VITERBI ALGORITHM

Introduction

- The Viterbi Algorithm (VA) was first proposed by **Andrew J. Viterbi** in 1967.
- The Viterbi algorithm is a dynamic programming algorithm.
- Use for finding the most likely sequence of hidden states-called the Viterbi path- that results in a sequence of observed events, especially in the context Hidden Markov Models.

Applications

- The algorithm has found its original application in communication for decoding such as in dial-up modems, satellite, deep-space communications and wireless LANs.
- It is now also commonly used in speech recognition, speech synthesis, natural language processing, computational linguistics and bioinformatics.

Hidden Markov Model

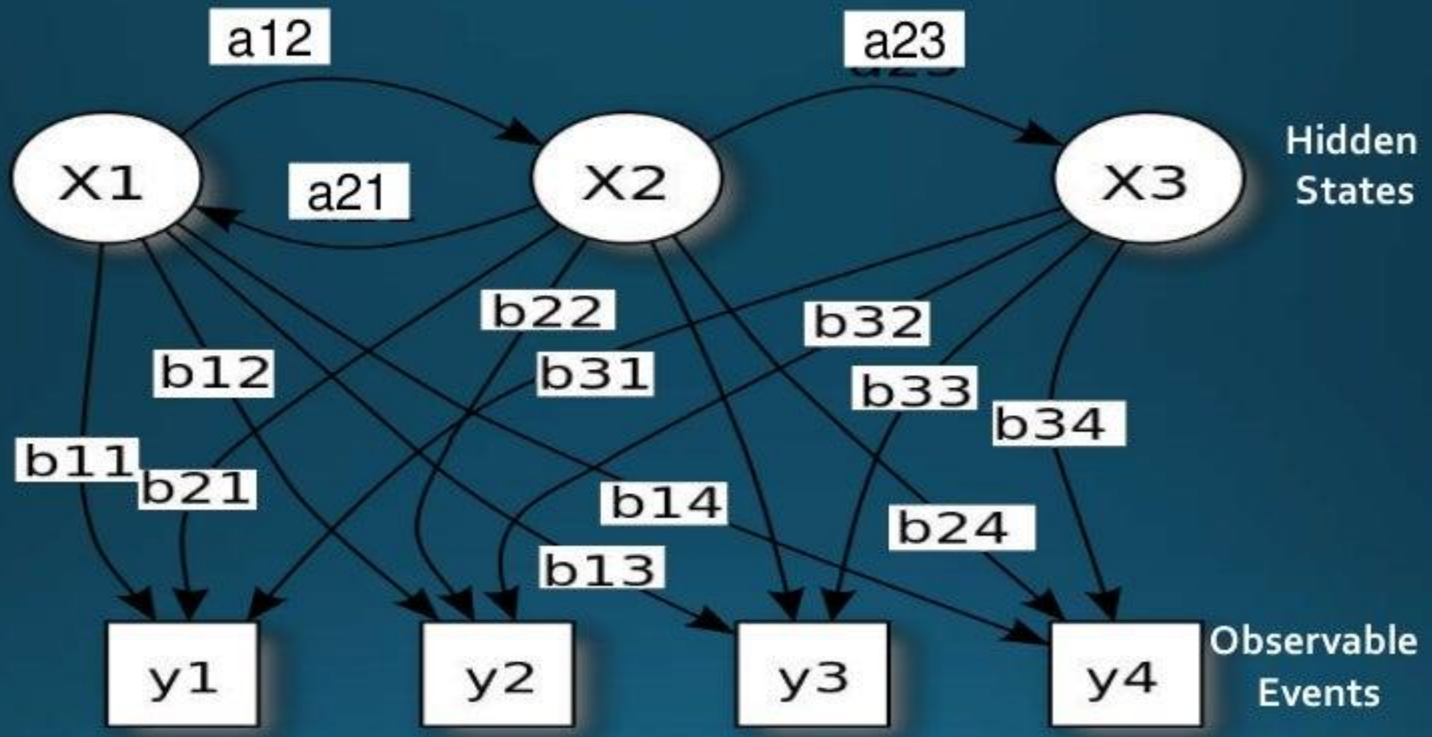
- Markov models are used to model sequences of events (or observations) that occur one after another .

- In a Hidden Markov model, the state is not directly visible, but the output/observations, dependent on the state, is visible.

- Each state has a probability distribution over the possible output .

