

HMM (1): Definition and two types of HMM

LING 570

Fei Xia

HMM

- Definition and properties of HMM
 - Two types of HMM
- Three basic questions in HMM

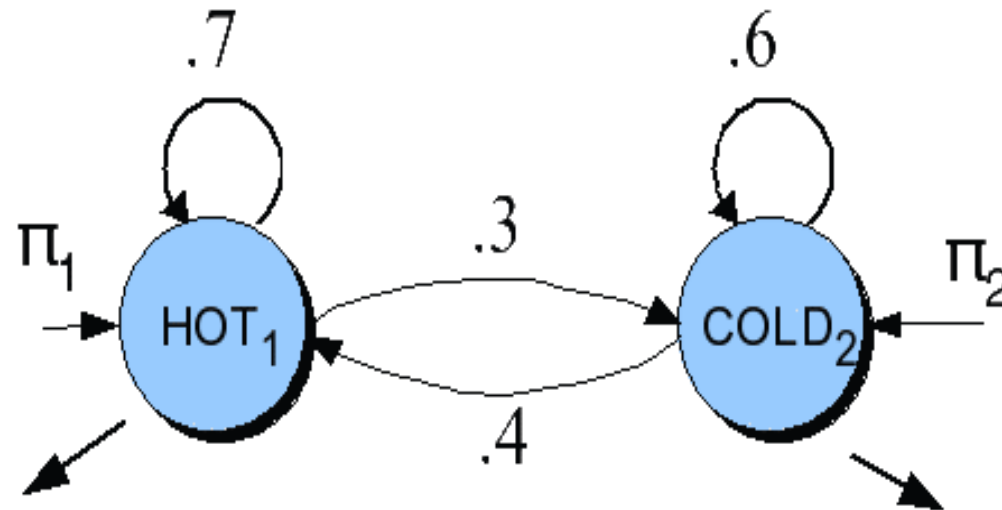
Definition of HMM

Hidden Markov Models

- There are n states s_1, \dots, s_n in an HMM, and the states are connected.
- The output symbols are produced by the states or edges in HMM.
- An observation $\mathbf{O}=(o_1, \dots, o_T)$ is a sequence of output symbols.
- Given an observation, we want to recover the hidden state sequence.
- An example: POS tagging
 - States are POS tags (in a bigram POS tagger)
 - Output symbols are words
 - Given an observation (i.e., a sentence), we want to discover the tag sequence.

An example: Ice Cream-Weather HMM

$$\pi = [.8, .2]$$



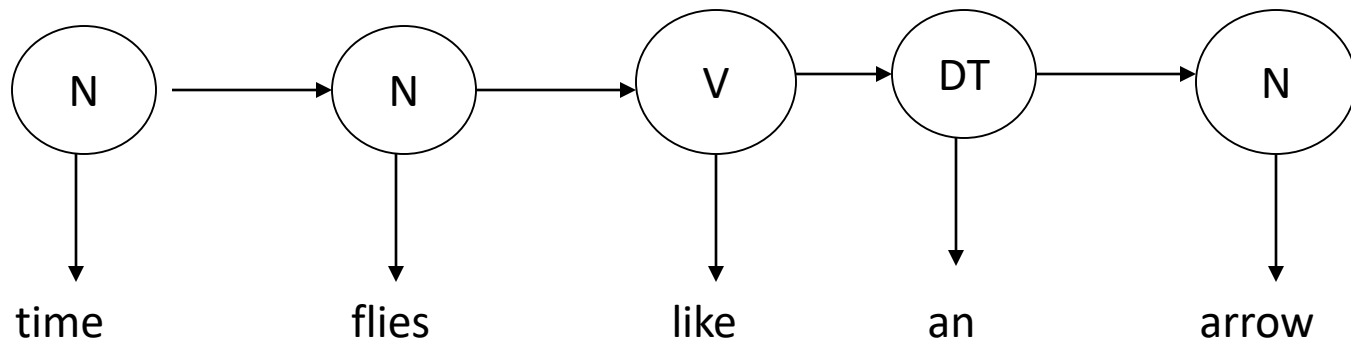
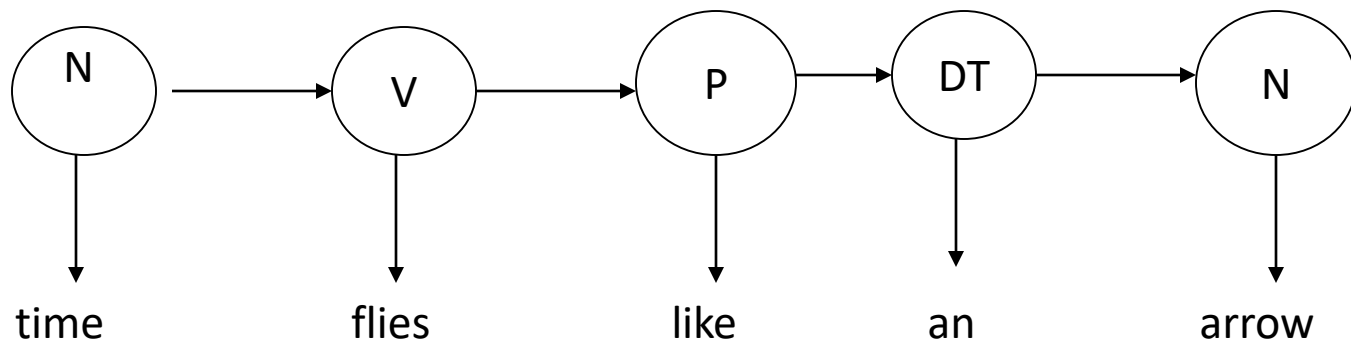
$$B_1 = \begin{bmatrix} P(1 | HOT) \\ P(2 | HOT) \\ P(3 | HOT) \end{bmatrix} = \begin{bmatrix} .2 \\ .4 \\ .4 \end{bmatrix}$$

