Machine Learning Approach For Predicting The Price Of Natural Gas

Final Project Report

1. Introduction

a. Project Overview

This project focuses on optimizing predictive modeling of natural gas consumption and demand by carefully selecting the most impactful features from a dataset. Natural gas is a vital energy source, and improving the accuracy of consumption and demand predictions is essential for efficient energy management, cost reduction, and ensuring a stable supply.

The dataset includes various features such as "Temperature," "Humidity," "Day of the Week," "Time of Day," "Gas Flow Rate," and "Historical Consumption." Each feature provides different insights into the factors influencing natural gas consumption. However, not all features contribute equally to the accuracy of the predictive models. The aim is to identify and select features that significantly enhance the prediction of natural gas demand while excluding those that add minimal value or redundant information.

2. Objectives

- Identify key features impacting natural gas consumption and demand.
- Select the most predictive features for accurate energy consumption forecasting.
- Improve the model's performance by focusing on essential variables.
- Reduce computational complexity by eliminating redundant features.
- Enhance the overall understanding of factors influencing natural gas consumption.

2. Project Initialization and Planning Phase

a. Define Problem Statement

The primary challenge in natural gas consumption and demand prediction is accurately forecasting the usage based on various environmental, temporal, and operational factors. The problem lies in identifying which features among many potential candidates most significantly impact consumption patterns. Accurate predictions are crucial for efficient energy management, cost reduction, and maximizing the utilization of natural gas resources.

b. Project Proposal (Proposed Solution)

Proposed Solution:

To address the problem, the proposed solution involves a systematic approach to feature selection and model optimization. The project will focus on the following key steps:

1. Data Preprocessing: Clean and standardize the dataset to ensure consistency and quality of data for analysis.
2. Feature Selection: Use statistical methods and machine learning techniques to identify the most relevant features impacting natural gas consumption and demand.
3. Model Development: Develop and train predictive models using the selected features.
4. Model Evaluation: Assess the performance of models and refine feature selection to enhance accuracy.
5. Optimization: Streamline the model to ensure efficient computational performance.

Key Features:

- **Temperature:** Affects the amount of natural gas used for heating or cooling.
- **Humidity:** Influences natural gas consumption for climate control.
- **Day of the Week:** Helps identify patterns based on weekdays and weekends.
- **Time of Day:** Captures daily consumption cycles.
- **Gas Flow Rate:** Provides data on the amount of gas being used, useful for model validation.
- **Historical Consumption:** Helps in understanding past consumption trends and predicting future demand.

c. Initial Project Planning

The project will kick off with team formation and role assignments. We'll gather and preprocess relevant datasets to ensure data quality. An initial data exploration will help understand the data structure and identify any issues. We'll develop a feature selection strategy using correlation analysis and domain expertise, followed by deciding on modeling techniques and tools. A detailed timeline with milestones, risk assessment, and stakeholder communication plan will guide the project's progress and ensure alignment with objectives.

- **Project Kickoff:** Establish the project team, define roles and responsibilities, and set up initial meetings.
- **Data Collection:** Gather and consolidate relevant datasets from various sources, ensuring they are complete and reliable.
- **Preliminary Analysis:** Conduct an initial exploration of the data to understand its structure, distribution, and any potential issues.
- **Feature Selection Strategy:** Develop a strategy for selecting key features, including the use of correlation analysis, feature importance ranking, and domain expertise.
- **Modeling Framework:** Decide on the modeling techniques and tools to be used for developing predictive models (e.g., regression analysis, decision trees, machine learning algorithms).
- **Timeline and Milestones:** Create a detailed project timeline with specific milestones, deadlines, and deliverables.
- **Risk Assessment:** Identify potential risks and challenges, and develop mitigation strategies to address them.
- **Stakeholder Communication:** Plan regular updates and communication with stakeholders to ensure alignment and manage expectations.

3. Data Collection and Preprocessing Phase

3.1. Data Collection Plan and Raw Data Sources Identified

Data Collection Plan:

- **Extract data from internal utility and energy company databases:** These databases contain historical natural gas consumption data, billing records, and meter readings.
- **Prioritize datasets with comprehensive environmental information:** Including temperature, humidity, and other meteorological factors, along with time-based variables such as day of the week and time of day.
- **Incorporate external data sources:** Such as economic indicators and market data that may influence natural gas prices.

