ECPX: Empowering Commodity Price Prediction Using XGBoost Algorithm

1. Introduction

Commodities are often the building blocks of the global economy and are categorized into several broad groups like energy, metals, financial, and agriculture. The foundation of industrial production and economic growth rests on eight core industries [1]. These essential sectors, as measured by the Index of Eight Core Industries (ICI), encompass coal, electricity generation, crude oil production, cement, natural gas, steel, refinery products, and fertilizer production [2]. Core industries collectively contribute 40.27% to India's Index of Industrial Production (IIP). Commodity prices in these sectors are influenced by factors like supply and demand, weather, geopolitics, and economic trends [3]. Accurate prediction of commodity prices is crucial for economic planning, inflation control, efficient supply chains, and informed investments, impacting energy sector stability, consumer budgets, and environmental considerations. The dynamic and volatile nature of commodity markets has sparked interest in predictive modeling, providing consumers, investors, and policymakers with reliable insights for budget planning, portfolio strategies, and effective economic policies [4].
In this paper, our goal is to develop predictive models for petrol and electricity prices, aiming to provide valuable insights into essential commodity price forecasting. Following recommendations from a previous study [5], we have chosen to utilize the XGBoost technique due to its demonstrated accuracy in predicting prices, such as gold futures [6]. Our research focuses on improving commodity price prediction accu- racy through careful input data selection and model choice, and based on our previous findings, we have opted for the XGBoost model [7].

2. Related Approaches

Historically, commodity price predictions have relied on traditional economic models and expert opinions. The emergence of machine learning and access to extensive datasets have transformed commodity price prediction. Traditional methods like time-series analysis using ARIMA and regression models have been common [8], but they may struggle to account for external factors and non-linear price changes in essential commodities like crude oil, petrol, electricity [9], and carbon price prediction [10]. XGBoost algorithm as a feature-selection technique can be used to extract features from high-dimensional time-series data [11]. Kamal Gulati's study focuses on predicting crude oil prices using Artificial Neural Networks (ANN) and a hybrid model (ANN-PSO) due to the increasing volatility caused by pandemics [12]. The primary potential drawback in the paper could be the limited ability of the ANN and ANN-PSO models to account for complex external factors and changing market dynamics beyond the dataset's timeframe.
[13] aims to assist rural Indian farmers with technology-driven price estimation for their crops. With the use of Decision Trees, Random Forest, and Linear Regression, the model predicts commodities prices with an accuracy of above 95%. Improving data for stock market prediction may increase accuracy [14]. A recurrent Neural Net- work model has been proposed by M. Chaitanya Lahari, emphasizes the importance

of forecasting fuel prices and its profound impact on various aspects of society and the economy to predict non-linear variations in fuel prices in major Indian cities, considering factors like international oil prices and exchange rates [15]. Shilong proposed an efficient and accurate sales forecasting model using machine learning[16]. K. Likitha's research focuses on predicting electricity prices, essential for understanding consumption and costs in the digital age[17]. Machine learning algorithms, including Random Forest, Logistic Regression, Support Vector, and Artificial Neural Network, are employed for regression analysis. The study finds that the ANN Regressor performs the best in electricity price prediction.

3. Proposed System

This section presents structured procedures for the system's use of data, model development and testing, and prediction visualization. Effective data preprocessing is key to ensuring that the data collected from a range of sources is accurate, uniform, and ready for model training. The process begins with data collection from various sources with different formats such as PDFs, CSV files, web data, as described in the Data and Variables section. Once the data is obtained, it needs to be accurately ex- tracted, and tools like Tabula or pdfplumber can be used to extract structured data from PDF files. After that, data cleaning is performed, which involves dealing with missing values, removing duplicates, and identifying and possibly handling outliers that may affect model performance. Data integration is critical, when merging data from different sources into a single dataset, requires alignment in terms of periods. Feature engineering is an important step in creating new features, such as lag vari- ables, rolling statistics, or percentage changes, to help the model detect important patterns.

