Chapter 6

Energy Prediction Using Extreme Weather Conditions for Smart Buildings

Energy prediction is crucial for smart buildings to maximize energy use and guarantee sustainable energy storage [146]. To estimate energy consumption in smart buildings, this chapter presents a novel method based on the Bidirectional Encoder Representations from Transformers (BERT) model. The study examines how weather and energy use are related, using data from the 'Manufacture of leather and related products' industry spanning six years. This study uses MSE, MAPE, and R-squared (R^2) metrics to compare the performance of LSTM, TCN, TFT, and BERT models in predicting energy consumption for 1-hour, 8-hour, and 24-hour intervals. The BERT model outperforms all other models with multivariate data, achieving superior outcomes such as an MAE of 0.011, MSE of 0.002, MAPE of 0.070, and R^2 of 0.979.

6.1 Introduction

The popularity of smart buildings has risen due to their ability to boost efficiency and enhance the comfort of people's lives [147], [148]. Accurate estimation of energy usage is essential for process optimization and guaranteeing sustainable and effective energy use [149], [150]. Using a variety of Machine Learning (ML) models, including RNN, LSTM, and GRU [151], [152], [153], researchers are looking at sophisticated data analysis techniques to forecast energy use. Because of the global impact of climate change, the relationship between energy consumption and weather parameters is of great importance [154], [155], [156]. Many other studies have looked at the connection between energy use and data from smart meters, occupancy, and environmental factors [157], [158], [159], [160], [161]. Early studies concentrated on machine learning techniques, including Support Vector Machines (SVM), Artificial Neural Networks (ANN), and Ensemble Learning (EL) [153]. More recent research has investigated hybrid models that combine multiple techniques, such as a recurrent neural network with an architecture inspired by the Time Location of the gating Mechanism in Long Short Term Memory (TL-MCLSTM), to increase accuracy [162]. More and more energy demand projections are being made using meteorological data [163], [164], [165].

In light of the increasing significance of energy prediction in smart buildings, this research explores the application of the BERT model for energy consumption prediction. It is compared with other sophisticated machine learning models, including Long Short-Term Memory (LSTM), Temporal Convolutional Networks(TCN), and Temporal Fusion Transformer (TFT). The research uses six years of data from the "Manufacture of leather and related products" industry to examine how weather patterns affect energy consumption. Metrics including MAE, MSE, MAPE, and R-squared (R^2) are used to measure the models' effectiveness. Despite their tremendous achievements, there are still a few issues with energy prediction models. Conventional models such as TFT, TCN, and LSTM frequently have trouble identifying complex patterns in the data and

may need a lot of hyperparameter adjustment. Additionally, these models may not fully leverage the contextual information present in the data, leading to suboptimal performance.

Numerous investigations have examined diverse ML methodologies to forecast energy use. These techniques include GRU, LSTM, and RNNs, which have demonstrated encouraging outcomes when used with smart meter electricity data [151], [152], [153]. A hybrid strategy that combines LSTM and CNN has also been proposed to predict building energy. Identifying long-term dependencies and spatial patterns in the data improves performance[166] [167]. Due to the impact of climate change on energy consumption, other researchers have looked at weather-based energy demand forecasting and emphasized the significance of taking weather characteristics like temperature, wind, and precipitation into account [154], [155], [156]. Research has also examined the sustainability of energy use and the role of future energy strategies in addressing global energy challenges [168]. Some studies focused on novel approaches, such as hybrid models combining various forecasting techniques to predict wind and temperature-based energy forecasting [163]. There is an increasing tendency in the literature to use hybrid models and advanced machine learning to improve the accuracy of energy forecasts in smart buildings. Researchers are working to create reliable forecasting techniques that can optimize energy use and support sustainable energy management by combining a variety of weather parameters and complex algorithms.

In this chapter, the following contributions are made to address these challenges:

- The Bidirectional Encoder Representations from Transformers (BERT) model is introduced for energy prediction because it captures intricate patterns and dependencies in the data.
- The proposed model is tested on a real-world dataset of six years collected from the 'Manufacture of leather and related products' industry, ensuring the applicability of this approach in practical settings.

- A comprehensive comparison between the BERT model and other state-of-the-art machine learning models, including LSTM, TCN, and TFT, highlights its superior performance.
- The predictive accuracy of the models is evaluated for different forecasting horizons, specifically 1 hour, 8 hours, and 24 hours ahead, demonstrating the robustness of this approach across various timeframes.
- The models are assessed using multiple performance metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2), providing a thorough evaluation of their effectiveness.

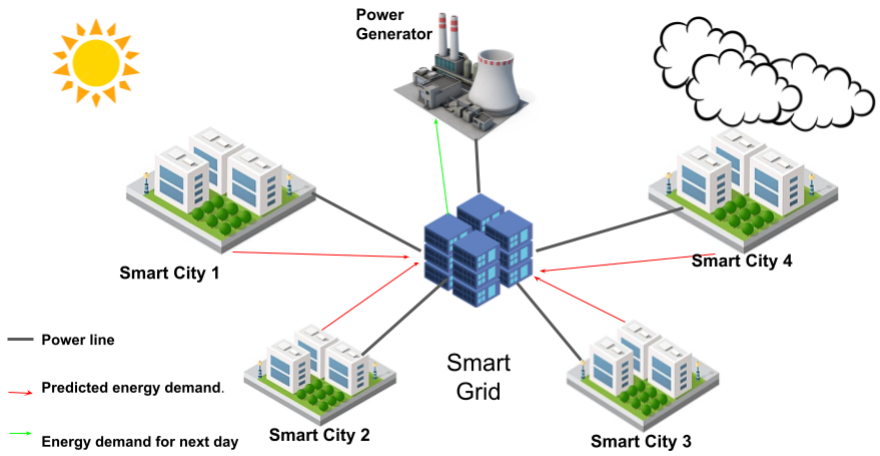

Figure 6.1: Application of the proposed model.

6.2 Application Scenario

The proposed model facilitates predicting energy requirements across different sectors, including buildings, industries, and smart grids. Figure 6.1 shows an energy storage system for smart buildings that store renewable energy according to expected energy consumption. Envision a scenario where these entities can accurately forecast their future energy needs and enable precise control over the timing and manner of energy

consumption. This capability leads to optimized resource utilization, cost savings, and a more sustainable approach. Moreover, it supports storing sustainable energy aligned with a building's forecasted consumption patterns. Practically, this research contributes to more informed scheduling of energy-intensive tasks, real-time adjustments of building systems based on projected demand, and efficient integration of renewable energy sources.