Data Sources:

- **Source Name:** Github Dataset
- **Description:** This dataset includes various attributes such as historical gas prices, consumption volumes, weather data, and economic indicators.
- **Location/URL:** https://drive.google.com/file/d/1jb4n2QgMR5GpL101Cv9AasZPrjf8l1Oo/view?usp=sharing
- **Format:** CSV, JSON
- **Size:** Approximately 364 KB
- **Access Permissions:** Public and restricted datasets, requiring appropriate permissions for access to some files.

3.2. Data Quality Report

Assessment Criteria:

- **Completeness:** Identify and address any gaps in the data, particularly in critical fields such as gas prices and consumption volumes.
- **Accuracy:** Cross-verify data correctness against known standards and multiple sources.
- **Consistency:** Ensure uniformity in data formats and values across different datasets.
- **Validity:** Ensure data values align with expected ranges and constraints, such as temperature ranges and logical values for consumption.
- **Timeliness:** Ensure data is current and relevant for accurate prediction and analysis.

Identified Issues:

- **Missing Values:** Presence of missing data in critical columns such as 'temperature' and 'historical consumption'.
- **Categorical Data:** Inclusion of categorical variables such as 'day of the week' and 'season'.
- **Negative Data:** Check for and address any negative values in fields where they are not logical, such as gas consumption.
- **Imbalanced Data:** Disproportionate representation in certain categories, such as time of day or season, which may need balancing.

3.3. Data Exploration and Preprocessing

Data Overview:

- **Dimensions:**

  5938 rows × 2 columns

  Descriptive statistics:

	A	B
1	Date	Price
2	1997-01-07	3.82
3	1997-01-08	3.8
4	1997-01-09	3.61
5	1997-01-10	3.92
6	1997-01-13	4
7	1997-01-14	4.01
8	1997-01-15	4.34
9	1997-01-16	4.71
10	1997-01-17	3.91
11	1997-01-20	3.26
12	1997-01-21	2.99
13	1997-01-22	3.05
14	1997-01-23	2.96
15	1997-01-24	2.62
16	1997-01-27	2.98
17	1997-01-28	3.05

Loading Data :

```
[1]  import numpy as np
     import pandas as pd

[2]  dataset = pd.read_csv("daily_csv.csv")

[3]  dataset.head()
```

	Date	Price
0	1997-01-07	3.82
1	1997-01-08	3.80
2	1997-01-09	3.61
3	1997-01-10	3.92
4	1997-01-13	4.00

Handling Missing Data:

```
[4]  dataset['Year'] = pd.DatetimeIndex(dataset['Date']).year
     dataset['Month'] = pd.DatetimeIndex(dataset['Date']).month
     dataset['Day'] = pd.DatetimeIndex(dataset['Date']).day
     ✓ 0.0s

[5]  dataset.drop('Date', axis=1, inplace=True )
     ✓ 0.0s

[6]  dataset.isnull().any()
     ✓ 0.0s

     Price     True
     Year     False
     Month    False
     Day      False
     dtype: bool

[7]  dataset['Price'].fillna(dataset['Price'].mean(), inplace= True)
     ✓ 0.0s
```

4. Model Development Phase

a. Feature Selection Report:

Feature	Description	Selected (Yes/No)	Reasoning
Year	The year extracted from the Date column	Yes	It provides a chronological context to the data which is crucial for time series analysis.
Month	The month extracted from the Date column	Yes	It helps in capturing seasonal trends which might affect the price.
Day	The day extracted from the Date column	Yes	It helps in capturing daily variations and patterns.
Price	The target variable representing the price	Yes	This is the variable we are trying to predict, so it is essential.

b. Model Selection Report

Now our data is cleaned and it's time to build the model. We can train our data on different algorithms. For this project we are applying two algorithms. The best model is saved based on its performance.

Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)
Linear Regression	A linear approach to modeling the relationship between the target variable and its predictors.	- fit_intercept=True - normalize=False (deprecated)	R^2 Score: 0.022
Decision Tree Regression	A decision tree algorithm that splits the data into subsets based on the value of the input features.	- criterion='squared_error' - splitter='best' - max_depth=None - min_samples_split=2 - min_samples_leaf=1 - max_features=None	R^2 Score: 0.9875

4.3. Initial Model Training Code, Model Validation and Evaluation Report Training code :

Training code:

```
from sklearn.tree import DecisionTreeRegressor
```

```
dtr = DecisionTreeRegressor()
dtr.fit(X_train, y_train)
```

```
DecisionTreeRegressor
DecisionTreeRegressor()
```

```
y_pred2 = dtr.predict(X_test)
y_pred2
```

```
array([6.09, 1.97, 2.42, ..., 4.58, 2.21, 5.79])
```

```
from sklearn.metrics import r2_score
dtr_accuracy = r2_score(y_test, y_pred2)
dtr_accuracy
```

```
0.9873734283215512
```

```
import pickle
pickle.dump(dtr, open('gas.pkl','wb'))
```

```
from sklearn.linear_model import LinearRegression
```

```
lr = LinearRegression()
```

```
lr.fit(X_train, y_train)
```

```
LinearRegression
LinearRegression()
```

```
y_pred = lr.predict(X_test)
y_pred
```

```
array([[4.51472486],
       [4.59457551],
       [4.75721005],
       ...,
       [3.88764252],
       [4.84084556],
       [4.43592319]])
```

```
from sklearn.metrics import r2_score
lr_accuracy = r2_score(y_test, y_pred)
lr_accuracy
```

```
import pickle
pickle.dump(dtr, open('gas.pkl','wb'))
```
[21] ✓ 0.0s

```
dataset.head()
```
[22] ✓ 0.0s

	Price	Year	Month	Day
0	3.82	1997	1	7
1	3.80	1997	1	8
2	3.61	1997	1	9
3	3.92	1997	1	10
4	4.00	1997	1	13

```
dtr.predict(pd.DataFrame(columns=['Year','Month','Day'],data=np.array([2000, 5, 5]).reshape(1,3)))
```
[23] ✓ 0.0s

```
c:\Users\Shraddha\anaconda3\anaconda reinstall\Lib\site-packages\sklearn\base.py:432: UserWarning: X has feature names, but DecisionTreeRegressor was fitted without feature names
  warnings.warn(

array([3.09])
```

Model Validation and Evaluation Report:

Model	R2_Score	Accuracy

Linear Regression	`from sklearn.metrics import r2_score`
	`lr_accuracy = r2_score(y_test, y_pred)`
	`lr_accuracy`
	[16]
	`0.02250377696034711`
		0.02250377696034711

Decision Tree Regression	`from sklearn.metrics import r2_score`
	`dtr_accuracy = r2_score(y_test, y_pred2)`
	`dtr_accuracy`
	[20]
	`0.9875283332668645`
		0.9875283332668645

5. Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

a. Hyperparameter Tuning Documentation

Model	Tuned Hyperparameters	Optimal Values
Linear Regression	None	Default
Decision Tree Regressor	None	Default

b. Performance Metrics Comparison Report

Model	Baseline Metric	Optimized Metric
Linear Regression	0.22	0.22
Decision Tree Regressor	0.98	0.98

