1. Import Required Libraries

Code:

import nltk

import pandas as pd

from nltk.stem import PorterStemmer, WordNetLemmatizer

* **nltk:** Natural Language Toolkit, a Python library used for text processing.
* **pandas:** A powerful data manipulation library used for handling structured data like CSV, Excel, etc.
* **PorterStemmer:** A class from nltk.stem used to perform stemming, which reduces words to their root form (e.g., "running" → "run").
* **WordNetLemmatizer:** A class from nltk.stem used for lemmatization, which reduces words to their dictionary form (e.g., "running" → "run").

1. Define File Paths

Code:

cleaned\_file\_path = 'D:/AIUB/Sentiment Analysis Research/Data Sets/SWR Hotel Reviews.xlsx'

lemmatized\_file\_path = 'D:/AIUB/Sentiment Analysis Research/Data Sets/Lemmatized Hotel Reviews.xlsx'

* **cleaned\_file\_path:** The file path where the cleaned reviews (after stopword removal) are stored.
* **lemmatized\_file\_path:** The file path where the lemmatized reviews will be saved after processing.

1. Load the Cleaned Reviews Data

Code:

cleaned\_reviews = pd.read\_excel(cleaned\_file\_path)

* **pd.read\_excel():** Loads the Excel file into a Pandas DataFrame, allowing further data manipulation.
* **cleaned\_reviews:** A variable holding the DataFrame that contains the cleaned reviews (after stopword removal).

1. Initialize Stemming and Lemmatization

Code:

stemmer = PorterStemmer()

lemmatizer = WordNetLemmatizer()

* **stemmer = PorterStemmer():** Initializes an instance of the PorterStemmer to be used for stemming words.
* **lemmatizer = WordNetLemmatizer():** Initializes an instance of WordNetLemmatizer for lemmatizing words..

1. Define the Function for Stemming

Code:

def stem\_words(tokens):

return [stemmer.stem(word) for word in tokens]

* **stem\_words(tokens):** A custom function that takes a list of words (tokens) as input and applies stemming to each word.
* **stemmer.stem(word):** Applies the stemming algorithm to each word in the input list. It reduces each word to its root form.

1. Define the Function for Lemmatization

Code:

def lemmatize\_words(tokens):

return [lemmatizer.lemmatize(word) for word in tokens]

* **lemmatize\_words(tokens):** A custom function that takes a list of words (tokens) as input and applies lemmatization to each word.
* **lemmatizer.lemmatize(word):** Applies the lemmatization algorithm to each word in the input list, reducing it to its dictionary (base) form.

1. Convert String Representation of Lists to Actual Lists

Code:

cleaned\_reviews['Cleaned Reviews'] = cleaned\_reviews['Cleaned Reviews'].apply(lambda x: eval(x))

* **cleaned\_reviews['Cleaned Reviews']:** Refers to the column in the DataFrame that contains tokenized words stored as strings (e.g., "[word1, word2]").
* **eval(x):** Converts the string representation of a list into an actual Python list, so further processing can be done on the tokenized reviews.
* .**apply(lambda x: eval(x)):** This applies the eval function to each row in the Cleaned Reviews column, converting the string to a list.

1. Perform Stemming or Lemmatization

Code:

# cleaned\_reviews['Stemmed Reviews'] = cleaned\_reviews['Cleaned Reviews'].apply(stem\_words)

cleaned\_reviews['Lemmatized Reviews'] = cleaned\_reviews['Cleaned Reviews'].apply(lemmatize\_words)

* **cleaned\_reviews['Stemmed Reviews']:** (commented out) This line would apply the stem\_words function to each review in the Cleaned Reviews column if stemming were being used. It creates a new column Stemmed Reviews in the DataFrame with stemmed words.
* **cleaned\_reviews['Lemmatized Reviews']:** Applies the lemmatize\_words function to each review in the Cleaned Reviews column. It creates a new column Lemmatized Reviews in the DataFrame containing the lemmatized words (words in their base form).

1. Save the Lemmatized Reviews to a New Excel File

Code:

cleaned\_reviews[['Lemmatized Reviews']].to\_excel(lemmatized\_file\_path, index=False, sheet\_name='Dhaka Regency')

* **[['Lemmatized Reviews']]:** Selects only the Lemmatized Reviews column to be saved to the new Excel file.
* **.to\_excel():** Saves the selected column(s) to the Excel file specified by lemmatized\_file\_path.
* **index=False:** Ensures that the row numbers (indices) are not saved in the Excel file.
* **sheet\_name='Dhaka Regency':** Specifies the sheet name where the data will be saved in the Excel file as "Dhaka Regency".

1. Print Confirmation

Code:

print(f"Lemmatized reviews saved to {lemmatized\_file\_path}")

* **print(f"..."):** This prints a confirmation message to the console, indicating that the lemmatized reviews have been successfully saved to the specified file path.

Summary of Workflow

1. Load the cleaned reviews from the Excel file.
2. Define and initialize a stemmer and a lemmatizer.
3. Convert the string representations of tokenized reviews to actual lists.
4. Apply either stemming or lemmatization to each review.
5. Save the processed reviews (lemmatized in this case) to a new Excel file.