6.3 Problem Formulation and Proposed Work

6.3.1 Problem Formulation

This chapter uses a dataset consisting of $DateandTime$ (D), $Temperature$ (T), $DewPoint$ (DP), $Humidity$ (H), and $EnergyConsumed$ (E). The primary objective of this work is to predict the energy consumed after 1 hour ($E_{\text{next\_hour}}$), 8 hours ($E_{\text{next\_8\_hours}}$), and 24 hours ($E_{\text{next\_day}}$). The dataset is represented as a collection of tuples $(D_i, T_i, DP_i, H_i, E_i)$, where i denotes the data instance index. The proposed BERT model utilizes the following input features:

$$\begin{bmatrix} D_{i-1\_b} & T_{i-1\_b} & DP_{i-1\_b} & H_{i-1\_b} & E_{i-1\_b} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ D_{i-1} & T_{i-1} & DP_{i-1} & H_{i-1} & E_{i-1} \\ D_i & T_i & DP_i & H_i & E_i \end{bmatrix}$$

The model learns to predict $E_{\text{next\_hour}}$, $E_{\text{next\_8\_hours}}$, and $E_{\text{next\_day}}$ by capturing temporal patterns and relationships between the weather parameters and energy consumed. The model architecture employs the BERT model, known for its effectiveness in handling time series data. BERT's self-attention mechanism enables it to identify critical dependencies and learn patterns in the data over various time horizons.

In this work, energy consumption predictions are made for 1-hour, 8-hour, and 24-hour intervals. These particular periods are selected for several significant reasons.

First, anticipating one hour in advance is essential for operational planning and making quick modifications. Building managers can respond quickly to occupancy or weather conditions changes by optimizing HVAC systems, lighting, and other equipment with short-term management. Participating in demand response programs, where buildings must swiftly modify their energy consumption in response to grid signals, also depends on this timeline. Second, an 8-hour estimate makes sense for shift scheduling and intermediate planning since it corresponds with regular work shifts in many industries. Overall, the weather conditions generally change within a short duration, which is another reason for forecasting the energy to be 1 hour, 8 hours, and 24 hours ahead.

Precise predictions facilitate the management of energy usage during various shifts, guaranteeing effective energy utilization all day and supporting resource distribution. Precise projections facilitate the management of energy consumption throughout various shifts, guaranteeing effective energy utilization throughout the day and supporting the distribution of resources for maintenance staff and operating tactics. Finally, a 24-hour forecast offers a thorough perspective for everyday energy management. Strategic planning for energy procurement and consumption is made possible by this long-term prediction, which also helps in energy rate negotiations and plans for energy storage or using alternative energy sources. Long-term forecasts also help achieve sustainability objectives by guaranteeing that energy use patterns coincide with more comprehensive environmental and economic measures during the day. Thus, these precise intervals address short-term, medium-term, and long-term requirements, offering a thorough method for energy management in smart buildings.

Real-valued vectors of dimension (look_back $\times$ 5) make up the model's input. The output, for each dimension 1×1, comprises the estimated energy usage for the next 1, 8, and 24 hours. To train the model and provide a more balanced assessment of prediction errors. The loss function is computed using mean absolute error, or MAE:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{E}_{\text{next}}^{(i)} - E_{\text{next}}^{(i)}| \tag{6.1}$$

where next(i) is the predicted energy consumption for the i-th instance after the given period, $E_{\text{next}}^{(i)}$ is the actual energy consumption for the same instance, and n is the total number of data instances. Figure 6.2 describes the proposed model's workflow. In the encoder part of BERT, several key mathematical operations are employed to process the input sequence and extract meaningful representations

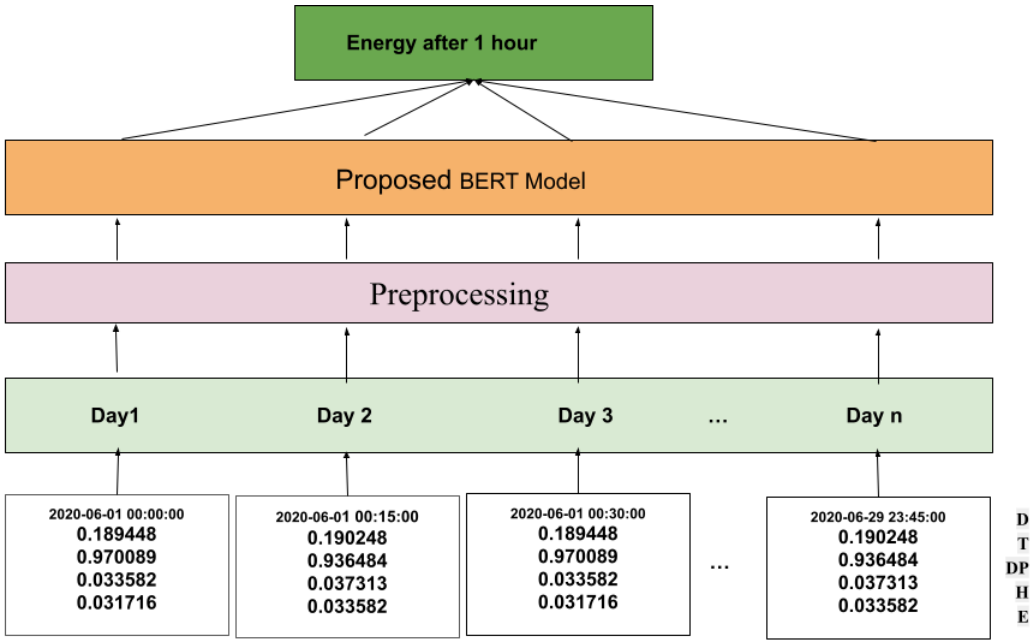

Figure 6.2: Workflow of the proposed model.