3.1. Data and Variables

The system worked on historical data related to two important commodities-petrol (gasoline or motor spirit) and electricity, with a focus on the Andhra Pradesh region. This approach makes the data more regionally relevant and offers valuable insights. The dataset consists of variables that cover a wide range of economic, financial, and geopolitical factors. For instance, to predict petrol prices, variables such as yearly crude oil prices, currency exchange rates, refining costs, and government taxation policies are considered. Similarly, to predict electricity prices, coal prices, local tariffs, and energy source mix data are considered.

The data analyzed in the study includes petrol prices from January 2016 to March 2022 and electricity prices from May 2000 to March 2022. The main area of focus is the price, which is the variable aimed to predict. Various factors that affect petrol prices, such as Crude Oil, USD/INR exchange rate, supply and demand dynamics, taxes and duties, and the annual inflation rate are considered. The system analysis also extends to predicting electricity

prices in Andhra Pradesh, with a specific emphasis on Electricity Prices as the target variable. Key factors that affect electricity pricing include Tariff Rates, Coal Price, and Supply and Demand dynamics. Table 1 and 2 provide data from reliable sources to support the predictive model and offer tailored forecasts for the Andhra Pradesh region. Through this research, proposed system aim's to contribute to enhanced energy management and data-driven decision-making in the local electricity market.

Table 1. Data and Variables of Petrol (Gasoline)

Variables	Signification	Source
Price	Petrol Price	https://mopng.gov.in/
Crude_Oil	Crude Oil Price	https://in.investing.com/
USD_INR	Exchange Rates	https://in.investing.com/
SD	Supply and Demand	https://mopng.gov.in/
Taxes	Taxes and Duties	https://mopng.gov.in/
Deal_Com	Dealer Commission	https://mopng.gov.in/
Inflation	Annual Inflation	https://macrotrends.net/

Table 2. Data and Variables of Electricity

Variables	Signification	Source
Price	Electricity Price	https://www.apspdcl.in/
Tariff	Tariff rates	https://www.apspdcl.in/
Coal	Coal Price	https://in.investing.com/
SD	Supply and Demand	https://cea.nic.in/

3.2. XGBoost Algorithm

XGBoost is a machine-learning technique that was created by Tianqi Chen in 2014. It is highly regarded for its exceptional predictive accuracy and robustness. This ensemble method aggregates predictions from weak learners, usually decision trees, to form a precise model. Unlike single decision tree models, XGBoost efficiently ensembles them and can use parallel processing to speed up the process. It offers control over tree depth (max_depth) for effective pruning and reduced overfitting. XGBoost can handle missing data and supports built-in cross-validation, which makes model tuning easier. The

algorithm starts with an initial prediction, iteratively creates decision trees to correct mistakes, and optimizes a specific loss function using gradient descent. It continues adding trees until it meets a stopping criterion, as depicted in Fig 1.

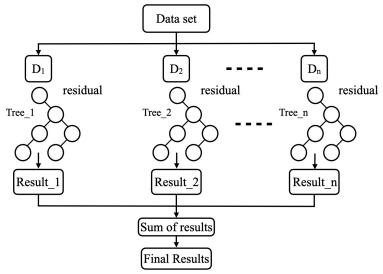

Fig. 1. Working of XGBoost Algorithm

XGBoost works as Newton-Raphson in function space unlike gradient boosting which works as gradient descent in function space, a second-order Taylor approximation is used in the loss function to make the connection to Newton-Raphson method [18].

General approach :

input: A training set $\{(x_i, y_i)\}_{i=1}^{N}$, a loss function $L(y, F(x))$, number of weak learners and a learning rate .

Procedure:

1. Set the model's initial value to a constant:

$$\hat{f}_{(0)}(x) = \underset{x}{argmin} \sum_{i=1}^{N} L(y_i, \theta).$$

2. for = 1 to :

1. Calculate ‘gradients’ and ‘hessians’:

$$\hat{g}_z(x_i) = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=\hat{f}_{(z-1)}(x)}.$$

$$\hat{h}_z(x_i) = \left[\frac{\partial^2 L(y_i, f(x_i))}{\partial f(x_i)^2}\right]_{f(x)=\hat{f}_{(z-1)}(x)}.$$

