STEPS IN INNOVATION:

To implement these techniques, you’ll need historical electricity price data and a programming environment like Python with libraries such as Prophet, Tensor Flow, or PyTorch. Here’s a high-level overview of the process:

1. Data Preparation:

- Collect historical electricity price data, ensuring it’s clean and properly formatted.

- Split the data into training and testing sets.

2. Prophet Forecasting:

- Use the Prophet library to fit a model to the training data.

- Tune hyperparameters and incorporate relevant seasonality factors.

- Make predictions on the test data and evaluate model performance.

3. Deep Learning Forecasting:

- Build a recurrent neural network (RNN) or long short-term memory network (LSTM) model using TensorFlow or PyTorch.

- Train the model on the training data, considering sequence length and batch size.

- Validate the model’s performance on the test data.

4. Evaluation and Tuning:

- Assess the forecasting accuracy using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE).

- Fine-tune model hyperparameters and architecture to improve accuracy.

5. Deployment and Monitoring:

- Once satisfied with the model’s performance, deploy it for real-time or batch forecasting.

- Continuously monitor and update the model as new data becomes available.

Keep in mind that the choice between Prophet and deep learning models depends on your specific data characteristics and forecasting goals. Prophet is more interpretable and suitable for datasets with strong seasonality, while deep learning models can capture intricate patterns but may require more data and computational resources.

**Done by**

**Gunendra k**

**Au720921244021**

**JCT COLLEGE OF ENGINEERING AND TECHNOLOGY**