1. **Self-Attention Mechanism**: Every token in the input sequence has an attention score determined by the self-attention mechanism. For every token j, the attention mechanism computes the attention scores $\alpha_{i,j}$ based on the input hidden states H_i at time i.

$$\alpha_{i,j} = \text{softmax}\left(\frac{(W_q H_i)_j \cdot (W_k H_i)_j^T}{\sqrt{d_k}}\right) \tag{6.2}$$

The attention scores between each pair of input tokens are calculated by the

self-attention mechanism given an input sequence of length n, represented by a matrix $X \in \mathbb{R}^{n \times d}$, where d is the dimensionality of each input vector. In order to accomplish this, the input sequence is linearly transformed by computing the dot product between the queries, keys, and values:

$$Q = XW_q, \quad K = XW_k, \quad V = XW_v$$

where the queries, keys, and values matrices are denoted, respectively, by Q, K, and V. Learnable weight matrices with dimensions $d \times d_k$, $d \times d_k$, and $d \times d_v$ are represented by W_q, W_k, and W_v, respectively. Scaled dot-product attention is used to calculate the attention scores:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{6.3}$$

The matrix of dot products between queries and keys in this case is $QK^\top \in \mathbb{R}^{n \times n}$, which has been scaled by $\sqrt{d_k}$ to avoid excessive values in the softmax calculation that could result in vanishing gradients. The attention scores are normalized throughout the sequence by the softmax function.

2. **Weighted Sum of Values**: The encoder calculates the weighted sum of the values ($W_v H_i$) for each token j using the attention scores. The expression for this operation is:

$$z_{i,j} = \alpha_{i,j} \cdot (W_v H_i)_j \tag{6.4}$$

The context vector is then created by first weighing the values using the attention weights (the softmax function's output):

$$C = \text{Attention}(Q, K, V) \tag{6.5}$$

The weighted total of values, which represents the outcome of the self-attention mechanism, is contained in the context vector $C \in \mathbb{R}^{n \times d_v}$.

3. **Multi-Head Attention**: Multiple attention heads are used by BERT to collect various elements of the input stream. The input sequence is projected into various queries, keys, and values several times in multi-head attention, and the attention mechanism is applied separately to each head:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(h_1, h_2, \ldots, h_H)W_o \tag{6.6}$$

The attention mechanism for each head is represented as $h_i = \text{Attention}(QW_q^i, KW_k^i, VW_v^i)$, where H is the number of heads. A weight matrix $W_o \in \mathbb{R}^{d_v \cdot H \times d}$ is used to concatenate attention outputs.

4. **Position-wise Feedforward Networks**: The BERT model captures complicated relationships and dependencies within the input sequence by utilizing self-attention and multi-head attention. After multi-head attention, BERT applies independent position-wise feedforward networks to each token. In view of the hidden states H_i, the feedforward network's output $\text{FFN}(H_i)$ can be expressed as follows:

$$\text{FFN}(H_i) = \text{ReLU}(H_i W_1 + b_1)W_2 + b_2 \tag{6.7}$$

5. **Layer Normalization**: BERT uses layer normalization to stabilize the training process after every sublayer, including the feedforward networks and the attention mechanism. The operation of layer normalization can be written as follows:

$$\text{LayerNorm}(H_i) = \frac{(H_i - \mu)}{\sigma} \odot \gamma + \beta \tag{6.8}$$

Where $\odot$ indicates element-wise multiplication, γ and β are learnable scaling and shifting parameters, and μ and σ represent the mean and standard deviation of H_i.

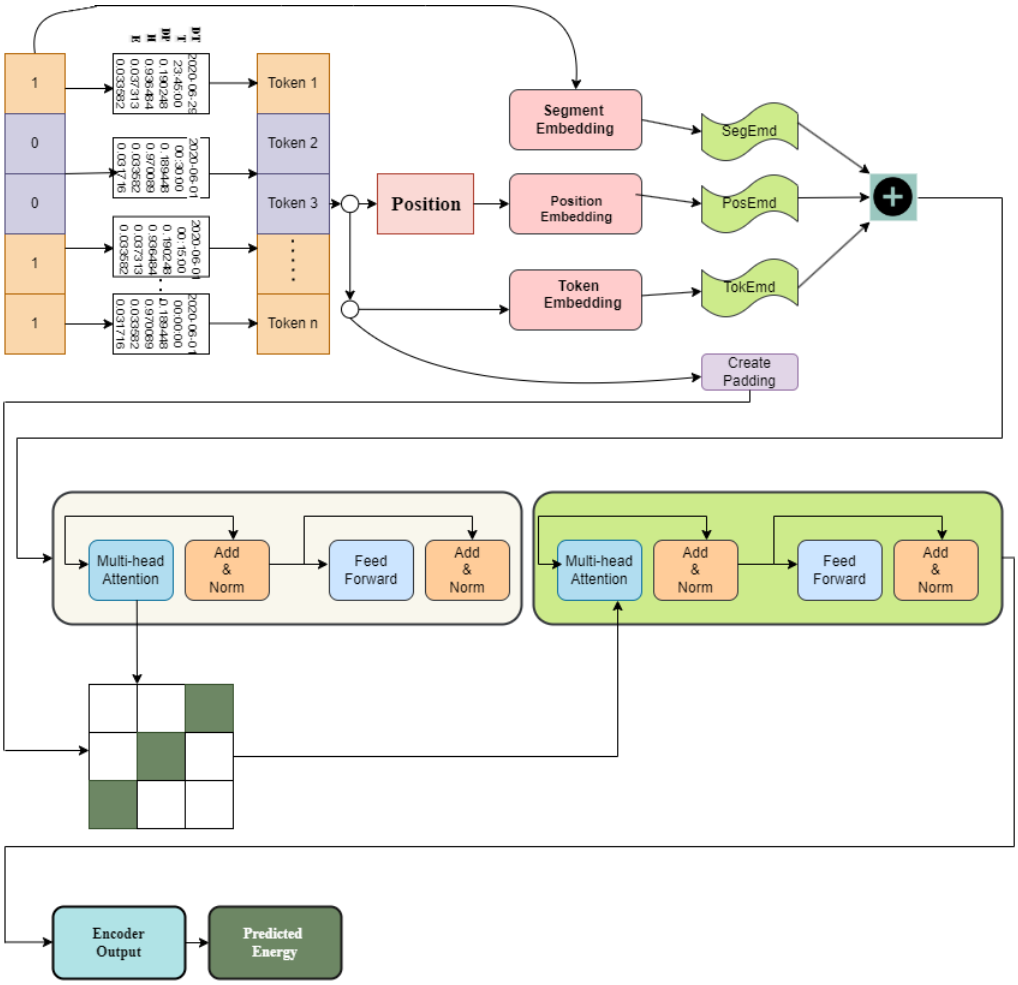

Figure 6.3: The proposed hybrid BERT architecture.