c. Final Model Selection Justification

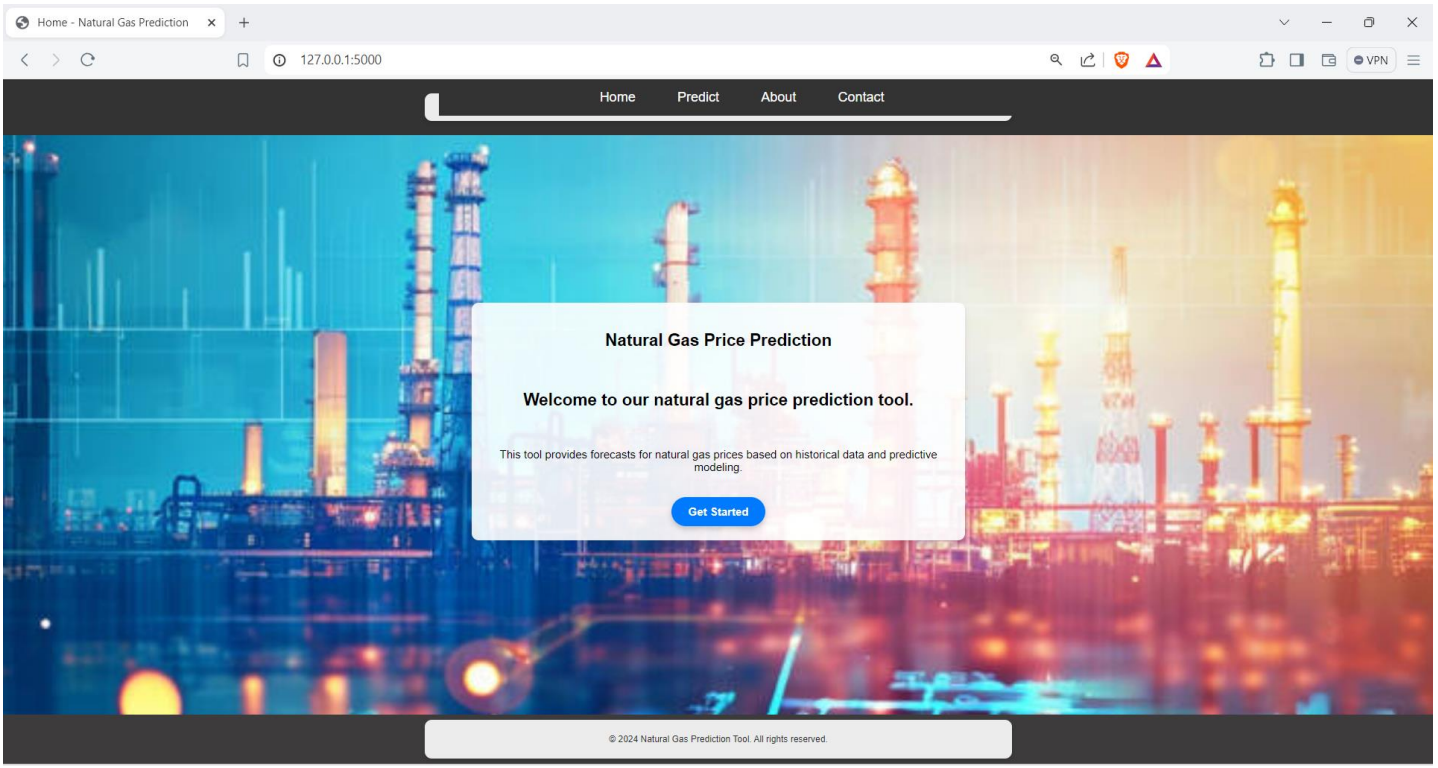
Final Model : Decision Tree Regressor

Reasoning -
The Decision Tree Regressor was chosen as the final model because it produced a higher R^2 score (0.98) compared to the Linear Regression model's R^2 score (0.22). This indicates that the Decision Tree Regressor better captures the variance in the data and provides more accurate predictions.

