ELECTRICITY DEMAND AND PRICE FORECASTING

INTERNSHIP PROJECT REPORT

Submitted by

KEERTHIVASAN S

To

INFOSYS SPRINGBOARD

NOV 2024

TABLE OF CONTENT

ABSTRACT

Accurate forecasting is a critical component in effective resource management and operational efficiency across various domains. This project focuses on utilizing advanced machine learning and deep learning techniques to enhance predictive accuracy. Traditional methods often struggle to handle the complexities of dynamic and large-scale data, leading to suboptimal outcomes. To address these challenges, this project evaluates the performance of models such as GRU, LSTM, CNN, and hybrid approaches like GRU-XGBoost and LSTM-Attention-XGBoost.

The models were tested on normalized datasets, with Mean Absolute Error (MAE) used as the primary metric for evaluation. The results show that hybrid models significantly improve predictive performance, achieving lower MAE scores compared to standalone machine learning and deep learning methods. For instance, hybrid approaches demonstrated superior accuracy, making them more effective in handling complex patterns and dependencies in the data.

These findings highlight the potential of integrating multiple advanced techniques to overcome the limitations of traditional predictive methods. The insights gained from this work can be applied in various fields, including resource planning, operations management, and decision-making processes, paving the way for smarter and more efficient systems.

Keywords: Forecasting, Machine Learning, Deep Learning, GRU, LSTM, CNN, Hybrid Models, XGBoost, Predictive Analytics, Mean Absolute Error (MAE), Resource Management.

ELECTRICITY DEMAND AND PRICE FORECASTING

1. INTRODUCTION

Efficient and accurate forecasting of electricity consumption is critical in modern energy systems. This project is designed to leverage advanced machine learning and deep learning models to predict energy demand more precisely. These predictions help utilities, grid operators, and policymakers optimize resource allocation, reduce operational costs, and ensure energy stability.

The project addresses the limitations of traditional forecasting methods, such as those used by Transmission System Operators (TSOs), by integrating state-of-the-art techniques. Various models, including GRU, LSTM, CNN, hybrid approaches like GRU-XGBoost, and LSTM-Attention-XGBoost, were employed to analyze and predict energy consumption patterns. These models were trained and evaluated using historical electricity consumption data, with Mean Absolute Error (MAE) serving as the primary performance metric.

The hybrid models demonstrated superior performance, achieving lower MAE scores compared to standalone machine learning and deep learning methods. For example, the Hybrid GRU-XGBoost model achieved an MAE of 0.014, significantly outperforming the TSO predictions, which had an MAE of 0.070. This improvement underscores the effectiveness of combining advanced deep learning techniques with robust machine learning frameworks.

By enhancing forecasting accuracy, this project contributes to reducing energy waste, optimizing grid operations, and supporting the integration of renewable energy sources. The results and insights from this project offer a promising direction for further research and practical applications in the energy sector, enabling smarter, more sustainable energy management.

2. ABOUT DATASET

The dataset used in this project is crucial for training, validating, and testing the forecasting models. It comprises historical data capturing key metrics related to the target prediction task. The following are the essential characteristics and details of the dataset:

1. Structure of the Dataset

- **Features (Independent Variables):**
    - The dataset contains features that influence the target variable. These may include:
        - Temporal features like date, time, or seasonality indicators.
        - External factors such as weather conditions, holidays, or events.
        - Historical consumption or demand data from previous time steps.
- **Target Variable (Dependent Variable):** The variable to be predicted, such as energy demand or consumption levels.

2. Size of the Dataset

- The dataset consists of a substantial number of records, providing sufficient data for training and evaluation. The size allows for robust model training and reduces the risk of overfitting or underfitting.

3. Data Preprocessing

- **Normalization:**
    - The raw data is normalized to ensure all features are scaled to similar ranges, making it suitable for machine learning and deep learning models. This step improves convergence and stability during training.
- **Handling Missing Values:**
    - Any missing data points were handled through interpolation or removal to maintain dataset integrity.
- **Feature Engineering:**
    - Additional features, such as lagged values or moving averages, were created to enhance the dataset's predictive power.