6.3.2 BERT Architecture

The BERT model is a transformer-based architecture that leverages self-attention mechanisms to process input sequences. The model consists of an encoder stack composed of multiple identical layers, allowing it to learn complex representations of the input data. Figure 6.3 demonstrates the proposed hybrid BERT architecture.

6.3.2.1 Encoder Layer

The two primary sub-layers of each encoder layer in the BERT model are a feed-forward neural network and a multi-head self-attention mechanism. The multi-head

self-attention mechanism enables the model to attend to multiple representation subspaces simultaneously. To accomplish this, attention scores are calculated for each input sequence, which is used to extract context vectors directly.

6.3.2.2 Feed-Forward Neural Network

The feed-forward neural network performs two linear transformations With a non-linear activation function and it is defined as follows:

$$\mathrm{FFN}(x) = \mathrm{ReLU}(xW_1 + b_1)W_2 + b_2, \tag{6.9}$$

where b_1 and b_2 are learnable bias vectors and W_1 and W_2 are learnable weight matrices.

6.3.2.3 Layer Normalization and Residual Connection

The sub-layer in the encoder has residual connections and layer normalization for stable training and enhanced performance. By adding the input x to each sub-layer, output, the residual connection helps to avoid the vanishing gradient issue:

$$\mathrm{Residual}(x, f) = x + f \tag{6.10}$$

where f is the output of a sub-layer.

6.3.2.4 Stacking Encoders

The BERT model can learn more intricate input data representations with its stack of L encoder layers. The final encoder layer's output provides the input sequence's contextualized representation.

6.3.2.5 Output Layers

The encoder stack's output passes through task-specific output layers for tasks like sequence and token classifications. The model uses a token in the last layer for sequence classification. The model employs the representation of each token for token classification. The BERT architecture offers a strong and adaptable foundation for various applications, including energy prediction tasks. BERT performs exceptionally well at comprehending and handling complex sequential material by fusing its intricate structure with attention mechanisms.

6.3.3 Data Input and Tokenization

The BERT model uses multivariate time series data as input when predicting energy. Features like $Dateand Time$ (D), $Temperature$ (T), $Dew\,Point$ (DP), $Humidity$ (H), and $EnergyConsumed$ (E) are among those included in the data. The goal is to predict energy usage for upcoming time intervals—such as 1 hour, 8 hours, or 24 hours. A set of tuples $(D_i, T_i, DP_i, H_i, E_i)$, where i indicates the index of the data instance, is used to represent the input data. Tokenizing and formatting data for the BERT model's input involves preprocessing. To do this, the features are transformed into fixed-length vectors representing the data's temporal trends and patterns.

- **Feature Tokenization:** Before anything further, the input features D, T, DP, H, and E are standardized and normalized to provide uniform scaling across various data types:

$$D' = \frac{D - \mu_D}{\sigma_D}, \quad T' = \frac{T - \mu_T}{\sigma_T},$$
$$DP' = \frac{DP - \mu_D P}{\sigma_D P}, \quad H' = \frac{H - \mu_H}{\sigma_H},$$
$$E' = \frac{E - \mu_E}{\sigma_E},$$

where μ and σ represent each feature's mean and standard deviation, respectively.

- **Input Sequences:** The model's input sequence comprises a window of data instances spanning from earlier time steps to the current instance:

$$\begin{bmatrix} D'_{i-\text{l\_b}} & T'_{i-\text{l\_b}} & DP'_{i-\text{l\_b}} & H'_{i-\text{l\_b}} & E'_{i-\text{l\_b}} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ D'_{i-1} & T'_{i-1} & DP'_{i-1} & H'_{i-1} & E'_{i-1} \\ D'_i & T'_i & DP'_i & H'_i & E'_i \end{bmatrix}$$

  In this case, the number of prior data points considered for forecasting is denoted by l_b. The windowing method captures the data's temporal context.

- **Token Embeddings:** The input sequence is then embedded using learned token embeddings for every feature, following the normalization of the features. This phase is essential for the model to effectively understand the correlations between various features in the input data. After passing through the BERT model, the output vectors are converted into higher-dimensional contextualized representations. Subsequently, the energy consumption for the designated future time intervals is predicted using these representations. To summarise, the multivariate time series data must be formatted appropriately for the model as part of the BERT data input and tokenization phase. The data is normalized, windowed, and embedded as part of this process to make sure the model can accurately represent temporal patterns and relationships in the data.

6.3.4 Training and Fine-Tuning

The BERT model's parameters are changed during training and fine-tuning to reduce prediction errors and optimize dataset performance for energy prediction. The procedure verifies the model's performance using training data, validation, and test datasets.

6.3.5 Loss Function

The optimization process is guided by the loss function, which calculates the difference between the predicted and actual energy consumption values. Because of its consistency in assessing prediction mistakes, the MAE is frequently selected as the loss function for energy prediction tasks:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{E}_{\text{pred}}^{(i)} - E_{\text{true}}^{(i)}| \tag{6.11}$$

where pred(i) is the predicted energy consumption for the i-th instance, $E_{\text{true}}^{(i)}$ is the actual energy consumption for the same instance, and n is the number of data instances.

6.3.5.1 Optimization Algorithm

An optimization algorithm like Adam, which combines momentum and adaptive learning rate techniques for effective convergence, is used to modify the model's parameters. The goal of the optimization procedure is to reduce the loss function:

$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta \text{Loss}(\theta_t) \tag{6.12}$$

The model's parameters are denoted by θ, the learning rate by α, and the gradient of the loss function with respect to the parameters at iteration t is represented by $\nabla_\theta \text{Loss}(\theta_t)$.

6.3.5.2 Training Process

The following steps are involved in the training process:

1. **Data Preparation:** The input data is preprocessed and tokenized as previously described. The data is then divided into training, validation, and test sets.
2. **Model Initialization:** The BERT model is initialized with random or pre-trained weights, depending on the approach.
3. **Forward Pass:** Future energy usage is predicted by running the model with the

supplied data.

4. **Loss Calculation:** The predicted and actual energy consumptions are compared to construct the loss function.
5. **Backward Pass:** Backpropagation is used to compute the gradients of the loss function concerning the model's parameters.
6. **Parameter Update:** The optimization algorithm updates the model's parameters.
7. **Validation:** The validation set evaluates the model's performance to track development and avoid overfitting.
8. **Iteration:** The process is repeated until the model converges or reaches a predetermined number of iterations.

