 6, 8)	d_{model} (8) dim_{hidden} (8)
C	dim_{hidden} (8, 16, 32, 64)	d_{model} (8) num_{layers} (2)

A. *Impact of Transformer Parameters*

1) *d_{model} – Hidden dimension of Tranformer:*

The d_{model} parameter, which controls the hidden dimension of the transformer, defines the size of the feature space for each time step. A higher d_{model} allows the model to learn more complex patterns, but it can also lead to overfitting if not tuned properly. Choosing the right d_{model} is crucial for capturing long-range dependencies without unnecessarily increasing the model's complexity. Simulations for scenario A, as illustrated in Fig. 3, show that lower values of d_{model} (such as 8 and 12) underfit the data resulting in high MAE of 53.74 and 49.36 respectively, failing to capture enough price fluctuations, while higher values (such as 28 and 32) result in higher complexity despite lower MAE of 31.06. An optimal d_{model} lies between 16 and 20 for this dataset, providing a balance between model expressiveness and generalization, with lowest MAE of 29.79 at d_{model} of 16.

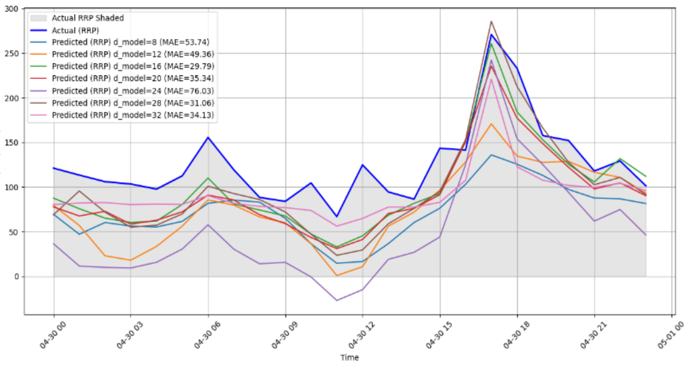

Fig. 3. Forecast of RRP with different D model configurations

2) num_{layers}– Transformer Layers:

The number of layers in the Transformer architecture, denoted by num_{layers}, controls how deeply the model can capture temporal relationships in the data. Increasing the num_{layers} improves the model's ability to recognize deeper patterns but introduces higher computational costs and risks overfitting. In the study of Scenario B, as illustrated in the Fig. 4, the model achieves relatively low error with 2 layers. Adding more layers introduces unnecessary complexity without improving performance. Further, it also increases the computational costs associated with the simulations.

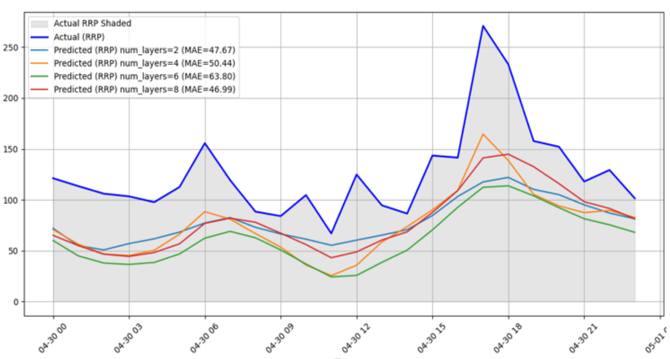

Fig. 4. Forecast of RRP with different Num Layer configurations

B. *Impact of DSSM dim_{hidden} – DSSM GRU Hidden Dimension*

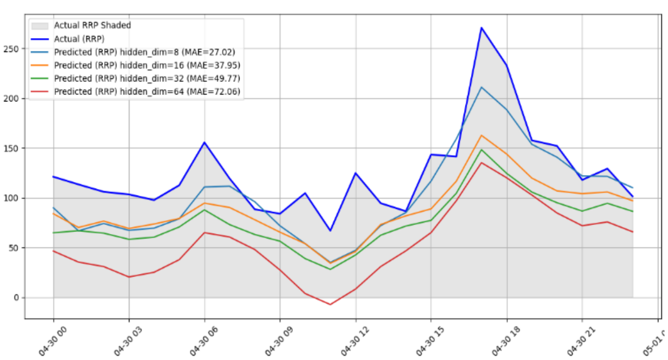

Fig. 5. Forecast of RRP with different Hidden Dimension configurations

As observed in Fig. 5, the configuration for Hidden Dimension (dim_{hidden}) = 8 produced the lowest MAE of 27.02, indicating best performance in terms of numerical accuracy. As the hidden dimension increases, the MAE rises with dim_{hidden} = 64 resulting in the largest MAE of 72.02. This trend suggests that increasing the hidden dimension introduces complexity that may lead to poor generalization and overfitting, when judging interdepenency of RRP solely on the demand.