4. Splitting the Dataset

- **Training Set:** Used to train the models on historical data.
- **Validation Set:** Used to tune hyperparameters and prevent overfitting.
- **Test Set:** Used to evaluate the final model performance on unseen data.

5. Time Resolution

- The dataset has a specific time resolution, such as hourly, daily, or monthly records, depending on the problem statement. This resolution determines the granularity of the predictions.

6. Domain Relevance

- The dataset is domain-specific and tailored to the forecasting task. It contains variables directly impacting the prediction outcome, ensuring meaningful and accurate model development.

This dataset forms the foundation of the project, allowing for effective model training, evaluation, and comparison. Proper preprocessing and understanding of the dataset are critical to achieving accurate and reliable results.

3. PREPROCESSING TECHNIQUES

The Data preprocessing techniques involves several processes, that can be listed below:

1. Data Loading

- **Objective:** Load the raw dataset into the workspace.
- **Steps:**
    - The dataset is imported into a pandas DataFrame.
    - The initial structure of the dataset is inspected using `.head()` and `.info()` to understand the columns, data types, and presence of missing values.

2. Handling Missing Values

- **Objective:** Address any incomplete or inconsistent data points in the dataset.
- **Steps:**
    - Missing values are identified using `.isnull().sum()`.
    - Common techniques such as imputation (mean/median filling) or interpolation are applied to fill missing values.
    - Rows with critical missing data (non-recoverable) are dropped if necessary to maintain data quality.

3. Feature Engineering

- **Objective:** Create new, meaningful features to improve model accuracy.
- **Steps:**
    - Temporal features are extracted from date-time columns, including:
        - Hour, Day of the Week, Month, or Season.
    - Rolling statistics (e.g., moving averages) are calculated for smoothing or capturing trends.
    - Lagged values of the target variable are added to capture temporal dependencies.
    - External features (if provided, such as weather data or holidays) are merged with the main dataset.

4. Data Normalization

- **Objective:** Scale numerical features to ensure uniformity and improve model performance.
- **Steps:**
    - The MinMaxScaler or StandardScaler from sklearn is applied to scale data between 0 and 1 or standardize it with a mean of 0 and standard deviation of 1.
    - Scaling ensures that no single feature disproportionately influences the model.

5. Data Splitting

- **Objective:** Divide the dataset for training, validation, and testing purposes.
- **Steps:**
  - Data is split into:
    - **Training Set:** To train the model.
    - **Validation Set:** To tune hyperparameters and evaluate during training.
    - **Test Set:** To evaluate the final model performance on unseen data.
  - The splitting is performed chronologically to avoid data leakage in time-series forecasting.

6. Data Transformation for Models

- **Objective:** Reshape or prepare data as required for specific models.
- **Steps:**
  - For recurrent neural networks (RNNs) like LSTM and GRU:
    - The data is reshaped into a 3D array (samples, timesteps, features) format.
  - For hybrid models like GRU-XGBoost:
    - Separate datasets are prepared for the GRU and XGBoost components, ensuring each receives the appropriate input format.

7. Target Variable Normalization

- **Objective:** Normalize the target variable to match the feature scaling.
- **Steps:**
  - The target variable (e.g., electricity demand) is scaled using the same scaler applied to the features.
  - Inverse scaling is applied after predictions to interpret the results in the original scale.

8. Data Inspection and Visualization

- **Objective:** Understand the dataset distribution and identify patterns.
- **Steps:**
  - Visualizations are generated to inspect feature distributions, trends, and correlations.
  - Plots such as line charts (for time-series trends) and heatmaps (for correlation analysis) are used.

9. Final Dataset Preparation

- **Objective:** Prepare the finalized datasets for model input.
- **Steps:**
  - Cleaned, engineered, and scaled data is exported into variables or files for the training pipeline.