6.3.5.3 Fine-Tuning

Optimal performance on the energy prediction challenge can be attained by fine-tuning the BERT model by modifying certain model hyperparameters. These variables could be the quantity of training epochs, learning rate, and batch size. Further methods to enhance generalization and avoid overfitting include dropout and early ending. Once the model has been trained and fine-tuned, its performance and possible real-world application may be tested using the test dataset. In summary, a methodical strategy is needed to optimize the loss function, modify the model's parameters, and guarantee reliable and accurate projections of future energy usage when training and fine-tuning the BERT model for energy prediction.

6.4 Experimental Setup

6.4.1 Dataset description

The dataset [169] includes time stamps, energy consumption data from 386 users spread over three cities, and information on key weather parameters such as temperature, dew

point, humidity, and pressure. The data spans a significant period from January 1, 2015, to November 1, 2022, encompassing nearly eight years of detailed records. Each city has unique weather conditions, providing diverse data to analyze. The data was collected at a high resolution of 15 minutes, allowing for a granular examination of energy usage patterns about weather changes over time. This level of detail facilitates the identification of trends and correlations between energy consumption and various weather factors. To ensure the consistency and accuracy of the data, overlapping timestamps are carefully consolidated, producing a single dataset with a uniform frequency of 15 minutes. By standardizing the data in this manner, the dataset becomes more reliable for analysis and modeling. Such meticulous data preparation is essential for effectively training predictive models, as it reduces the potential for inconsistencies and inaccuracies arising from misaligned or duplicated timestamps. The consolidated dataset offers a solid foundation for exploring the relationships between energy usage and weather conditions over an extended period, contributing to developing more precise and effective energy prediction models.

6.4.2 Deep Learning Training

To take advantage of the continuity and lack of breakdowns or gaps during this time, this work uses July data for training and testing the predictive model. 30% of the September data is set aside for testing, and 70% is allocated to training. This distribution ensures a balance between the data accessible for model training and the data utilized for model evaluation. The BERT model is used in a two-step approach for model training. The BERT model successfully recognizes spatial and temporal patterns in the input data and is first trained with the training set. This covers successive patterns in energy usage over time and relationships between weather and energy consumption. The BERT model is optimized to learn and forecast energy usage patterns more precisely after being trained using the training dataset. After that, the trained model's accuracy and performance

are assessed using a different testing dataset. This method fully uses the BERT model to capture complex spatial and temporal linkages in the data and enable more accurate estimates of energy use.

6.5 Results and Discussions

A workstation equipped with an 11th Gen Intel(R) Core(TM) i5-11300H @ 3.10GHz 3.11 GHz CPU and 16GB RAM used to run the simulations. Python 3.11's Hugging Face Transformers library implements the BERT model. Ten repetitions of the experiments were carried out with the same random number generator seed to improve uniformity and reproducibility.

6.5.1 Evaluation Metrics

Evaluating the models' accuracy and performance is crucial to predictive modeling. Standard evaluation measures like R-squared (R^2) and MAE, MSE, MAPE, and BERT are used to assess the performance of the suggested model.

- **MAE:** The average absolute difference between the expected and actual values is measured by the mean absolute difference (MAE), and it is as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|, \tag{6.13}$$

  Where the number of data points is indicated by n, the actual value is indicated by y_i, and the predicted value for the i-th data point is indicated by $\hat{y}_i$. Better model performance is indicated by a lower MAE number.

- **MSE:** The mean squared error (MSE) calculates the average of the deviations between the expected and actual values. It is calculated as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{6.14}$$

A lower MSE value indicates better model performance; nevertheless, the squaring of errors makes MSE more susceptible to outliers.

- **MAPE:** By calculating the average percentage difference between the expected and actual values, MAPE assesses the relative error of the model. It is calculated as follows:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100, \tag{6.15}$$

Better model performance is indicated by a lower MAPE score, which is especially helpful when evaluating the model error in percentage terms.

- **$\mathbf{R^2}$:** The coefficient of determination, or R^2, expresses how much of the variance in the dependent variable can be accounted for by the independent variables. It is computed as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{6.16}$$

Where $\bar{y}$ represents the mean of the actual values. A model that explains a greater variance in the dependent variable has a R^2 score closer to 1. Combined, these measures offer a thorough assessment of the BERT model's predictive capability and point out areas needing development.

6.5.2 Basic Results

The BERT model performed exceptionally well in the univariate scenario throughout all prediction horizons. The model produced an R-squared (R^2) value of 0.979 for the 1-hour forecast, with an MAE of 0.011, MSE of 0.002, and MAPE of 0.070. The BERT model can successfully estimate energy consumption within a short time period and well captures the variance in the data, as evidenced by its high R-squared value. With a R^2 value of 0.935, the 8-hour forecast demonstrated similarly remarkable results, with an MAE of 0.019, MSE of 0.006, and MAPE of 0.130. With a R^2 score of 0.917, an MAE of 0.021, an MSE of 0.008, and a MAPE of 0.151, the 24-hour forecast likewise

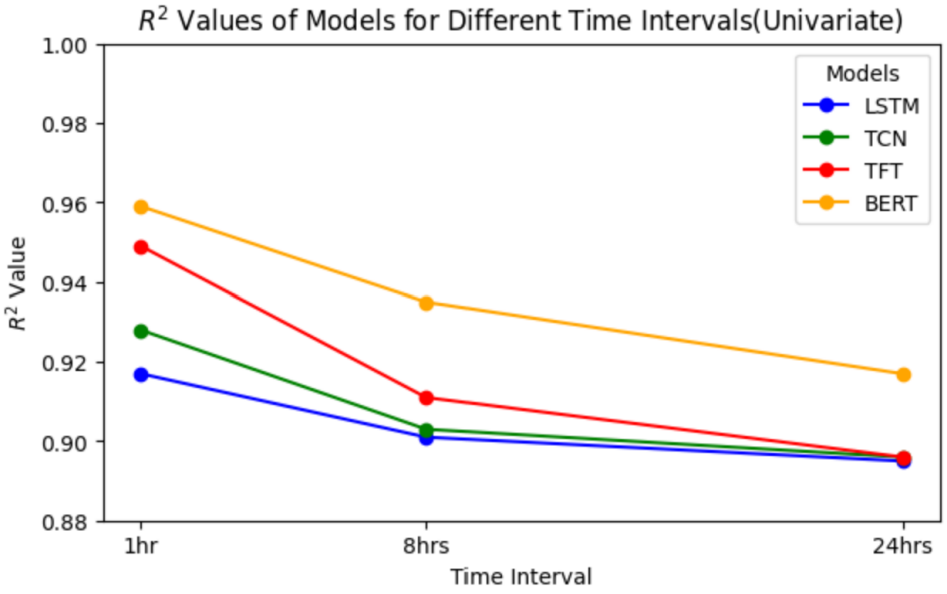

Figure 6.4: R^2 comparison of models in the univariate scenario.

showed good performance. The R^2 of all models for 1-hour, 8-hour, and 24-hour hours is plotted in Figures 6.4 (univariate) and Figure 6.5 (multivariate).