- The sequence of observations generated by a HMM gives some information about the sequence of states.

An example of Hidden Markov Model (State Diagram)



a_{ij} -> Probability of transition from one state to another
b_{ij} -> Probability of an observation for a state

The Viterbi Algorithm

Input:

- The state space $\boldsymbol{S=\{s_1, s_2, \ldots s_N\}}$.
- The observation space $\boldsymbol{O=\{o_1, o_2, \ldots o_K\}}$.
- Transition matrix $\boldsymbol{A}$ of size $\boldsymbol{N.N}$ such that $\boldsymbol{A_{ij}}$ stores the transition probability of transiting from state $\boldsymbol{s_i}$ to $\boldsymbol{s_j}$ state.
- Emission matrix $\boldsymbol{B}$ of size $\boldsymbol{N.K}$ such that $\boldsymbol{B_{ij}}$ stores the probability of observing $\boldsymbol{o_j}$ from state $\boldsymbol{s_i}$.
- An array of initial probabilities π of size N such that π_i stores the probability of state s_i at time $t=1$.
- Sequence of observations $y_1, y_2 \ldots y_T$.

Output :

The most likely hidden state sequence ***$X = \{x_1, x_2, \ldots\ldots x_T\}$.***

Algorithm:

function VITERBI(O, S, π, A, T,B) : X

for each state s *from* 1 *to* N *do*

Viterbi[s,1] $\leftarrow \pi_s * B_{s,o1}$

Backpointer[s,1] $\leftarrow$ 0

for each time step t *from* 2 *to* T *do*

for each state s *from* 1 *to* N *do*

Viterbi[s, t] $\leftarrow \max_{k=1}^{N}$ (Viterbi [k , t-1] $* A_{k,s} * B_{s,o_t}$)

Backpointer[s, t] $\leftarrow \operatorname{argmax}_{k=1}^{N}$ (Viterbi [k , t-1] $* A_{k,s} * B_{s,o_t}$)

End *for*

End *for*

Cont....

$Z_T \leftarrow \underset{S=1}{\overset{N}{\operatorname{argmax}}}(\text{Viterbi}[s, T])$

$X_T \leftarrow S_{Z_T}$

for i $\leftarrow$ T,T-1….2 *do*

$Z_{i-1} \leftarrow \text{Backpointer}[Z_i, i]$

$X_{i-1} \leftarrow S_{Z_{i-1}}$

End *for*

Return X

End *function*

The complexity of the algorithm is $O(T * N^2)$

Implementation Example

- Consider a doctor diagnoses fever by asking patients how they feel. The patients may only answer that they feel normal, dizzy, or cold.

- There are two states, "Healthy" and "Fever", but the doctor cannot observe them directly, they are *hidden* from him.

- On each day, there is a certain chance that the patient will tell the doctor he/she is "normal", "cold", or "dizzy", depending on his/her health condition.

Inputs:

- ❑ States (S)='Healthy' , 'Fever'.
- ❑ Observation (O)='Normal' , 'Cold' , 'Dizzy'.
- ❑ Start_probability (π) = Healthy: 0.6, Fever: 0.4