2. Utilizing the training set, fit a base learner (or weak learner, such as a tree) $\left\{x_i, -\frac{\hat{g}_z(x_i)}{h_z(x_i)}\right\}_{i=1}^{N}$ by resolving the following optimization problem:

$$\hat{\phi}_z = \underset{\phi \in \Phi}{argmin} \sum_{i=1}^{N} \frac{1}{2} \hat{h}_z(x_i) \left[\phi(x_i) - \frac{\hat{g}_z(x_i)}{\hat{h}_z(x_i)} \right]^2 .$$

$$f_z(x) = \alpha \hat{\phi}_z(x) .$$

3. Revise the model:

$$\hat{f}_{(z)}(x) = \hat{f}_{(z-1)}(x) + f_z(x) .$$

3. Output:

$$\hat{f}(x) = \hat{f}_{(Z)}(x) = \sum_{z=0}^{Z} \hat{f}_z(x) .$$

3.3. System Design

The architecture of the system is illustrated in Fig. 2. The system initiates with the collection of data from diverse sources such as government databas- es, financial markets, and Kaggle datasets in various formats. Preprocessing of data is a crucial step and it encompasses cleaning, feature engineering, and encoding of categorical variables. Integration of data consolidates histor- ical price data, influencing factors, and external variables into a consistent dataset, ensuring uniformity in units and scales through normalization to prevent feature dominance in modeling.

Algorithm:

Input:

Petrol_data, Electricity_data

Number of loops (*N*) for experimentation

Procedure:

Divide the data into feature sets (*X*) and target variable (*y*)

Initialize models for petrol *(XG_Petrol)* and electricity *(XG_Elec- tricity)*

Initial reference error : $e_{0p} = 1$, $e_{0e} = 1$

for $i = 1$ to N

Formulate training sets specific to petrol and electricity prediction.

Train *XG_Petrol_i* and *XG_Electricity_i*

Compute *e_petrol_i* and *e_electricity_i* on validation sets

Calculate mean measure index *e_petrol* and *e_electricity*

if *e_petrol_i* < $e_{0p} = 1$ then

e_{0p} = e_ petrol_i

Update XG_Petrol to XG_Petrol_i
if $e\_electricity\_i < e_{0e} = 1$ then
e_{0e} = e_electricity_i
Update *XG_Electricity* to *XG_Electricity_i*
end for
Calculate prediction error *(e_ petrol, e_electricity)* and measure index (M_p, M_e)
Output: *XG_Petrol, XG_Electricity, e_ petrol, e_electricity,* M_p, M_e

The procedure explains how to train models to make predictions. Two models are created as a result: XG_Electricity and XG_Petrol. Metrics like MAE, MSE, RMSE, and R2 are used to assess the model's correctness. Iterating to identify the optimal model involves altering the dataset and adjusting parameters as needed. Finally, Table 3 presents the trained models together with the relevant errors and metrics.

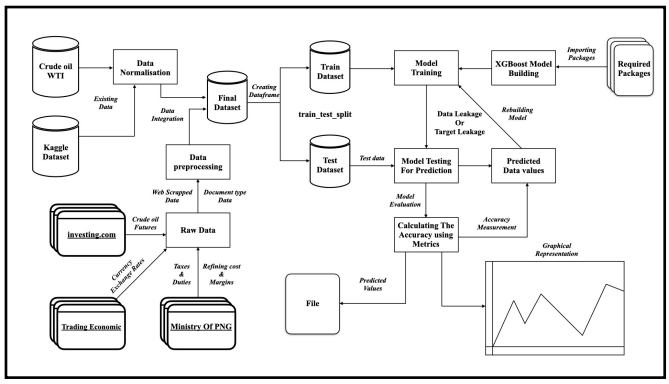

Fig. 2. Architecture of Petrol price prediction

Data quality assurance is an ongoing process to ensure that data sets remain reliable. Meanwhile, feature engineering helps to improve model performance by creating and transforming features. The train-test split technique is crucial in this process. It involves dividing the dataset into training and testing sub-datasets, typically 70-80% and 20-30% respectively. The training dataset trains the model, while the testing dataset evaluates its performance. This ensures that the model's ability to generalize to new, unseen data is assessed impartially. During the model training phase, the algorithm learns from historical data using the X_train dataset. Once the model has been trained, it can then be used to make predictions on new data. Model testing is then performed to assess its ability to generalize to unseen data, revealing any potential overfitting issues or real-world performance concerns. Model evaluation techniques are then employed to assess the model's strengths and

weaknesses, including hyperparameter fine-tuning aimed at maximizing performance metrics. Common regression evaluation metrics like MSE, RMSE, MAE, and R2 are used to measure model performance.