The BERT model produced reliable findings in the multivariate scenario throughout the forecast interval. More data like temperature, dew point, and humidity helped the model better represent the intricate connections between meteorological factors and energy use. Improved prediction performance resulted from the 1-hour, 8-hour, and 24-hour time frames. Particularly over longer time periods, the multivariate method substantially enhanced the model's accuracy and reliability. These findings demonstrate how flexible the model is and how well it can use various data sources to provide predictions that lead to more accurate and effective energy management plans.

6.5.3 Performance comparison

The BERT model performed better than every other model in the 1-hour forecast in every metric. Its lowest MAE of 0.011 is far lower than that of TFT, the next best model with an MAE of 0.017. Additionally, BERT's MAPE of 0.070 and lowest MSE of 0.002 showed that its forecasts were highly accurate. Additionally, with the greatest R-

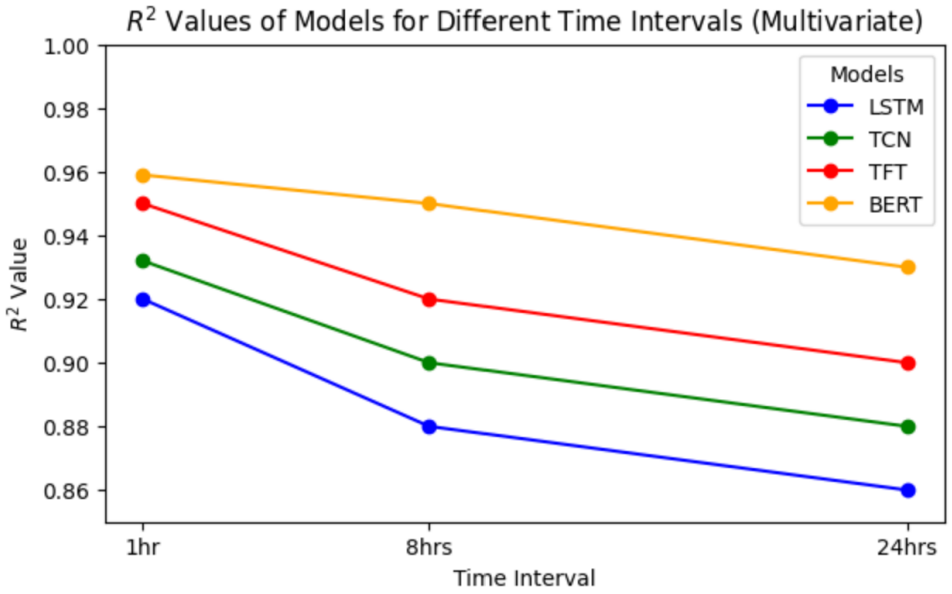

Figure 6.5: R^2 comparison of models in the multivariate scenario.

squared (R^2) value of 0.979, BERT was the most effective in explaining the short-term energy consumption data variance.

The BERT model remained the top performer among the models for the 8-hour forecast. With an MAE of 0.019, it performed worse than any other models, including TFT (0.023) and TCN (0.023). Among the models compared, BERT's MSE of 0.006 and MAPE of 0.130 indicated its superior performance. Furthermore, BERT continued to exhibit a high R^2 value of 0.935, demonstrating its capacity to consider the intricacies and variations in energy consumption patterns over the medium term.

Table 6.1: Univariate model performance comparison across 1-hour, 8-hour, and 24-hour intervals.

	1hr				**8hrs**				**24hrs**			
	MAE	**MSE**	**MAPE**	**R2**	**MAE**	**MSE**	**MAPE**	**R2**	**MAE**	**MSE**	**MAPE**	**R2**
LSTM	0.026	0.013	0.197	0.869	0.031	0.016	0.242	0.834	0.032	0.017	0.252	0.823
TCN	0.026	0.012	0.194	0.885	0.029	0.016	0.230	0.849	0.032	0.016	0.243	0.845
TFT	0.022	0.009	0.161	0.909	0.027	0.013	0.203	0.878	0.032	0.018	0.256	0.829
BERT	**0.018**	**0.006**	**0.122**	**0.943**	**0.022**	**0.018**	**0.157**	**0.918**	**0.024**	**0.011**	**0.176**	**0.896**

BERT is the most successful model, with an MAE of 0.021 and an MSE of 0.008 in the 24-hour prediction. Both values were less than the same measurements for every

