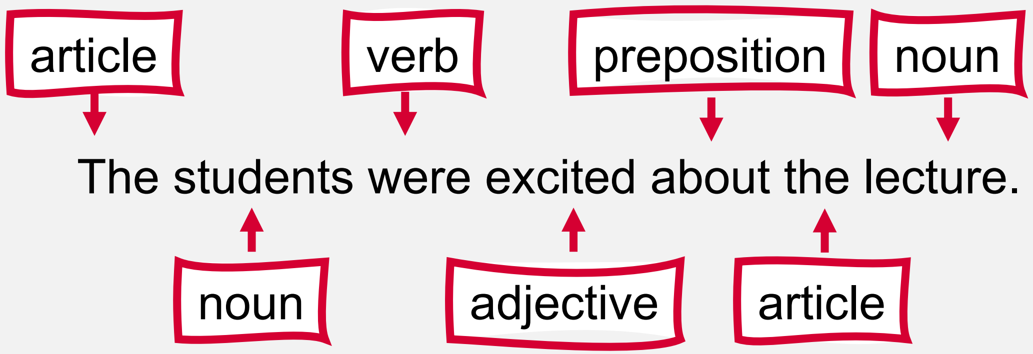
Hidden Markov Models and POS Tagging

Natalie Parde

UIC CS 421

Sequence Modeling and Sequence Labeling

In general: assigning labels to individual tokens or spans of tokens given a longer string of input



Sequence Labeling

- Objective: Find the label for the next item, based on the labels of other items in the sequence.

verb determiner

Give me a **break**!

pronoun noun

verb noun

Did the window **break**?

determiner verb

Why perform sequence labeling?

- In document-level text classification, models assume that the individual datapoints being classified are disconnected and independent
- Many NLP problems do not satisfy this assumption! Instead, they involve
  - Interconnected decisions
  - Each of which are mutually dependent
  - Each of which resolve different ambiguities

Example Sequence Labeling Applications

- Named entity recognition
- Semantic role labeling

person organization

Natalie Parde works at the **University of Illinois at Chicago** and lives in **Chicago, Illinois**.

location

agent source destination

Natalie drove for 15 hours from **Dallas** to **Chicago** in her hail-damaged **Honda Accord**.

instrument

This Week's Topics

Tuesday

- Hidden Markov Models
- Forward Algorithm
- Viterbi Algorithm
- Forward-Backward Algorithm

Thursday

- Parts of Speech
- POS Tagsets
- POS Tagging

This Week's Topics

Hidden Markov Models
Forward Algorithm
Viterbi Algorithm
Forward-Backward Algorithm

