Implementation of Hidden Markov Model with Python Programming Language

Sashi Novitasari[1]
Informatics Study Program
School of Electrical Engineering and Informatics
Bandung Institute of Technology, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13514027@std.stei.itb.ac.id

***Abstract* – Python is one of high-level programming laguages that widely used in various kinds of program. It can be use to implement the solution of various kind of problem, such as pattern recognition. Pattern recognition is widely used in machine learning domain to learn the given data and output an result based on the learning process. One of widely used model in machine learning is hidden Markov model. This model is widely used because it can be applied in various domains of problem, mainly to recognize hidden pattern based on givem information or model.**

Index Terms – Python, general-purpose programming, hidden Markov model, patten recognition.

I. INTRODUCTION

Nowadays, various programming language had been developed and used in various field that uses computer to support it works. As in 2013, there were 256 programming that exist in this world [1]. Among those languages, one of programming languages that widely used is Python programming language. In TIOBE Index for April 2017, Python ranked as 5th in the popularity index of programming language [2]. Python is a programming language that started as scripting language like PHP. Now it is widely used in web and desktop application. Python categorized as dynamic language and has many standard library that can be used to build complex application. In some cases, Python works more efficiently than other language.

Python is a general-purpose programming language, so it can be used in various application domain. One of domain that it can be applied is machine learning. Machine learning is an algorithm or system than can figure out how to perform important tasks by generalizing from given examples [4]. (Domigos, 2012, p. 78) Machine learning uses model from example data to output a result. One of model that can be used in machine learning is hidden Markov model. Hidden Markov model is a statistical model that widely used in pattern recognition such as speech recognition and bioinformatics[5].

This paper mainly discuss the implementation of hidden Markov model to solve a simple problem using Python programming language. The implemented hidden Markov model only focused on solving two of three fundamental problems that can be solved by using hidden Markov model, that is the determination of likehood of observed sequence of pattern based of given data model and determination of optimal sequence of state in given Markov process.

II. LITERATURE STUDY

A. Python Programming Language

Python is a high-level programming language that widely used by programmers for general purpose. Python was first developed by Guido Van Rossum and released its first version, which is Python 1.6, in 2000 [3]. The developer of Python continuously releases its newer version and in March 2017 they released the newest one, Python 3.6.1.

As in April 2017, Python denoted as 5th most popular programming language that used by programmers according to TIOBE Index for April 2017[2]. Python is widely used because of its characteristics. The characteristics or features of Python are listed as follows.

a. Fast and powerful
   Python has internal standard library that provides all things that needed to create program with basic to advanced operation [3]. It allows programmers to write short lines of codes to create advanced function. For example, we can create a web server only by writing 3 lines of codes.
b. Supports other technologies
   Python support other technologies such as COM and .Net so it will be easier to integrate Python with those technologies [3].
c. Portable
   Python script can be used in various operating system, for example Windows, Linux, and Mac.
d. Simple
   Python's language structure is simple and minimalistic, so programmers able concentrate to the solution of the problem rather than the language structure[3][6].
e. Open source
   Python can be used, changed, and distributed without a limit.

Python allows programmers to develop programs with any kind of programming paradigm because Python is a multi-paradigm programming laguage [3]. In other word, programmers able to develop an application with object-oriented programming and other application with structural

programming using Python. Figure 1 is an example of code that written in Python.

```
class HelloWorld:
    msg = "Hello World!"

    def __init__(self):
        print(self.msg)

Hello = HelloWorld()
```

Figure 1. Example of code written in Python

Python can be used to develop various kinds of program. Among those, the most common programs that can be developed using Python are system programming, graphical user interface, network programming, components integrity, database programming, and numerical programming [3].

B. *Hidden Markov Model*

Hidden Markov model or HMM is a statistical Markov model that widely used in various field. It is a ubinquintous tool that used to modelling time series data[7]. (Ghahramani, 2001, p. 1) Usually HMM is used in machine learning and pattern recognition fields such as speech recognition systems and molecular biology computational system. Rabiner (1986, p. 5) stated that HMM is a doubtly stochastic process with an underlying stochastic process that is not observable, but can only be observed through another set of stochastic processes that produce the sequence of observed symbols[8]. For example, we can find sequence of weather of a day, without knowing the actual weather first, based on number of ice creams eaten by certain person on corresponding day [9].

To compute the result, HMM use following data throughout the process[10].

a. Lenght of the observation sequence (T)
b. Number of states in the model (N)
c. Number of observation symbols (M)
d. Set of distinct states of Markov process (Q)
e. Set of possible observations (V)
f. State transition probabilities matrix (A)
g. Observation probability matrix (B)
h. Initial state probability distribution (π)
i. Set of observed sequence (O)

In previous example, weather acts as state and number of ice creams acts as observed symbols.