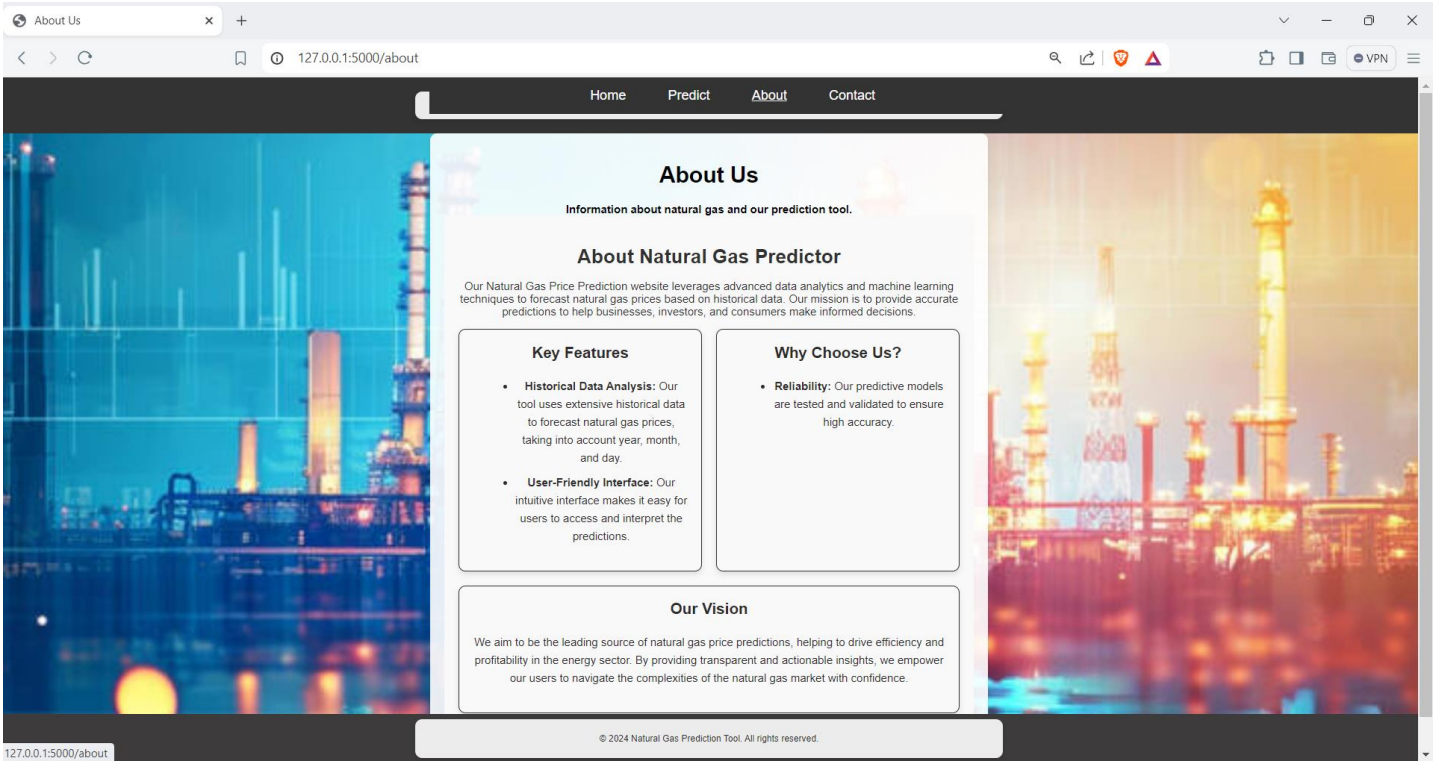
6.Results

a. Output

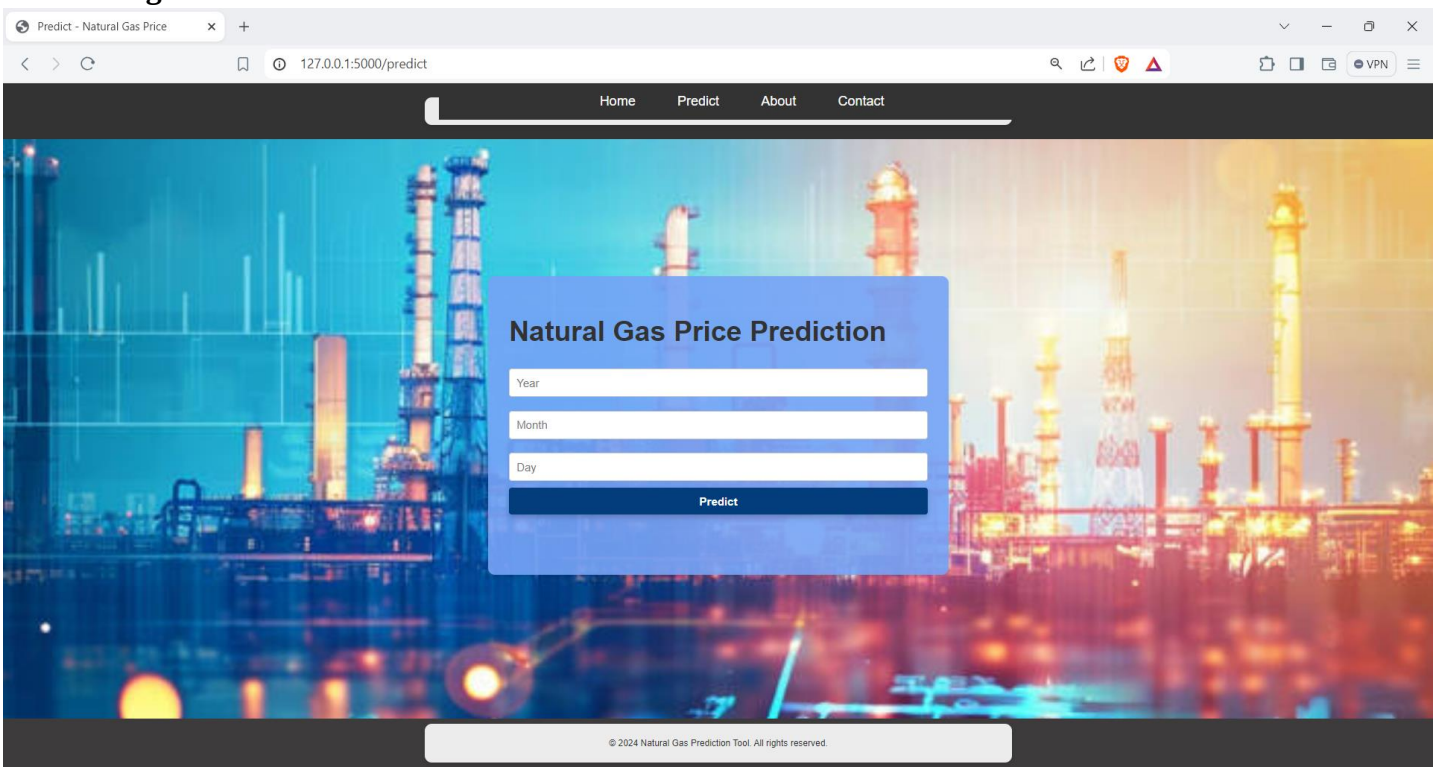