4. ABOUT THE MODELS

In this project, multiple machine learning and deep learning models were used to predict electricity demand. Here's a detailed explanation of each model and how they work:

1. Transmission System Operator (TSO) Prediction

- **What It Is:**
  This is the baseline forecasting approach used by energy Transmission System Operators. It typically relies on statistical methods or simple historical averages to predict electricity demand.
- **How It Works:**
  - Uses past demand data to calculate trends, seasonal patterns, and averages.
  - Provides forecasts based on historical data without advanced feature extraction or deep learning capabilities.
- **Limitations:**
  - Does not handle complex, non-linear patterns in data.
  - Performs poorly compared to machine learning models for dynamic scenarios.

2. XGBoost (Extreme Gradient Boosting)

- **What It Is:**
  XGBoost is an ensemble learning method that uses boosted decision trees for regression and classification tasks.
- **How It Works:**
  - Builds a series of decision trees sequentially.
  - Each new tree corrects the errors of the previous trees by focusing on the residuals (difference between actual and predicted values).
  - Combines the predictions of all trees for the final output.
- **Strengths:**
  - Handles missing data well.
  - Highly efficient and fast.
  - Works well with structured/tabular data.
- **Applications in Forecasting:**
  Can model relationships between various features and predict outcomes with high accuracy.

3. GRU (Gated Recurrent Unit)

- **What It Is:**
  GRU is a type of recurrent neural network (RNN) designed for sequential data. It is simpler than LSTM but equally powerful for many tasks.
- **How It Works:**
  - Processes input sequences step-by-step, maintaining a hidden state to capture temporal dependencies.
  - Uses gates (reset and update gates) to control how much information to keep or discard at each time step.
- **Strengths:**
  - Efficient for time-series data.
  - Reduces computational overhead compared to LSTM.
- **Applications in Forecasting:**
  Ideal for capturing patterns in electricity demand over time.

4. LSTM (Long Short-Term Memory)

- **What It Is:**
  LSTM is a type of RNN specifically designed to learn long-term dependencies in sequential data.
- **How It Works:**
  - Introduces memory cells that retain information over long periods.
  - Uses three gates (input, forget, and output gates) to control the flow of information.
  - Allows the model to focus on relevant parts of the sequence while ignoring irrelevant parts.
- **Strengths:**
  - Excellent for long-range dependencies in time-series data.
  - Reduces issues like vanishing gradients common in traditional RNNs.
- **Applications in Forecasting:**
  Useful for capturing seasonal trends and long-term patterns in electricity demand.

5. CNN (Convolutional Neural Network)

- **What It Is:**
  CNNs are primarily used for image processing but can also be applied to time-series data for feature extraction.
- **How It Works:**
  - Applies convolutional layers to extract features from input data.
  - Captures local patterns and dependencies effectively.
  - Uses pooling layers to reduce dimensionality and enhance efficiency.
- **Strengths:**
  - Fast and efficient for extracting spatial or temporal features.
  - Works well with structured time-series data.
- **Applications in Forecasting:**
  Can identify patterns and relationships in time-series data, such as peak demand times.

6. CNN-LSTM

- **What It Is:**
  A hybrid model combining CNN for feature extraction and LSTM for sequence modeling.
- **How It Works:**
  - CNN extracts features from the input data.
  - LSTM processes these extracted features over time to capture temporal dependencies.
- **Strengths:**
  - Combines the strengths of both CNN and LSTM.
  - Captures both spatial and temporal patterns effectively.
- **Applications in Forecasting:**
  Useful for modeling complex patterns in time-series data.

7. LSTM-Attention

- **What It Is:**
  An extension of LSTM with an attention mechanism to focus on relevant parts of the sequence.
- **How It Works:**
  - LSTM processes the sequence to capture dependencies.
  - The attention mechanism assigns weights to specific time steps, emphasizing their importance in predictions.
- **Strengths:**
  - Enhances the interpretability of the model by highlighting critical inputs.
  - Improves performance by focusing on relevant parts of the sequence.
- **Applications in Forecasting:**
  Identifies and prioritizes significant patterns or time periods in electricity demand.