There are three fundamental problems that can be solve by using HMM. The three fundamental problems with its solution are listed below.

a. Determine likehood of the observed sequence given the model $\lambda = (A, B, \pi)$
   In this kind of problem, we would like to find the probability $P(O|\lambda)$. To find it, we will use forward algorithm or α-pass ($\alpha_t(i)$) to compute the relevant probability of state q_i up to time t. For time or t = 0,1,2...,T-1 and state or i = 0,1,...N-1, the algorithm consist of recursive computational as follows [10].
   1. If $t = 0$, then $\alpha_t(i) = \pi_i b_i(O_0)$
   2. If t = 1,2...T-1, then
   $$\alpha_t(i) = \left[\sum_{j=0}^{N-1} \alpha_{t-1}(j) a_{ji}\right] b_i(\mathcal{O}_t)$$
   3. From equation above, $P(O|\lambda)$ can be calculated as
   $$P(\mathcal{O}|\lambda) = \sum_{i=0}^{N-1} \alpha_{T-1}(i).$$

b. Determine an optimal state sequence of underlying Markov process.
   In this kind of problem, we want to uncover the hidden state in hidden Markov model or find the most likely sequence of state with given model $\lambda = (A, B, \pi)$. To solve the problem, first we will use backward algorithm or β-pass ($\beta_t(i)$) to compute the relevant probability of state q_i after time t [10]. The algorithm consist of following recursive computation.
   1. If $t = T$-1, then $\beta_t(i) = 1$
   2. If $t = T$-2, T-3...0 then
   $$\beta_t(i) = \sum_{j=0}^{N-1} a_{ij} b_j(\mathcal{O}_{t+1}) \beta_{t+1}(j).$$
   To find the most likely state in a certain time, define for t = 0,1,..T-1 and i = 0,1...,N-1
   $$\gamma_t(i) = P(x_t = q_i | \mathcal{O}, \lambda).$$
   The most likely state at time t is state q_i with maximum value $\gamma_t(i)$ in corresponding time.

c. Determine the model $\lambda = (A, B, \pi)$ that maximize the probability of O with given observed sequence O and dimensions N and M.

Among three fundamental problems that can be solved by HMM, the conducted experiment focused on the first and second problem.

III. METHODS

A. *Problem*

The experiment will be based on following case. Suppose you have two friends that live with you in campus dorm. One day, they plan to go out to celebrate one of your friends' birthday. Unfortunately, you were forced to stay at dorm because you caught a cold. You didn't want them to cancel their plan, so you told them to go. Before they left you, you told them to get a picture of a flower each time they go to a park, or get a picture of a fountain each time they go the a mall. Your friends listed the places that they plan to visit; the places were consisting of several parks and malls. Their destination will be depend on weather and their mood. After they go, you noticed that the weather was sunny first, and then went dark, sunny, and finally the weather was rain when your friends came back. Before your friend gave you the pictures, they asked you to guess the route and the picture. Because you know them well and you are computer science student, you decided to find the answer by using HMM.

B. Model

From the given problem, the instruments of hidden Markov model are follows:

Q = {Park, Mall}

V = {Sunny, Cloudy, Rain}

O = {Sunny, Cloudy, Sunny, Rain}

N = 2

M = 4

T = 4

Because you know your friends very well, you define the following probabilities matrices

A =

	Park	Mall
Park	0.4	0.6
Mall	0.2	0.8

B =

	Sunny	Cloudy	Rain
Park	0.6	0.3	0.1
Mall	0.1	0.2	0.7

π =

Park	Mall
0.6	0.4

We would like to determine the sequence of places that visited by our friends based on given information that consisting of sequence of weather and probability matrices that you define. In this case, we will approach the answer by using solution in second fundamental problem that can be solved by HMM, that is determining the most likely sequence of states.

C. Python Code

In the following Python code, the numerical representation of states and observed symbols are follow.

a. State
   0 : Park
   1 : Mall
b. Observed symbol
   0 : Sunny
   1 : Cloudy
   2 : Rain

