

# CPSC 322: Introduction to Artificial Intelligence

## Uncertainty: Markov Models and Hidden Markov Models

Textbook reference: [8.5]

Instructor: Varada Kolhatkar  
University of British Columbia

Credit: These slides are adapted from the slides of the previous offerings of the course. Thanks to all instructors for creating and improving the teaching material and making it available!

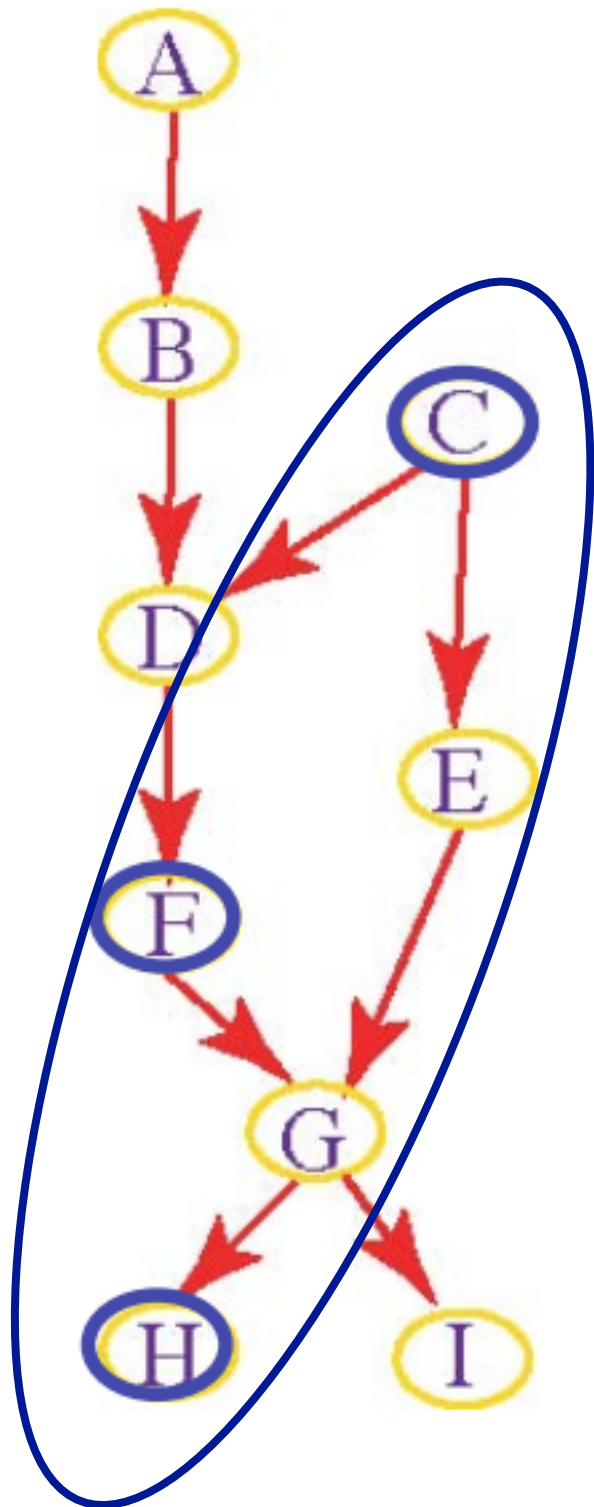
# Announcements

- Teaching evaluations are open. You should have received an email.
- I am teaching an undergrad course for the first time and I will very much appreciate constructive feedback.
- Final exam
  - **Time:** Dec 9 at 7:00pm and **Location:** SRC A
- Assignment 4 due on **Nov 29th, 11:59 PM**

# Lecture outline

- Recap 🙌
- Temporal probabilistic models
- Markov chains
- Markov chains in Natural Language
- Hidden Markov models introduction

# VE and conditional independence



Can we use conditional independence to make VE simpler?

Before running VE, we can prune all variables  $Z_i$  that are conditionally independent of the query  $Y$  given evidence  $E$ :  $Z_i \perp\!\!\!\perp Y \mid E$

In particular, any node that has no observed or queried descendants and is itself not observed or queried may be pruned.

Example: Which variables can we prune for the query  $P(G = g \mid C = c_1, F = f_1, H = h_1)$ ?

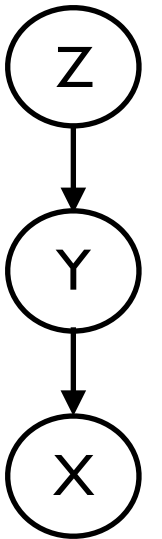
A, B, D, I can be pruned.

# Recap: Independencies in BNs

Given  $Y$ , does learning the value of  $Z$  tell us nothing new about  $X$ ?  
I.e., is  $P(X|Y, Z)$  equal to  $P(X | Y)$ ?

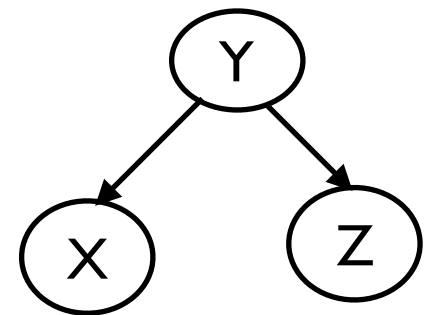
Yes. Since we know the value of all of  $X$ 's parents (namely,  $Y$ ), and  $Z$  is not a descendant of  $X$ ,  $X$  is conditionally independent of  $Z$ .

Also, since independence is symmetric,  $P(Z|Y, X) = P(Z|Y)$ .



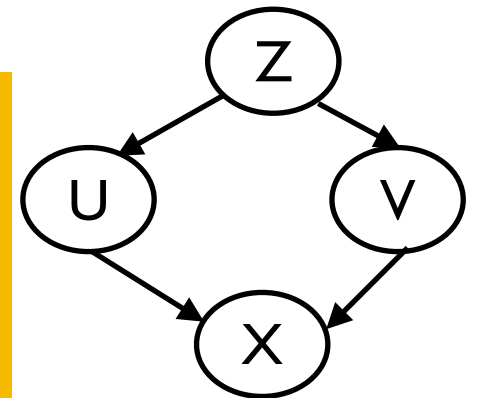
Is  $X$  conditionally independent of  $Z$  given  $Y$ ?

Yes. All  $X$ 's parents are given and  $Z$  is not a descendent of  $X$

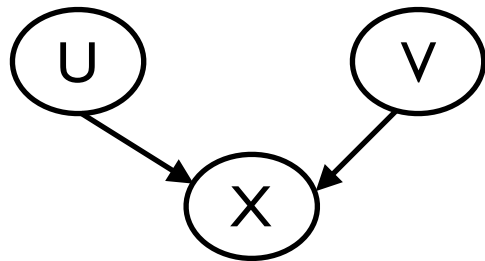


Is  $Z$  conditionally independent of  $X$  given  $U$ ? No.

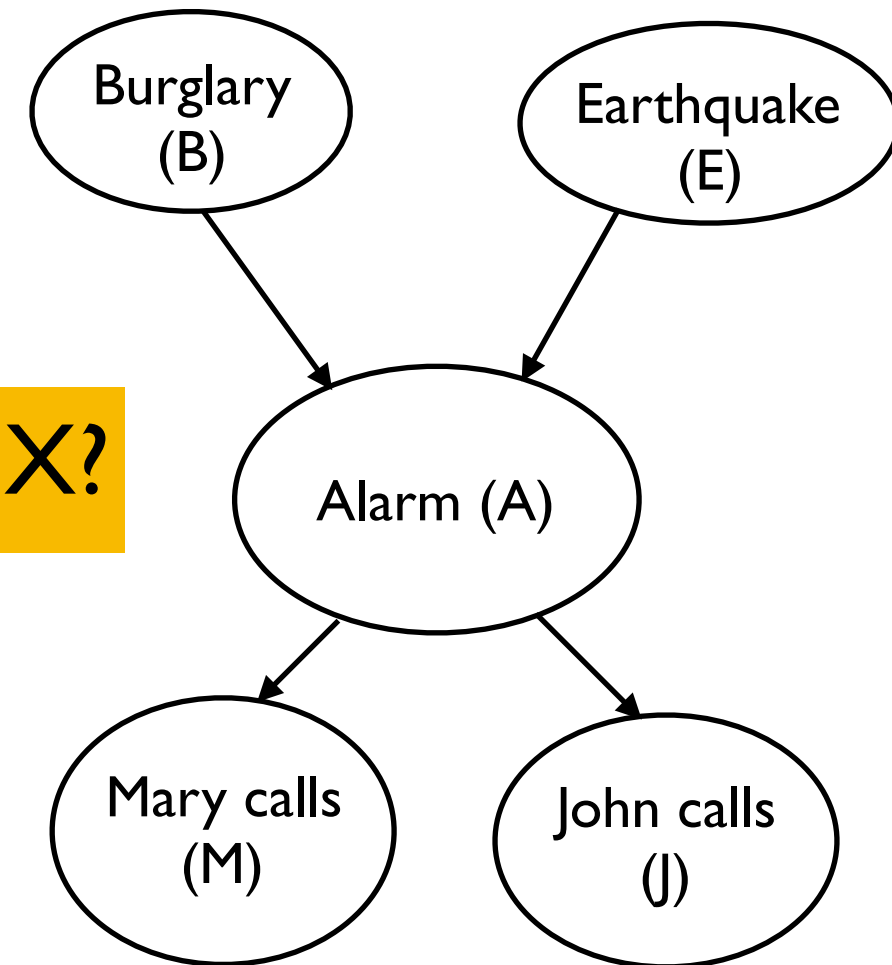
Is  $Z$  conditionally independent of  $X$  given  $U$  and  $V$ ? Yes.