Tuesday

Thursday

Parts of Speech
POS Tagsets
POS Tagging

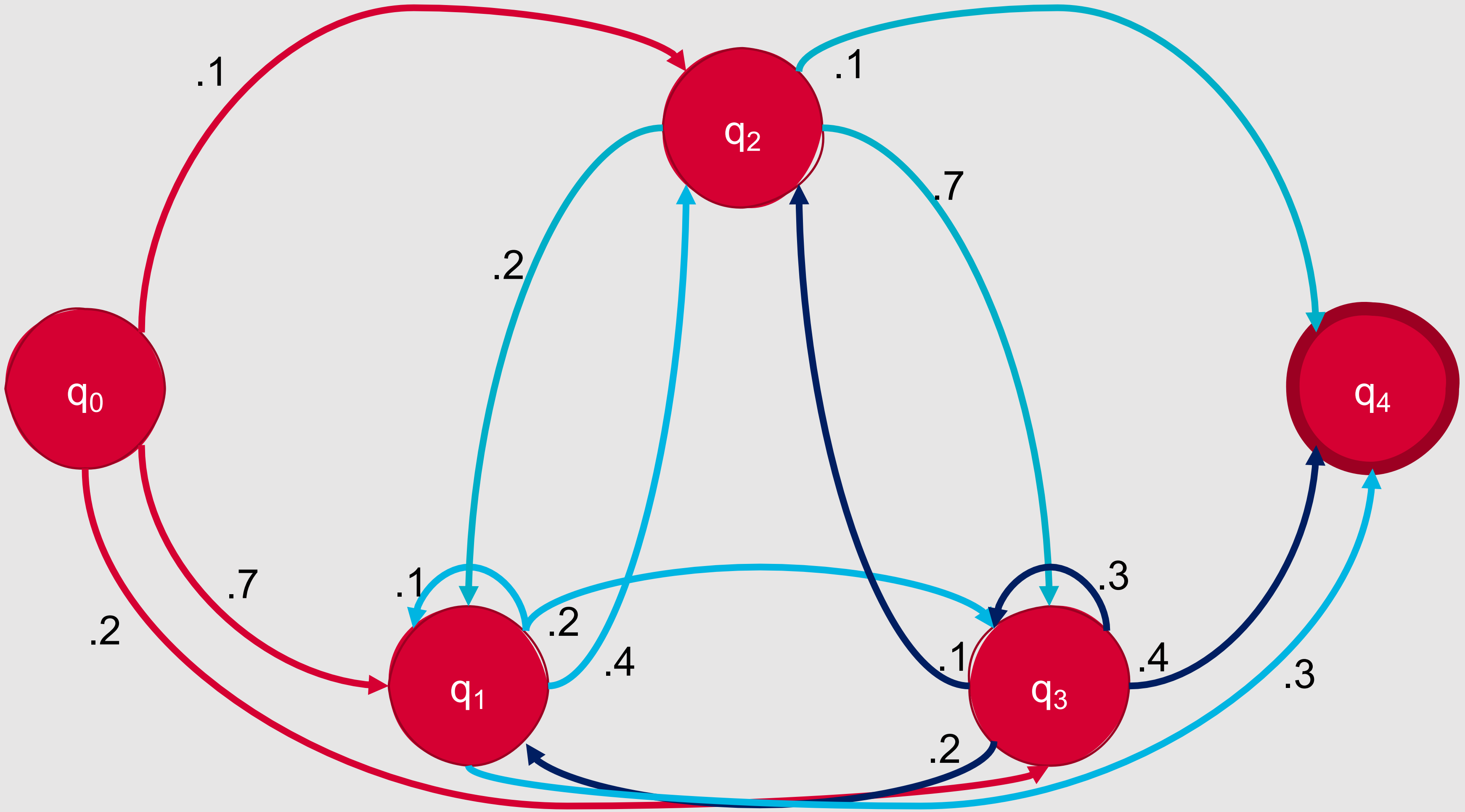
Probabilistic Sequence Models

- We can perform multiple, interdependent classifications to address a greater problem using probabilistic sequence models
- These models can be neural networks, but they can also be lighter-weight alternatives closer to finite state automata known as **hidden Markov models**
- Hidden Markov models are **probabilistic generative models for sequences** that make predictions based on an underlying set of **hidden states**