The solution of problem implemented as a Python class. The code of solution is follow.

```
class CalcHMM:
    B = [[0 for x in range(3)] for y in range(2)]
    A = [[0 for x in range(2)] for y in range(2)]
    O       = [0,1,0,2]
    state   = [0,1]
    observed= [0,1,2]
    T       = 0
    N       = 0
    M       = 0
    phi     = [0.6, 0.4]

    def __init__(self):
        #Initialize B,A,T,N, and M

    #Forward algorithm or a-pass; compute at(i)
    def FWAlgorithm(self,timeT,stateI):
        if timeT == 0:
            #Adjusted
            return self.phi[stateI]
                    *self.B[stateI][self.O[0]]
        else:
            ati = 0
            for X in range (0,self.N):
                #Adjusted
                ati += self.FWAlgorithm(timeT-1,X)
                       *self.A[X][stateI]
            ati *= self.B[stateI][self.O[timeT]]
            return ati

    #Backward algorith or b-pass; compute bt(i)
    def BWAlgorithm(self,timeT,stateI):
        if timeT == len(self.O)-1:
            return 1
        else:
            bti = 0
            for x in range (0,self.N):
                #Adjusted
                bti+= self.A[stateI][x]
                       *self.B[x][self.O[timeT+1]]
                       *self.BWAlgorithm(timeT+1,x)
            return bti

    #Compute probability of P(O|model)
    def prob_likehoodOfO(self):
        retVal = 0
        for X in range(0,self.N):
            retVal += self.FWAlgorithm(3,X)
        return retVal

    #Compute probability of a state
    #in certain t or yt(i)
    def prob_stateInT(self,timeT,stateI):
        a = self.FWAlgorithm(timeT,stateI)
        b = self.BWAlgorithm(timeT,stateI)
        return (a*b)/self.prob_likehoodOfO()

    #Determine the highest value in an array
    def maxValue(self,array):
        min = 0
        for x in range (1,len(array)):
            if array[x]>array[min]:
                min = x
        return min
    #Determine the most likely sequence of
    #state given model
    def mostLikelyStatePath(self):
        probOfState = [0,0]
        state = [0 for y in range(len(self.O))]
        for x in range (0,len(self.O)):
            print("Path ", x+1,":")
            for y in range (0, len(probOfState)):
                #Adjusted
                probOfState[y] =
                          self.prob_stateInT(x,y)
                print(" ",y,"-",probOfState[y])
            state[x] = self.maxValue(probOfState)
        return state

H = CalcHMM()
array = H.mostLikelyStatePath()
print(array)
```

The program will show the result by showing the probability of each state in each sequence of time and the most likely sequence or path of states.

IV. RESULT

The answer of given problem based on implemented solution in Python are follow.

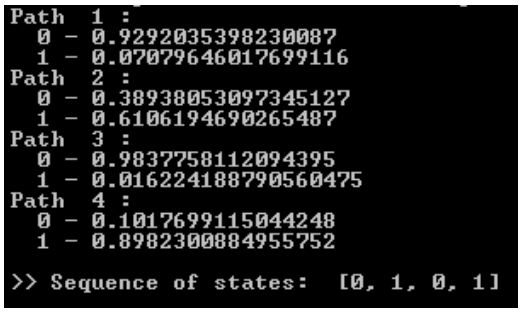

Figure 2. The solution of given problem

Based on HMM that implemented in Python, the sequence of place that visited by your friends is 0-1-0-1 or park-mall-park-mall and you received two pictures of flower and two pictures of fountain.

V. CONCLUSION

Python is one of most-used programming language that offers rich characteristics that ables programmer to develop different kind of program with ease. Python can be used to implement solution of various kind of problem, including solution to compute or recognize a pattern using HMM approach. Similar to Python that can be use in general-purpose programming, HMM can be used to solve various kind of problem too especially to reconize a pattern.

ACKNOWLEDGMENT

First, the author would like to express her highest praise to God for His guidance that allows her to complete this paper. Author would also like to express her gratitude to her parents for their support, and also Mr. Rinaldi Munir, Mrs. Ayu Purwarianti, and Mrs. Dessi Puji Lestari for their encouragement and knowledges that they gave to author so author able to complete this paper.

REFERENCES

[1] Diana, Robert. (2013). *The Big List of 256 Programming Languages*. Retrieved May 2, 2017, from https://dzone.com/articles/big-list-256-programming.

[2] *TIOBE Index for April 2017*. (2017). Retrieved May 2, 2017, from https://www.tiobe.com/tiobe-index/

[3] M. Nosrati, 'Python: An appropriate language for real world programming', *World Applied Programming*, vol. 1, no. 2, p 110-114, 2011.

[4] P. Domigos, 'A Few Useful Things to Know About Machine Learning', *Communication of the ACM*, no. 55, p. 78-87, 2012.

[5] L.E. Baum and T. Petrie, 'Statistical Inference for Probabilistic Function of Finite State Markov Chains', *The Annals of Mathematical Statistics*, vol. 37, no. 6, p. 1554-1563, 1966.

[6] A Byte of Python: Features of Python. (2005). Retrieved May 2, 2017, from https://www.ibiblio.org/g2swap/byteofpython/read/features-of-python.html

[7] Z. Ghahramani, *An Introduction to Hidden Markov Models and Bayesian Networks*. New Jersey: World Scientific Publishing Co., 2002, p. 1-2

[8] L.R. Rabiner and B.H. Juang, 'An Introduction to Hidden Markov Models', *IEEE ASSp Magazine*, no. 3, p 4-16, 1986.

[9] D. Jurafsky and J.H. Martin. *Speech and Language Processing*. *New Jersey:* Prentice Hall, 2016.

[10] M. Stamp. (2015). *A Revealing Introduction to Hidden Markov Model*. Retrieved 3 May, 2017, from https://www.cs.sjsu.edu/~stamp/RUA/HMM.pdf

Bandung, May 4 2017

Sashi Novitasari