### Outcome Likelihood Problem

*Find the probability that an HMM emits a given string.*

**Input:** A string *x* emitted by an HMM (*Σ*, *States*, *Transition*, *Emission*).

**Output:** The probability Pr(*x*) that the HMM emits *x*.

**Input Format.** The first line of the input contains the outcome string *x*. The second line of the input is “--------” (a delimiter). The third line of the input is the list of symbols in the alphabet *Σ* (space-separated). The fourth line of the input is “--------” (a delimiter). The fifth line of the input is the list of states *States* (space-separated). The sixth line of the input is “--------” (a delimiter). The next |*States*|+1 lines are the transition matrix *Transition*, depicted as a tab-delimited |*States*| by |*States*| matrix as shown in the sample dataset. The next line is “--------” (a delimiter). The remaining lines are the emission matrix *Emission*, depicted as a tab-delimited |*States*| by |*Σ*| matrix as shown in the sample dataset. You may assume that transitions from the initial state occur with equal probability.

**Output Format.** The probability Pr(*x*) that the HMM emits *x* to at least 3 significant figures.

**Constraints.** |*x*| = |*π*| = 100; 2 ≤ |*States*| ≤ 4; |*Σ*| = 3

**SAMPLE DATASET:**

Input:

xzyyz

--------

x y z

--------

A B

--------

A B

A 0.303 0.697

B 0.831 0.169

--------

x y z

A 0.533 0.065 0.402

B 0.342 0.334 0.324

Output:

1.544e-03

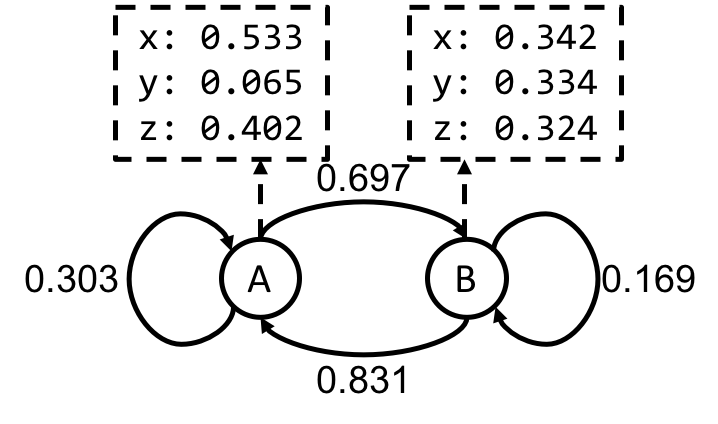
The probability that the given HMM emits the string xzyyz is 1.54×10-3.

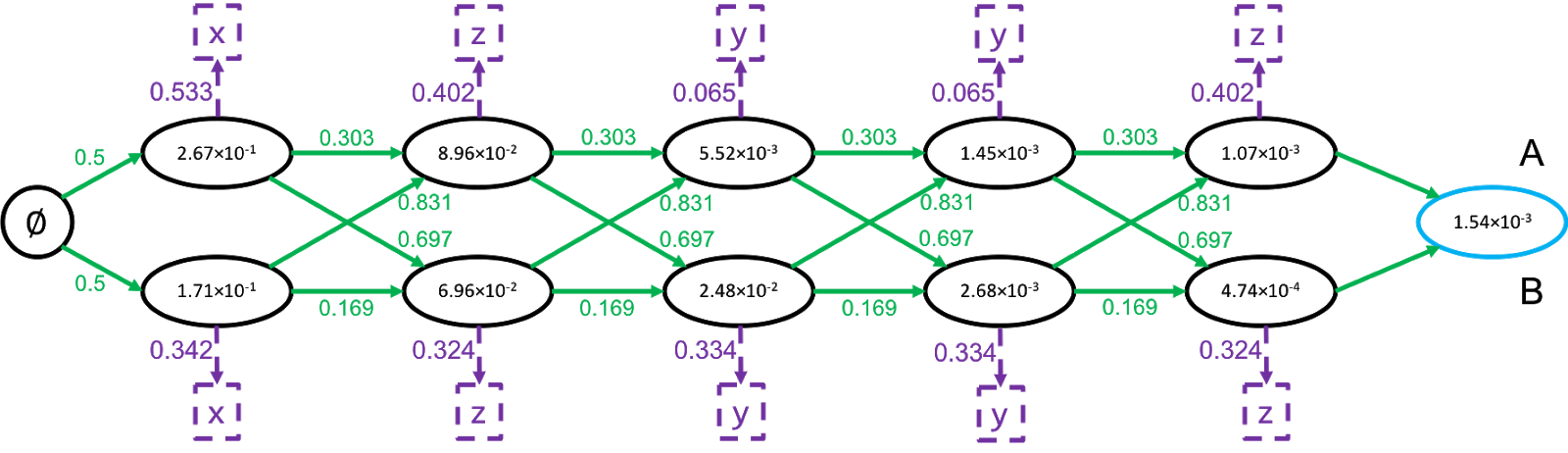
The first figure below is a visualization of this HMM, in which transitions are shown as solid edges (labeled by transition probabilities) and emissions are shown as dashed lines (labeled by emission probabilities).