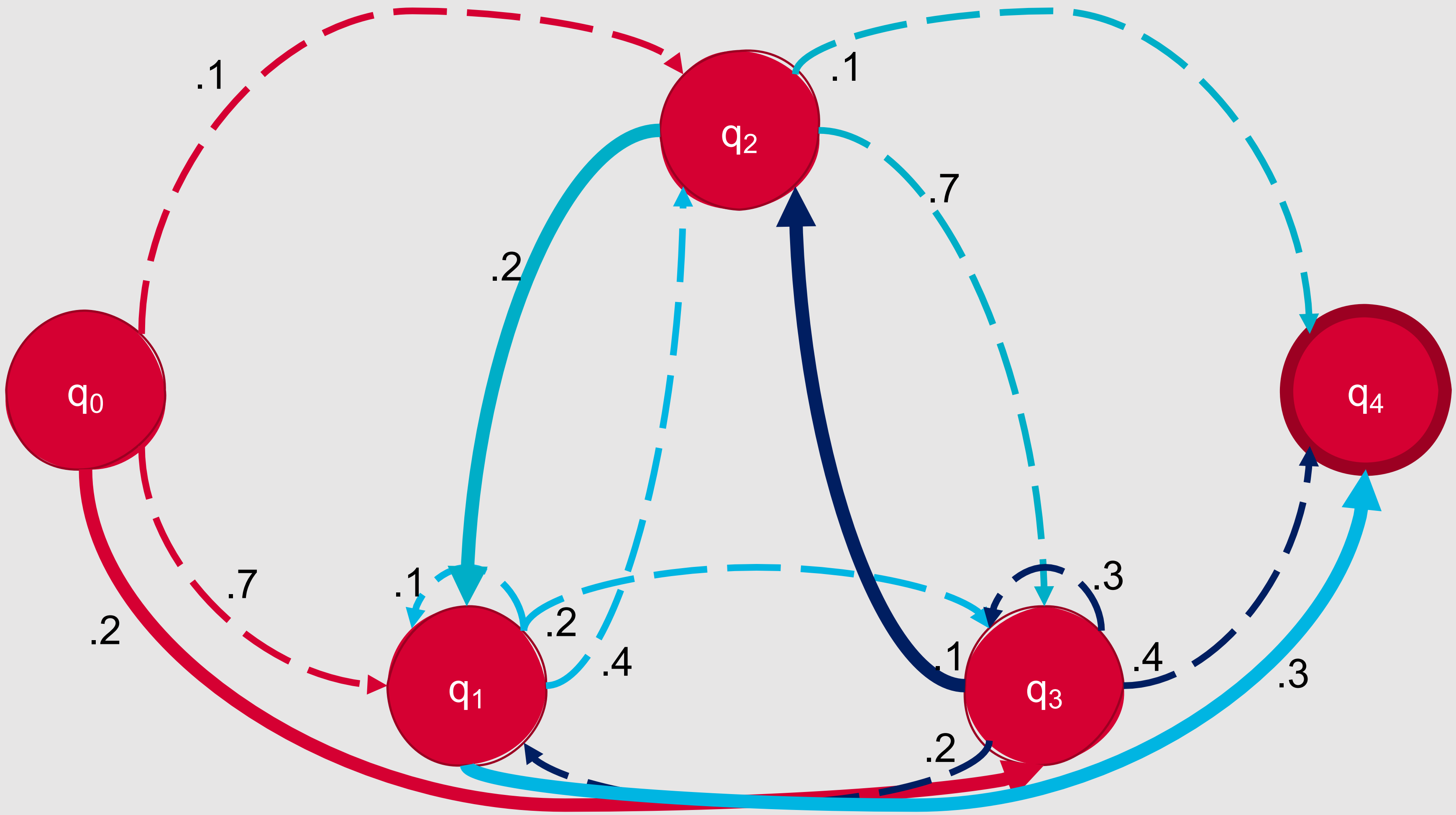
What are Markov Models?

- **Finite state automata with probabilistic state transitions**
- Markov Property: The future is independent of the past, given the present.
  - In other words, the next state only depends on the current state …it is independent of previous history.
- Also referred to as **Markov Chains**