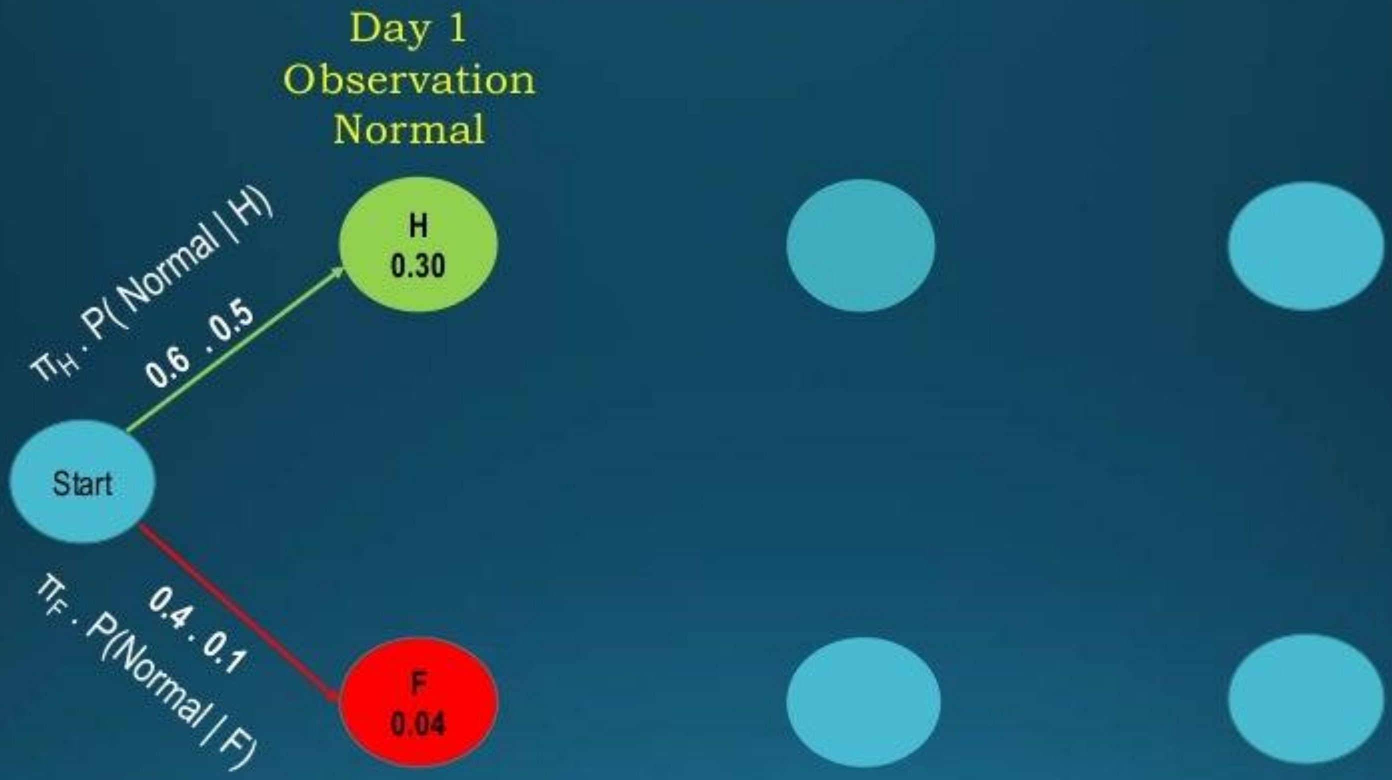
- ❑ Transition Probability(A)=

	Healthy	Fever
Healthy	0.7	0.3
Fever	0.4	0.6

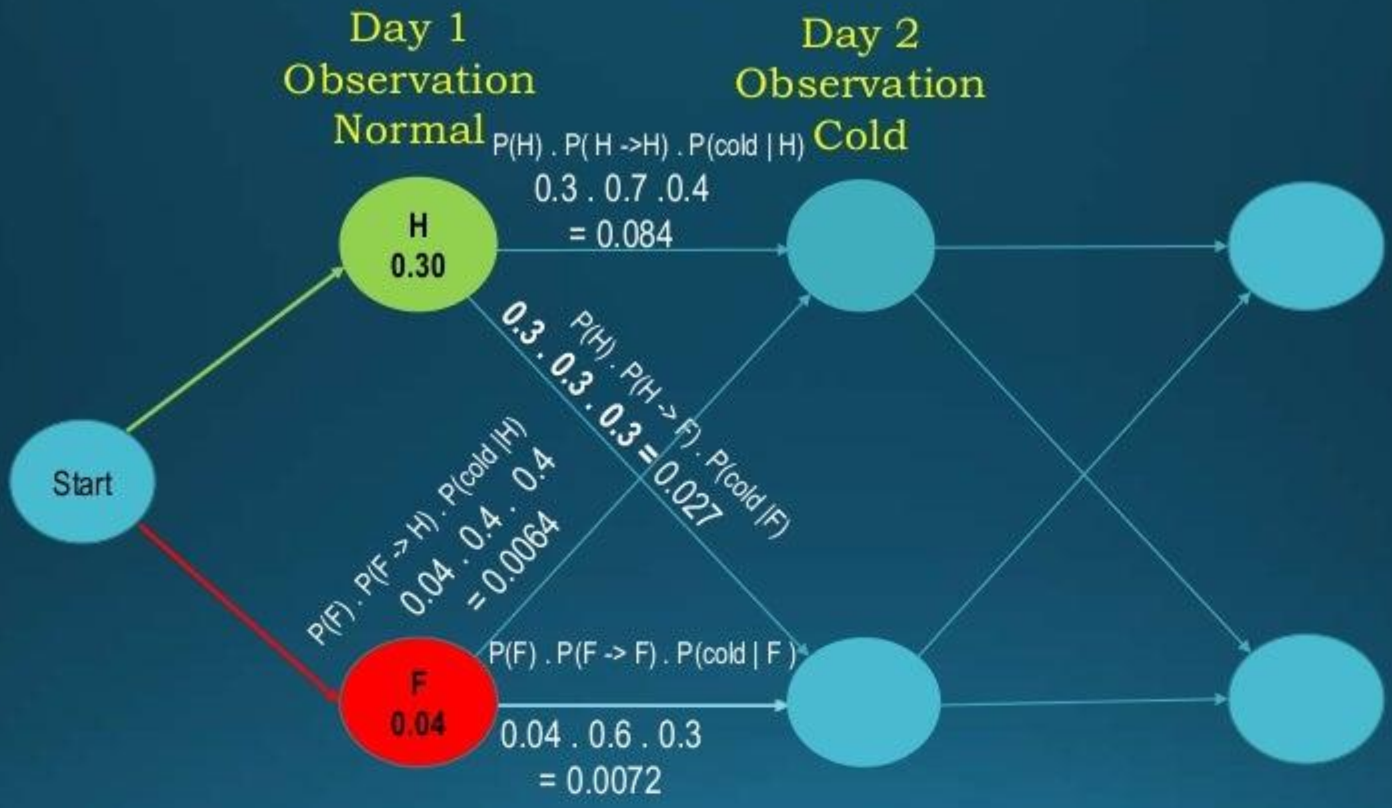