The second figure below represents the Dynamic Programming graph solving the problem, in which a node *v* contains the total probability of all paths from the start node to v emitting the corresponding prefix of *x*. Green edges denote probabilities *transition*(*πi*-1,*πi*), and purple edges denote probabilities *emission*(*πi*,*xi*). The actual weight on a given edge from node (*l*,*i*-1) to node (*k*,*i*) has *weighti*(*l*,*k*) = *transition*(*πi*-1,*πi*) \* *emission*(*πi*,*xi*). Thus, the solution is denoted by the value in the end node (highlighted in blue), which is the sum of the values in the nodes in the last column.

For example, the value in node (A,2) is

(0.267×0.303×0.402) + (0.171×0.831×0.402) = 8.96×10-2





**TEST DATASET 1:**

Input:

x

--------

x

--------

A B

--------

A B

A 0.5 0.5

B 0.5 0.5

--------

x

A 0.5

B 0.5

Output:

0.5

This dataset makes sure that your code is correctly applying transitions from the initial state. In this problem we assume that transitions form the initial state occur with equal probability. Since |x| = 1 in this dataset the only contributions to the likelihood calculation come from the initial state transitions and the emission probabilities at each state. If your code is outputting 0.25 instead of 0.5 it’s possible that your code is not actually calculating the total probability of emitting x. It’s likely that you are implementing the Viterbi algorithm instead of the dynamic programming approach to outcome likelihood. If your code is outputting a value larger than 0.5 your code probably is not correctly considering transitions from the initial state.

**TEST DATASET 2:**

Input:

xy

--------

x y

--------

A B

--------

A B

A 0.4 0.6

B 0.6 0.4

--------

x y

A 0.7 0.3

B 0.4 0.6

Output:

0.252

This dataset makes sure that your code is correctly parsing string *x*. If your code outputs 0.55 then it’s likely that your code ignores the last character in the emitted string. If your code outputs 0.45 then it’s likely that your code ignores the first character in the emitted string. Double check that your code considers all characters in the emitted string.

**TEST DATASET 3:**

Input:

xzywyxw

--------

w x y z

--------

A B C

--------

A B C

A 0.7 0.1 0.2

B 0.5 0.3 0.2

C 0.1 0.4 0.5

--------

w x y z

A 0.34 0.24 0.42 0

B 0.17 0.49 0.34 0

C 0.22 0.22 0.56 0

Output:

0

This dataset makes sure that your code does not assign a non-zero likelihood to impossible outputs. This dataset is incapable of emitting a z character, even though it is in the alphabet of the HMM. Since the string *x* has a z character in it the likelihood of emitting *x* should be zero. If your code outputs a non-zero likelihood you are likely failing to apply the emission probabilities for the character z.

**TEST DATASET 4:**

Input:

xxxxyxxxzz

--------

x y z

--------

A B C

--------

A B C

A 0.7 0.1 0.2

B 0.5 0.3 0.2

C 0.1 0.4 0.5

--------

x y z

A 0.24 0.41 0.01

B 0.49 0.33 0.01

C 0.22 0.55 0.01

Output:

9.17e-09

This dataset checks to make sure your output is to at least three significant figures. This is not the same as three digits past the decimal point. If your output is incorrect make sure that your code doesn’t round the final answer to three digits past the decimal point. Scientific notation using a lowercase e (as in the example output) is accepted. Do not output scientific notation in the style of 9.17\*10-09 or 9.17x10-09.