

Sentiment Analysis of Commodity News (Gold) Project Hand-out, Faculty Development Program – NaanMudhalvan

SmartInternz

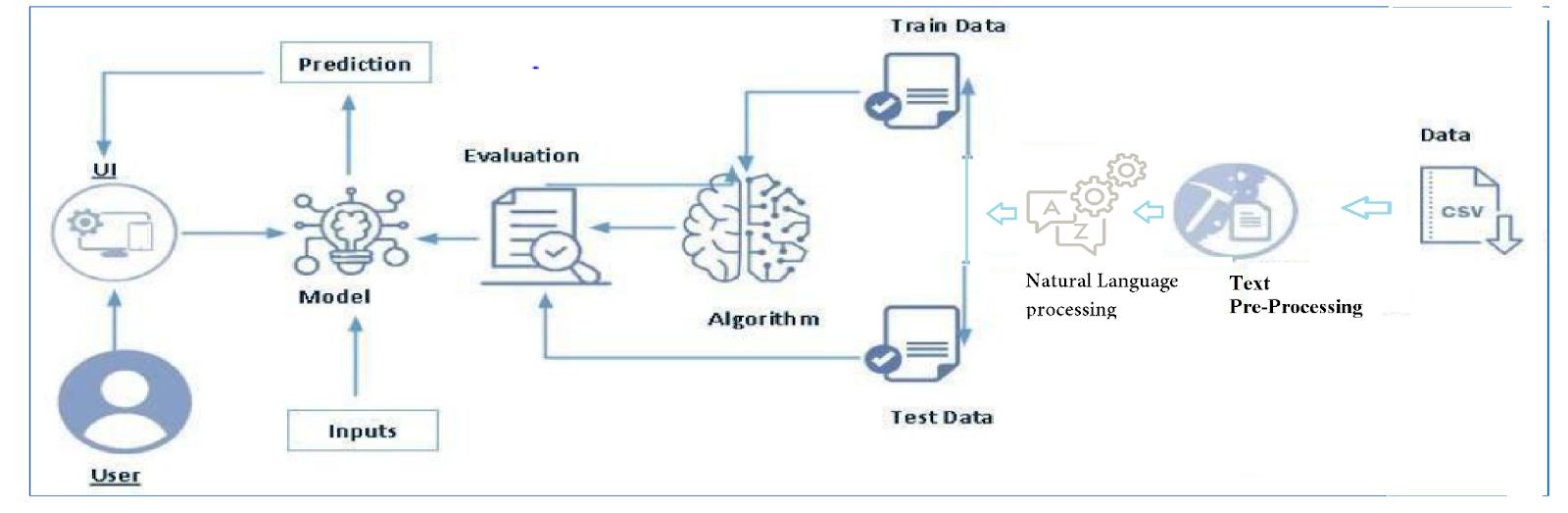
www.smartinternz.com

**Sentiment Analysis of Commodity News (Gold)**

Sentiment Analysis of Commodity News (Gold) is a process of using natural language processing and machine learning techniques to determine the emotional tone of news articles or other text related to the gold commodity market. The goal of sentiment analysis is to understand how people feel about a particular topic, in this case gold, by analyzing the words and phrases used in the text

Sentiment analysis can help to determine whether news about gold is generally positive, negative, or neutral, giving traders and investors an idea of how the market is reacting to the latest developments in the gold industry. For example, positive sentiment in news articles about gold might indicate increasing demand for the precious metal, which could drive up prices. On the other hand, negative sentiment could indicate a decrease in demand or a downturn in the market. By conducting sentiment analysis on a large corpus of news articles about gold, it is possible to gain insights into the overall sentiment of the market and make informed decisions about buying and selling gold..

**Technical Architecture:**



**Project Flow:**

* User interacts with the UI to enter the input.
* Entered input is analysed by the model which is integrated.
* Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities listed below,

* Define Problem / Problem Understanding
  + Specify the business problem
  + Business requirements
  + Literature Survey
  + Social or Business Impact.
* Data Collection & Preparation
  + Collect the dataset
  + Data Preparation
* Exploratory Data Analysis
  + Descriptive statistical
  + Visual Analysis
* Model Building
  + Training the model in multiple algorithms
  + Testing the model
* Performance Testing
  + Testing model with multiple evaluation metrics
* Model Deployment
  + Save the best model
  + Integrate with Web Framework

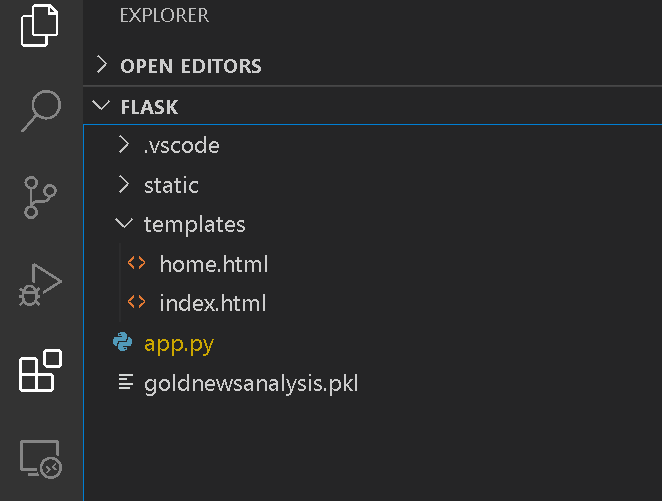
**Prior Knowledge:**

You must have prior knowledge of following topics to complete this project.

* ML Concepts
  + Supervised learning: <https://www.javatpoint.com/supervised-machine-learning>
  + Unsupervised learning: <https://www.javatpoint.com/unsupervised-machine-learning>
* Decision tree: [https://www.javatpoint.com/machine-learning-decision-tree-classification-](https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm) [algorithm](https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm)
* Random forest: <https://www.javatpoint.com/machine-learning-random-forest-algorithm>
* KNN: <https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning>
* Xgboost: [https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-](https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/) [understand-the-math-behind-xgboost/](https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/)
* Evaluation metrics: [https://www.analyticsvidhya.com/blog/2019/08/11-important-model-](https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/) [evaluation-error-metrics/](https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/)
* NLP:-https://www.tutorialspoint.com/natural\_language\_processing/natural\_language\_processing\_python.htm
* Flask Basics: <https://www.youtube.com/watch?v=lj4I_CvBnt0>

**Project Structure:**

Create the Project folder which contains files as shown below



* We are building a flask application which needs HTML pages stored in the templates folder and a python script app.py for scripting.
* goldnewsanalysis.pkl is our saved model. Further we will use this model for flask integration.
* Training folder contains a model training file.

# Milestone 1: Define Problem / Problem Understanding

**Activity 1: Specify the business problem**

Refer Project Description

## Activity 2: Business requirements

To successfully implement sentiment analysis of commodity news, particularly gold, there are several business requirements that need to be met. Some of the key requirements include:

* Accurate and up-to-date information: The project should use the most recent and reliable data to classify news titles, in order to ensure that the information is accurate and relevant to current commodity new .
* Flexibility: The classification system should be flexible and able to see the news updated news and changing information as it becomes available.
* User-friendly interface: The classification system should be easy to use and understand for both technical professionals and non-technical peoples.

## Activity 3: Literature Survey

* Overview of Sentiment Analysis: An overview of the concept of sentiment analysis and its applications in various domains, including finance and commodity markets applications of Sentiment Analysis in Commodity News: A review of existing research on the use of sentiment analysis in the analysis of commodity news, including gold. This could include studies that have applied sentiment analysis to news articles, social media, or other sources of text data related to gold
* Limitations and Challenges of Sentiment Analysis: A discussion of the challenges and limitations of sentiment analysis, such as subjectivity, ambiguity, and the need for large annotated datasets.
* Future Research Directions: A discussion of the future research directions for sentiment analysis of commodity news, including the development of more advanced algorithms and the integration of sentiment analysis with other financial market analysis tools.

**Activity 4: Social or Business Impact.**

Improved Investment Decisions: Sentiment analysis of commodity news can provide investors and traders with valuable insights into market sentiment and trends, helping them to make more informed investment decisions. For example, positive sentiment in news articles about gold might indicate increasing demand for the precious metal, which could drive up prices, while negative sentiment could indicate a decrease in demand or a downturn in the market.

Improved Market Stability: By providing a more complete and accurate picture of market sentiment, sentiment analysis can help to stabilize the gold commodity market. This can reduce the risk of sudden price swings and improve the overall efficiency of the market.

