**Program:  
  
import numpy as np**

**from hmmlearn import hmm**

**# Step 1: Define Model Parameters**

**n\_states = 2 # Number of hidden states (Rainy and Sunny)**

**# Transition matrix (A): Probability of transitioning from one state to another**

**trans\_matrix = np.array([[0.7, 0.3], [0.4, 0.6]])**

**# Emission matrix (B): Probability of observing an emission given the current state**

**emission\_matrix = np.array([[0.1, 0.4, 0.5], [0.6, 0.3, 0.1]])**

**# Initial state probabilities (π): Probability distribution of starting in each state**

**initial\_probs = np.array([0.6, 0.4])**

**# Step 2: Create HMM Model**

**model = hmm.MultinomialHMM(n\_components=n\_states,**

**startprob\_prior=initial\_probs,**

**transmat\_prior=trans\_matrix,**

**n\_iter=100)**

**# Step 3: Generate Training Data (for simplicity, you can use a pre-existing dataset)**

**# Observations: 0 - Umbrella, 1 - Jacket, 2 - T-shirt**

**train\_data = np.array([[0, 1, 2, 0, 1, 2, 0, 2, 1]])**

**# Reshape the array if needed**

**train\_data = train\_data.reshape(-1, 1)**

**# Step 4: Fit the Model**

**model.fit(train\_data)**

**# Step 5: Predict States for a New Sequence**

**new\_data = np.array([[0, 2, 1]]) # Umbrella, T-shirt, Jacket**

**new\_data = new\_data.reshape(-1, 1)**

**predicted\_states = model.predict(new\_data)**

**# Map numerical predictions to weather states**

**weather\_states = ['Rainy', 'Sunny']**

**predicted\_states\_text = [weather\_states[state] for state in predicted\_states]**

**# Display Results**

**print("Predicted Weather States:", predicted\_states\_text)**

**Output:   
  
Predicted Weather States: ['Rainy', 'Sunny', 'Rainy']**