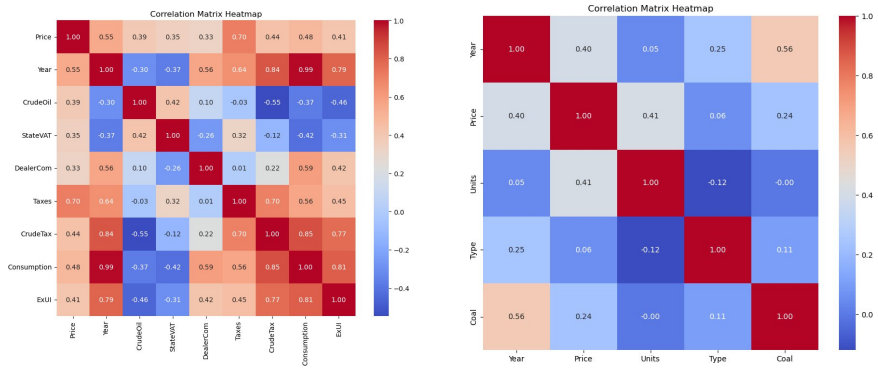

Fig. 3. Correlation matrices of Petrol and Electricity datasets

Data leakage and target leakage are two serious concerns in machine learning tasks, which can significantly impact the reliability of the model. Data leakage occurs when inadvertent future or external information influences the training dataset, leading to artificially high training performance but poor generalization to new data. Target leakage, on the other hand, arises when the features used in model training contain target variable information that is not accessible during prediction, resulting in inflated testing performance but poor generalization to new data. Both types of leakage can lead to models that perform well during development but fail in real-world scenarios or on new data.

Correlation matrix is an effective statistical tool for measuring and visualizing relationships between variables in a dataset, uncovering patterns and dependencies. Fig. 3 illustrates the correlations between features and target variable price for petrol and electricity datasets. These matrices are fundamental in feature selection and model building, identifying critical features with strong positive or negative correlations to the target variable. Correlation coefficients range from -1 to 1. Visualizations of testing data and predictions facilitate communication of model accuracy, enabling stakeholders to evaluate the dependability of predictions and make informed decisions in commodity markets.

4. Implementation and Results

The system utilized various Python libraries in the Jupyter Notebook environment to perform machine learning and data analysis work. The primary tools were NumPy for numerical operations and Pandas for data handling and organization. Scikit-Learn was crucial for machine learning tasks, with XGBoost serving as prediction algorithm. For

data visualization, Matplotlib and Seaborn libraries are used. This selection of tools streamlined workflow, enabling efficient data preprocessing, model training, and insightful visualizations, all of which were well-documented for reproducibility. To conduct the proposed research, the system works with data from diverse sources in different formats, such as .txt and .csv. Pandas library methods were used to read and create structured DataFrames from various file formats, facilitating efficient data manipulation and transformation. System used pandas file reader and writer methods to handle data in different formats, ensuring consistency for analysis. To create a comprehensive dataset containing necessary features and target variables, data is merged and integrated into single dataset and then exported it in CSV format for model training and testing. The next step involved parameter tuning for the XGBoost model. The model is trained with different parameters and features through an iterative process, which helped to identify the best model configuration. Validation sets were used to assess performance and prevent overfitting or underfitting. Mean Squared Error (MSE) and R-squared (R2) were used to evaluate model performance. To gain in- sights into the model's forecasting capabilities, sample predictions generated by the model were considered, and visualized data patterns using line plot. Test outputs, in- cluding MSE, R2, and sample predictions, were summarized in Table 5.