# Milestone 2: Data Collection & Preparation

ML depends heavily on data. It is the most crucial aspect that makes algorithm training possible. So, this section allows you to download the required dataset.

## Activity 1: Collect the dataset

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc.

In this project we have used .csv data. This data is downloaded from kaggle.com. Please refer to the link given below to download the dataset.

Link: <https://www.kaggle.com/datasets/ankurzing/sentiment-analysis-in-commodity-market-gold>

As the dataset is downloaded. Let us read and understand the data properly with the help of some visualisation techniques and some analysing techniques.

**Note:** There are a number of techniques for understanding the data. But here we have used some of it. In an additional way, you can use multiple techniques.

**Activity 1.1: Importing the libraries**

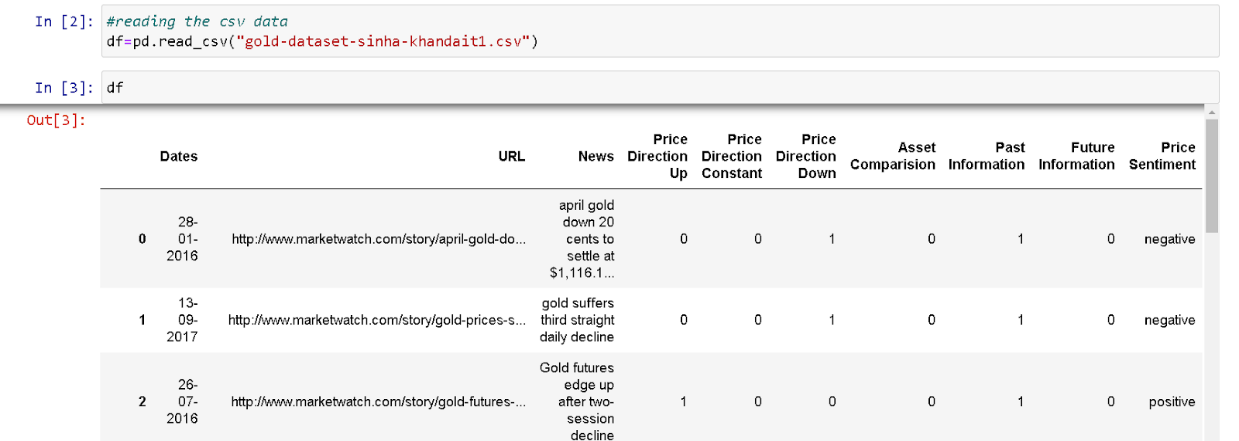
Import the necessary libraries as shown in the image.



## Activity 1.2: Read the Dataset

Our dataset format might be in .csv, excel files, .txt, .json, etc. We can read the dataset with the help of pandas.

In pandas we have a function called read\_csv() to read the dataset. As a parameter we have to give the directory of the csv file.



## Activity 2: Data Preparation

As we have understood how the data is, let's pre-process the collected data.

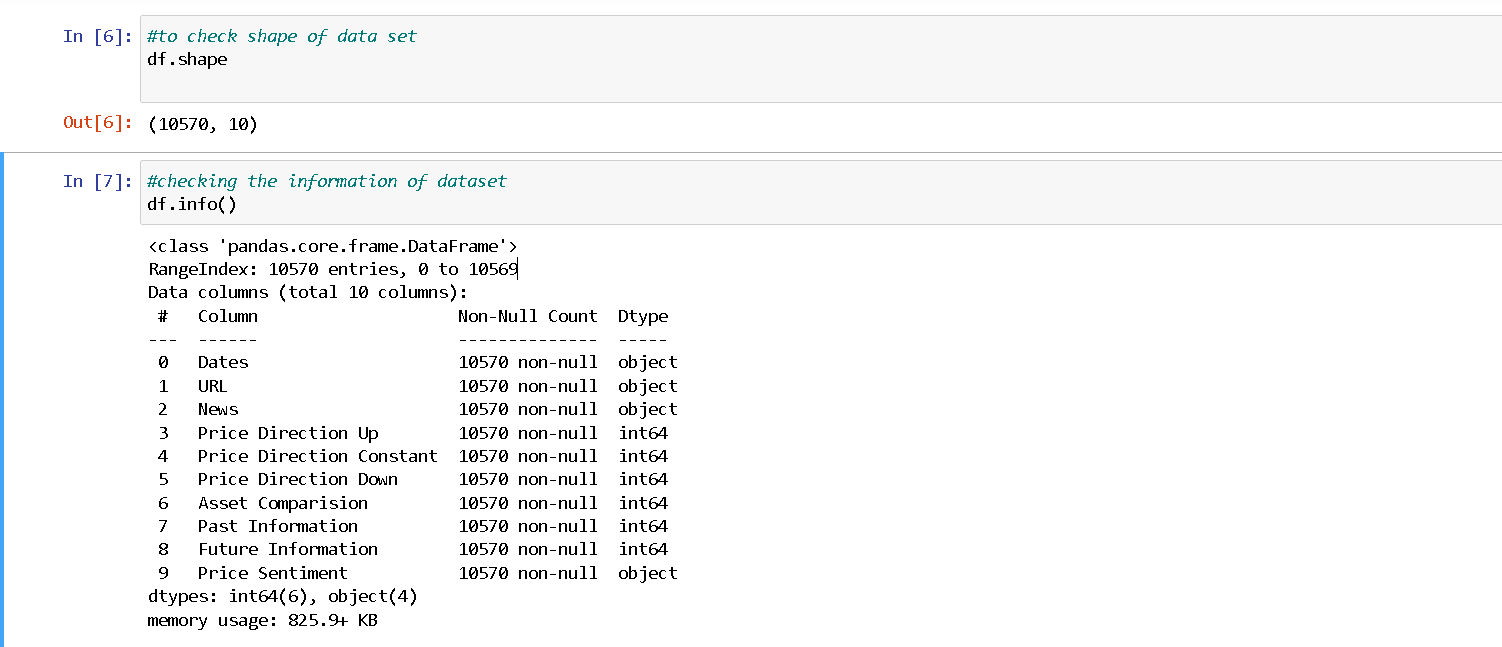
The download data set is not suitable for training the machine learning model as it might have so much randomness so we need to clean the dataset properly in order to fetch good results. This activity includes the following steps.

* Handling missing values
* Handling categorical data
* Handling Outliers

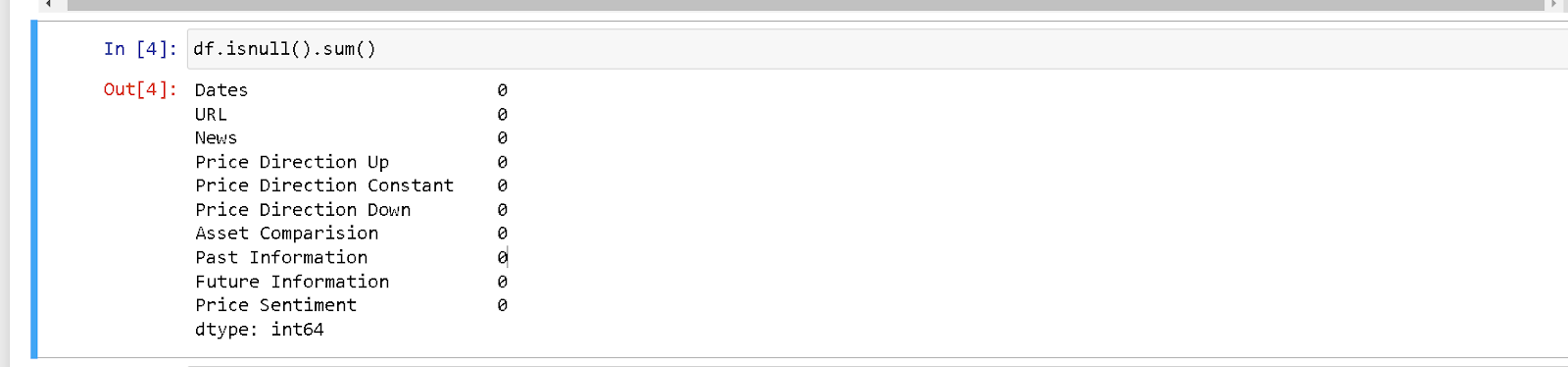
Note: These are the general steps of pre-processing the data before using it for machine learning. Depending on the condition of your dataset, you may or may not have to go through all these steps.

