HMM-Forward Algorithm

BEIJING INSTITUTE OF TECHNOLOGY

PIMENOV GLEB

1820243077

I hereby promise that the experimental report and codes written by me are independently completed without any plagiarism or copying of others' homework. All the views and materials involving other classmates have been annotated. If there is any plagiarism or infringement of others' intellectual property rights, I will bear all the consequences arising from it.

HMM-Forward Algorithm

1 Experiment Introduction

Hidden Markov Models (HMMs) are probabilistic models widely used in various fields, including speech recognition, bioinformatics, and natural language processing. The Forward algorithm is a fundamental technique for computing the probability of observing a sequence of symbols given an HMM.

In this experiment, we aim to apply the Forward algorithm to compute the probability of observing a specific sequence, TAGA, using a given HMM. We will implement the Forward algorithm and use it to calculate the probability of the sequence TAGA based on the transition and emission probabilities provided.

2 Experiment Objectives

1. To understand the concept and implementation of the Forward algorithm in Hidden Markov Models.
2. To implement the Forward algorithm using Python programming language.
3. To compute the probability of observing a sequence (TAGA) based on a given Hidden Markov Model.
4. To validate the correctness of the implemented Forward algorithm by comparing the computed probability with known results or theoretical expectations.

3 Relevant Theories and Knowledge

Hidden Markov Models (HMMs):

Hidden Markov Models are probabilistic models used to model sequences of observable symbols (emissions) generated by a sequence of hidden states. The model consists of transition probabilities between hidden states and emission probabilities for each hidden state generating observable symbols.

Forward Algorithm:

The Forward algorithm is a dynamic programming algorithm used to compute the probability of observing a sequence of symbols given an HMM. It calculates the forward probabilities, which represent the probability of being in a particular state at a specific time step and observing the sequence up to that point.

The Forward algorithm proceeds iteratively through each time step, updating the forward probabilities based on the previous probabilities and the transition and emission probabilities. It utilizes the principle of dynamic programming to efficiently compute the probabilities.

4 Experimental Tasks and Grading Criteria

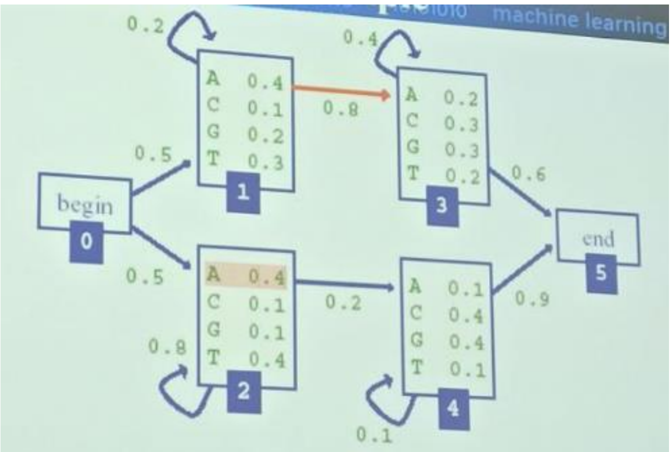
No.	**Task Name**	**Specific Requirements**	**Grading Criteria (100-point scale)**
1	HMM-Forward Algorithm	Development language: Python	100

5 Experimental Conditions and Environment

Requirements	Name	Version	Remarks
Programming Language	Python	3.12	
Development Environment	windows	11	
Third-party toolkits/libraries/plugins			

Other Tools	Jupyter notebook		
Hardware Environment	I5 12XXX 8GB RAM		

6 Experimental Data and Description

Attribute (Entry)	Content
Dataset Name	**TAGA sequence from class**
Dataset Origin	Class work 
Main Contents of the Dataset	Transition probabilities, emission probabilities, initial probabilities, and a sequence of observations
Dataset File Format	List

7 Experimental Steps and Corresponding Codes

Step number	1
Step Name	Defining probabilities
Step Description	Define transition probabilities, define emission probabilities, define the sequence
Code and Explanation	transition_probs = [[0.0, 0.5, 0.5, 0.0, 0.0, 0.0], # From state 0

	[0.0, 0.2, 0.0, 0.8, 0.0, 0.0], # From state 1 [0.0, 0.0, 0.8, 0.0, 0.2, 0.0], # From state 2 [0.0, 0.0, 0.0, 0.4, 0.0, 0.6], # From state 3 [0.0, 0.0, 0.0, 0.0, 0.1, 0.9], # From state 4 [0.0, 0.0, 0.0, 0.0, 0.0, 0.0], # From state 5 (ending state)] emission_probs = [{'T': 0.0, 'A': 0.0, 'G': 0.0, 'C': 0.0}, # State 0 (starting state) {'T': 0.3, 'A': 0.4, 'G': 0.2, 'C': 0.1}, # State 1 {'T': 0.4, 'A': 0.4, 'G': 0.1, 'C': 0.1}, # State 2 {'T': 0.2, 'A': 0.2, 'G': 0.3, 'C': 0.3}, # State 3 {'T': 0.1, 'A': 0.1, 'G': 0.4, 'C': 0.4}, # State 4 {'T': 0.0, 'A': 0.0, 'G': 0.0, 'C': 0.0}, # State 5 (ending state)] sequence = ['T', 'A', 'G', 'A'] # TAGA

