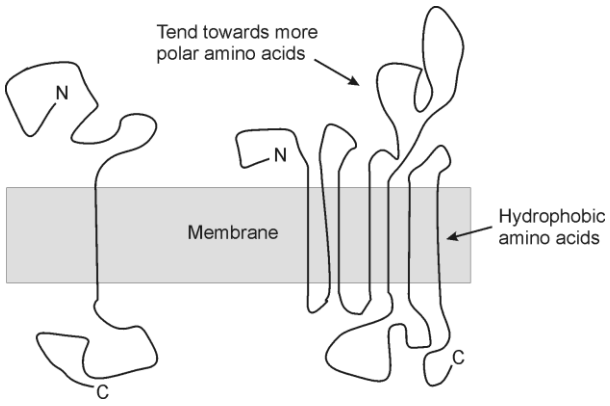
BCH394P/364C Systems Biology/Bioinformatics **Marcotte**
Problem Set #2 **due before 10PM 2/23/2026**

Don't forget to turn in the code/formulas/calculations that you performed to get your answers.

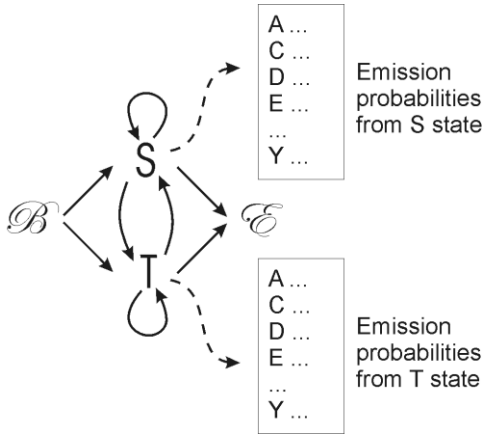
A Hidden Markov Model for finding transmembrane protein sequences

1. Proteins in a cell are often synthesized by a ribosome at one location in a cell (for example, in the cytosol or attached to the endoplasmic reticulum), then shipped someplace else in the cell where the proteins actually do their work. About a fourth to a third of the proteins get attached to membranes, often with a portion of the protein on one side of a membrane, a portion penetrating the membrane, and a portion on the other side of the membrane:



In this problem, we'll train a hidden Markov model to detect the regions of these so-called "integral membrane proteins" that end up buried in the membrane. The reason we can do this is because the amino acids in these regions tend to be much greasier in character than the amino acids in the non-membrane-associated part of the protein.

We'll construct a very simple hidden Markov model of the following architecture:



where $\mathscr{B}$ is the beginning state, S is the hidden state for an amino acid in a water-soluble segment of the protein, T is the hidden state for an amino acid in a transmembrane segment, and $\mathscr{E}$ is the end state. In this exercise, we won't model the end state, so you can ignore it. So, the protein sequence on the left in the example above would be something like SSSSSSSSSSSSSSTTTTTTTSSSSSSSSSSS and the protein sequence on the right would have seven runs of T's, flanked and interspersed with runs of S's.

From the course web page, download 3 data sets:
soluble_sequences, transmembrane_sequences, and state_sequences

These files have, respectively, a set of amino acid sequences from soluble segments of proteins, a set of amino acid sequences from trans-membrane regions of proteins, and a set of state sequences for all of the known integral membrane proteins of yeast.

Calculate the frequencies of amino acids in the first two files (remember your programs from the first problem set?). Use these frequencies to generate the emission probability tables of the S and T states of the HMM.

Calculate the frequencies of beginning a sequence with each of the 2 states (S or T) in the third file. Use these to calculate the transition probabilities from the beginning state $\mathscr{B}$ of the HMM.

Calculate the number of occurrences of each *digram* in the state file. A digram is a 2-character string, e.g., SS or TT or ST or TS. In general, a string of *n* characters is an *n*-gram. For somewhat longer strings, you'll also hear them called *k*-mers, where *k* is the length.) You can use your other program from problem set#1 for this. From these numbers of observations, you should now be able to construct the transition probability matrix for the HMM.

Now, take the amino acid sequence `KNSFFFFFFFLIII` and calculate the most likely state sequence using the Viterbi algorithm. You can code this if you wish, but it is not required, and it's short enough that if you really prefer not to code it, you can even work out this problem in a spreadsheet.

Turn in the transition probability matrix, emission probability matrix, the Viterbi matrix, and the most likely state sequence, along with your supporting code/formulas/calculations to solve this.

2. How many transmembrane amino acid segments (*i.e.*, segments of all T's) does the average integral membrane protein have? Show your calculation.

3. Draw a reasonable alternate architecture for the transmembrane sequence predicting HMM. Write why you might prefer one of the two HMM architectures over the other.

4. Draw a reasonable architecture for an HMM that would identify protein secondary structures (e.g., alpha helix, beta strand, random coil) within protein sequences.

5. Suggest an application (ANY application!) of HMM's to a biological problem other than the applications we have discussed (*i.e.*, not gene finding, CpG islands, protein secondary structure or transmembrane segment prediction, sequence alignment). Draw a potential HMM architecture that could be applied to that problem.