Final Year B.Tech (Electrical Engineering)

Semester: VII **Subject: Smart Grid Systems**

Name- Shubham Kishan And Devesh Khanna

Class- Final Year BTech EE

Roll No - PA28 PA09

Batch- A2 And A1

Experiment No: 5

Name of the Experiment: Study and implement Load Forecasting using AI Technique

Performed on:

Submitted on: 08/12/2023

Marks	Teacher's Signature with date

Aim: Study and implement Load Forecasting using AI Technique

Objective:
Apply artificial intelligence techniques to analyze the data and forecast load.

Prerequisite:
Knowledge of AI models

Theory:
Load Forecasting
Load forecasting, also known as demand forecasting or electric load forecasting, is the process of predicting the future electricity consumption or demand for a specific geographic area over a certain period of time. Accurate load forecasting is crucial for utilities, energy providers, and grid operators to efficiently plan and manage the generation, transmission, and distribution of electricity.

Methods of Load Forecasting:

1. Statistical Methods: Use historical data and mathematical models to identify patterns and trends. Common techniques include time series analysis, regression analysis, and artificial neural networks.

2. Machine Learning: Utilizes algorithms to analyze large datasets, identify patterns, and make predictions. Support Vector Machines, Random Forests, and deep learning approaches can be employed.
3. Simulation Models: Represent the power system and its components to simulate different scenarios and assess their impact on electricity demand.
4. Expert Judgment: Involves input from industry experts and stakeholders to consider qualitative factors that may not be captured by quantitative models.

Load Forecasting Process

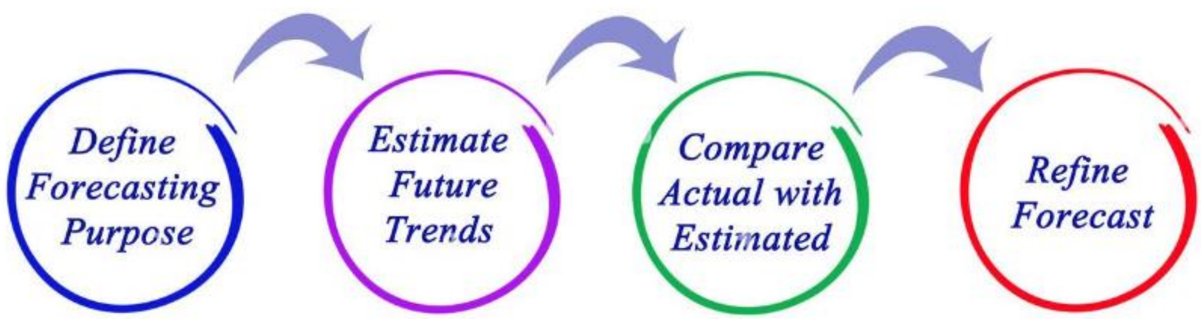


Fig: Load forecasting Process

The steps include:

1. To analyse data to identify patterns and relationships.
2. To identify, extract and select features to include in the forecast model.
3. Then to split data for training, validation and testing.
4. The different types of forecasts and their features performed
5. Popular statistical and machine learning models for point and probabilistic forecasting.
6. To select error measures and scores to assess your forecast accuracy.

Advantages of Load Forecasting:

1. Optimal Resource Planning: Load forecasting helps utilities and grid operators plan for the optimal mix of generation sources and capacities, ensuring that resources are allocated efficiently to meet expected demand.
2. Grid Reliability: Accurate load forecasts contribute to the reliability of the electrical grid by enabling proactive measures to maintain the balance between electricity supply and demand.
3. Cost Savings: Utilities can avoid unnecessary costs associated with over-generation or under-generation of electricity by aligning their resources with forecasted demand, leading to cost savings.

4. Infrastructure Investment: Long-term load forecasting assists in planning for future infrastructure development, including the construction of new power plants, transmission lines, and distribution systems.
5. Energy Efficiency: Load forecasting allows for the optimization of energy resources, promoting energy efficiency and reducing environmental impact by avoiding excessive generation during low-demand periods.
6. Demand Response Programs: Utilities can use load forecasts to implement demand response programs, encouraging consumers to adjust their electricity usage during peak times and reducing the overall demand on the grid.