$$B_2 = \begin{bmatrix} P(1 | COLD) \\ P(2 | COLD) \\ P(3 | COLD) \end{bmatrix} = \begin{bmatrix} .5 \\ .4 \\ .1 \end{bmatrix}$$

Highlights of HMM

- Can't directly observe states
 - Those states are said to be “hidden”
 - Distinguished from observations: ice cream vs weather
- Allows us to capture ambiguity
- Characteristic of many real-world problems
 - POS Tagging:
 - Observe words, not the underlying tag sequence
 - Speech recognition:
 - Observe acoustic signal, not underlying word/meaning sequence
- Widely used sequence labeler

Same observation, different state sequences



HMM assumptions

- Independence assumptions:
 - Markov assumption: Finite history
 - Typically 1st order
 - $P(q_i | q_1 \dots q_{i-1}) = P(q_i | q_{i-1})$
 - Output independence:
 - Output probability depends only on state that produced it
 - $P(o_i | q_1 \dots q_i \dots q_T, o_1 \dots o_i \dots o_T) = P(o_i | q_i)$
- Typically:
 - Ergodic: Fully connected; every state transition to all others
 - Stationary: All probabilities are independent of time step

Two types of HMMs

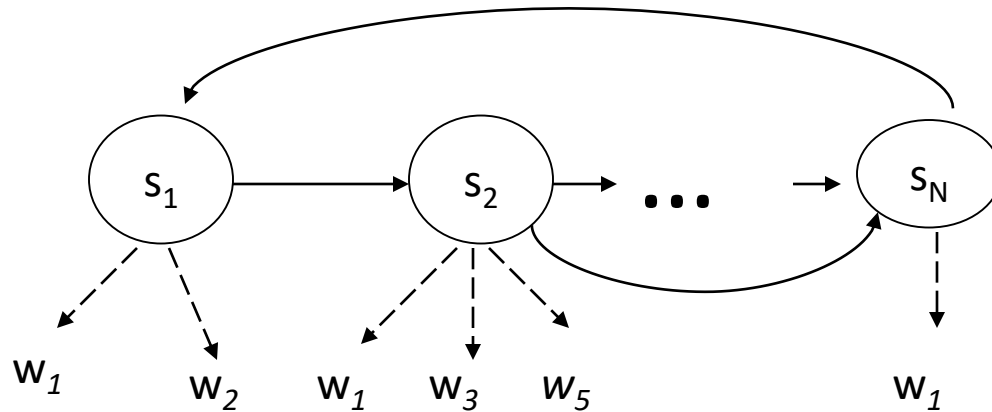
- State-emission HMM (Moore machine):
 - The output symbol is produced by states:
 - By the from-state
 - By the to-state
- Arc-emission HMM (Mealy machine):
 - The output symbol is produce by the edges; i.e., by the (from-state, to-state) pairs.