# Recap: Independencies in BNs



Is  $U$  conditionally independent of  $V$  given  $X$ ?



ASIDE:  $d$ -separation algorithm to determine whether two variables in Bayes net are conditional independent or not.

# Recap: Independencies in BNs

To-do when you get a chance

Are  $X$  and  $Z$  conditionally independent given  $Y$ ?

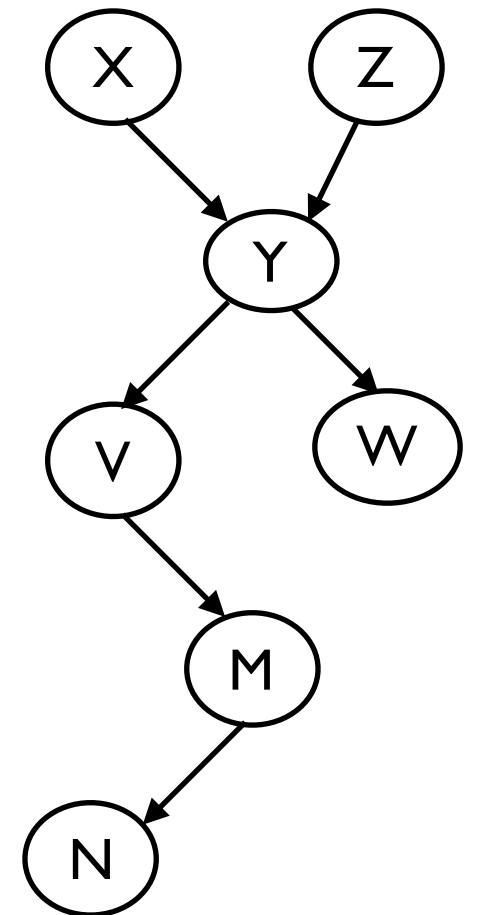
Are  $X$  and  $Z$  conditionally independent given  $V$  and  $W$ ?

Are  $V$  and  $W$  conditionally independent given  $Y$ ?

Are  $V$  and  $W$  conditionally independent given  $X$  and  $Z$ ?

Credit: d-separation algorithm

Refer to this for the algorithm for a more systematic way to determine which variables are independent in a Bayesian network



# Lecture outline

- Recap
- Temporal probabilistic models 📌
- Markov chains
- Stationary distribution
- Markov chains in Natural Language
- Hidden Markov models introduction



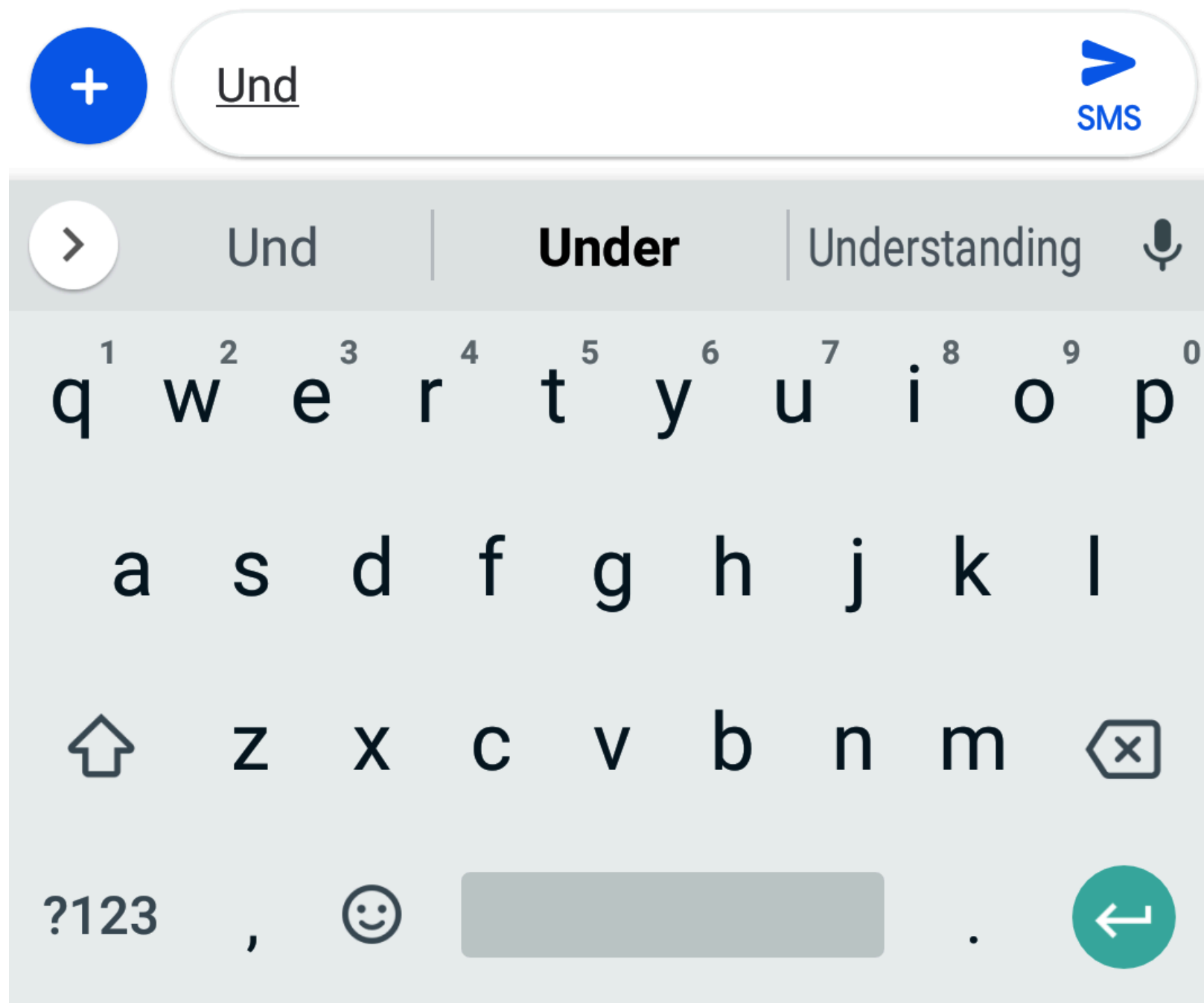
# Today: Learning outcomes

From this lecture, students are expected to be able to:

- Specify a Markov chain and compute the probability of a sequence of states
- Explain the general idea of stationary distribution
- Justify and apply Markov chains to compute the probability of a natural language sentence
- Specify the components of a hidden Markov model

# Everyday example of Markov chains

Autocomplete in text messaging



# Modelling static environments


- So far we have used Bayesian networks to perform inference in **static environments**
- For instance, the system keeps collecting evidence to diagnose the cause of a fault in a system (e.g., a car).
- The environment (values of the evidence, the true cause) does not change as we gather new evidence
- What does change?
  - The system's belief over possible values

# Modelling evolving environments

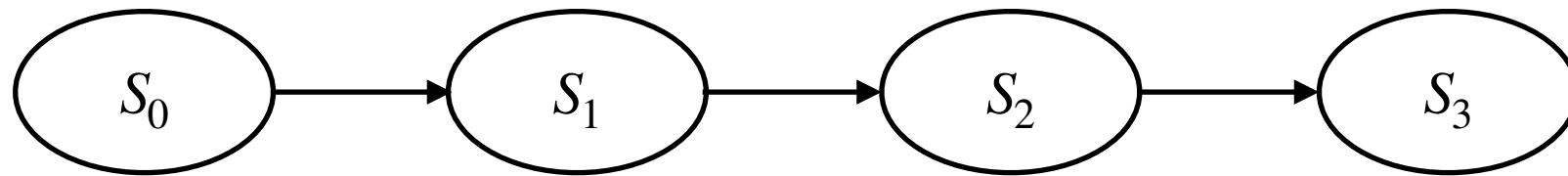
Often we need to make inferences about evolving environments.