8. Hybrid GRU-XGBoost

- **What It Is:**
  A hybrid model combining GRU for feature extraction and XGBoost for final predictions.
- **How It Works:**
  - GRU processes the input sequence to extract temporal features.
  - XGBoost uses the extracted features to make the final predictions.
- **Strengths:**
  - Combines the sequence modeling strength of GRU with the structured data handling capabilities of XGBoost.
  - Provides high accuracy and robustness.
- **Applications in Forecasting:**
  Captures temporal patterns while leveraging XGBoost's efficiency.

9. Hybrid LSTM-Attention-XGBoost

- **What It Is:**
  A hybrid model integrating LSTM with attention and XGBoost for robust and accurate predictions.
- **How It Works:**
  - LSTM processes the input sequence to capture dependencies.
  - The attention mechanism identifies and emphasizes critical parts of the sequence.
  - XGBoost uses the processed features for final predictions.
- **Strengths:**
  - Combines LSTM's temporal modeling, attention's focus mechanism, and XGBoost's efficiency.
  - Achieves high accuracy and interpretability.
- **Applications in Forecasting:**
  Particularly effective in identifying significant time periods and leveraging them for precise predictions.

Inference Obtained from the Models

- **Standalone Models (XGBoost, GRU, LSTM, CNN):** Effective for individual tasks but may not fully capture the complexities of electricity demand.
- **Hybrid Models:** Combine the strengths of multiple approaches, delivering superior performance and accuracy.

5. RESULTS

The evaluation of various models was conducted using the Mean Absolute Error (MAE) metric to compare their predictive accuracy. The key findings are as follows:

- **TSO Prediction:** Baseline model with an MAE of **0.070**.
- **XGBoost:** Achieved a significantly lower MAE of **0.016**, demonstrating better performance than the baseline.
- **GRU:** Performed slightly better with an MAE of **0.015**.
- **LSTM:** Achieved an MAE of **0.018**, showcasing its ability to capture long-term dependencies effectively.
- **CNN:** Yielded an MAE of **0.025**, indicating reasonable accuracy for feature extraction-based forecasting.
- **CNN-LSTM:** Improved performance with an MAE of **0.019**, leveraging the strengths of both architectures.
- **LSTM-Attention:** Matched GRU with an MAE of **0.015**, highlighting the benefit of attention mechanisms in identifying critical patterns.
- **Hybrid GRU-XGBoost:** Delivered the best results with an MAE of **0.014**, showcasing the advantage of combining sequential learning and gradient boosting.
- **Hybrid LSTM-Attention-XGBoost:** Also achieved an MAE of **0.015**, demonstrating robust performance through hybridization.

6. CONCLUSION

This study demonstrates the effectiveness of advanced machine learning and deep learning models for electricity demand forecasting. While traditional TSO prediction methods offer a baseline, modern models like GRU, LSTM, and XGBoost significantly enhance forecasting accuracy. Furthermore, hybrid architectures, such as GRU-XGBoost and LSTM-Attention-XGBoost, emerge as superior solutions by combining the strengths of multiple approaches.

Key Takeaways:

1. Hybrid models excel by capturing both temporal dependencies and non-linear relationships in the data.
2. The integration of attention mechanisms improves interpretability and ensures focus on critical data points.
3. The findings emphasize the potential of hybrid architectures in addressing complex forecasting challenges, surpassing traditional methods and standalone models.
4. This work underscores the importance of leveraging state-of-the-art techniques to meet the growing demands of precision in energy management systems.

7. REFERENCES

1. OpenWeatherMap API: https://openweathermap.org/api

2. ENTSOE Transparency Platform: https://transparency.entsoe.eu/dashboard/show

3. Spanish TSO: https://www.esios.ree.es/en/market-and-prices?date=27-03-2023#

4. GitHub Repo : https://github.com/KeerthiVasan-ai/Electricity_Demand_and_Price_Forecasting