State-emission HMM

Definition of state-emission HMM

- A HMM is a tuple (S, Σ, Π, A, B) :
 - A set of states $S = \{s_1, s_2, \dots, s_N\}$.
 - A set of output symbols $\Sigma = \{w_1, \dots, w_M\}$.
 - Initial state probabilities $\Pi = \{\pi_i\}$
 - Transition prob: $A = \{a_{ij}\}$
 - Emission prob: $B = \{b_{jk}\}$
- We use s_i and w_k to refer to what is in an HMM **structure**.
- We use X_i and O_i to refer to what is in a particular HMM **path** and its **output**

An example: the HMM structure



Two kinds of parameters:

- Transition probability: $P(s_j | s_i)$
 - Emission probability: $P(w_k | s_i)$
- ➔ # of Parameters: $O(NM + N^2)$

Constraints in a state-emission HMM

$$\sum_{i=1}^N \pi_i = 1$$

$$\forall i \quad \sum_{j=1}^N a_{ij} = 1$$

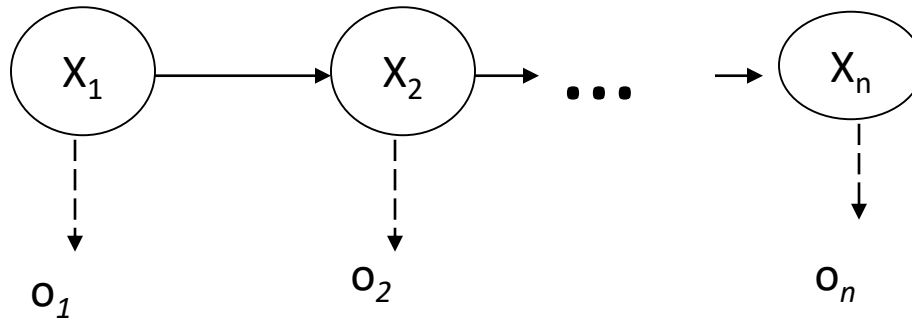
$$\forall i \quad \sum_{k=1}^M b_{ik} = 1$$



For any integer n and any HMM

$$\sum_{|O|=n} P(O \mid HMM) = 1$$

Output symbols are generated by the from-states

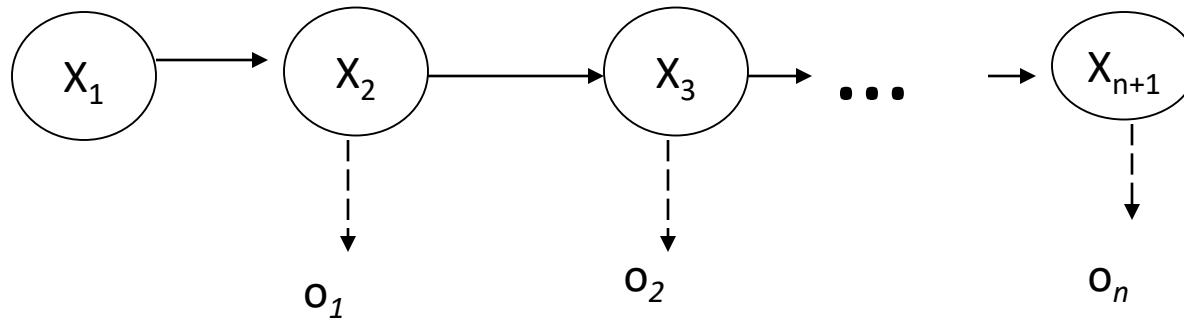


- State sequence: $X_{1,n}$
- Output sequence: $O_{1,n}$

$$P(O_{1,n}, X_{1,n}) = \pi(x_1) \left(\prod_{i=1}^{n-1} P(x_{i+1} | x_i) \right) \left(\prod_{i=1}^n P(o_i | x_i) \right)$$