Represent the state of the world at each specific point in time via a series of snapshots, or time slices

# Lecture outline

- Recap
- Temporal probabilistic models
- Markov chains 
- Stationary distribution
- Markov chains in Natural Language
- Hidden Markov models introduction

# Simplest possible Dynamic Bayesian Net



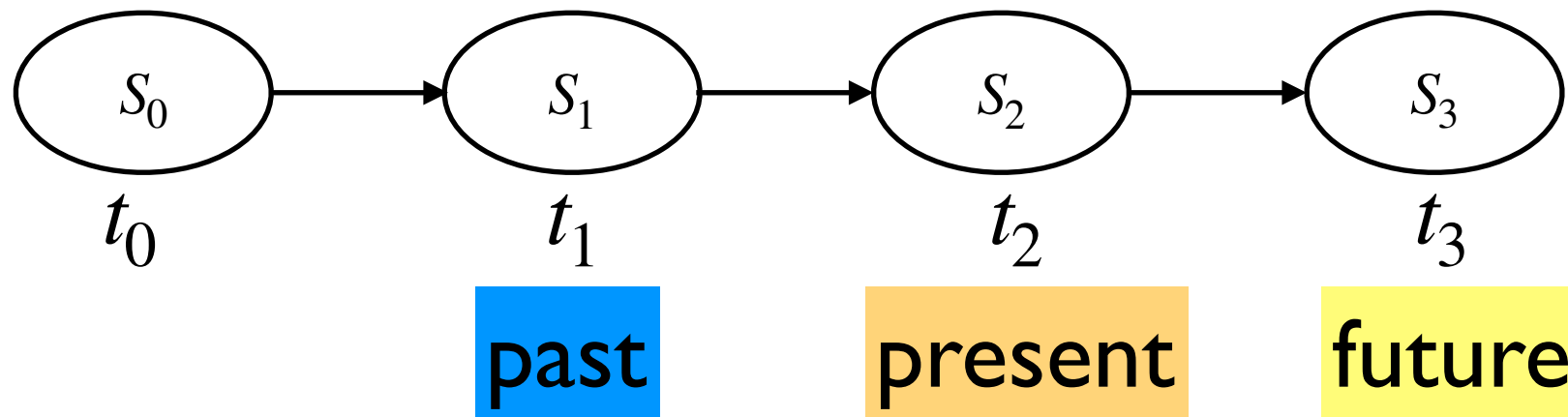
One random variable for each time slice: let's assume  $S_t$  represents the state at time  $t$  with domain  $\{V_1, \dots, V_n\}$

Each random variable depends only on the previous one

Intuitively,  $S_t$  conveys all of the information about the history that can affect the future states.

$$P(S_{t+1} | S_0, \dots, S_t) = P(S_{t+1} | S_t)$$

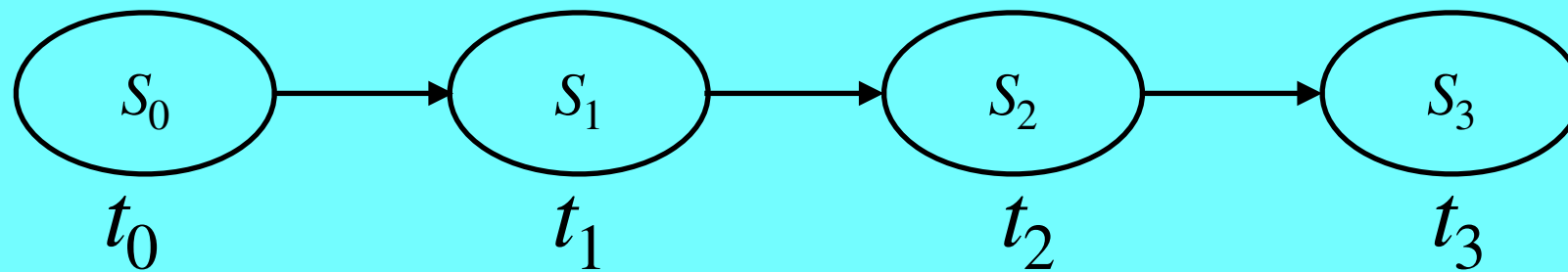
# Markov assumption



The independence assumption conveyed by the Bayesian network:  $P(S_{t+1} | S_0, \dots, S_t) = P(S_{t+1} | S_t)$

“The **future** is conditionally independent of the **past** given the **present**.”

# How many CPTs?



How many CPTs do we need to specify?