Table 3. Petrol Dataset

Date	Price	CrudeOil	StateVAT	CrudeTax	Consumption	ExUI
23-10-2021	113.54	81.78	5	0.103	1321.9	74.935
24-10-2021	113.89	81.78	5	0.103	1321.9	74.935
25-10-2021	113.89	81.78	5	0.103	1321.9	74.935
26-10-2021	114.24	81.78	5	0.103	1321.9	74.935
27-10-2021	114.24	81.78	5	0.103	1321.9	74.935

Table 3. displays sample dataset used to train the model, in that the target variable is price and is stored in Y. The remaining are features, which are stored in X. The proposed model has experimented with 5 years of historical data for fuel prices in Andhra Pradesh, India. For the petrol price prediction model, the data is set to have 80% as a training dataset and 20% as a testing dataset. The observed accuracy is above 95% on testing data. A similar procedure is applied to the electricity price prediction model, the correlation matrix of the electricity dataset is shown in Fig. 3. and the sample dataset format is shown in Table 4. The dataset contains 23 years of annual data of electricity prices, tariff rates, type of consumer and price of coal. The observed accu- racy is above 90% on testing data.

Table 4. Electricity Dataset

Year	Price	Units	Type	Coal
2022	9.05	81.78	5	0.103
2022	9.60	81.78	5	0.103
2022	10.15	81.78	5	0.103
2022	114.24	81.78	5	0.103
2022	114.24	81.78	5	0.103

Table 5. Parameter Optimization results

	Petrol	Electricity
Best Perms	[n_estimators=600, learning_rate=0.02, max_depth=3, random_state=42]	[n_estimators=500, learning_rate=0.02, max_depth=7, random_state=42]
Best R-Squared score	0.976514340012527	0.960227743127287
mean_square_error	2.05928441628387	0.291912879718121
y_test (first five)	[76.31, 92.42, 74.46, 74.65, 77.37]	[6.90, 6.90, 1.54, 2.60, 5.11]
y_pred	[76.844124, 91.427574, 76.45471 76.65402, 76.65402]	[6.923163, 6.564761, 2.5438266 2.332338, 5.305329]

The Petrol model demonstrates high accuracy with a low Mean Squared Error (MSE) of 2.0592, indicating close alignment between predictions and actual prices. Examining the test dataset, the model's predictions for the first five instances are in good agreement with the actual values. Similarly, the Electricity model exhibits accuracy, boasting a low MSE of 0.2919, signifying precise predictions that closely match actual prices for the initial five instances in the test dataset. Visualizations effectively illustrate model performance by comparing actual and predicted values, revealing strengths and areas for potential improvement, as shown in Fig. 4.

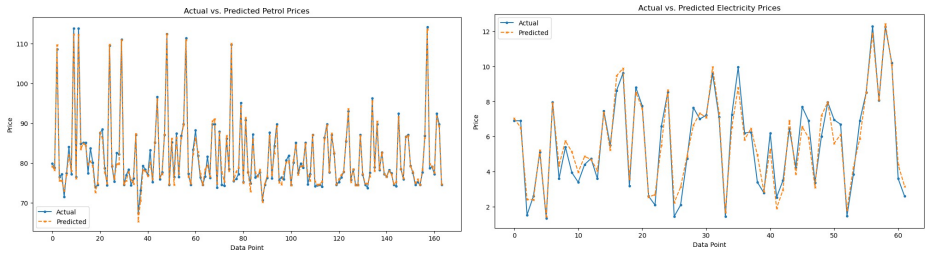

Fig. 4. Comparison between predicted vs actual petrol prices of Petrol and Electricity

5. Conclusion

Commodity price prediction, with a particular focus on petrol and electricity prices in Andhra Pradesh. The system analysis involved various stages, including data collection, preprocessing, feature engineering, model training, and evaluation, and system utilized the powerful XGBoost algorithm. The work findings suggest that machine learning is effective in forecasting commodity prices, as the XGBoost models exhibited impressive predictive accuracy, with R-squared (R2) scores of 0.9765 for petrol and 0.9459 for electricity, indicating their ability to capture complex relationships between variables. Additionally, the predictions demonstrated high precision, as reflected by the low Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) values. This research has practical significance for the commodities market in Andhra Pradesh, benefiting various stakeholders such as investors, policymakers, and energy consumers by improving energy management, economic planning, and investment decisions. Ongoing commitments to improving commodity price prediction, such as integrating data sources and utilizing alternative modeling techniques, will further enhance the project in the future.