$$P(O_{1,n}) = \sum_{X_{1,n}} P(O_{1,n}, X_{1,n})$$

Output symbols are generated by the to-states



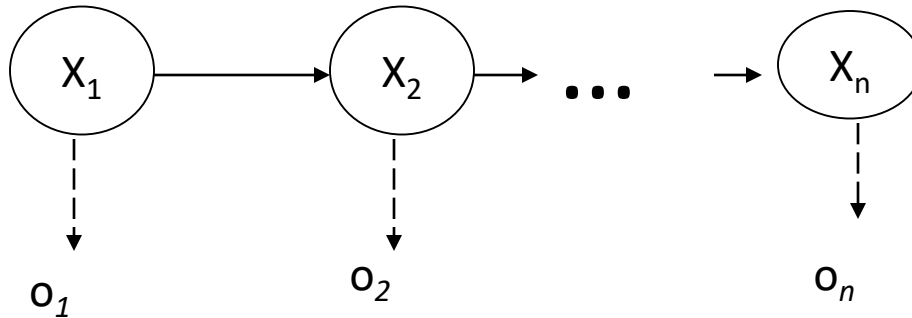
- State sequence: $X_{1,n+1}$
- Output sequence: $O_{1,n}$

$$P(O_{1,n}, X_{1,n+1}) = \pi(x_1) \prod_{i=1}^n (P(x_{i+1} | x_i) P(o_i | x_{i+1}))$$

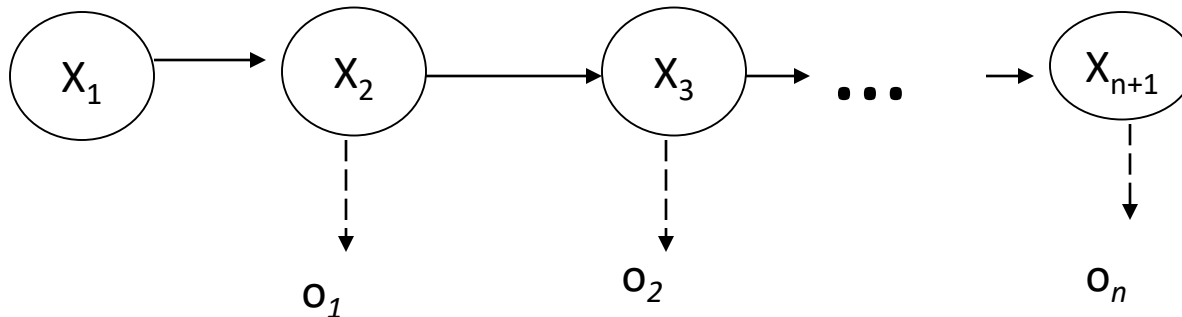
$$P(O_{1,n}) = \sum_{X_{1,n+1}} P(O_{1,n}, X_{1,n+1})$$

A path in a state-emission HMM

Output symbols are produced by the from-states:



Output symbols are produced by the to-states:



Properties of HMM

- Markov assumption (Limited horizon):

$$P(X_{t+1} \mid X_1, X_2, \dots, X_t) = P(X_{t+1} \mid X_t)$$

- Stationary distribution (Time invariance): the probabilities do not change over time:

$$P(X_{t+1} \mid X_t) = P(X_{t+1+m} \mid X_{t+m})$$

- The states are **hidden** because we know the structure of the machine (i.e., S and Σ), but we don't know which state sequences generate a particular output.

Arc-emission HMM

Definition of arc-emission HMM

- A HMM is a tuple (S, Σ, Π, A, B) :
 - A set of states $S = \{s_1, s_2, \dots, s_N\}$
 - A set of output symbols $\Sigma = \{w_1, \dots, w_M\}$
 - Initial state probabilities $\Pi = \{\pi_i\}$
 - Transition prob: $A = \{a_{ij}\}$
 - Emission prob: $B = \{b_{ijk}\}$