- ❑ Emission Probability(B)=

	Normal	Cold	Dizzy
Healthy	0.5	0.4	0.1
Fever	0.1	0.3	0.6

Operations

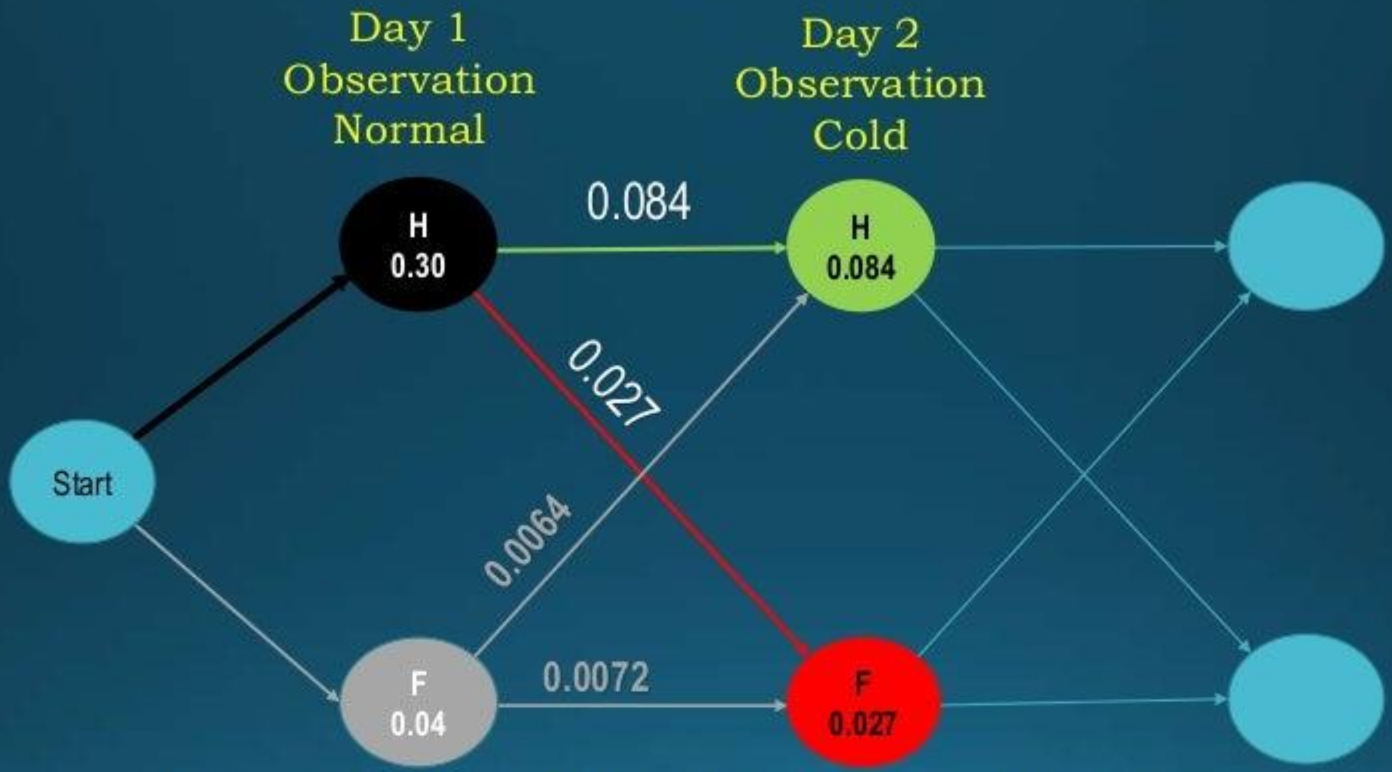