## Activity 2.1: Handling missing values

* Let’s find the shape of our dataset first. To find the shape of our data, the df.shape method is used. To find the data type, df.info() function is used



* For checking the null values, df.isnull() function is used. To sum those null values we use .sum() function. From the below image we found that there are no null values present in our dataset. So we can skip handling the missing values step.

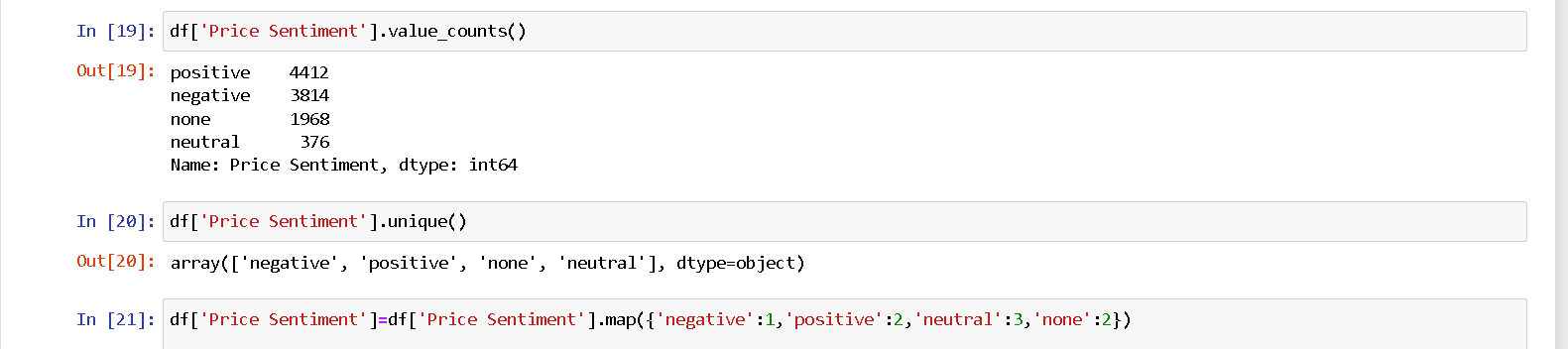


## Activity 2.2: Handling Categorical Values

As we can see our dataset has categorical data we must convert the categorical data to integer encoding or binary encoding.

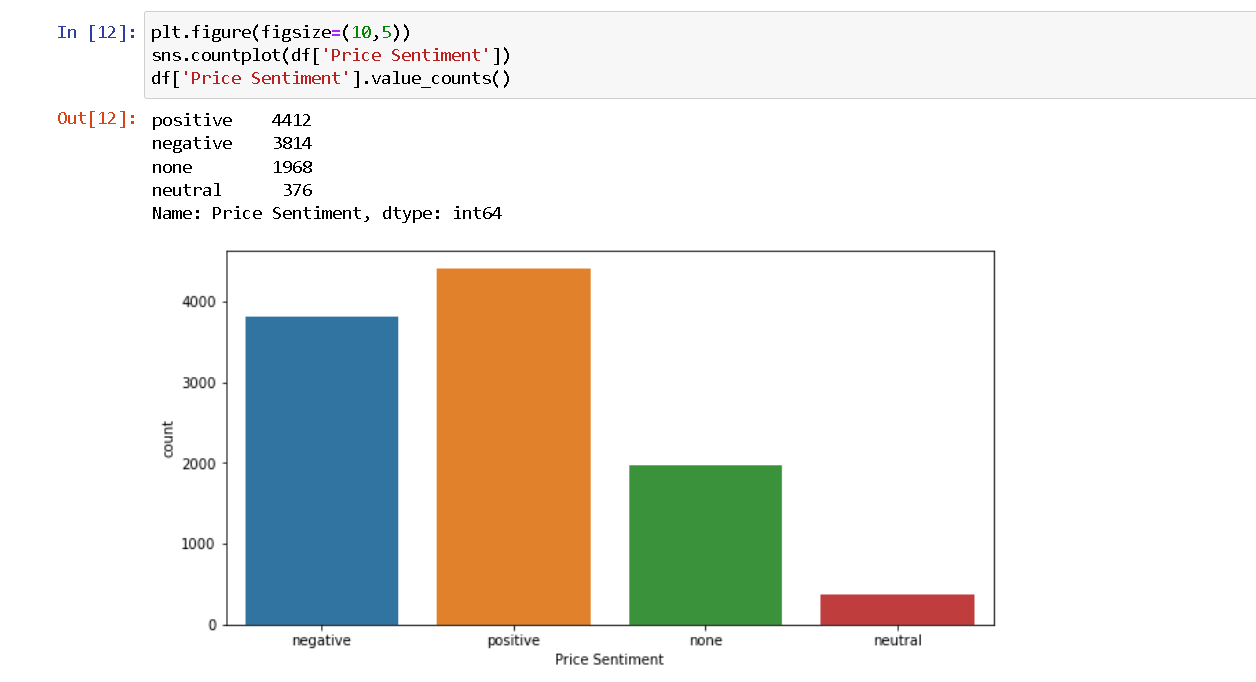
To convert the categorical features into numerical features we use encoding techniques. There are several techniques but in our project we are using manual encoding with the help of list comprehension.

• In our project, categorical features is Price sentiment column but no need to convert in this project because the we are doing here natural language processing but I have shared how to convert the categorical values to numerical



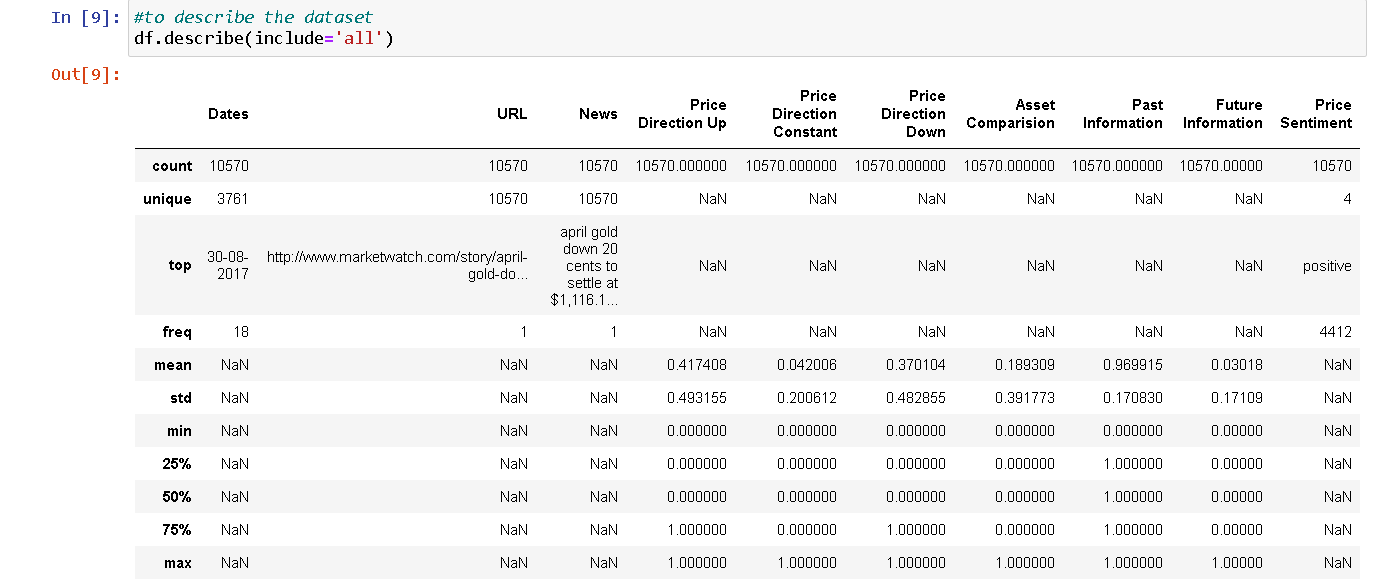
## Activity 2.3: Handling Imbalance Data

With the help of count plot are visualized. And here we are going to find bars with imbalance data of Price sentiment. but in this dataset no need to perfom any analysis as we are using data as it is



# Milestone 3: Exploratory Data Analysis

## Activity 1: Descriptive statistical

Descriptive analysis is to study the basic features of data with the statistical process. Here pandas has a worthy function called describe. With this describe function we can understand the unique, top and frequent values of categorical features. And we can find mean, std, min, max and percentile values of continuous features.