Constraints in an arc-emission HMM

$$\sum_{i=1}^N \pi_i = 1$$

$$\forall i \quad \sum_{j=1}^N a_{ij} = 1$$

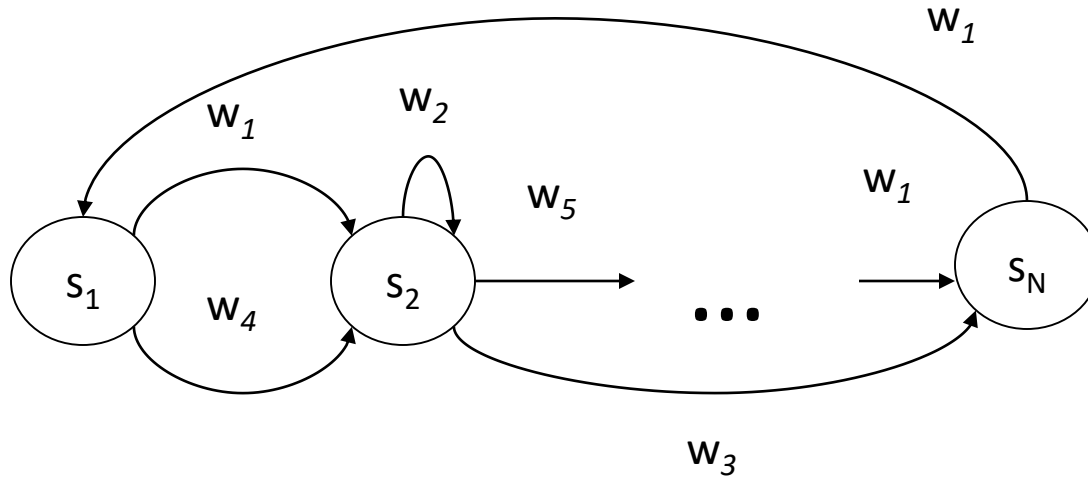
$$\forall i, j \quad \sum_{k=1}^M b_{ijk} = 1$$



For any integer n and any HMM

$$\sum_{|O|=n} P(O \mid HMM) = 1$$

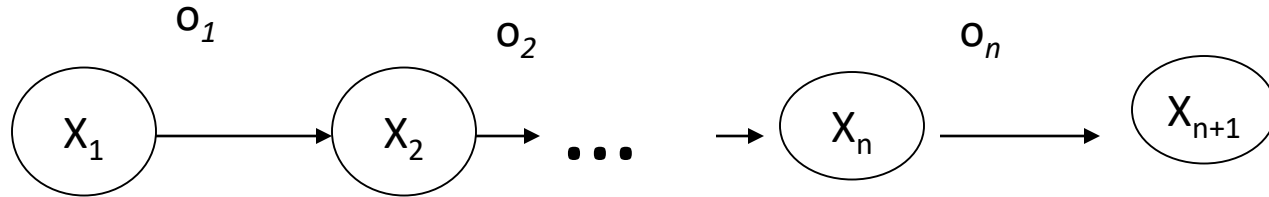
An example: HMM structure



Same kinds of parameters but the emission probabilities depend on both states: $P(w_k \mid s_i, s_j)$

➔ # of Parameters: $O(N^2M + N^2)$.

A path in an arc emission HMM



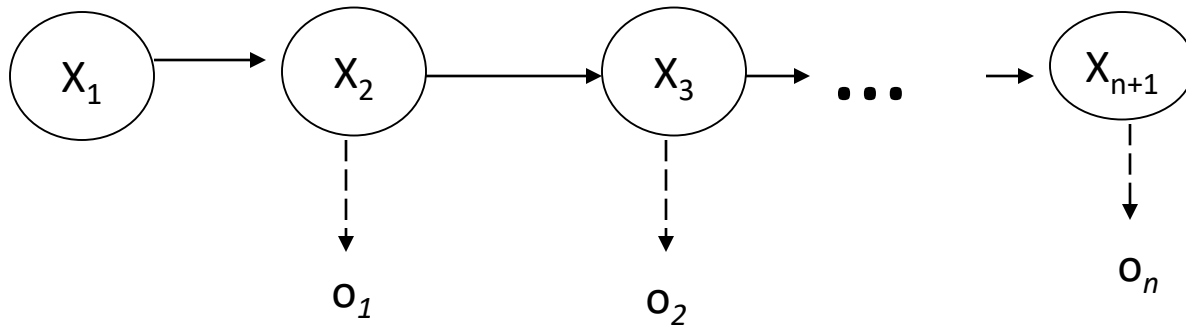
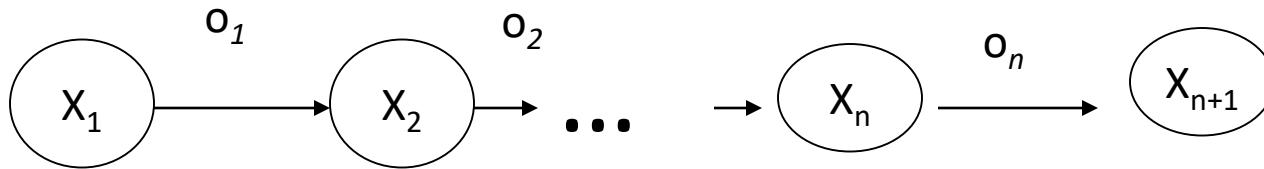
State sequence: $X_{1,n+1}$