Disadvantages and Challenges of Load Forecasting:

1. Uncertainty: Load forecasting is inherently uncertain due to factors such as weather variations, economic changes, and unforeseen events, making it challenging to predict future demand accurately.
2. Data Quality: The accuracy of load forecasts depends on the quality and completeness of historical and real-time data. Inaccurate or missing data can lead to less reliable predictions.
3. Technological Changes: Advances in technology, such as the integration of renewable energy sources and energy storage, introduce complexities that traditional forecasting models may struggle to account for.
4. External Factors: Events like economic downturns, policy changes, or unforeseen natural disasters can significantly impact electricity demand, and predicting such external factors is challenging.
5. Short-Term Variability: Short-term load forecasting can be particularly challenging due to sudden changes in demand patterns, especially during special events, emergencies, or unexpected shifts in consumer behavior.

Data Description:

List of Columns:
'time', 'generation biomass', 'generation fossil brown coal/lignite', 'generation fossil gas', 'generation fossil hard coal', 'generation fossil oil', 'generation hydro pumped storage consumption', 'generation hydro run-of-river and poundage', 'generation hydro water reservoir', 'generation nuclear', 'generation other', 'generation other renewable', 'generation solar', 'generation waste', 'generation wind onshore', 'forecast solar day ahead', 'forecast wind onshore day ahead', 'total load forecast', 'total load actual', 'price day ahead(Rs)', 'price actual(Rs)', 'price day ahead(Euro)', 'price actual(Euro)'

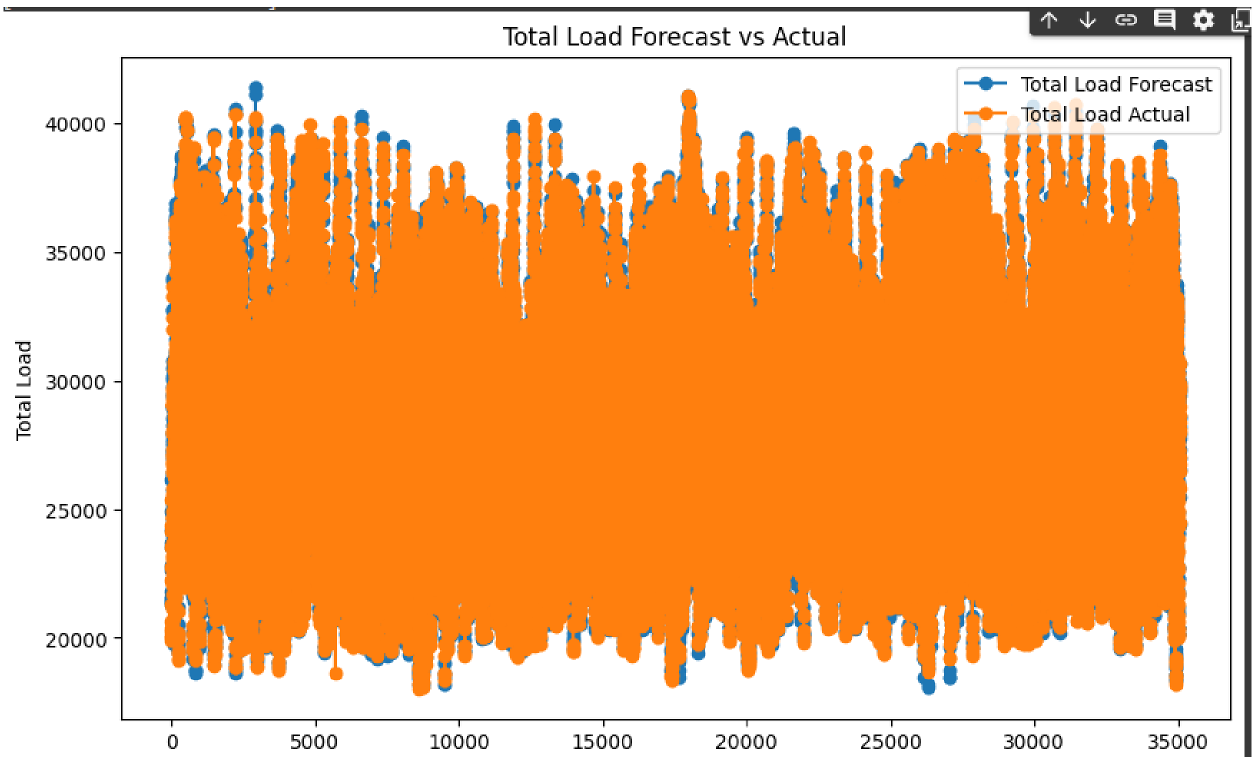
Topic:

Energy Load Forecasting Using Machine Learning: A Comparative Analysis

Program and Result:

```
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.impute import SimpleImputer

# Load the dataset from CSV
file_path = '/content/energyL.csv'  # Correct the path to your CSV file
df = pd.read_csv(file_path)

# Handle date-time features
df['time'] = pd.to_datetime(df['time'], utc=True)
df['timestamp_numeric'] = (df['time'] - pd.Timestamp("1970-01-01", tz="UTC")) // pd.Timedelta('1s')  # Convert timestamp

# Convert 'time' to Timestamp before sorting
df['time'] = pd.to_datetime(df['time'], utc=True)

# Sort by 'time'
df = df.sort_values(by='time')

# Handle missing values (fill NaN values with zero for simplicity)
df = df.fillna(0)

# Train a simple linear regression model
X = df.drop(['total load actual', 'time'], axis=1)
y_load = df['total load actual']
y_price = df['price actual(Rs)']  # Include the price in Rs as the target variable

# Impute missing values in the features used for training
imputer = SimpleImputer(strategy='mean')  # Use 'mean' or 'median' as needed
X_imputed = pd.DataFrame(imputer.fit_transform(X), columns=X.columns)

model_load = LinearRegression()
model_price = LinearRegression()
```

```python
# Function to predict load and price for a given timestamp
def predict_load_and_price(timestamp):
    timestamp = pd.to_datetime(timestamp, utc=True)
    timestamp_numeric = (timestamp - pd.Timestamp("1970-01-01", tz="UTC")) // pd.Timedelta('1s')

    # Create a DataFrame with the same features as used during training
    input_data = pd.DataFrame(columns=X.columns)
    input_data['timestamp_numeric'] = [timestamp_numeric]  # Assuming 'timestamp_numeric' is one of the features

    # Impute missing values in the input features
    input_data_imputed = pd.DataFrame(imputer.transform(input_data), columns=input_data.columns)

    # Predict load and price using the trained models
    load_prediction = model_load.predict(input_data_imputed)
    price_prediction = model_price.predict(input_data_imputed)

    # Return the predicted load and price
    return load_prediction[0], price_prediction[0]

# Get user input for the timestamp
user_input_timestamp = input("Enter the timestamp in 'YYYY-MM-DD HH:MM:SS' format: ")

try:
    # Predict load and price
    load_prediction, price_prediction = predict_load_and_price(user_input_timestamp)

    # Display the predictions
    print(f'Predicted Load: {load_prediction}')
    print(f'Predicted Price (Rs): {price_prediction}')
except Exception as e:
    print(f"Error: {e}")
```



```
Enter the timestamp in 'YYYY-MM-DD HH:MM:SS' format: 2023-12-11  08:25:00
Predicted Load: 28699.410347545676
Predicted Price (Rs): 4629.533555726626
```

Conclusion:

1. **The experiment's goal was to use AI algorithms to load prediction datasets. The data analysis made predicting the load required easy.**
2. **The practice of electric load forecasting assists in predicting future electric load demand and peak load by examining historical data.**
3. **Numerous research have shown that inaccuracies in load predictions result in higher operational expenditures.**

Post Lab Question:

1) How does incorporating weather data improve the accuracy of load forecasting models?

2) Can AI models adapt to sudden changes in the load profile?

Devesh Khanna PA09
SGS Prac-5
Post Lab :-

Q1] How does incorporating weather data improve the accuracy of load forecasting models?

⇒ → Temperature impact on heating and cooling loads.

→ Seasonal variations.

→ Renewable energy generation forecasting.

→ Humidity & air quality.

→ Storms & extreme weathers.

→ Demand response planning.

→ Load profile adjustments.

→ Supervisory Control & Data Acquisition.

Q2] Can AI models adapt to sudden changes in the load profile?

⇒ AI models adapt well to variations as it is real-time learning.

→ They can detect anomalies.

→ They can ensemble models.

→ They perform feature selection.

→ They make use of RNN & LSTM networks.

→ Continuous monitoring, feedback loop and transfer learning.

Effectiveness of adaptation depends on the model's training data, the quality of features.