Step number	**2**
Step Name	Forward matrix
Step Description	Defining forwarding matrix, initialize first row of forward matrix using initial probabilities and emission probabilities

Code and Explanation	num_states = len(transition_probs)
	num_obs = len(sequence)
	forward_matrix = [[None] * num_states for _ in range(num_obs + 1)]
	for state in range(num_states):
	    # Initial state has probability 1, others 0
	    if state == 0:
	        forward_matrix[0][state] = 1.0
	    else:

	forward_matrix[0][state] = 0.0

Step number	3
Step Name	Forward algorithm
Step Description	Code takes transition probabilities, emission probabilities, initial probabilities, and a sequence of observations as input, and computes the total probability of the sequence using the Forward algorithm. Recursively compute forward probabilities for each position in the sequence. Compute the total probability of the sequence by summing forward probabilities of all states at the last position

Code and Explanation	for t in range(1, num_obs + 1):
	for j in range(num_states):
	# Compute the sum of probabilities from all previous states to the current state
	prob_sum = sum(forward_matrix[t - 1][i] * transition_probs[i][j] for i in range(num_states))
	forward_matrix[t][j] = prob_sum * emission_probs[j][sequence[t - 1]]
	# Using special "if" because fifth state is the end state without new letters
	if t == 4 and j == 5:
	forward_matrix[t][j] = sum(forward_matrix[t][i] * transition_probs[i][j] for i in range(num_states))

Output results and Interpretation	Probability of sequence TAGA: 0.00046224

8 Experiment Difficulties and Precautions

9 Experiment Results and Interpretation

Handwrite calculations:

$$f_1(1) = 0{,}3 \cdot (1 \cdot 0{,}5 + 0 \cdot 0{,}2) = 0{,}15$$

$$f_1(2) = 0{,}4 \cdot (0 \cdot 0{,}5 + 0{,}15 \cdot 0{,}2) = 0{,}012$$

$$f_1(3) = 0{,}2 \cdot (0 \cdot 0{,}5 + 0{,}012 \cdot 0{,}2) = 0{,}00048$$

$$f_2(1) = 0{,}4 \cdot (1 \cdot 0{,}5 + 0 \cdot 0{,}8) = 0{,}2$$

$$f_2(2) = 0{,}4 \cdot (0 \cdot 0{,}5 + 0{,}2 \cdot 0{,}8) = 0{,}064$$

$$f_2(3) = 0{,}1 \cdot (0 \cdot 0{,}5 + 0{,}064 \cdot 0{,}8) = 0{,}00512$$

$$f_3(2) = 0{,}2 \cdot (0 \cdot 0{,}4 + 0{,}15 \cdot 0{,}8) = 0{,}024$$

$$f_3(3) = 0{,}3 \cdot (0{,}012 \cdot 0{,}8 + 0{,}024 \cdot 0{,}4) = 0{,}00576$$

$$f_3(4) = 0{,}2 \cdot (0{,}00576 \cdot 0{,}4 + 0{,}00048 \cdot 0{,}8) = 0{,}0005376$$

$$f_4(2) = 0{,}1 \cdot (0{,}2 \cdot 0{,}2 + 0 \cdot 0{,}1) = 0{,}004$$

$$f_4(3) = 0{,}4 \cdot (0{,}064 \cdot 0{,}2 + 0{,}004 \cdot 0{,}1) = 0{,}00528$$

$$f_4(4) = 0{,}1 \cdot (0{,}00528 \cdot 0{,}1 \; + 0{,}00512 \cdot 0{,}2) = 0{,}0001552$$

$$f_5(4) = 0{,}6 \cdot 0{,}0005376 + 0{,}9 \cdot 0{,}0001552 = 0{,}00046224$$

Answer: 0,00046224

The HMM-forward algorithm was implemented, data from the class work was used, to calculate the final probability, all intermediate probabilities had to be calculated even in cases where they were obviously multiplied by 0. To obtain the result, it was necessary to take only the probability from the last state, since in theory it is possible to switch from other states to the TAGA sequence and the program calculates such probabilities, but we need such a sequence only in the final state.

What is more, the results of manual calculations and the work of the program converged.

10 References

11 Experiment-related Metadata

Metadata Item	Content
Case name	
Applicable course name	Machine learning Fundamentals
Keyword/Search Term	HMM, forward algorithm
AliTianchi URI	

12 Remarks and Others