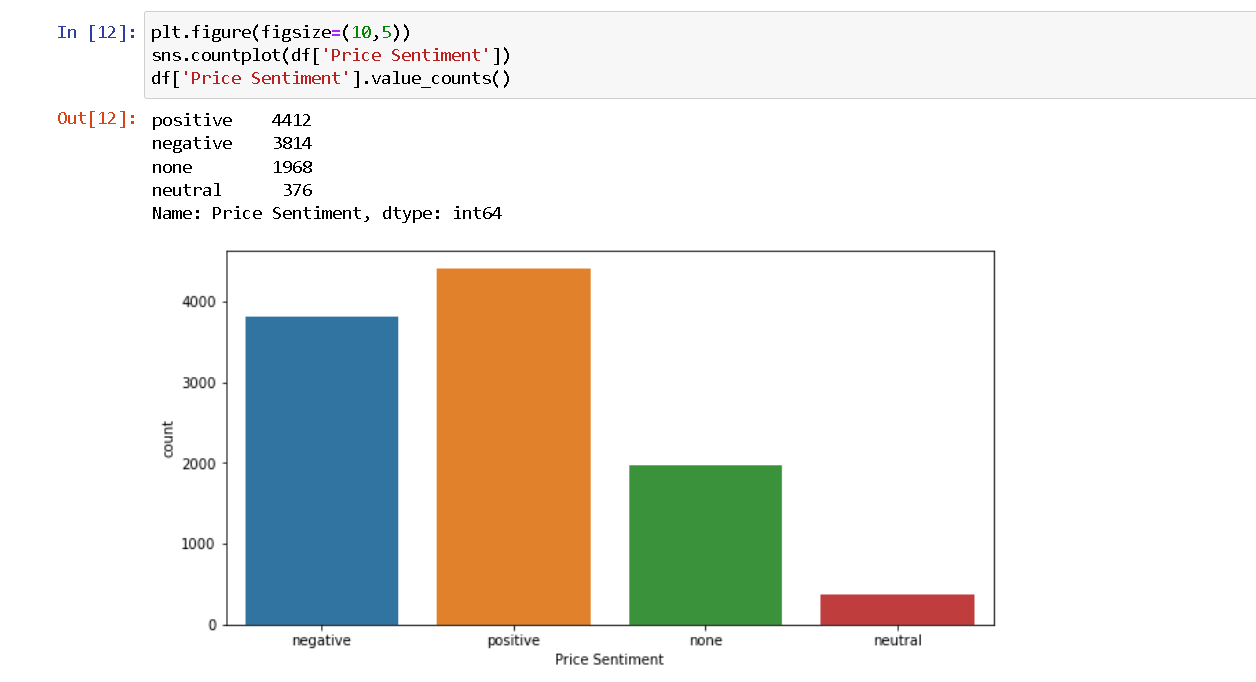
## Activity 2: Visual analysis

Visual analysis is the process of using visual representations, such as charts, plots, and graphs, to explore and understand data. It is a way to quickly identify patterns, trends, and outliers in the data, which can help to gain insights and make informed decisions.

**Activity 2.1: Univariate analysis**

In simple words, univariate analysis is understanding the data with single feature. Here we have displayed two different graphs such as distplot and countplot.

In our dataset we have some categorical features. With the countplot function, we are going to count the unique category in those features.



**Activity 2.2: Bivariate analysis**

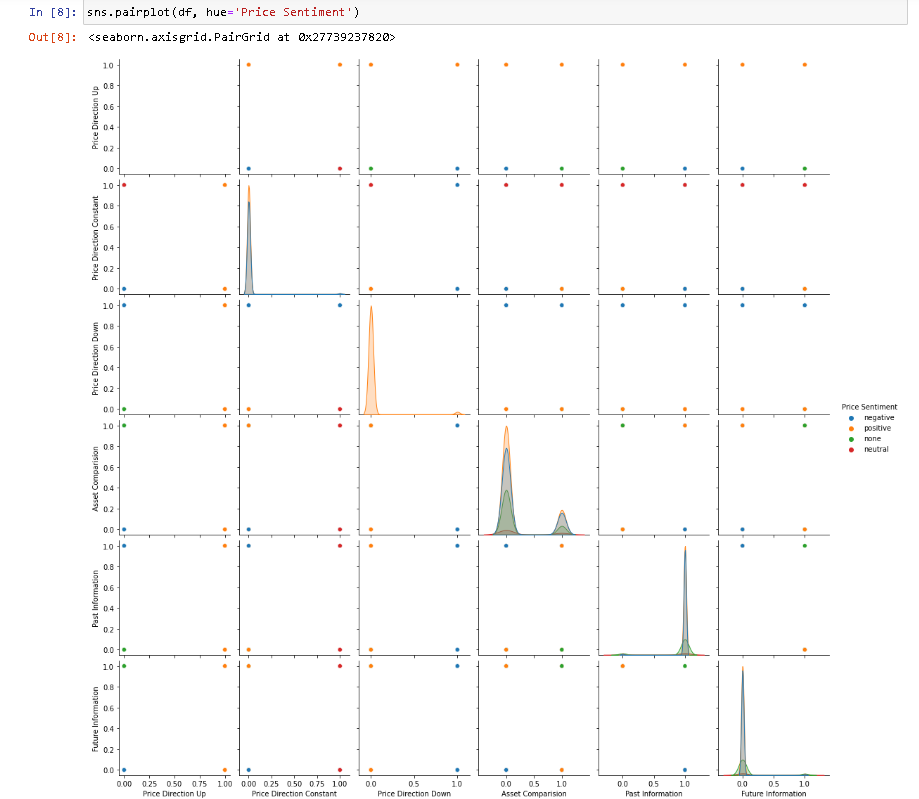
To find the relation between two features we use bivariate analysis. Here we are  
visualizing the relationship between price sentimnet & price direction up, price sentimnet & past Infromation.

Countplot is used here. As a 1st parameter we are passing x value and as a 2nd  
parameter we are passing hue value.



## Activity 2.3: Multivariate analysis

In simple words, multivariate analysis is to find the relation between multiple features. Here we have used pairplot from seaborn package.



**Text pre-processing (python packages)**

Text pre-processing is a crucial step in Natural Language Processing (NLP) and Information Retrieval (IR) tasks. The goal is to convert raw text into a more meaningful and manageable representation for further analysis.

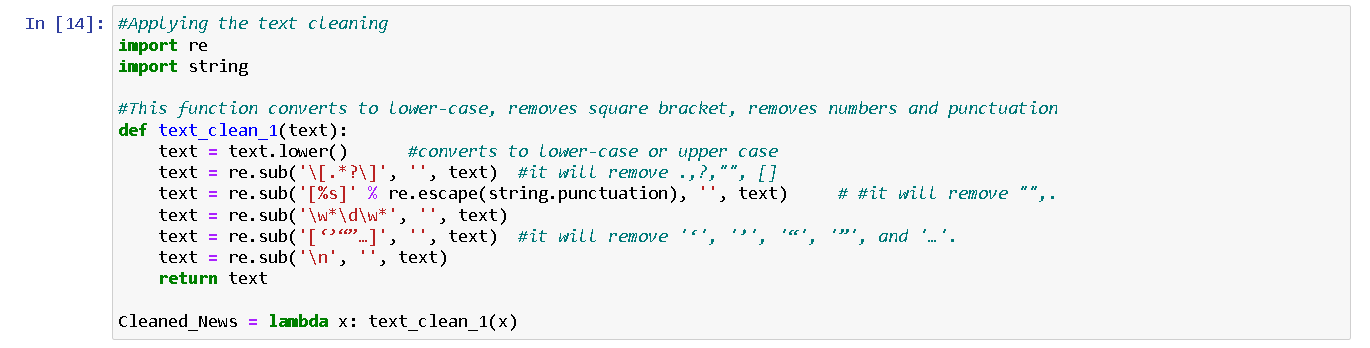
In Python, there are several packages that provide support for text pre-processing operations. Some of the most common ones are:

* **NLTK (Natural Language Toolkit)**-

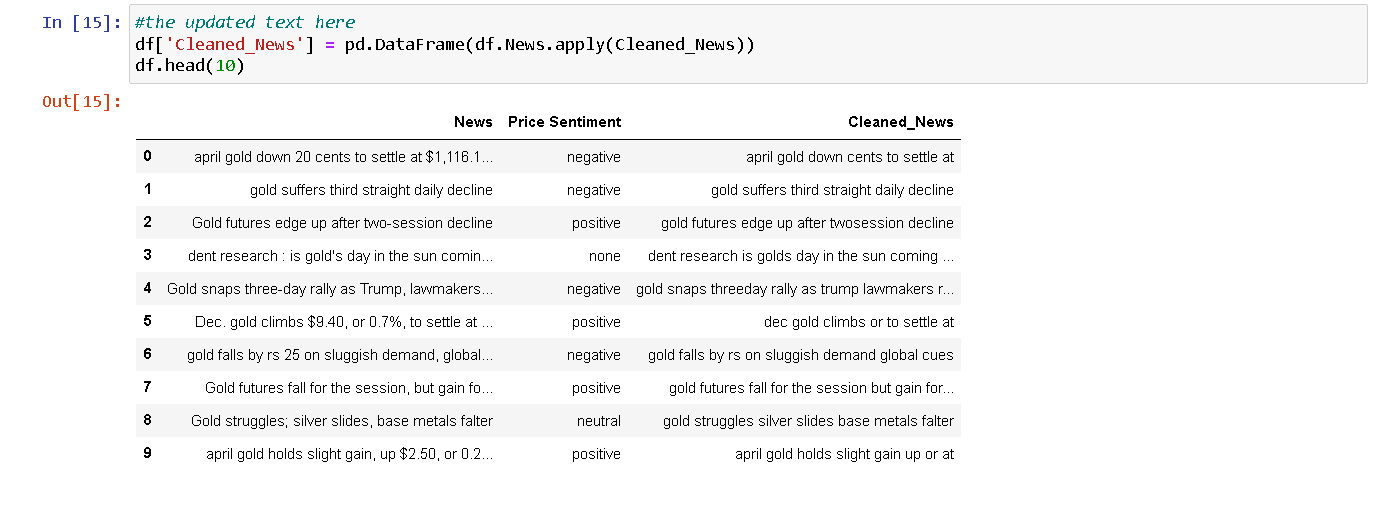
It is one of the most widely used NLP libraries in Python. It provides tools for tokenization, stemming, lemmatization, stop-word removal, and more.

* **Re (regular expressions)** –

This module provides support for working with regular expressions. It is commonly used to perform string operations such as removing punctuation, white spaces, and other non-alphanumeric characters.



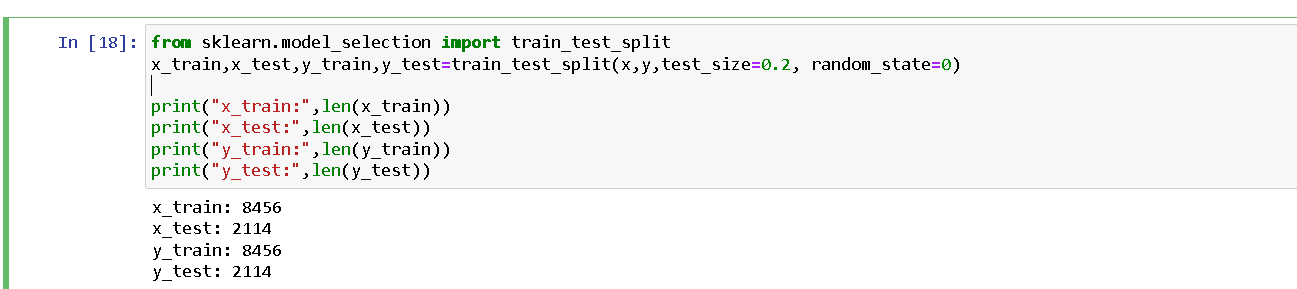
**After Text Pre-Processing:-**

After text pre-processing, the data will typically look much cleaner and more manageable. The specific form it takes will depend on the specific operations performed during pre-processing and here is below fig after cleaning the text

**Splitting data into train and test**

Now let’s split the Dataset into train and test sets. First split the dataset into x and y and then split the data set

Here x and y variables are created. On x variable, df is passed with dropping the target variable. And on y target variable is passed. For splitting training and testing data we are using train\_test\_split() function from sklearn. As parameters, we are passing x, y, test\_size, random\_state.



# Milestone 4: Model Building

## Activity 1: Training the model in multiple algorithms

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 three classification algorithms. The best model is saved based on its performance.

## Activity 1.1: Logistic Regression model

A function named Logistic regression is created and train and test data are passed as the parameters. Inside the function, Logistic Regression algorithm is initialised and training data is passed to the model with the .fit() function. Test data is predicted with .predict() function and saved in a new variable. For evaluating the model with confusion matrix and accuracy score

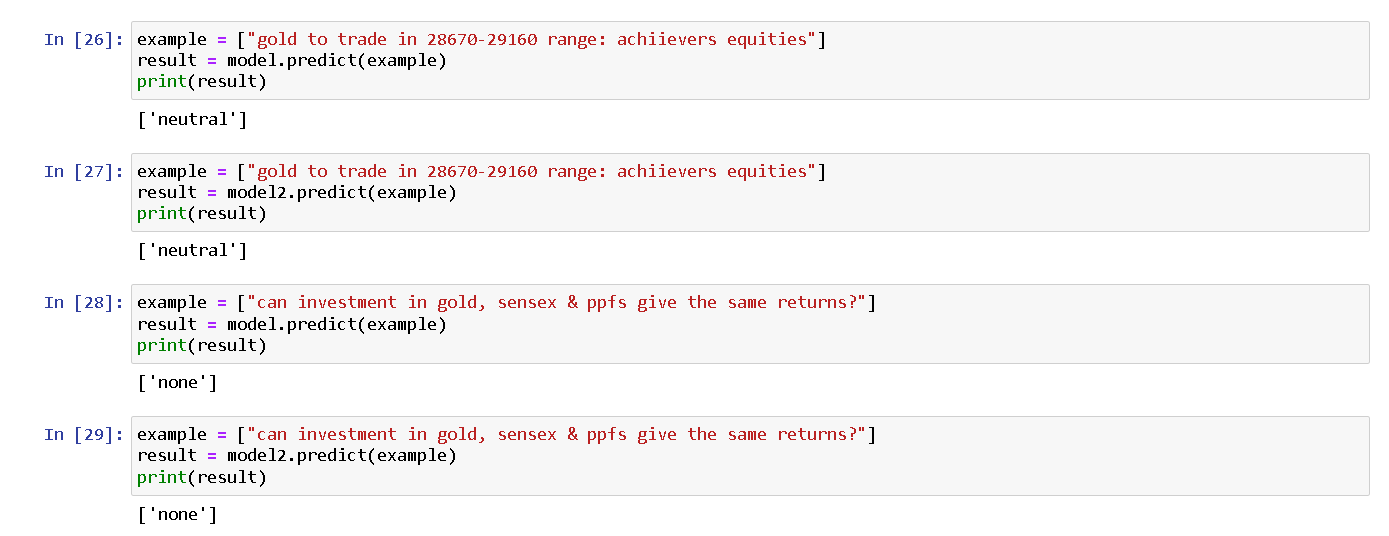
The Tf-idf (Term frequency- inverse document frequency) is a commonly used numerical representation of text data that is    used in NLP and IR tasks. The goal of Tf-idf is to represent the importance of a word in a document while taking into account the frequency of the word in the entire corpus.

  Using a pipeline has several benefits. First, it makes the process of building a machine learning model more efficient, as the steps can be automated and repeated with ease. Second, it reduces the risk of errors, as each step in the pipeline is clearly defined and less prone to manual mistakes.