V. Discussion and Conclusions

The empirical results demonstrate that the hybrid GPI-DSSM model can effectively forecast RRP and Total demand, making it a valuable tool for market participants who need to anticipate price fluctuations in state-wide energy systems. By focusing solely on RRP and Total demand data, the model bypasses the spatial-temporal complexities introduced by external variables like weather, offering a more straightforward and robust approach to large-scale energy forecasting. This approach is particularly flexible when dealing with aggregate state-wide data where incorporating weather data is unreliable due to its Spatio-temporal characteristics across many regions.

Additionally, hyperparameter tuning is critical in optimizing the model's performance, especially when working with datasets that exclude external variables like weather across multiple regions. Since no weather-related data is used, the model is heavily reliant on historical RRP and Total demand to learn patterns and hidden relationships, making careful tuning of these internal parameters essential to ensuring capturing hidden demand-price relationship across state-wide aggregate data forecasts. One hidden potential of this research is related to computing power. To reach exponentially limited error potential, the model may need to undergo training with even larger datasets with significantly larger number of epochs. Future work could include advanced versions of the GPI-DSSM model to better address geo-spatial temporal challenges, where the model can further incorporate hybrid structures that consider both local and state-wide influences on demand and price.

References

[1] Z. Su, G. Zheng, M. Hu, L. Kong, and G. Wang, "Short-term load forecasting of regional integrated energy system based on spatio-temporal convolutional graph neural network," Electric Power Systems Research, vol. 232, Jul. 2024, doi: 10.1016/j.epsr.2024.110427.

[2] J. Liu and Y. Fu, "Renewable energy forecasting: A self-supervised learning-based transformer variant," Energy, vol. 284, Dec. 2023, doi: 10.1016/j.energy.2023.128730.

[3] Q. Zhang, J. Chen, G. Xiao, S. He, and K. Deng, "TransformGraph: A novel short-term electricity net load forecasting model," Energy Reports, vol. 9, pp. 2705–2717, Dec. 2023, doi: 10.1016/j.egyr.2023.01.050.

[4] M. C. Chiu, H. W. Hsu, K. S. Chen, and C. Y. Wen, "A hybrid CNN-GRU based probabilistic model for load forecasting from individual household to commercial building," *Energy Reports*, vol. 9, pp. 94–105, Oct. 2023, doi: 10.1016/j.egyr.2023.05.090.

[5] C. Rao, Y. Zhang, J. Wen, X. Xiao, and M. Goh, "Energy demand forecasting in China: A support vector regression-compositional data second exponential smoothing model," *Energy*, vol. 263, Jan. 2023, doi: 10.1016/j.energy.2022.125955.

[6] A. Srinivasan, R. Wu, P. Heer, and G. Sansavini, "Impact of forecast uncertainty and electricity markets on the flexibility provision and economic performance of highly-decarbonized multi-energy systems," *Appl Energy*, vol. 338, May 2023, doi: 10.1016/j.apenergy.2023.120825.

[7] "AEMO | NEM data dashboard." Accessed: Sep. 23, 2024. [Online]. Available: https://aemo.com.au/energy-systems/electricity/national-electricity-market-nem/data-nem/data-dashboard-nem

[8] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, Sep. 2021, doi: 10.1016/j.neucom.2021.03.091.

[9] N. Bacanin *et al.*, "Multivariate energy forecasting via metaheuristic tuned long-short term memory and gated recurrent unit neural networks," *Inf Sci (N Y)*, vol. 642, Sep. 2023, doi: 10.1016/j.ins.2023.119122.

[10] C. Yang, Y. Wang, B. Yang, and J. Chen, "GRAformer: A gated residual attention transformer for multivariate time series forecasting," *Neurocomputing*, vol. 581, May 2024, doi: 10.1016/j.neucom.2024.127466.