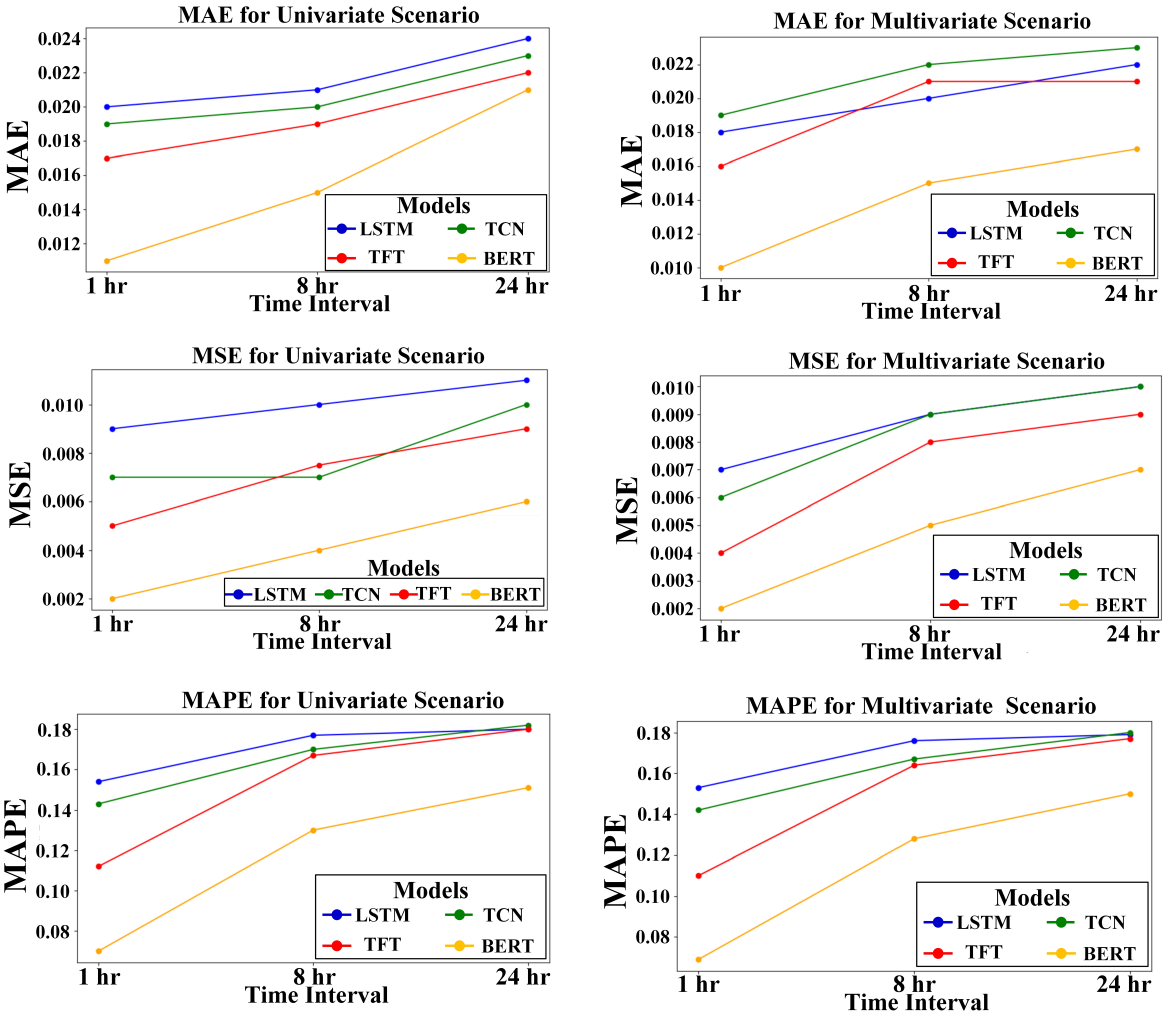

Figure 6.6: MAE, MSE, and MAPE comparisons across models for univariate and multivariate scenario.

other model. With the lowest percentage error in its predictions, BERT's MAPE of 0.151 is likewise the best of the models. The higher ability of BERT to simulate the longer-term fluctuations in energy use is further shown by the R^2 value of 0.917. As demonstrated in Tables 6.1 (Univariate) and, 6.2 (multivariate) BERT consistently produced the most accurate and dependable predictions across all forecast intervals, surpassing the other models and demonstrating its potential as a top choice for energy forecasting tasks. Figure 6.6 mentions the graphs that illustrate this comparison.

Table 6.2: Multivariate model performance comparison across 1-hour, 8-hour, and 24-hour intervals.

	1hr				8hrs				24hrs			
	MAE	**MSE**	**MAPE**	**R2**	**MAE**	**MSE**	**MAPE**	**R2**	**MAE**	**MSE**	**MAPE**	**R2**
LSTM	0.091	0.017	0.154	0.917	0.024	0.010	0.177	0.901	0.024	0.011	0.180	0.895
TCN	0.020	0.007	0.143	0.928	0.023	0.010	0.170	0.903	0.025	0.011	0.182	0.896
TFT	0.017	0.005	0.112	0.949	0.023	0.009	0.167	0.911	0.024	0.011	0.180	0.896
BERT	**0.011**	**0.002**	**0.070**	**0.979**	**0.019**	**0.006**	**0.130**	**0.935**	**0.021**	**0.008**	**0.151**	**0.917**