## Activity 1.2: SVM (Support Vector machine)

A function named svm is created and train and test data are passed as the parameters. Inside the function, svc algorithm is initialized and training data is passed to the model with fit() function. Test data is predicted with predict() function and saved in a new variable. For evaluating the model, a confusion matrix and classification report is done. and it performed same as logistic regression model here we using the svm algorithms remaining all are the same process as above

## Activity 2: Testing the model

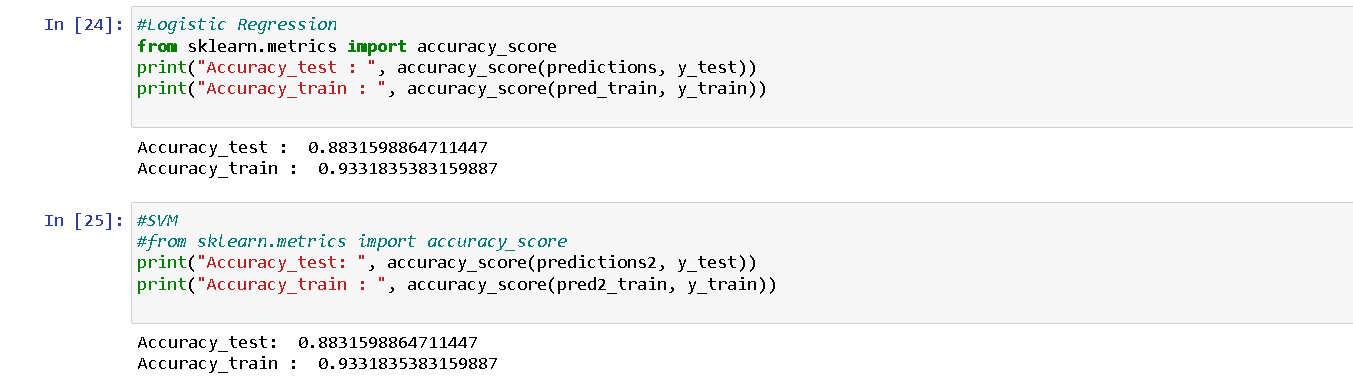
Here we have tested with Logistic regression and Svm algorithms. With the help of predict() function.

# Milestone 5: Performance Testing

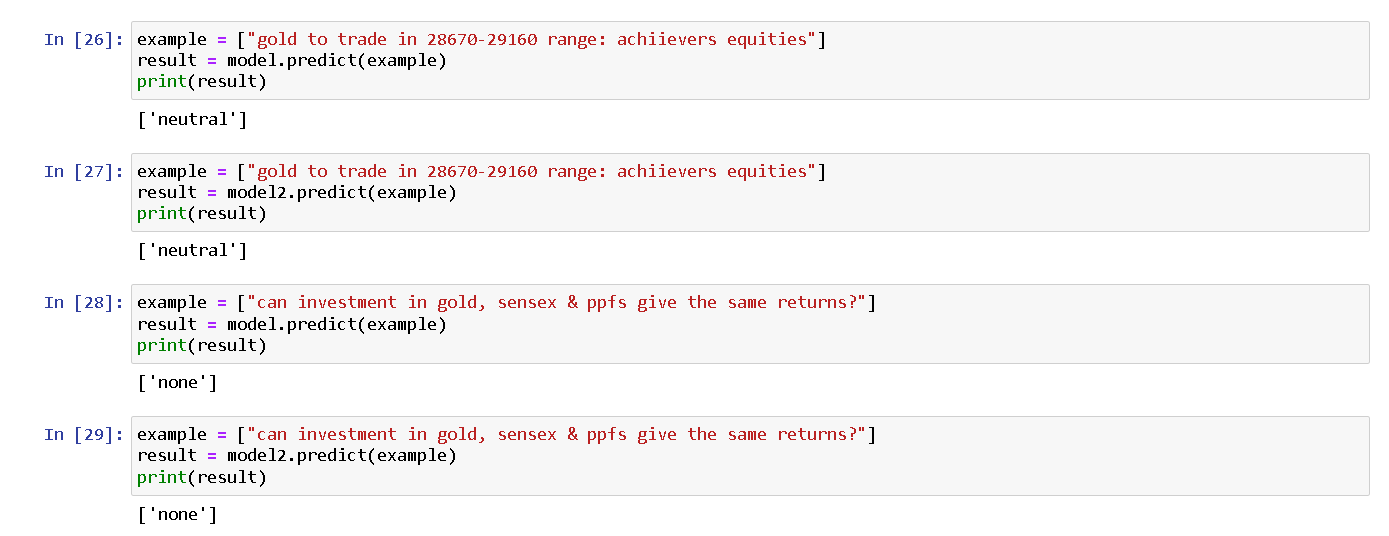
## Activity 1: Testing model with multiple evaluation metrics

Multiple evaluation metrics means evaluating the model's performance on a test set using different performance measures. This can provide a more comprehensive understanding of the model's strengths and weaknesses. We are using evaluation metrics for classification tasks including accuracy, precision, recall, support and F1-score.

## Activity 1.1: Compare the model

For comparing the below two models, with their accuracy score

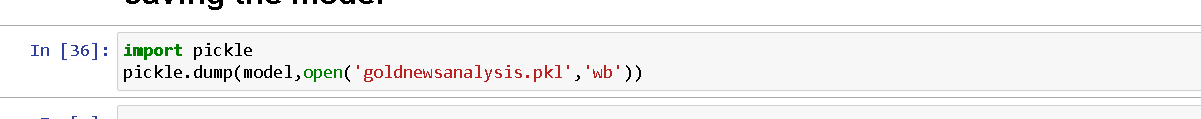
## Activity 2: Comparing model accuracy before & after applying hyperparameter tuning

After seeing, the results of models are displayed as output. From the two models which are Logistic Regression and SVM both models are performing well & Hyperparameter tuning For this project (it is not required)

# Milestone 6: Model Deployment

## Activity 1: Save the best model

Saving the best model after comparing its performance using different evaluation metrics means selecting the model with the highest performance This can be useful in avoiding the need to retrain the model every time it is needed and also to be able to use it in the future.



## Activity 2: Integrate with Web Framework

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

* Building HTML Pages
* Building server-side script
* Run the web application

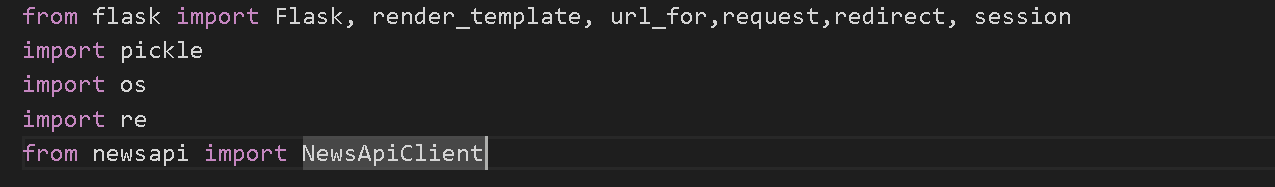
**Activity 2.1: Building Html Pages:**

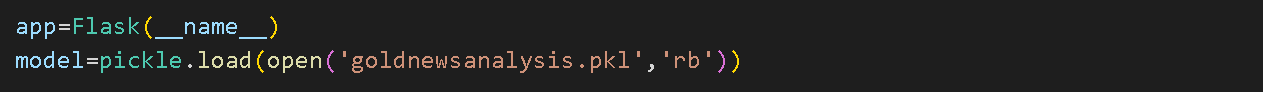
For this project create two HTML files namely

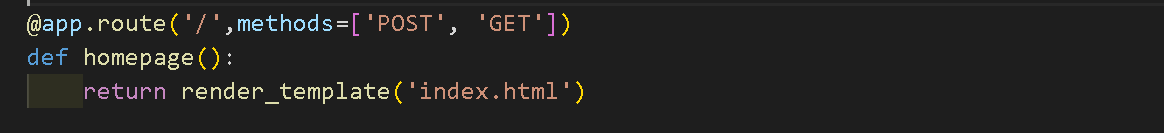
* Index.html

and save them in the templates folder. Refer this [entire files link](https://drive.google.com/file/d/1y1X2Cl506vQMZpeWACwOux6tcXzLWwVO/view?usp=sharing) for templates, static and python file