A. 4

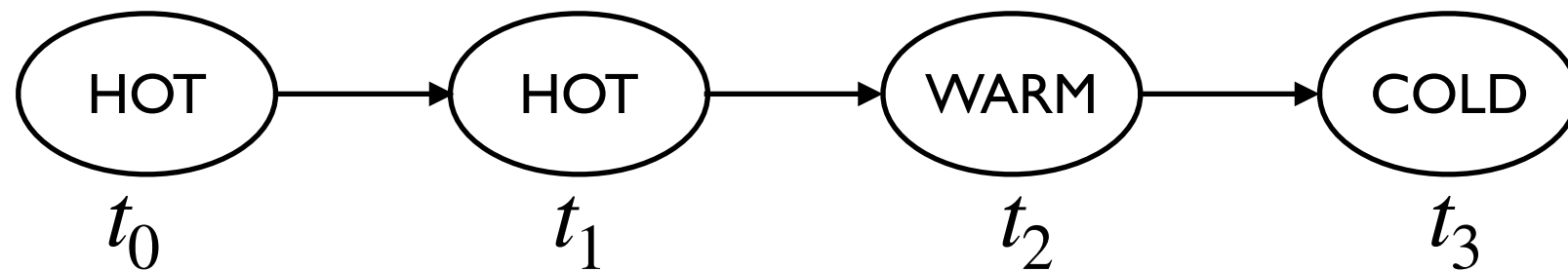
B. 3 

C. 2

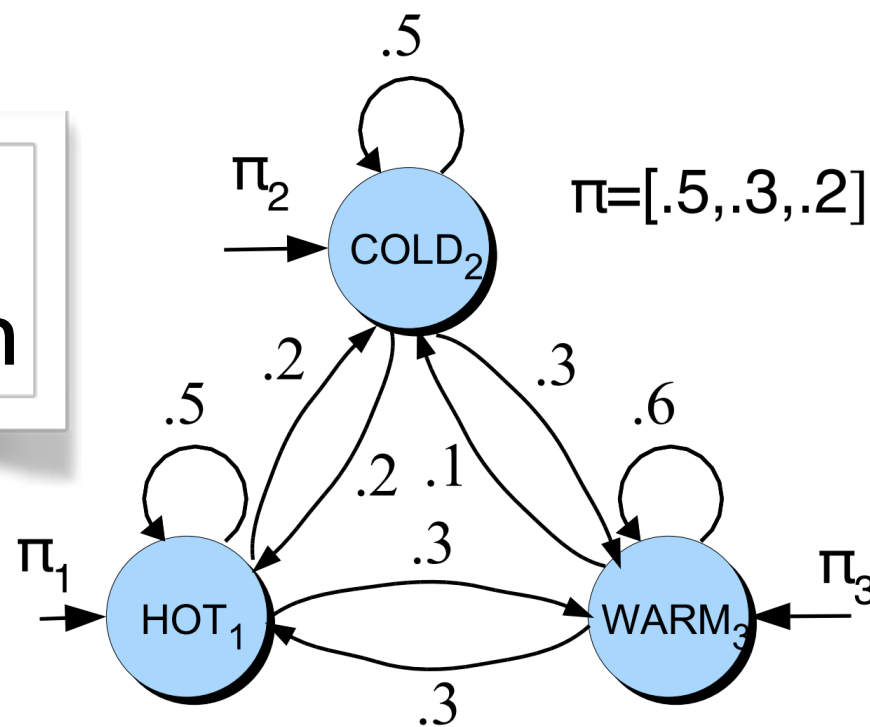
D. 1



# Markov chain: Weather example



Alternative  
representation



Credit: <https://web.stanford.edu/~jurafsky/slp3/a>

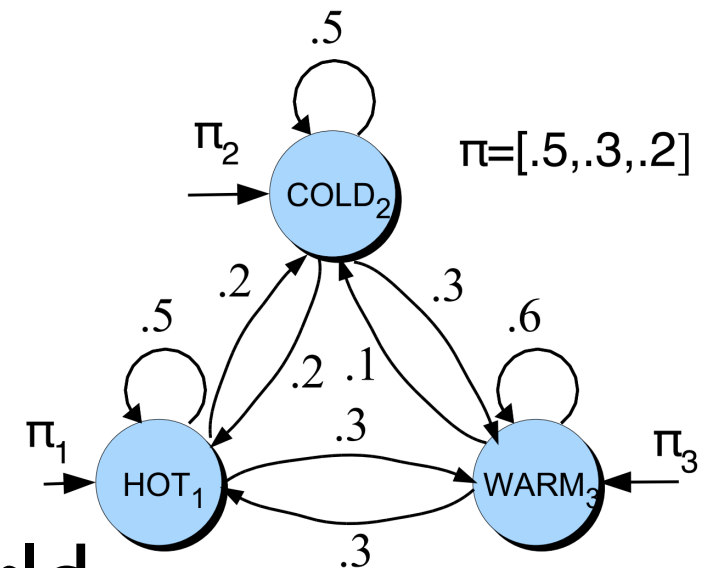
# Markov chain ingredients

State space:

$$S = \{HOT, COLD, WARM\}$$

Set of possible states we can be in at time  $t$

Represent the unique observations in the world.



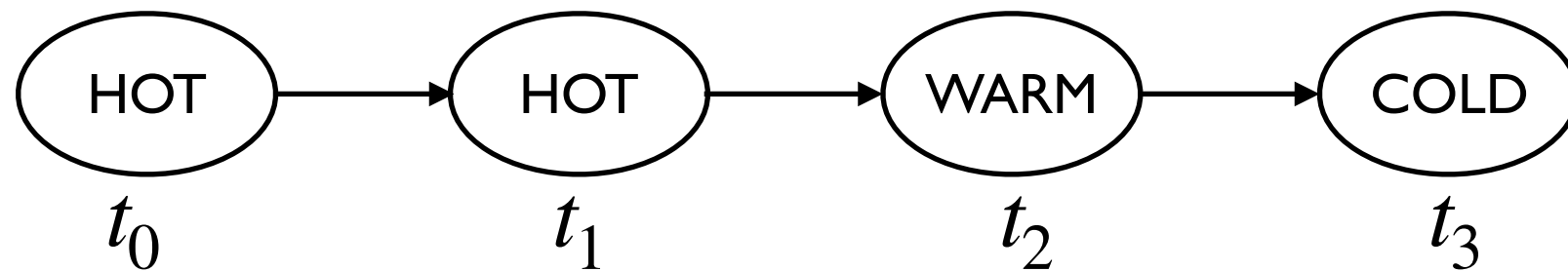
An initial probability distribution over states:

$$\pi_0 = [P(\text{HOT at time 0}) \quad P(\text{COLD at time 0}) \quad P(\text{WARM at time 0})] = [0.5 \quad 0.3 \quad 0.2]$$

Transition probability matrix  $A$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & P(\text{HOT}|\text{HOT}) & P(\text{COLD}|\text{HOT}) & P(\text{WARM}|\text{HOT}) \\ \text{COLD} & P(\text{HOT}|\text{COLD}) & P(\text{COLD}|\text{COLD}) & P(\text{WARM}|\text{COLD}) \\ \text{WARM} & P(\text{HOT}|\text{WARM}) & P(\text{COLD}|\text{WARM}) & P(\text{WARM}|\text{WARM}) \end{bmatrix} = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.2 & 0.5 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{bmatrix}$$

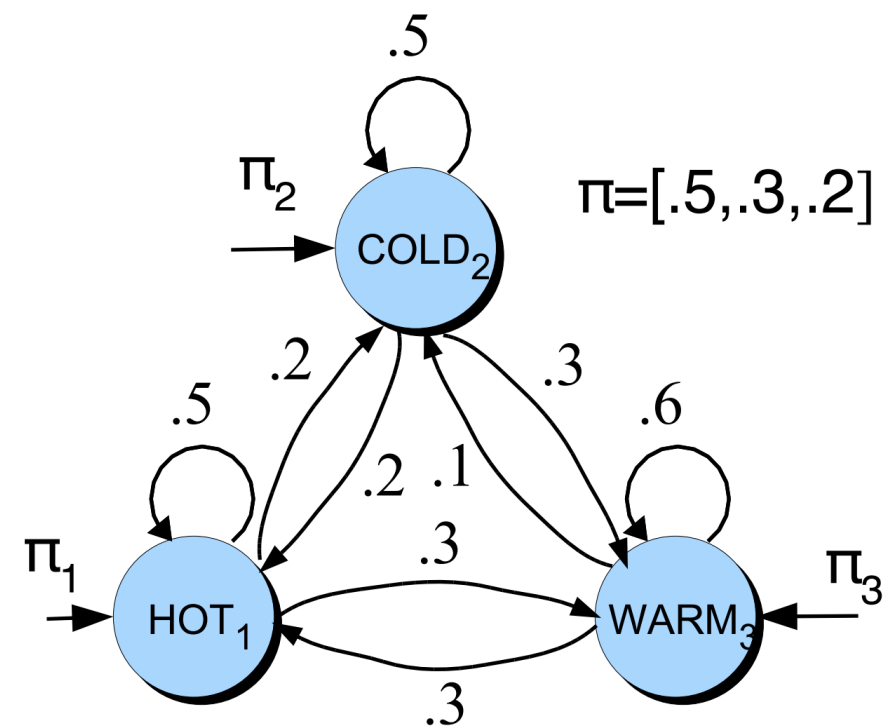
# Markov chain ingredients



$$S = \{\text{HOT}, \text{COLD}, \text{WARM}\},$$

$$\pi_0 = [0.5 \quad 0.3 \quad 0.2]$$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & 0.5 & 0.2 & 0.3 \\ \text{COLD} & 0.2 & 0.5 & 0.3 \\ \text{WARM} & 0.3 & 0.1 & 0.6 \end{bmatrix}$$



# What can we do with Markov chains?

- Generation: generate sequences that follow the probabilities of the states.
- **Predict probabilities of sequences** of states.  $P(\text{COLD, HOT, COLD, HOT})$
- **Inference**: compute probability of being in a particular state at time  $t$
- **Stationary distribution**: Find the steady state after running for a long time
- Decoding: compute most likely sequences of states

# Compute the probability of sequences: Pair-share

1.  $P(\text{HOT}, \text{WARM}, \text{WARM}, \text{COLD})$

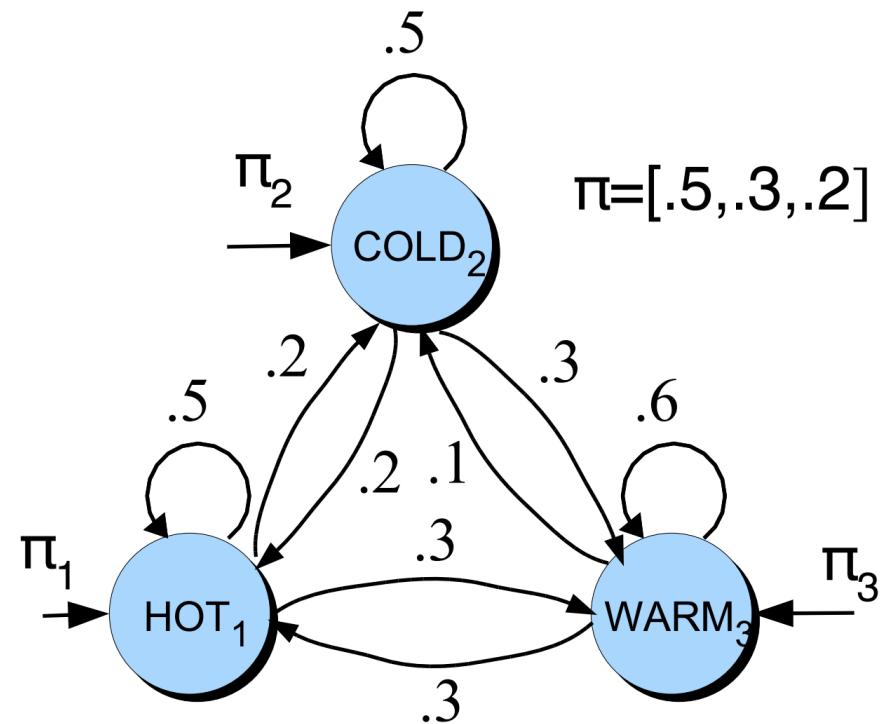
2.  $P(\text{HOT}, \text{COLD}, \text{HOT}, \text{COLD})$

Hint: If we want to predict the future, all that matters is present.