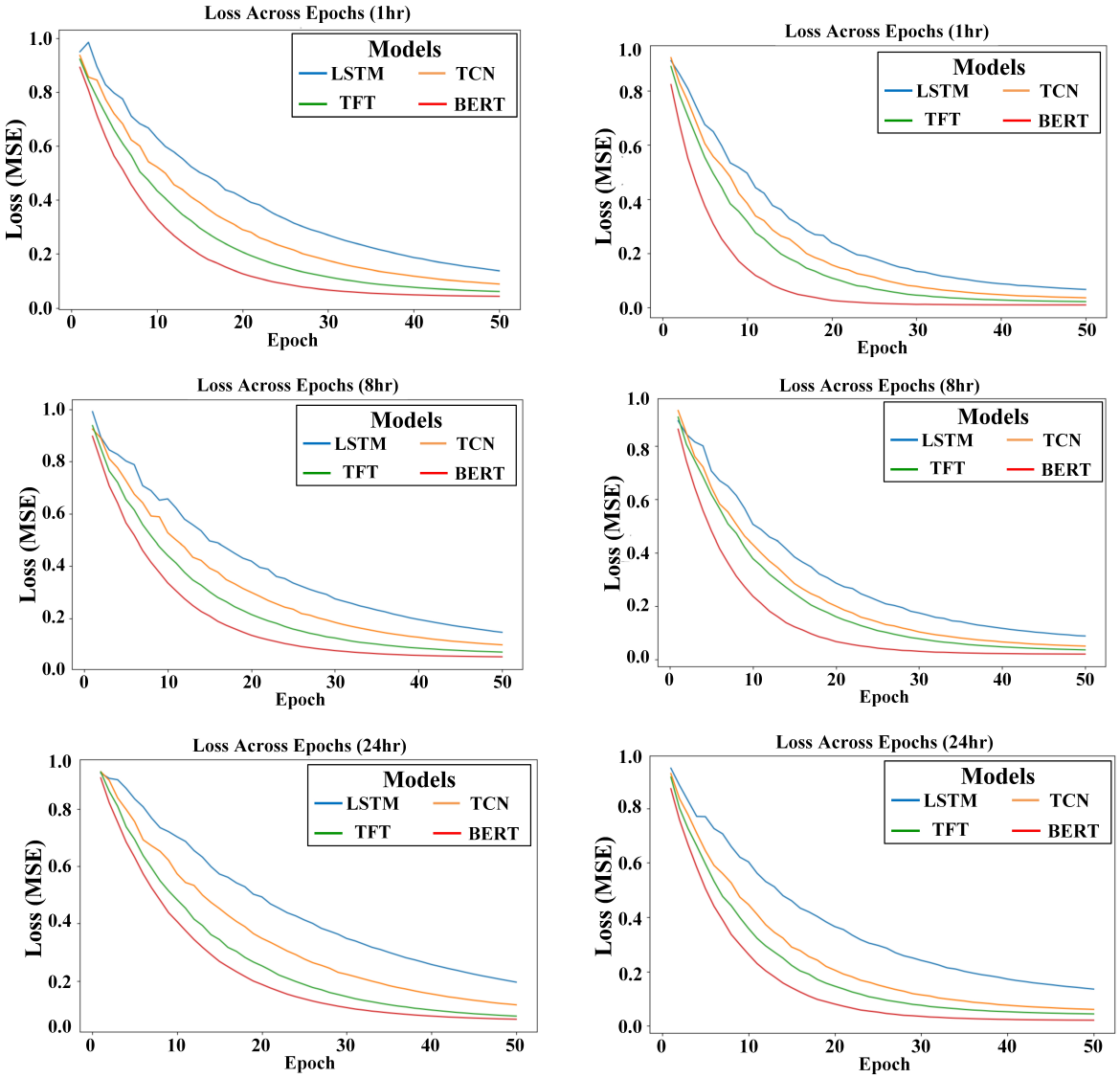

Figure 6.7: Loss across epochs in univariate vs. multivariate cases for 1hr, 8hrs, and 24hrs.

6.5.4 Univariate and Multivariate comparison

The comparison between univariate and multivariate prediction mechanisms reveals notable insights into their respective performance characteristics. Firstly, the learning rate observed in multivariate prediction models surpassed that of their univariate counterparts. This distinction is particularly evident in the speed at which the models converged to their optimal states. For instance, when predicting energy consumption one hour ahead, multivariate models achieved minimal loss in merely 10 epochs, while their univariate counterparts required twice as many epochs, i.e., 20, to reach a comparable level of optimization as shown in Figure 6.7. This discrepancy underscores the advantage of incorporating multiple input features, such as temperature, humidity, and other environmental factors, to enhance the model's learning efficiency and predictive accuracy. Additionally, as Figure 6.8 illustrates, assessing R^2 values clarifies the advantages of multivariate prediction models over univariate ones. Multivariate models demonstrated improved predictive performance, as evidenced by their consistently higher R^2 values over a range of prediction horizons, from one to twenty-four hours ahead. This superiority is assigned to multivariate models' capacity to identify more complex dependencies and correlations in the input data. Multivariate models produce more accurate forecasts because they provide a deeper knowledge of the underlying patterns influencing the dynamics of energy use by taking into account numerous input features at once.

Interestingly, the difference in R^2 values between univariate and multivariate models grew increasingly noticeable as the prediction horizon grew. Univariate models showed higher R^2 values than multivariate equivalents, especially for 24-hour forward forecasts. This discrepancy implies that multivariate models might have trouble maintaining predictive accuracy over longer time horizons, even though they are excellent at capturing short-term dependencies and fluctuations. This finding emphasizes the importance of carefully weighing the trade-offs between predictive performance and model complexity,

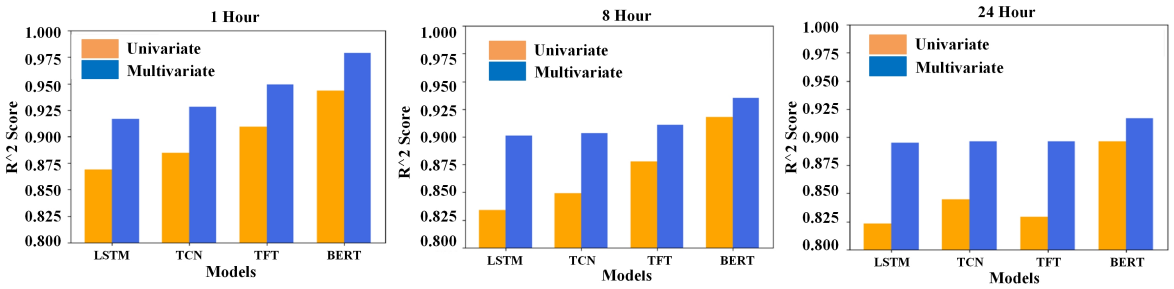

Figure 6.8: R^2 comparison of univariate and multivariate model for 1hr, 8hrs and 24hrs.

particularly when working on long-term forecasting problems related to energy projection. BERT outperforms the state-of-the-art models in both univariate and multivariate scenarios.

6.5.5 Comparsion with Existing works

Accurately estimating energy use is key for optimizing processes and ensuring efficient energy consumption. Researchers are exploring advanced data analysis methods, using models like RNN, LSTM, and GRU, to forecast energy use. Climate change has made the link between energy consumption and weather more important. Many studies have examined energy use in relation to smart meter data, occupancy, and environmental factors [165]. Earlier studies focused on machine learning models such as SVM, ANN, and EL, while newer research combines multiple techniques, like hybrid models with TL-MCLSTM, to improve accuracy. Weather data is increasingly used for energy demand predictions. This research compares BERT's performance in predicting energy use with other models like LSTM, TCN, and TFT, using six years of data from the leather industry and metrics like MAE, MSE, and R-squared[167]. Despite progress, challenges remain in energy prediction.

6.6 Conclusion and Future work

The BERT model uses self-attention processes to extract complex patterns and connections from the data and make extremely precise energy forecasts. The model provides accurate projections for one, eight, and twenty-four hours, effectively capturing the intricate links between weather and energy consumption. The model's improved performance and accuracy in energy prediction are shown by the low values of MAE, MSE, and MAPE that BERT attained together with high R-squared (R^2) values. According to the results, state-of-the-art algorithms and existing models are less effective than the BERT model. It also shows the model's generalizability and dependability in real-world scenarios because it can be adjusted to other periods.

This chapter describes the short-term energy prediction of smart buildings by considering weather events using the BERT model. The other influential factor considered for the increase in energy consumption is heat. Heat and electricity both equally contribute to the energy consumption of smart buildings. Hence, the next chapter proposes a heat energy prediction in smart buildings.