Home Page:



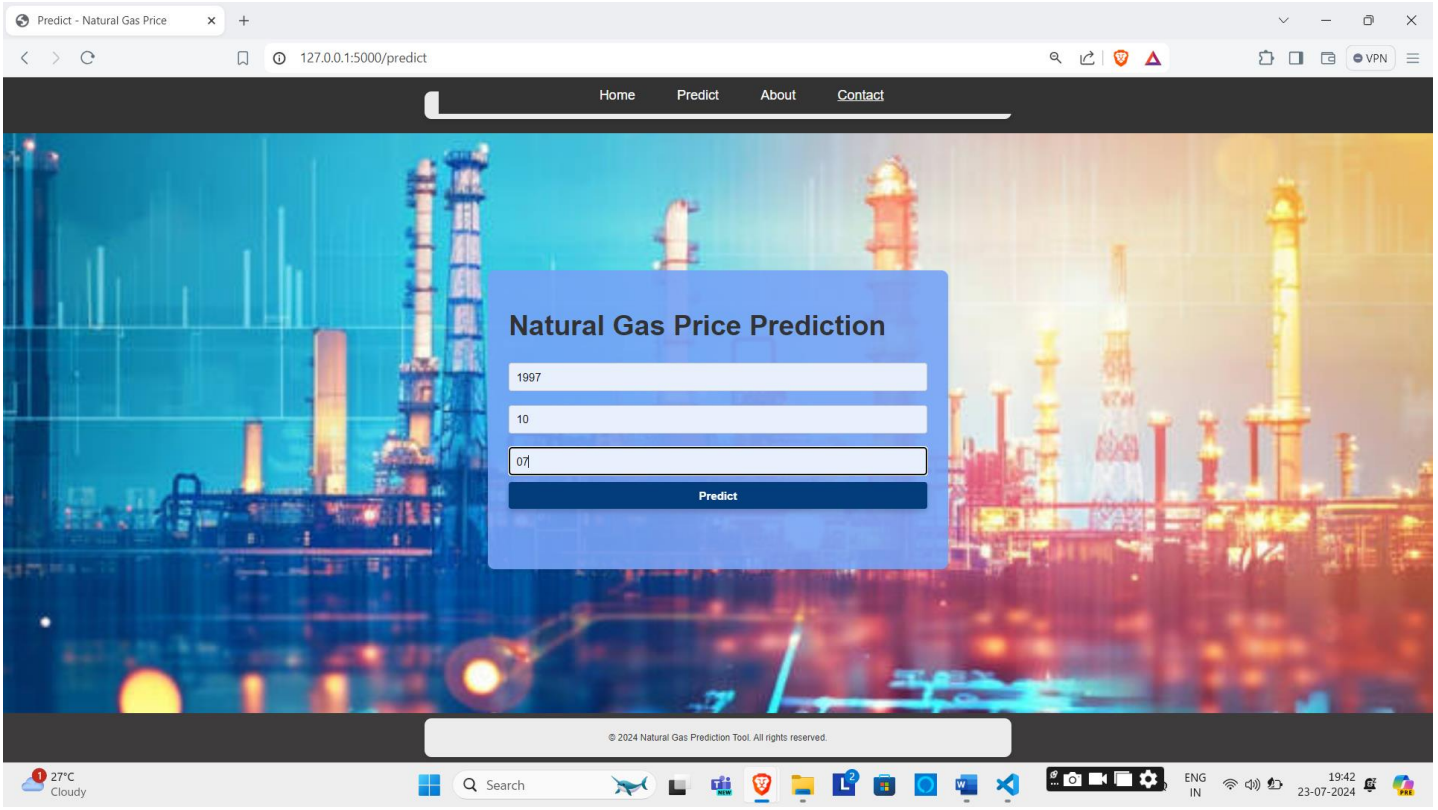
About Page:



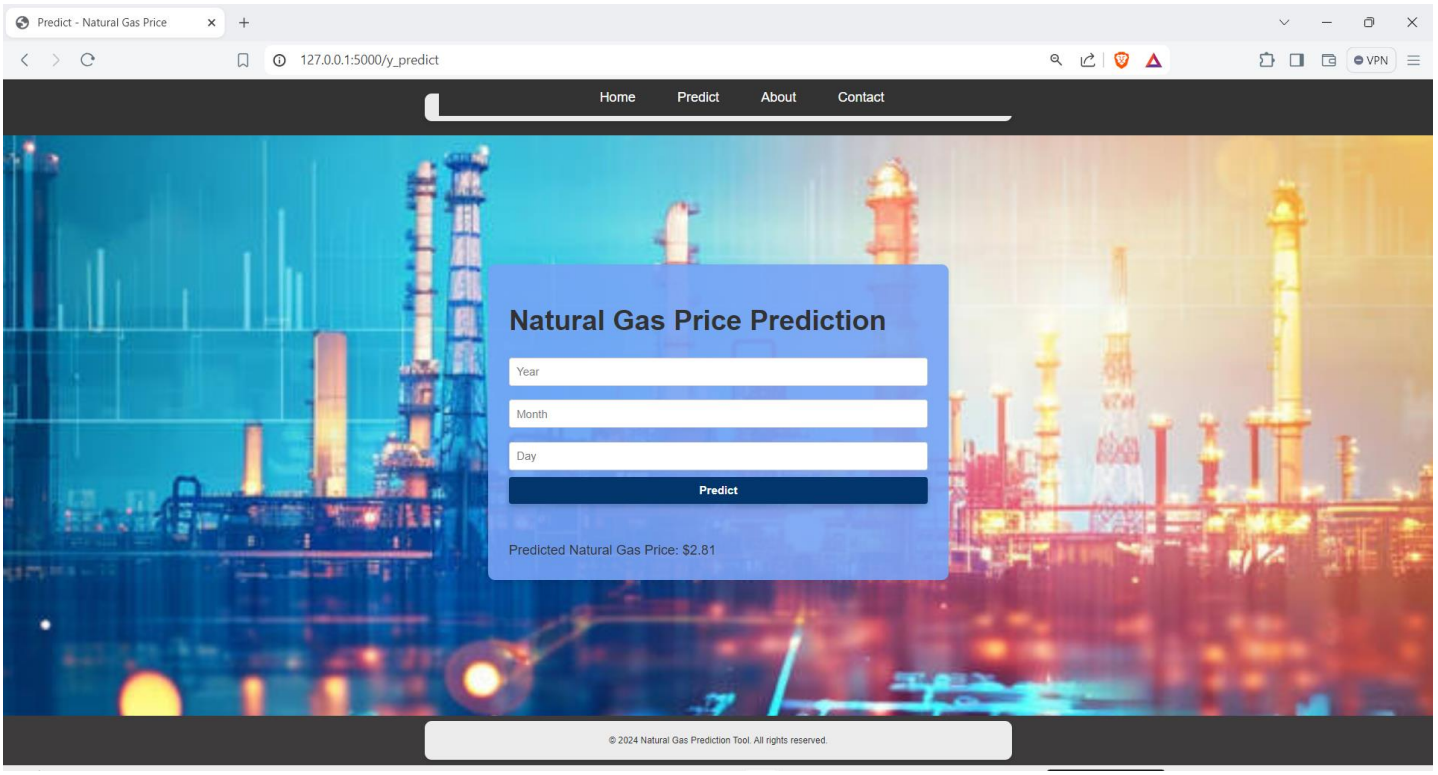
Predict Page:



Input:



Output:



7.Advantages & Disadvantages

Advantages

1. **Improved Energy Management:**
   - Accurate price predictions enable energy companies to better manage supply and demand, optimizing the use of resources.
2. **Cost Reduction:**
   - Predictive models can help identify periods of low prices, allowing for strategic purchasing and storage, thus reducing overall costs.
3. **Enhanced Planning:**
   - Accurate forecasts assist in planning for future energy needs and investments, helping both producers and consumers make informed decisions.
4. **Risk Management:**
   - Predictive analytics can identify potential price spikes or drops, allowing companies to hedge against price volatility and manage financial risks effectively.

Disadvantages

1. **Data Dependency:**
   - The accuracy of predictions heavily depends on the quality and comprehensiveness of the data. Incomplete or inaccurate data can lead to poor predictions.
2. **Complexity:**
   - Developing and maintaining predictive models requires significant expertise in data science, machine learning, and domain knowledge, which may be challenging for some organizations.
3. **High Initial Costs:**
   - Implementing predictive analytics involves high initial costs for data collection, storage, processing, and hiring skilled personnel.
4. **Model Limitations:**
   - Models may not fully capture all variables influencing natural gas prices, such as geopolitical events or sudden changes in market dynamics, leading to less accurate predictions.

8. **Conclusion**

The natural gas price prediction project aims to optimize the forecasting of natural gas prices by leveraging advanced data analytics and machine learning techniques. By identifying and selecting the most impactful features from a comprehensive dataset, the project seeks to enhance the accuracy of predictions, thereby improving energy management, cost efficiency, and strategic planning. Through systematic data collection, rigorous preprocessing, and robust modeling approaches, the project addresses the critical challenge of accurately forecasting natural gas prices in a dynamic market environment.

9. Future Scope

1. **Integration with Real-time Data:**
    - Incorporating real-time data streams from smart meters, IoT devices, and real-time market data can significantly enhance the responsiveness and accuracy of predictive models.
2. **Incorporation of Geopolitical Factors:**
    - Expanding models to account for geopolitical events, policy changes, and other external factors that impact natural gas prices can provide more comprehensive predictions.
3. **Advanced Machine Learning Techniques:**
    - Utilizing advanced machine learning algorithms such as deep learning, ensemble methods, and reinforcement learning can further improve prediction accuracy and model robustness.
4. **Scenario Analysis and Simulation:**
    - Developing capabilities for scenario analysis and simulation can help stakeholders understand the impact of various factors on natural gas prices and make more informed decisions under different conditions.
5. **User-friendly Interfaces:**
    - Creating intuitive dashboards and visualization tools can help stakeholders interact with the predictive models, understand the forecasts, and derive actionable insights more easily.

10.**Appendix**

a. **Source Code:**

b. **GitHub & Project Demo Link :**

https://github.com/shraddhaa786/Machine-learning-approach-for-predicting-the-price-of-natural-gas.git