Calculate
P(start) * P(normal | state)

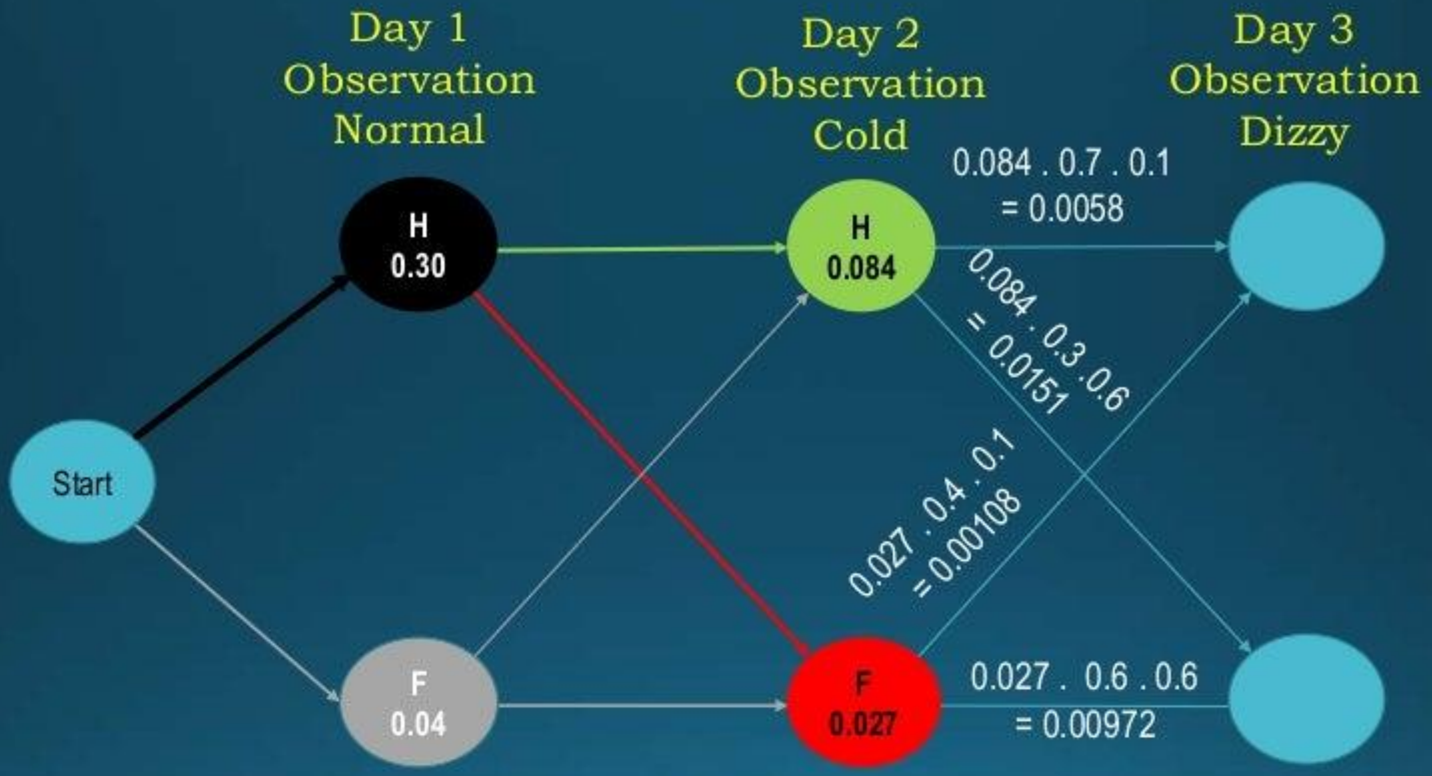


Calculate

P(old_state) * P(old_state -> new_state) * P(cold | new_state)