Output sequence: $O_{1,n}$

$$P(O_{1,n}, X_{1,n+1}) = \pi(x_1) \prod_{i=1}^n P(x_{i+1} | x_i) P(o_i | x_i, x_{i+1})$$

$$P(O_{1,n}) = \sum_{X_{1,n+1}} P(O_{1,n}, X_{1,n+1})$$

Arc-emission vs. state-emission



Are the two types of HMMs equivalent?

- For each state-emission HMM_1 , there is an arc-emission HMM_2 , such that for any sequence O , $P(O | \text{HMM}_1) = P(O | \text{HMM}_2)$.
- The reverse is also true.
- How to prove that?

Applications of HMM

- N-gram POS tagging
 - Bigram tagger: o_i is a word, and s_i is a POS tag.
- Other tagging problems:
 - Word segmentation
 - Chunking
 - NE tagging
 - Punctuation predication
 - ...
- Other applications: ASR,

Additional slides

PFA recap

Formal definition of PFA

A PFA is $(Q, \Sigma, I, F, \delta, P)$

- Q : a finite set of N states
- Σ : a finite set of input symbols
- $I: Q \rightarrow \mathbb{R}^+$ (initial-state probabilities)
- $F: Q \rightarrow \mathbb{R}^+$ (final-state **probabilities**)
- $\delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q$: the transition relation between states.
- $P: \delta \rightarrow \mathbb{R}^+$ (transition probabilities)

Constraints on function:

$$\sum_{q \in Q} I(q) = 1$$

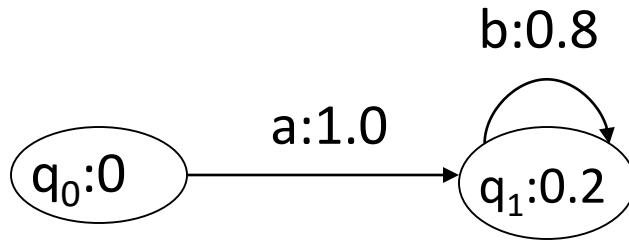
$$\forall q \in Q \quad F(q) + \sum_{\substack{a \in \Sigma \cup \{\varepsilon\} \\ q' \in Q}} P(q, a, q') = 1$$

Probability of a string:

$$P(w_{1,n}, q_{1,n+1}) = I(q_1) * F(q_{n+1}) * \prod_{i=1}^n p(q_i, w_i, q_{i+1})$$

$$P(w_{1,n}) = \sum_{q_{1,n+1}} P(w_{1,n}, q_{1,n+1})$$

An example of PFA



$$F(q_0)=0$$

$$F(q_1)=0.2$$

$$I(q_0)=1.0$$

$$I(q_1)=0.0$$

$$\begin{aligned} P(ab^n) &= I(q_0) * P(q_0, ab^n, q_1) * F(q_1) \\ &= 1.0 * 1.0 * 0.8^n * 0.2 \end{aligned}$$

$$\sum_x P(x) = \sum_{n=0}^{\infty} P(ab^n) = 0.2 * \sum_{n=0}^{\infty} 0.8^n = 0.2 * \frac{0.8^0}{1-0.8} = 1$$

PFA vs. Arc-emission HMM

A PFA is $(Q, \Sigma, I, F, \delta, P)$

- Q : a finite set of N states
- Σ : a finite set of input symbols
- $I: Q \rightarrow R^+$ (initial-state probabilities)
- $F: Q \rightarrow R^+$ (final-state probabilities)
- $\delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q$: the transition relation between states.
- $P: \delta \rightarrow R^+$ (transition probabilities)

A HMM is a tuple (S, Σ, Π, A, B)

- A set of states $S = \{s_1, s_2, \dots, s_N\}$
- A set of output symbols $\Sigma = \{w_1, \dots, w_M\}$
- Initial state prob: $\Pi = \{\pi_i\}$
- Transition prob: $A = \{a_{ij}\}$
- Emission prob: $B = \{b_{ijk}\}$