$S = \{\text{HOT}, \text{COLD}, \text{WARM}\},$

$\pi_0 = [0.5 \quad 0.3 \quad 0.2],$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & 0.5 & 0.2 & 0.3 \\ \text{COLD} & 0.2 & 0.5 & 0.3 \\ \text{WARM} & 0.3 & 0.1 & 0.6 \end{bmatrix}$$



# Compute the probability of sequences: Pair-share

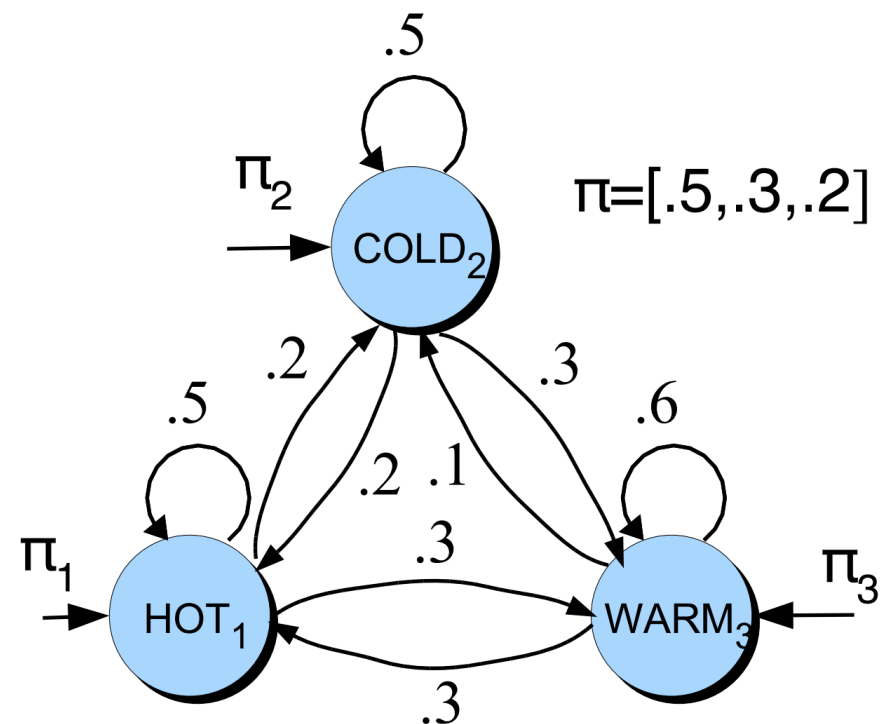
$$\begin{aligned} P(\text{HOT}, \text{WARM}, \text{WARM}, \text{COLD}) &= P(\text{HOT}) \times P(\text{WARM}|\text{HOT}) \\ &\quad \times P(\text{WARM}|\text{WARM}) \times P(\text{COLD}|\text{WARM}) \\ &= 0.5 \times 0.3 \times 0.6 \times 0.1 \\ &= 0.0090 \end{aligned}$$

Hint: If we want to predict the future, all that matters is present.

$$S = \{\text{HOT}, \text{COLD}, \text{WARM}\},$$

$$\pi_0 = [0.5 \quad 0.3 \quad 0.2],$$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & 0.5 & 0.2 & 0.3 \\ \text{COLD} & 0.2 & 0.5 & 0.3 \\ \text{WARM} & 0.3 & 0.1 & 0.6 \end{bmatrix}$$



# Markov chain: Inference

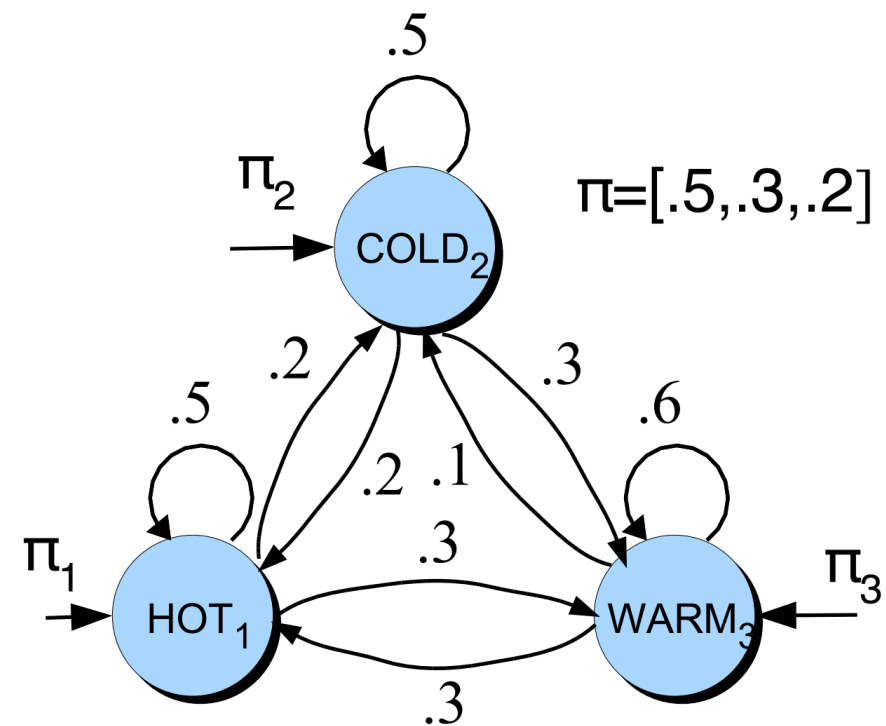
What is the probability of being in a particular state at time  $t$ ?

Example: What is the probability of HOT at time 1?

$P(\text{HOT at time zero}) \times P(\text{HOT}|\text{HOT}) + P(\text{WARM at time zero}) \times P(\text{HOT}|\text{WARM}) + P(\text{COLD at time zero}) \times P(\text{HOT}|\text{COLD})$

$S = \{\text{HOT, COLD, WARM}\},$

$\pi_0 = [0.5 \quad 0.3 \quad 0.2]$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & 0.5 & 0.2 & 0.3 \\ \text{COLD} & 0.2 & 0.5 & 0.3 \\ \text{WARM} & 0.3 & 0.1 & 0.6 \end{bmatrix}$$


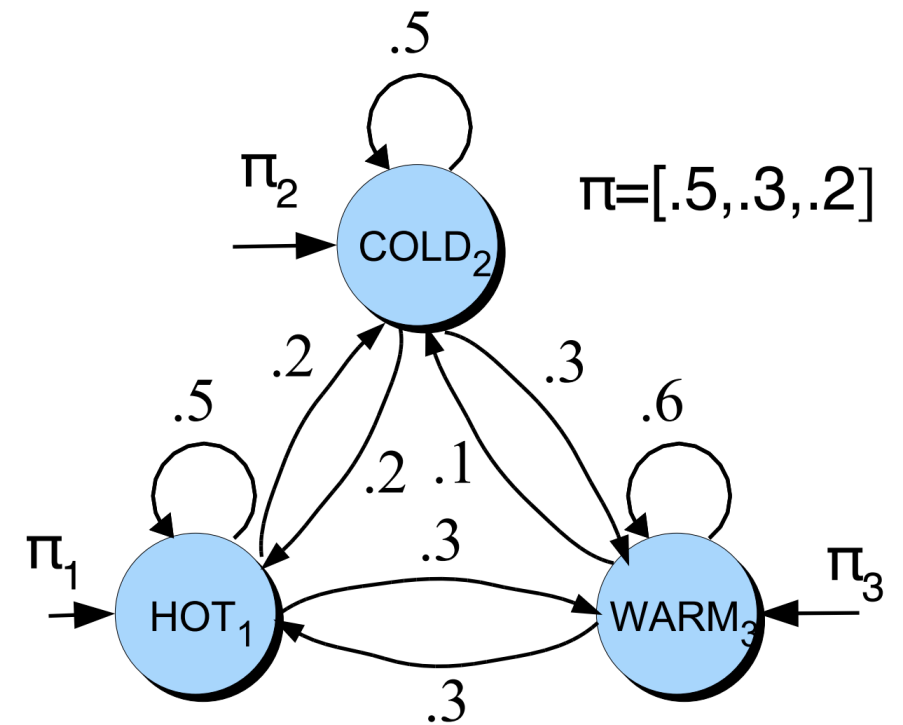
# Markov chain: Inference

What is the probability of being in a particular state at time  $t$ ?