**Activity 2.2: Build Python code:**

Import the libraries

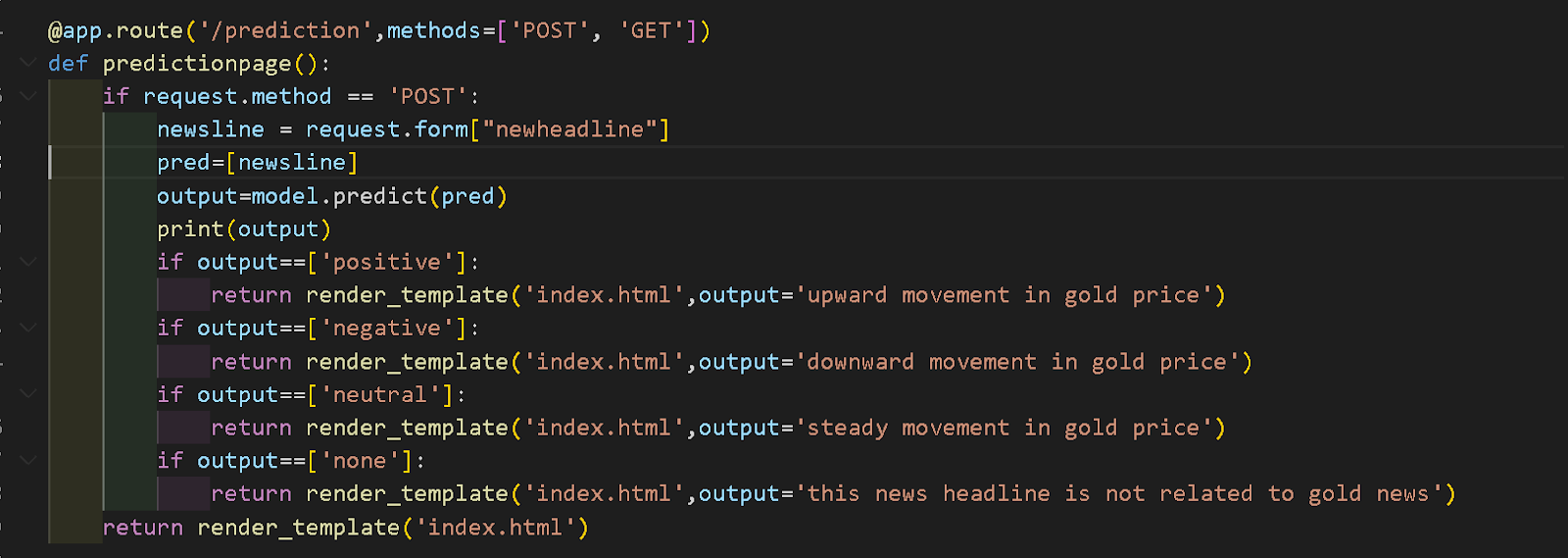
Load the saved model. Importing the flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as argument.

Render HTML page:

Here we will be using a declared constructor to route to the HTML page which we have created earlier.

In the above example, ‘/’ URL is bound with the index.html function. Hence, when the home page of the web server is opened in the browser, the html page will be rendered.

Whenever you enter the values from the html page the values can be retrieved using POST Method.

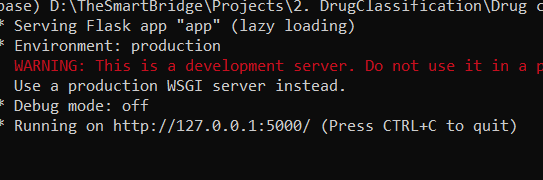
Retrieves the value from UI:

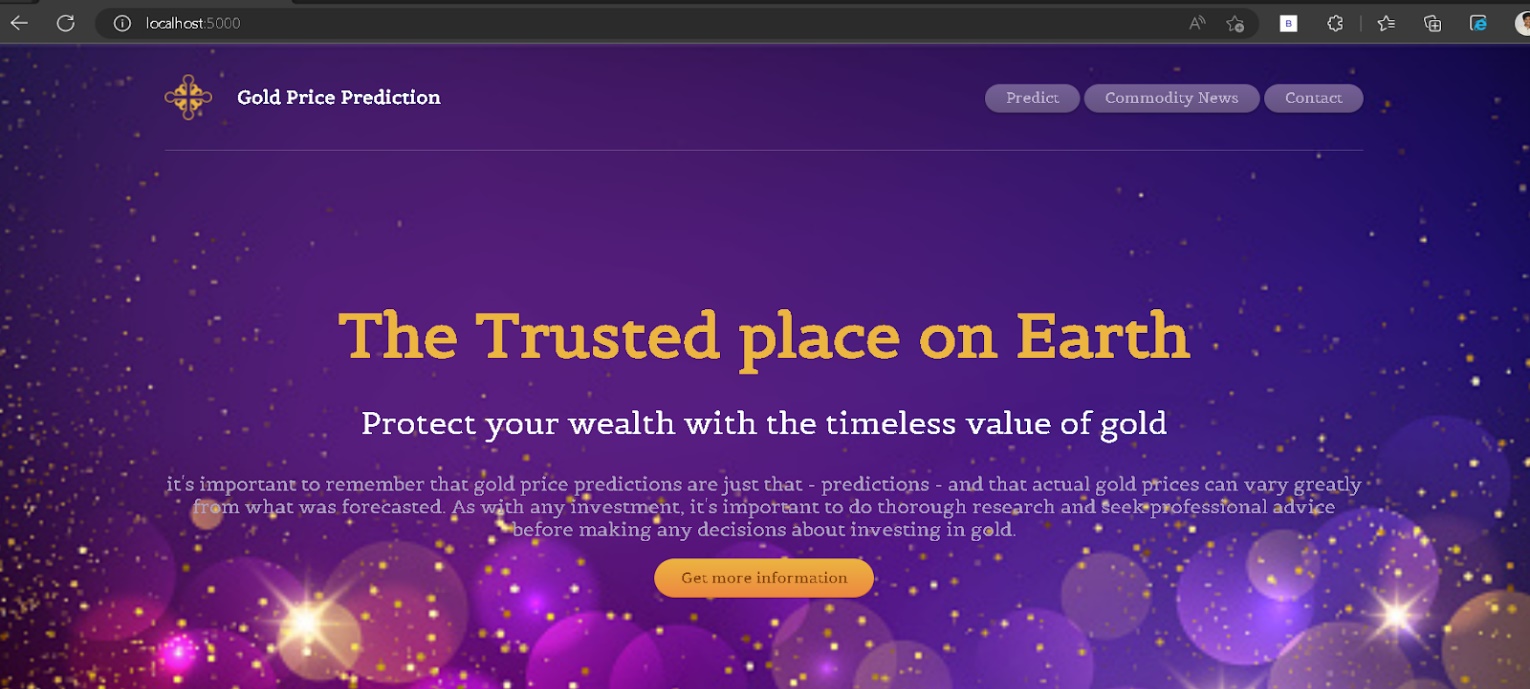
Here we are routing our app to predict() function. This function retrieves all the values from the HTML page using Post request. That is stored in an array. This array is passed to the model.predict() function. This function returns the prediction. And this prediction value will be rendered to the text that we have mentioned in the submit.html page earlier.

Main Function:

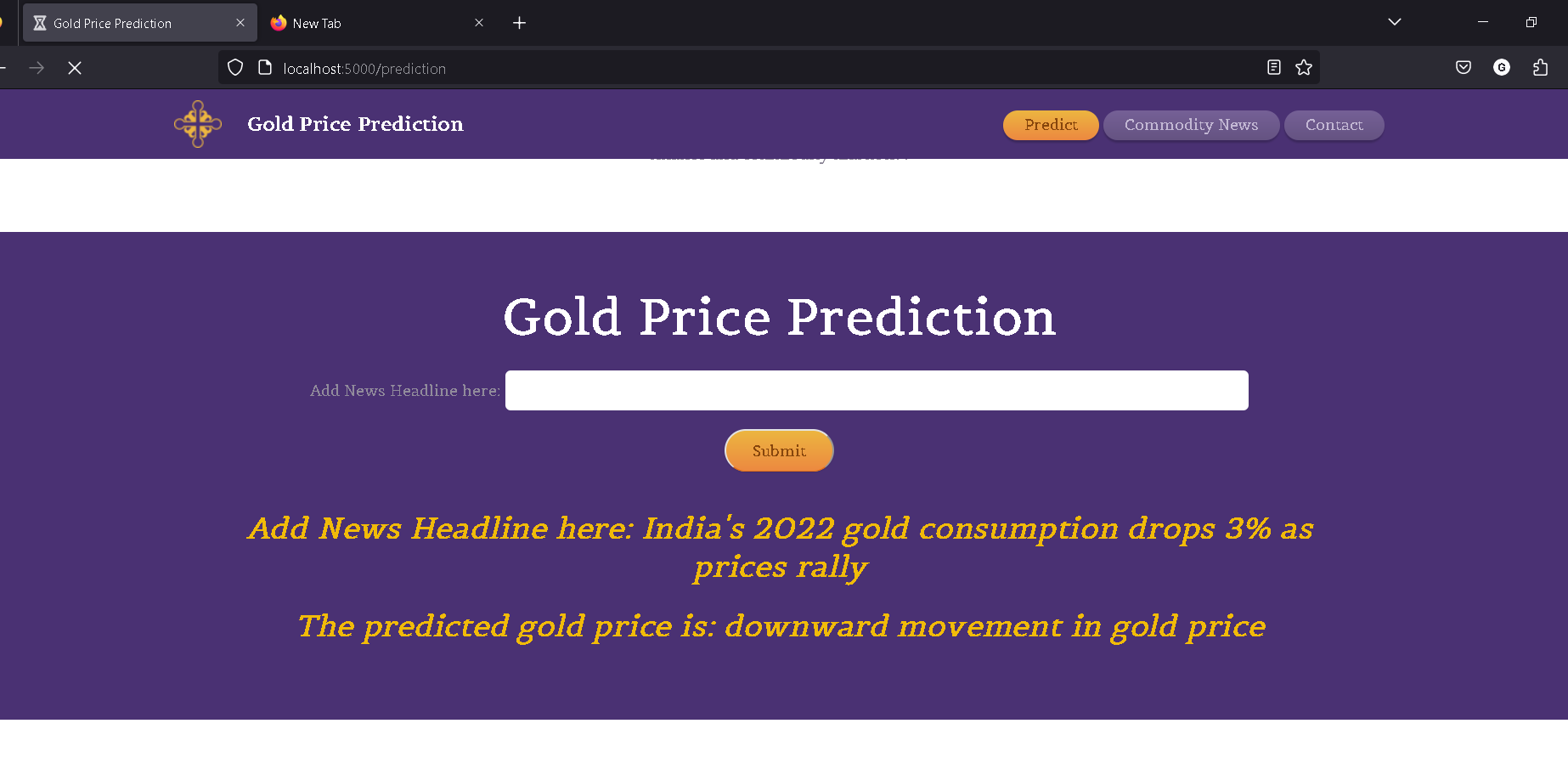
## Activity 2.3: Run the web application