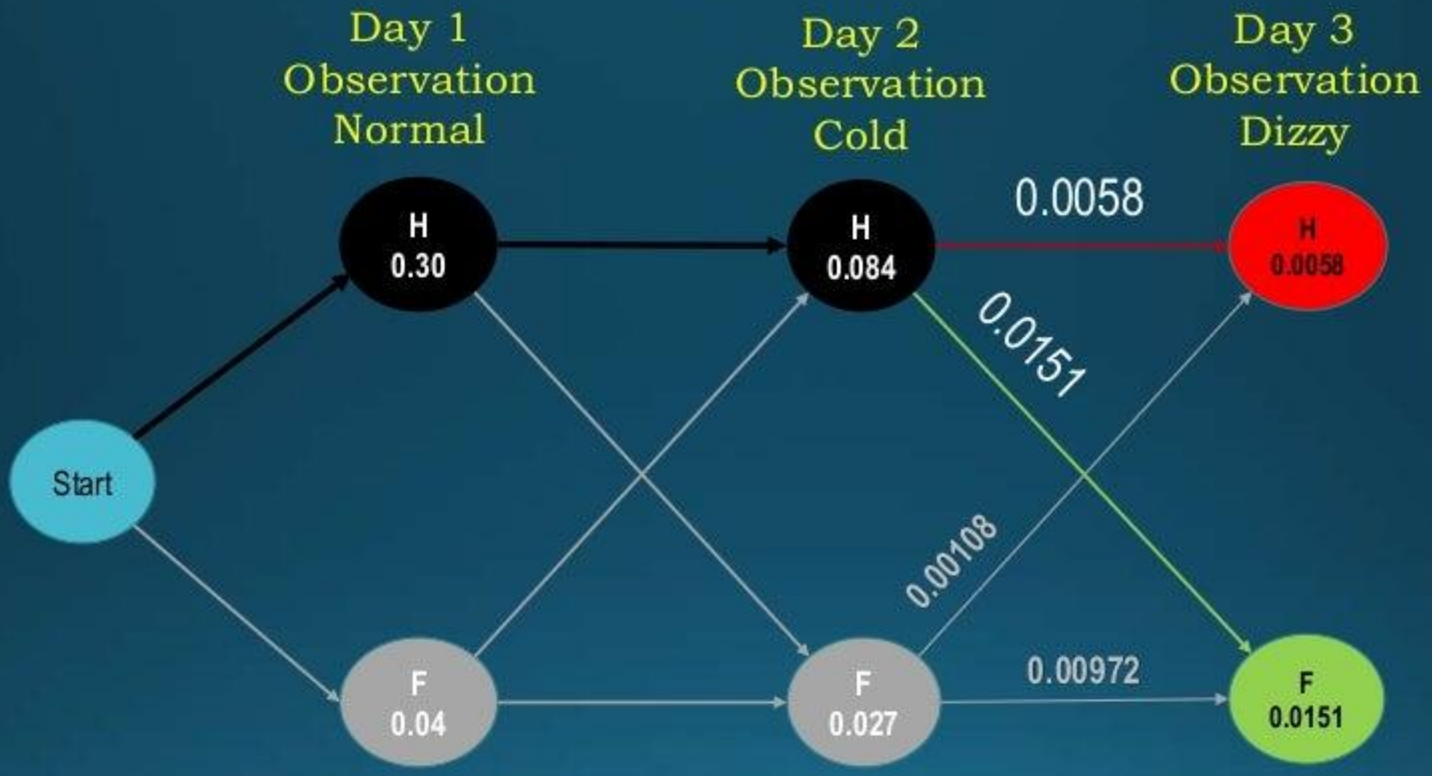


For each State H/F, Select the path with the Highest probability

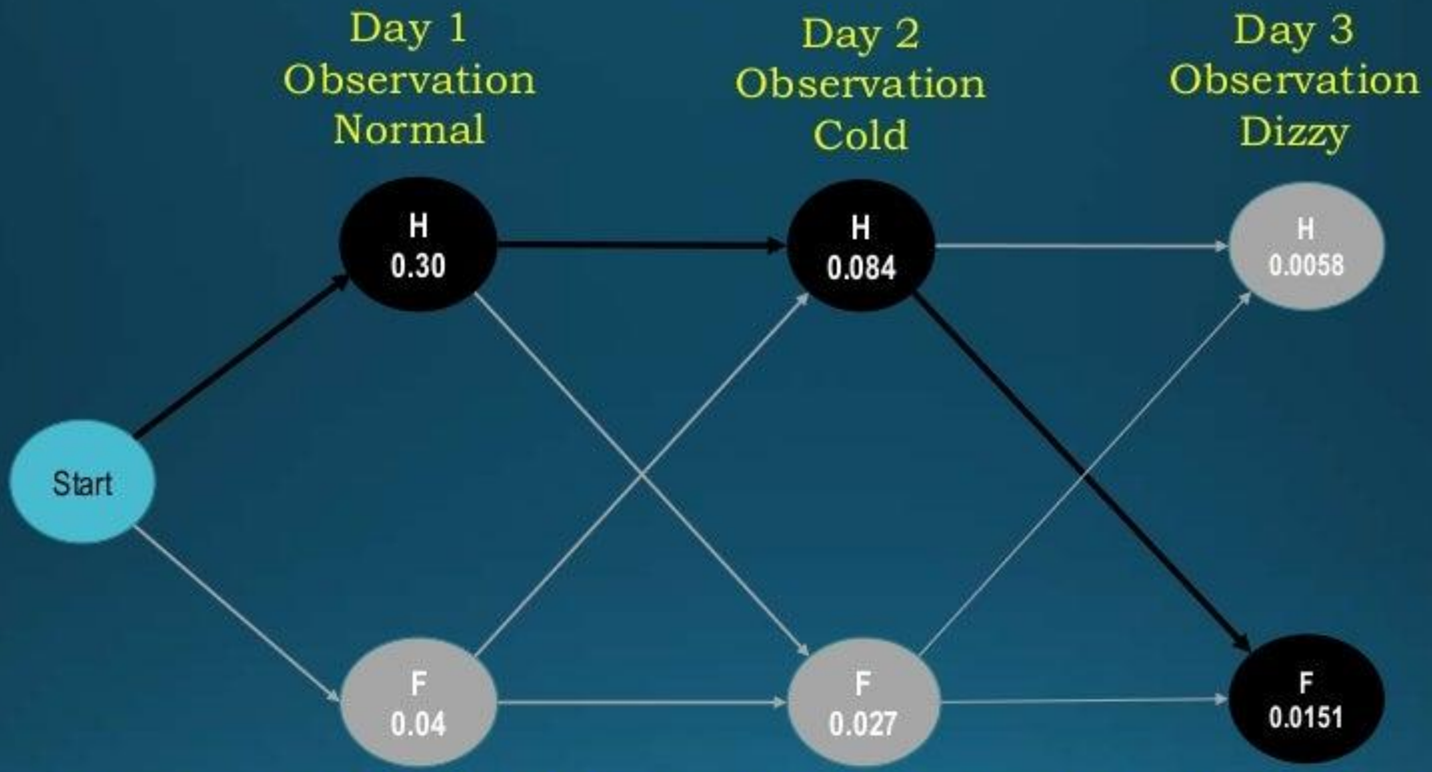


Calculate

P(old_state) * P(old_state -> new_state) *P(Dizzy | new_state)



For each State H/F, Select the path with the Highest probability



For time step T, select the state that has the highest probability and backtrack to the path that produced the highest probability using the backpointer and return the states.

Result

Day 1 Observation Normal	Day 2 Observation Cold	Day 3 Observation Dizzy
(0.30) "HEALTHY"	(0.084) "HEALTHY"	(0.0151) "FEVER"

Advantages

1. Ability to correct wrong bits transmitted by adding redundant information.
2. The State diagram offers a complete description of the system.
3. It is possible to reconstruct lost data.

Disadvantages

1. Computation becomes complex for large number of states.
2. More bandwidth needed for redundant information.

CONCLUSION

- Viterbi algorithm is widely used in communication.
- Use to find the hidden states of finite states Hidden Markov Model.
- Also used extensively in recognition problems

THANK YOU