$$S = \{\text{HOT}, \text{COLD}, \text{WARM}\},$$

$$\pi_0 = [0.5 \quad 0.3 \quad 0.2]$$

$$A = \begin{bmatrix} & \text{HOT} & \text{COLD} & \text{WARM} \\ \text{HOT} & 0.5 & 0.2 & 0.3 \\ \text{COLD} & 0.2 & 0.5 & 0.3 \\ \text{WARM} & 0.3 & 0.1 & 0.6 \end{bmatrix}$$



Example: What is the probability of HOT at time 1?


Dot product between  $\pi$  at time 0 and the first column of the transition matrix.

$$P(\text{HOT at time zero}) \times P(\text{HOT}|\text{HOT}) + P(\text{WARM at time zero}) \times P(\text{HOT}|\text{WARM}) + P(\text{COLD at time zero}) \times P(\text{HOT}|\text{COLD})$$

$$\begin{matrix} & \pi_0 \\ [0.5 & 0.3 & 0.2] \end{matrix} \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.2 & 0.5 & 0.3 \\ 0.3 & 0.1 & 0.6 \end{bmatrix} = \begin{matrix} & \pi_1 \\ [0.37 & 0.27 & 0.36] \end{matrix}$$



# Lecture outline

- Recap
- Temporal probabilistic models
- Markov chains
- Stationary distribution 
- Markov chains in Natural Language
- Hidden Markov models introduction

# Stationary process assumption

**A stationary distribution of a Markov chain** is a probability distribution that remains unchanged in the Markov chain as time progresses.

A distribution  $\pi$  on states  $S$  is stationary when  $\pi A = \pi$ , where  $A$  is the transition matrix.

The mechanism that regulates how state variables change overtime is stationary, that is it can be described by a single transition model.

$P(S_t | S_{t-1})$  is the same for all  $t$

# Stationary distribution

Suppose TransLink launches Downtown to UBC SkyTrain. In the first month of operation it was found that 20% of the commuters going to UBC started using it and 80% of the commuters were still using other modes of transportation. The following transition matrix was determined from the records of other transit systems.

$$S = \{\text{SkyTrain}, \text{Other}\}$$

$$\pi = [0.20 \quad 0.80], A = \begin{bmatrix} & \text{SkyTrain} & \text{Other} \\ \text{SkyTrain} & 0.9 & 0.1 \\ \text{Other} & 0.4 & 0.6 \end{bmatrix}$$

# Stationary distribution

1. What percentage of the commuters will be using the SkyTrain after two months?
2. What about after three months?
3. What's the percentage of the commuters using the system after the service has been in place for a long time?

# Stationary distribution

What percentage of the commuters will be using the SkyTrain after two months, three months?

$$\pi_0 = [0.20 \quad 0.80]$$

$$\pi_1 = [0.20 \quad 0.80] \begin{bmatrix} 0.9 & 0.1 \\ 0.4 & 0.6 \end{bmatrix} = [0.5 \quad 0.5]$$

$$\pi_2 = [0.5 \quad 0.5] \begin{bmatrix} 0.9 & 0.1 \\ 0.4 & 0.6 \end{bmatrix} = [0.65 \quad 0.35]$$

Big improvement at each time step!! How long does this continue?

# Stationary distribution

The chain reach a steady state  $\pi A = \pi$  after the 26<sup>th</sup> time step, which is  $[0.80 \quad 0.20]$ .

What's the percentage of the commuters using the SkyTrain after the service has been in place for a long time?

- In the long run we can expect 80% of the commuters using the SkyTrain.

# Stationary Markov chain (SMC)

A stationary Markov Chain : for all  $t > 0$

$$P(S_{t+1} | S_0, \dots, S_t) = P(S_{t+1} | S_t) \text{ (Markov assumption)}$$

$P(S_{t+1} | S_t)$  is the same for all  $t$  (Stationary)


So we only need to specify:  $P(S_{t+1} | S_t)$  and  $P(S_0)$

Simple Model, easy to specify and often the natural model

The network can extend indefinitely

Variations of SMC are at the core of many Natural Language Processing (NLP) applications! (E.g., PageRank)

# Lecture outline

- Recap
- Temporal probabilistic models
- Markov chains
- Stationary distribution
- Markov chains in Natural Language 
- Hidden Markov models introduction



# Language models

Suppose your states are words instead of weather and you are computing probabilities of sequences of words.

What does it tell us?

Which sequence of words is more likely to occur in English?

$P(\text{in the age of data algorithms have the answers})$

$p(\text{answers age data of in algorithms the the have})$

# Language model

Compute the probability of a sentence or a sequence of words

$$P(w_1, w_2, \dots, w_t)$$

A related task: What's the probability of an upcoming word?

$$P(w_t | w_1, w_2, \dots, w_{t-1})$$

Example: Your smartphone's or Gmail's feature of next word(s) suggestion

A model that computes either of these probabilities is called a **language model**.

# Language models

Powerful idea in natural language processing and helps in many tasks.

## Machine translation

Example:  $P(\text{In the age of data algorithms have the answer}) > P(\text{the age data of in algorithms answer the have})$

## Spelling correction

Example: My office is a 20 minuet bike ride from my home.

$P(20 \text{ minute bike ride from my home}) > P(20 \text{ minuet bike ride from my home})$

## Speech recognition

$P(\text{I read a book}) > P(\text{Eye red a book})$

# Language modeling: Apply chain rule

Example: Suppose we want to calculate the probability of the following sequence of words:

$$\begin{aligned} P(\text{In the age of data algorithms have the answer}) &= P(\text{In}) \times P(\text{the}|\text{In}) \\ &\quad \times P(\text{age}|\text{In the}) \times P(\text{of}|\text{In the age}) \\ &\quad \times P(\text{data}|\text{In the age of}) \\ &\quad \times P(\text{algorithms}|\text{In the age of data}) \\ &\quad \times P(\text{have}|\text{In the age of data algorithms}) \\ &\quad \dots \end{aligned}$$

What if we just count occurrences of these sequences in large amount of text to get conditional probabilities?

**BAD IDEA!!** The counts will be tiny and the model will be very sparse.

# Calculating probability of a sentence

**Markov assumption:** When predicting future the past doesn't matter only the present.

$$P(\text{algorithms} | \text{In the age of data}) \approx P(\text{algorithms} | \text{data})$$

$$\begin{aligned} P(\text{In the age of data algorithms have the answer}) &= P(\text{In}) \times P(\text{the} | \text{In}) \\ &\quad \times P(\text{age} | \text{the}) \\ &\quad \times P(\text{of} | \text{age}) \\ &\quad \times P(\text{data} | \text{of}) \\ &\quad \times P(\text{algorithms} | \text{data}) \\ &\quad \times P(\text{have} | \text{algorithms}) \\ &\quad \times P(\text{the} | \text{have}) \\ &\quad \times P(\text{answer} | \text{the}) \end{aligned}$$

Estimating conditional probabilities

$$P(\text{algorithms} | \text{data}) = \frac{\text{Count}(\text{data algorithms})}{\text{Count}(\text{data})}$$

# Language model: pair-share

Given the information below, what are the probabilities of the following sequences?

