Electricity Prices Prediction:

Problem Statement:

Create a predictive model that exploits historical electricity prices and relevant factors to forecast future electricity prices, assist energy providers and consumers in making informed decisions regarding consumption and investment.

Design Thinking:

The design thinking process to tackle the challenge of electricity price prediction are as follows:

1. **Empathize:**
    - Understand the needs of beneficiaries , which may include utility companies, energy traders, and consumers who want to manage their electricity costs.
    - Research and gather data to understand the historical patterns and factors affecting electricity prices.
    - **Data Availability**: Ensure that the dataset that is chosed is readily available and accessible , include historical electricity price data as well as relevant factors like date, demand, supply, weather conditions, and economic indicators.
2. **Define:**
    - **Frame the problem**: Define the specific aspects of electricity price prediction , consider the accuracy of short-term price forecasts or the ability to identify peak demand periods.
    - Set clear targets and performance benchmarks for prediction model.
3. **Ideate:**
    - Consider machine learning algorithms, incorporate real-time data sources, or develop predictive models based on weather patterns, market dynamics, and energy supply.
4. **Prototype:**
    - Develop prototype models or software tools for electricity price prediction with varying features and data inputs.

- **Data Format**: Common formats include CSV, Excel, or structured databases. Ensure that the data can be easily imported into the chosen data analysis tools or programming language.

5. **Test:**
    - Gather feedback from relevant beneficiaries, such as energy analysts, traders, and utility companies to provide insights on the usability and effectiveness of your prototypes.
    - Use the feedback to refine and improve the electricity price prediction models.
    - Continuously test and validate models using historical and real-time data to ensure they are accurate and reliable.
    - iterate and adjust your models and approaches based on the feedback and insights.

Dataset used:

Dataset: https://www.kaggle.com/datasets/chakradharmattapalli/electricity-price-prediction/

Predicting electricity prices is complex and involves various factors that influence price fluctuations, the key components and considerations in electricity price prediction:

1. **Historical data:**
   This data can include hourly or sub-hourly price observations over an extended period, often spanning months or years.

	DateTime	Holiday	HolidayFlag	DayOfWeek	WeekOfYear	Day	Month	Year	PeriodOfDay	Fore
0	01/11/2011 00:00	None	0	1	44	1	11	2011	0	
1	01/11/2011 00:30	None	0	1	44	1	11	2011	1	
2	01/11/2011 01:00	None	0	1	44	1	11	2011	2	

2. **Factors Affecting Prices:**

- Supply and Demand: Fluctuations in supply and demand for electricity can have a significant impact on prices due to High demand during peak hours or supply shortages can lead to price spikes.

- Weather Conditions: Weather plays a critical role in electricity pricing, affecting both supply (e.g., renewable energy generation) and demand (e.g., heating or cooling needs).
- Fuel Prices: The cost of fuels used in power generation, such as natural gas or coal, can directly impact electricity prices.
- Regulatory Policies: Government policies, subsidies, and regulations can influence prices.
- Market Dynamics: Competitive electricity markets and trading practices can also impact pricing.

3. **Data Features:**
    - Uses features or input variables, such as historical price data, weather data (temperature, wind speed, etc.), market data, and more, to make predictions.
4. **Modeling Techniques:**
    - Machine learning and statistical modeling techniques are commonly used to build electricity price prediction models. Time series analysis, regression, neural networks, and ensemble methods like random forests are often applied.
5. **Evaluation Metrics:**
    - Success criteria for electricity price prediction models are based on specific evaluation metrics, such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), or other accuracy and reliability measures.

Data Processing:

1. Import the libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM
from statsmodels.tsa.arima.model import ARIMA
from sklearn.model_selection import train_test_split
import seaborn as sns
from sklearn.metrics import mean_absolute_error, mean_squared_error, mean_absolute_percentage_
```

2. Load the dataset:

```
# Load your time series data (replace 'your_data.csv' with the path to your dataset)
data = pd.read_csv("/content/Electricity (1) (1) (1).csv")
```

3. Display the first five rows of the dataset:

```
<bound method NDFrame.head of                DateTime         Holiday  HolidayFlag  DayOfWeek  
WeekOfYear  \
0      01/11/2011 00:00          None            0         1          44
1      01/11/2011 00:30          None            0         1          44
2      01/11/2011 01:00          None            0         1          44
3      01/11/2011 01:30          None            0         1          44
4      01/11/2011 02:00          None            0         1          44
```

4. Display the information of the dataframe:
    - The number of non-null entries in each column.
    - The data type of each column (e.g., int64, float64, object).
    - The memory usage of the DataFrame.
    - The number of columns.
    - The names of the columns.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38014 entries, 0 to 38013
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   DateTime                38014 non-null  object
 1   Holiday                 38014 non-null  object
 2   HolidayFlag             38014 non-null  int64 
 3   DayOfWeek               38014 non-null  int64 
 4   WeekOfYear              38014 non-null  int64 
 5   Day                     38014 non-null  int64 
 6   Month                   38014 non-null  int64 
 7   Year                    38014 non-null  int64 
 8   PeriodOfDay             38014 non-null  int64 
 9   ForecastWindProduction  38014 non-null  object
 10  SystemLoadEA            38014 non-null  object
 11  SMPEA                   38014 non-null  object
 12  ORKTemperature          38014 non-null  object
 13  ORKWindspeed            38014 non-null  object
 14  CO2Intensity            38014 non-null  object
 15  ActualWindProduction    38014 non-null  object
 16  SystemLoadEP2           38014 non-null  object
 17  SMPEP2                  38014 non-null  object
dtypes: int64(7), object(11)
memory usage: 5.2+ MB
```

5. Coorelation matrix:
    - Creates a correlation matrix of the data using the Pearson correlation coefficient. The Pearson correlation coefficient measures the linear correlation between two variables. A correlation of 1 indicates a

perfect positive correlation, a correlation of -1 indicates a perfect negative correlation, and a correlation of 0 indicates no correlation.

- Variables that are highly correlated with electricity prices.
- Variables that are highly correlated with each other.
- Relationships between variables that may not be immediately obvious.

6. Input Data:

   Day: The day of the month.
   Month: The month of the year.
   ForecastWindProduction: The forecast wind production for the day.
   SystemLoadEA: The forecast system load for the day.
   SMPEA: The forecast electricity price for the day.
   ORKTemperature: The observed temperature for the day.
   ORKWindspeed: The observed wind speed for the day.
   CO2Intensity: The carbon dioxide intensity of the electricity generation for the day.
   ActualWindProduction: The actual wind production for the day.
   SystemLoadEP2: The actual system load for the day.

```
x = data[["Day", "Month", "ForecastWindProduction", "SystemLoadEA",
"SMPEA", "ORKTemperature", "ORKWindspeed", "CO2Intensity",
"ActualWindProduction", "SystemLoadEP2"]]
y = data["SMPEP2"]
```

7. Feature scaling:
   - It is a data preprocessing technique that is used to normalize the features of a dataset. This is important for machine learning models because it helps to ensure that all of the features are treated equally by the model.
8. Evaluation Metrics:
   - Mean absolute error (MAE):
   - Mean squared error (MSE):
   - Mean absolute percentage error (MAPE):
   - R^2 score

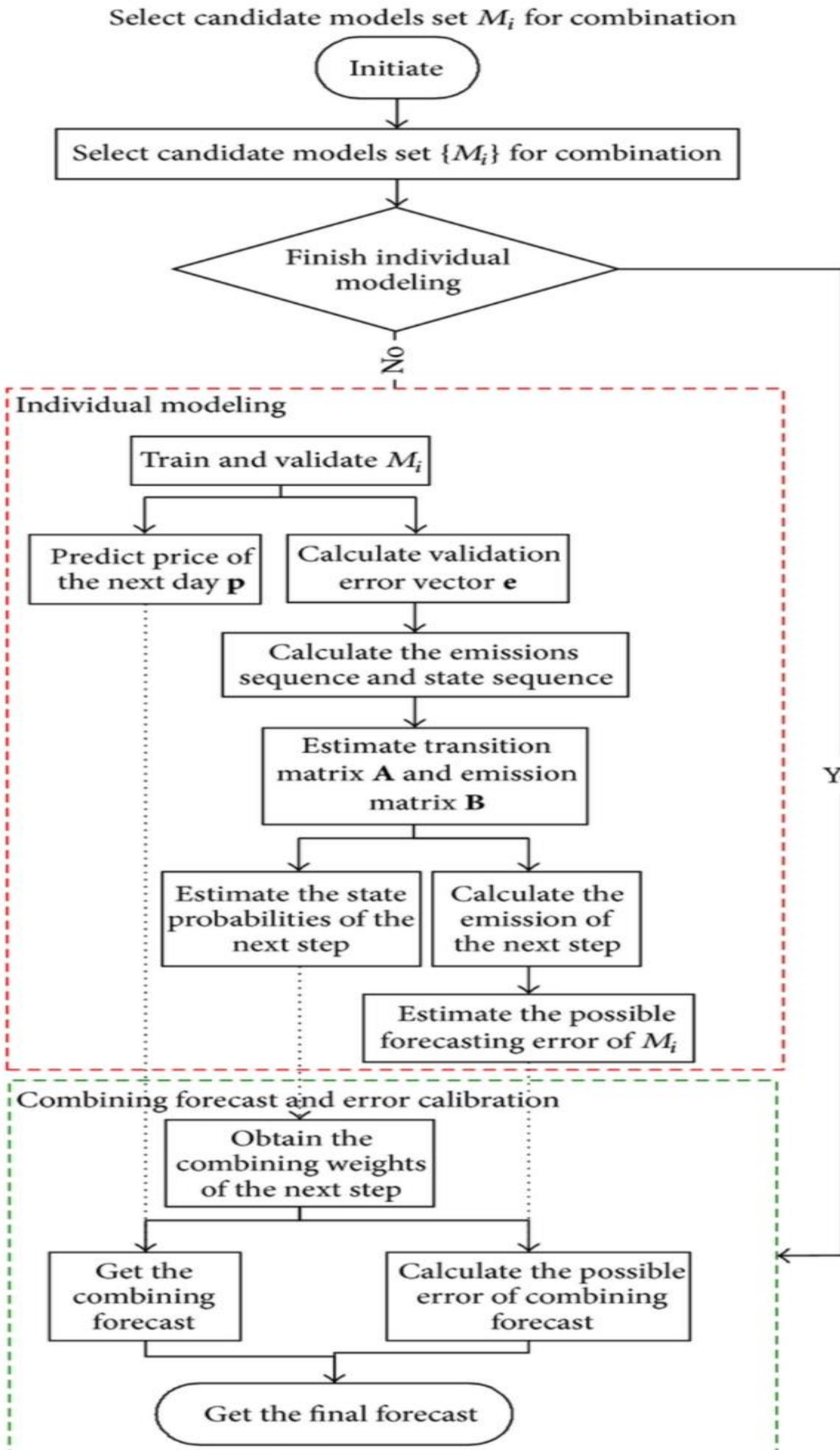
Time Series Forecasting algorithm:

Time series forecasting algorithms are used to predict future values of a time series based on its past values. There are many different time series forecasting algorithms, each with its own strengths and weaknesses.

The choice of time series forecasting algorithm for electricity prices prediction depends on a number of factors.

- **ARIMA (Autoregressive Integrated Moving Average) models:** class of linear time series forecasting models that are based on the assumption that the future values of the time series can be predicted from its past values and errors. ARIMA models are relatively simple to implement and interpret, and they can be effective for forecasting stationary time series.
- **SARIMA (Seasonal Autoregressive Integrated Moving Average) models:** They are a type of ARIMA model that is specifically designed for forecasting seasonal time series. SARIMA models can be effective for forecasting electricity prices, which often exhibit seasonal patterns.
- **GARCH (Generalized Autoregressive Conditional Heteroskedasticity) models:** They are a class of time series forecasting models that are designed to forecast time series with volatile error variances. GARCH models can be effective for forecasting electricity prices, which can be volatile.
- **Exponential smoothing models:** Exponential smoothing models are a class of time series forecasting models that are based on the assumption that the future values of the time series are a weighted average of its past values. Exponential smoothing models are relatively simple to implement and interpret, and they can be effective for forecasting both stationary and non-stationary time series.
- **Machine learning models:** Machine learning models, such as support vector machines, decision trees, and random forests, can also be used to forecast electricity prices. Machine learning models can be

more accurate than traditional time series forecasting models, but they are also more complex to train and deploy.



Evaluation metrics:

- Mean absolute error (MAE): average of the absolute differences between the predicted values and the actual values. It is a good measure of the overall accuracy of the model, and it is easy to interpret.
- Mean squared error (MSE): average of the squared differences between the predicted values and the actual values. Root mean squared error (RMSE): The RMSE is the square root of the MSE. It is a good measure of the magnitude of the error, and it is often used to compare the performance of different machine learning models.
- Mean absolute percentage error (MAPE): average of the absolute percentage differences between the predicted values and the actual values. It is a good measure of the relative accuracy of the model, and it is often used to evaluate the performance of forecasting models.
- R^2 score: The R^2 score is a measure of the goodness of fit of the model. It ranges from 0 to 1, with higher values indicating a better fit. An R^2 score of 1 indicates that the model perfectly fits the data.

```
Mean Absolute Error (MAE): 20.74485129895697
Mean Squared Error (MSE): 1301.282173838463
Root Mean Squared Error (RMSE): 36.073288924611006
Mean Absolute Percentage Error (MAPE): 149207971484931.8
R² Score: -0.0018503434967058752
```

Reference:

- Collab: https://colab.research.google.com/drive/1DA8txYdHVLOMotL4SGruHnsNsOLTuVJj#scrollTo=6MuZXYfBA350
- Dataset: https://www.kaggle.com/datasets/chakradharmattapalli/electricity-price-prediction/