**Intensity Analysis (Build your own model using NLP and Python)**

**Project Setup and Data Preparation**

**Import Necessary Libraries:**

import pandas as pd

import numpy as np

import nltk

from nltk.tokenize import word\_tokenize

from nltk.corpus import stopwords

from nltk.stem import PorterStemmer

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.model\_selection import train\_test\_split

from sklearn.naive\_bayes import MultinomialNB

from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score

Download NLTK Data:

nltk.download('punkt')

nltk.download('stopwords')

nltk.download('wordnet')

Load and Preprocess Data:

def preprocess\_text(text):

# Tokenization

words = word\_tokenize(text)

# Remove stop words

stop\_words = set(stopwords.words('english'))

words = [word for word in words if word not in stop\_words]

# Stemming

stemmer = PorterStemmer()

words = [stemmer.stem(word) for word in words]

return ' '.join(words)

# Assuming your data is in a CSV file with columns 'text' and 'intensity'

data = pd.read\_csv('your\_data.csv')

data['text'] = data['text'].apply(preprocess\_text)

Feature Engineering and Model Training

Create TF-IDF Features:

vectorizer = TfidfVectorizer()

X = vectorizer.fit\_transform(data['text'])

y = data['intensity']

Split Data into Training and Testing Sets:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

Train a Naive Bayes Model:

model = MultinomialNB()

model.fit(X\_train, y\_train)