1.  $P(I \text{ like } AI)$
2.  $P(AI \text{ like } I)$

Conditional probabilities

	<i>I</i>	<i>like</i>	<i>AI</i>
<i>I</i>	0.002	0.30	0.002
<i>like</i>	0.03	0.0001	0.2
<i>AI</i>	0.001	0.002	0.0001

Suppose

$$P(I) = 0.25$$

$$P(AI) = 0.01$$

# N-gram language models

First order Markov model

$$P(w_t | w_1, w_2, \dots, w_{t-1}) = (w_t | w_{t-1})$$

Second order Markov model

$$P(w_t | w_1, w_2, \dots, w_{t-1}) = (w_t | w_{t-2}, w_{t-1})$$

...

Fourth order Markov model

$$P(w_t | w_1, w_2, \dots, w_{t-1}) = (w_t | w_{t-4}, w_{t-3}, w_{t-2}, w_{t-1})$$

# ASIDE: Google Ngram release

All Our N-gram are Belong to You

<https://ai.googleblog.com/2006/08/all-our-n-gram-are-belong-to-you.html>

*“Here at Google Research we have been using word n-gram models for a variety of R&D projects, such as statistical machine translation, speech recognition, spelling correction, entity detection, information extraction, and others.*

*That's why we decided to share this enormous dataset with everyone. We processed **1,024,908,267,229** words of running text and are publishing the counts for all **1,176,470,663** five-word sequences that appear at least 40 times. There are **13,588,391** unique words, after discarding words that appear less than 200 times.”*



# Markov chain applications

## 18 Applications

18.1 **Physics**

18.2 Chemistry

18.3 Testing

18.4 Speech recognition

18.5 Information and computer science

18.6 Queueing theory

18.7 Internet applications

18.8 Statistics

18.9 Economics and finance

18.10 Social sciences

18.11 Mathematical biology

18.12 Genetics

18.13 Games

18.14 Music

18.15 Baseball

18.16 Markov text generators

18.17 Bioinformatics

# Does this look like Python to you?

```
import sys
import warnings
import ast
import numpy.core.overrides import set_module
# While not in __all__, matrix_power used to be defined here, so we import
# it for backward compatibility
    getT = T.fget
    getI = I.fget

def _from_string(data):
    for char in '[]':
        data = data.replace(char, '')

    rows = str.split(';')
    rowtup = []
    for row in rows:
        trow = newrow
        coltup.append(thismat)
        rowtup.append(concatenate(coltup, axis=-1))
    return self[0, 0]
    elif ndim == 0:
        [ 2. +2.j,  6. +6.j, 10.+10.j],
        Ncols = len(newrow)
    else:
        return NotImplemented

def __ipow__(self, other):
    self[:] = self * other
    ret : ndarray
        If `self` is singular.
    See `amin` for complete descriptions
    See Also
```

Not written by a programmer.  
Generated by a Bayesian network  
(Markov chain)!

# Other fun things with Markov chains

The Life and Work of A.A. Markov

Markov chains “explained visually”


Snakes and ladders

Candyland

Yahtzee

Chess pieces returning home and K-pop vs. ska

# Lecture outline

- Recap
- Temporal probabilistic models
- Markov chains
- Stationary distribution
- Markov chains in Natural Language
- Hidden Markov models introduction 

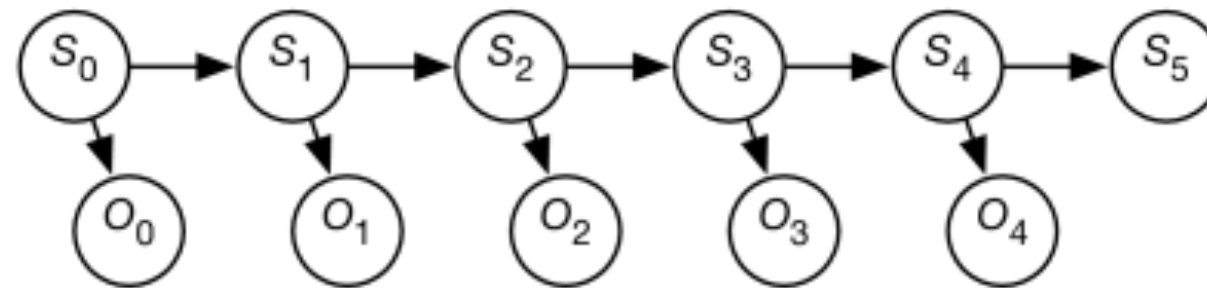
# Motivation: Hidden Markov models

Very often the things you observe in the real world are only a function of some other **hidden** variable.

- Speech sounds are the outputs of hidden phonemes
- Words are the outputs of hidden parts-of-speech
- Encrypted symbols are outputs of hidden messages
- Genes are outputs of functional relationships
- Weather is the output of hidden climate conditions
- Stock prices are the output of market conditions

# Hidden Markov model

Augmentation of a Markov chain to include observations



The assumptions behind an HMM as depicted in the Bayesian network

- The state at time  $t + 1$  only directly depends upon the state at time  $t$ :  $P(S_i | S_0, S_1, \dots, S_{i-1}) = P(S_i | S_{i-1})$
- The observation at time  $t$  only directly depends upon the state at time  $t$ :  $P(O_i | S_0, S_1, \dots, S_{i-1}, O_0, O_1, \dots, O_{i-1}) = P(O_i | S_i)$

# HMM example

Suppose you have a little robot that is trying to estimate the posterior probability that you are **Happy (H or 😊)** or **Sad (S or 😞)**, given that the robot has observed whether you are doing one of the following activities: Learning AI (L or 📖), Eat (E or 🍏), Cry (C or 🐱💧), Social media (F or 📘)

The robot is trying to estimate the unknown (hidden) state  $Q$ , where  $Q = H$  when you are happy (😊) and  $Q = S$  when you are sad (😞).

The robot is able to observe the activity you are doing:  
 $O = L, E, C, F$

# HMM example

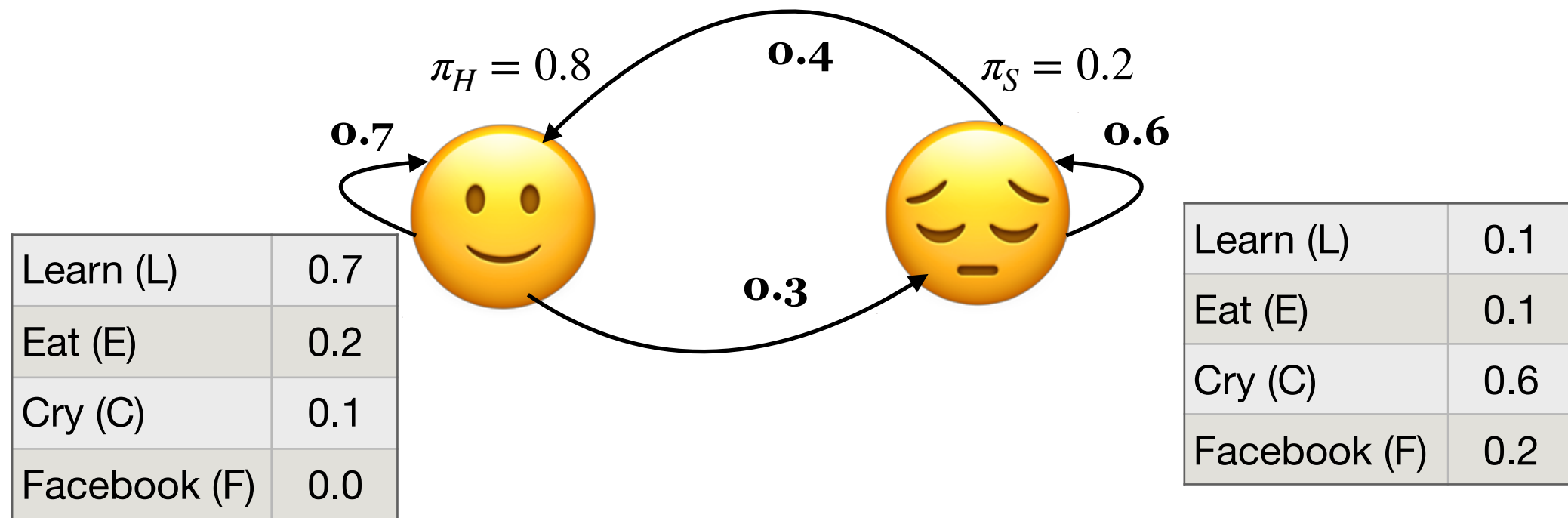
Example questions we could answer

What is  $P(Q = \text{😞} | O = F)$ ?