5.1. Future Enhancement

This research offers valuable insights into predicting petrol and electricity prices in Andhra Pradesh. Future enhancements could involve integrating alternative data sources like satellite imagery and social media sentiment analysis, exploring advanced machine learning techniques, and expanding the project's scope to cover other critical commodities and a broader geographical region, further improving prediction accuracy and applicability.

References

1. Chen, Z., Goh, H. S., Sin, K. L., Lim, K., Chung, N. K. H., & Liew, X. Y. (2021). Automated agriculture commodity price prediction system with machine learning techniques. arXiv preprint arXiv:2106.12747.

2. Herrera, G. P., Constantino, M., Tabak, B. M., Pistori, H., Su, J. J., & Naranpanawa, A. (2019). Long-term forecast of energy commodities price using machine learning. Energy, 179, 214-221.
3. Zhao, H. (2021). Futures price prediction of agricultural products based on machine learning. Neural computing and applications, 33, 837-850.
4. Yun, K. K., Yoon, S. W., & Won, D. (2021). Prediction of stock price direction using a hybrid GA-XGBoost algorithm with a three-stage feature engineering process. Expert Systems with Applications, 186, 115716.
5. Avanijaa, J. (2021). Prediction of house price using xgboost regression algorithm. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(2), 2151-2155.
6. Jabeur, S. B., Mefteh-Wali, S., & Viviani, J. L. (2021). Forecasting gold price with the XGBoost algorithm and SHAP interaction values. Annals of Operations Research, 1-21.
7. Zhao, X., Li, Q., Xue, W., Zhao, Y., Zhao, H., & Guo, S. (2022). Research on ultra-short-term load forecasting based on real-time electricity price and window-based XGBoost model. Energies, 15(19), 7367.
8. Yucong, W., & Bo, W. (2020, April). Research on EA-xgboost hybrid model for building energy prediction. In Journal of Physics: Conference Series (Vol. 1518, No. 1, p. 012082). IOP Publishing.
9. Lu, H., Ma, X., Ma, M., & Zhu, S. (2021). Energy price prediction using data-driven models: A decade review. Computer Science Review, 39, 100356.
10. Sun, W., & Zhang, J. (2020). Carbon price prediction based on ensemble empirical mode decomposition and extreme learning machine optimized by improved bat algorithm considering energy price factors. Energies, 13(13), 3471.
11. Vuong, P. H., Dat, T. T., Mai, T. K., & Uyen, P. H. (2022). Stock-price forecasting based on XGBoost and LSTM. Computer Systems Science & Engineering, 40(1).
12. Gulati, K., Gupta, J., Rani, L., & kumar Sarangi, P. (2022, October). Crude Oil Prices Predictions in India Using Machine Learning based Hybrid Model. In 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 1-6). IEEE.
13. Rani, S., Kumar, S., Jain, A., & Swathi, A. (2022, October). Commodities Price Prediction using Various ML Techniques. In 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS) (pp. 277-282). IEEE.
14. Han, Y., Kim, J., & Enke, D. (2023). A machine learning trading system for the stock market based on N-period Min-Max labeling using XGBoost. Expert Systems with Applications, 211, 118581.
15. Lahari, M. C., Ravi, D. H., & Bharathi, R. (2018, September). Fuel price prediction using RNN. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 1510-1514). IEEE.
16. Shilong, Z. (2021, January). Machine learning model for sales forecasting by using XGBoost. In 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE) (pp. 480-483). IEEE.
17. Chowdary, K. L., Krishna, C. N., Manaswini, K. S., & Jithendra, B. (2023, February). Electricity Price Prediction using Machine Learning. In 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS) (pp. 611-615). IEEE.
18. XGBoost Homepage, https://en.wikipedia.org/wiki/XGBoost,last accessed 2023/09/27.