* Open anaconda prompt from the start menu
* Navigate to the folder where your python script is.
* Now type “python app.py” command
* Navigate to the localhost where you can view your web page.
* Click on the predict button from the top left corner, enter the inputs, click on the submit button, and see the result/prediction on the web.



 Now, Go the web browser and write the localhost url (http://127.0.0.1:5000) to get the below result

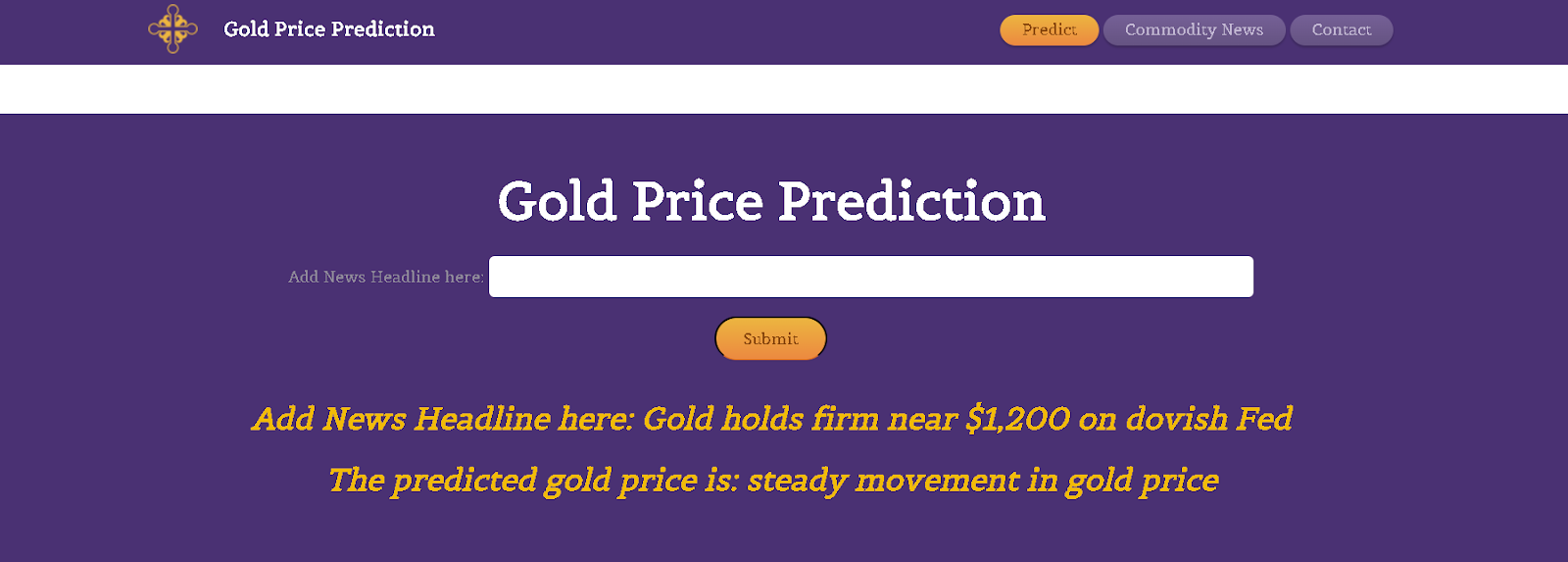
Downward movement results:-



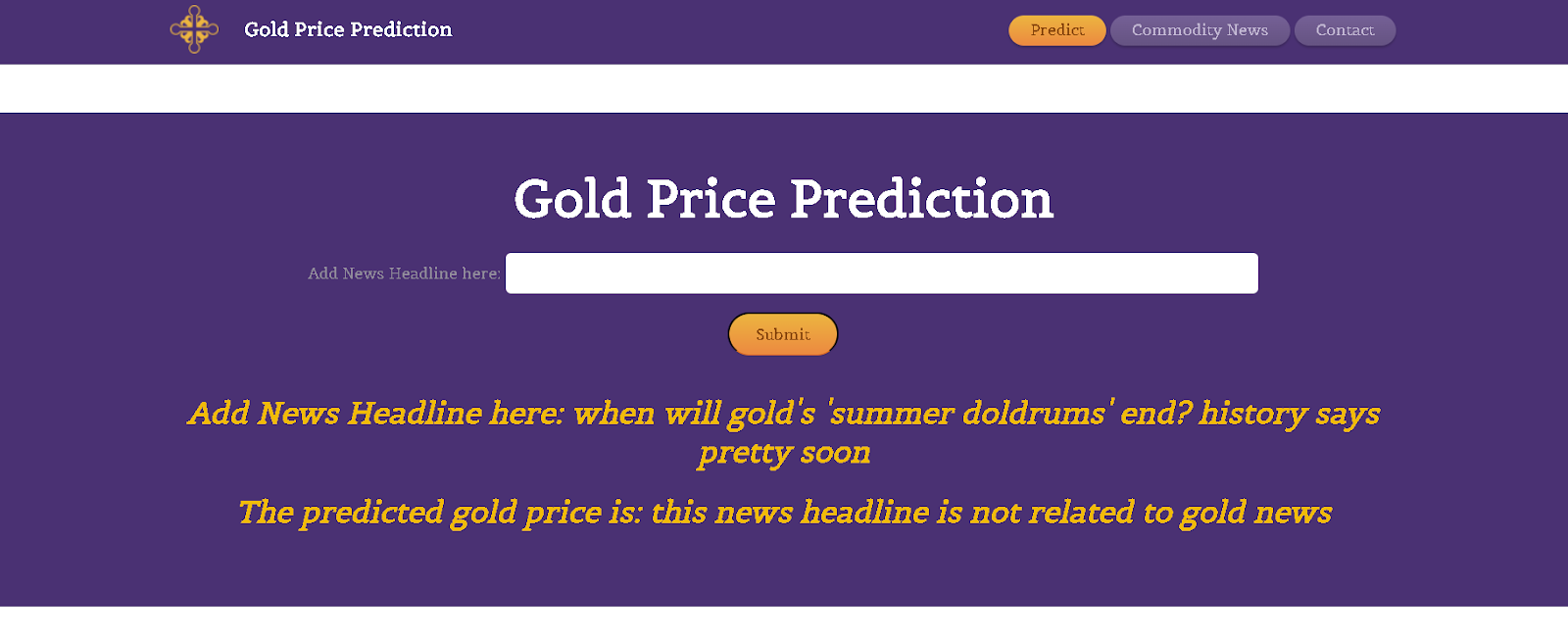
Upward movement results: -



Steady movement results: -



Title is not related results:-



Current commodity news results:-