What is the best possible sequence of state of mind (e.g., 😊, 😞, 😞, 😊 ) given an observation sequence?



# Components of an HMM



State space (e.g., 😊 (H), 😞 (S))

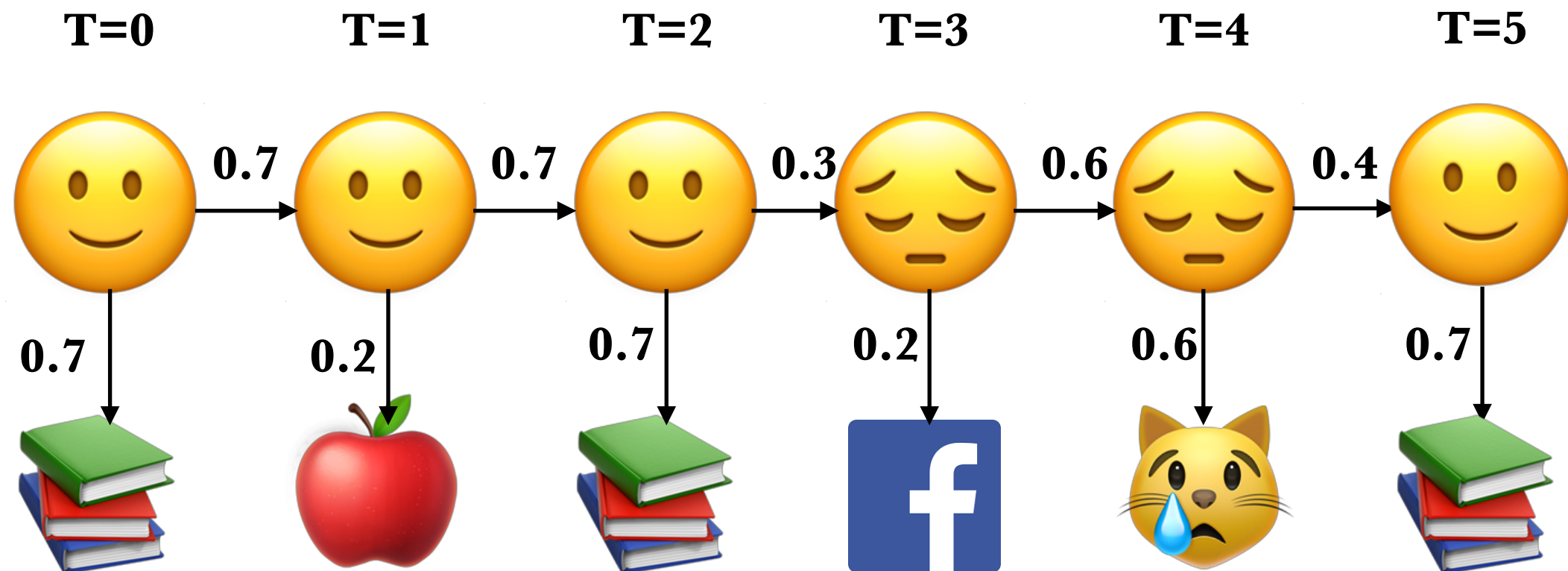
An initial probability distribution over the states

$P(S_{t+1} | S_t)$ : Transition probabilities (dynamics)

$P(O_t | S_t)$ : Emission probabilities (sensor model)

# Components of an HMM

Yielding the state sequence and the observation sequence



# HMM: Definition

A hidden Markov model (HMM) is specified by the 5-tuple:  
 $\{S, O, \pi, A, B\}$

$S = \{s_1, s_2, \dots, s_n\}$  is a set of states (e.g., moods)

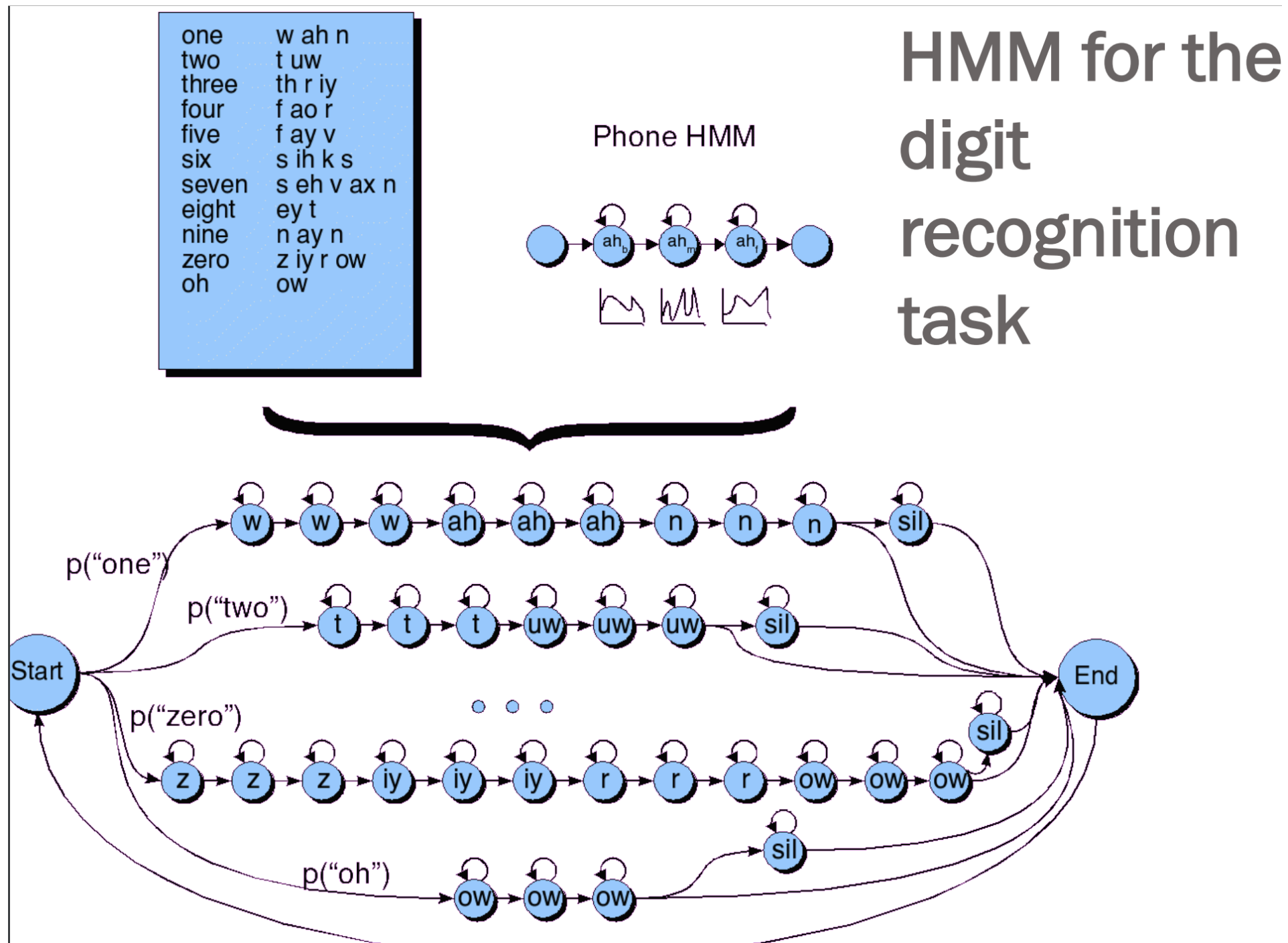
$Y = \{y_1, y_2, \dots, y_k\}$  is output alphabet (e.g., set of activities)

$\pi = \pi_1, \pi_2, \dots, \pi_n$  is initial state discrete probability distribution

Transition probability matrix  $A$ , where each  $a_{ij}$  represents the probability of moving from state  $s_i$  to state  $s_j$

Emission probabilities  $B = b_i(o), i \in S, o \in Y$

# HMM for digit recognition



# Revisit: Learning outcomes

From this lecture, students are expected to be able to:

- Specify a Markov chain and compute the probability of a sequence of states
- Explain the general idea of stationary distribution
- Justify and apply Markov chains to compute the probability of a natural language sentence
- Specify the components of a hidden Markov model

# To do

You are responsible to read 8.5 Sequential Probability Models from the textbook.

# Coming up

## 9.2 One-Off Decisions