Sample Markov Model



Sample Markov Model



P(q_3 q_2 q_1 q_4)
= .2 * .1 * .2 * .3
= .0012

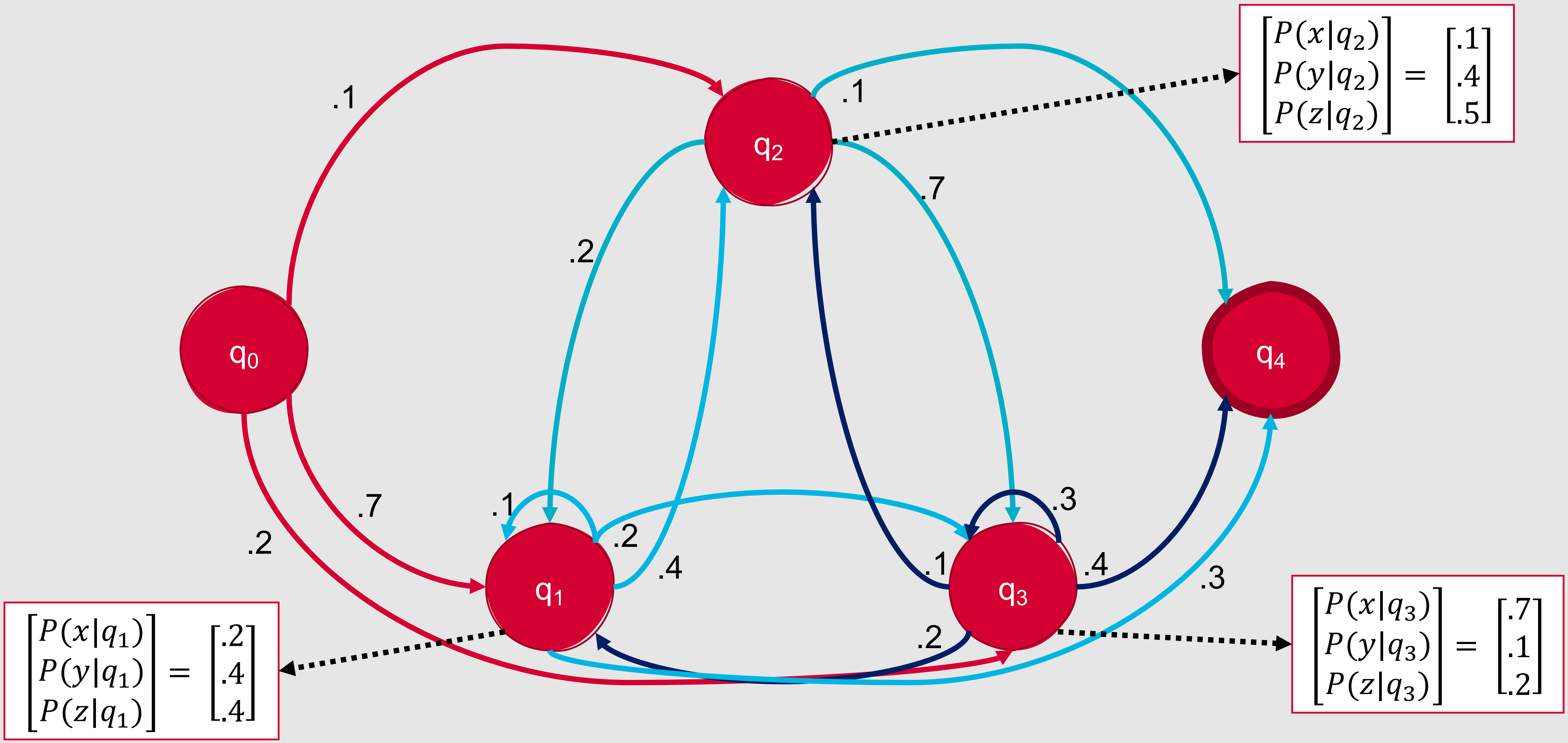
Hidden Markov Models

- Markov models that assume an underlying set of hidden (unobserved) states in which the model can be
- Assume probabilistic transitions between states over time
- Assume probabilistic generation of items (e.g., tokens) from states

Formal Definition

- A Hidden Markov Model can be specified by enumerating the following properties:
  - The set of states, ***Q***
  - A sequence of observation likelihoods, ***B***, also called emission probabilities, each expressing the probability of an observation being generated from a state *i*
  - A start state, q_0, and final state, q_F, that are not associated with observations

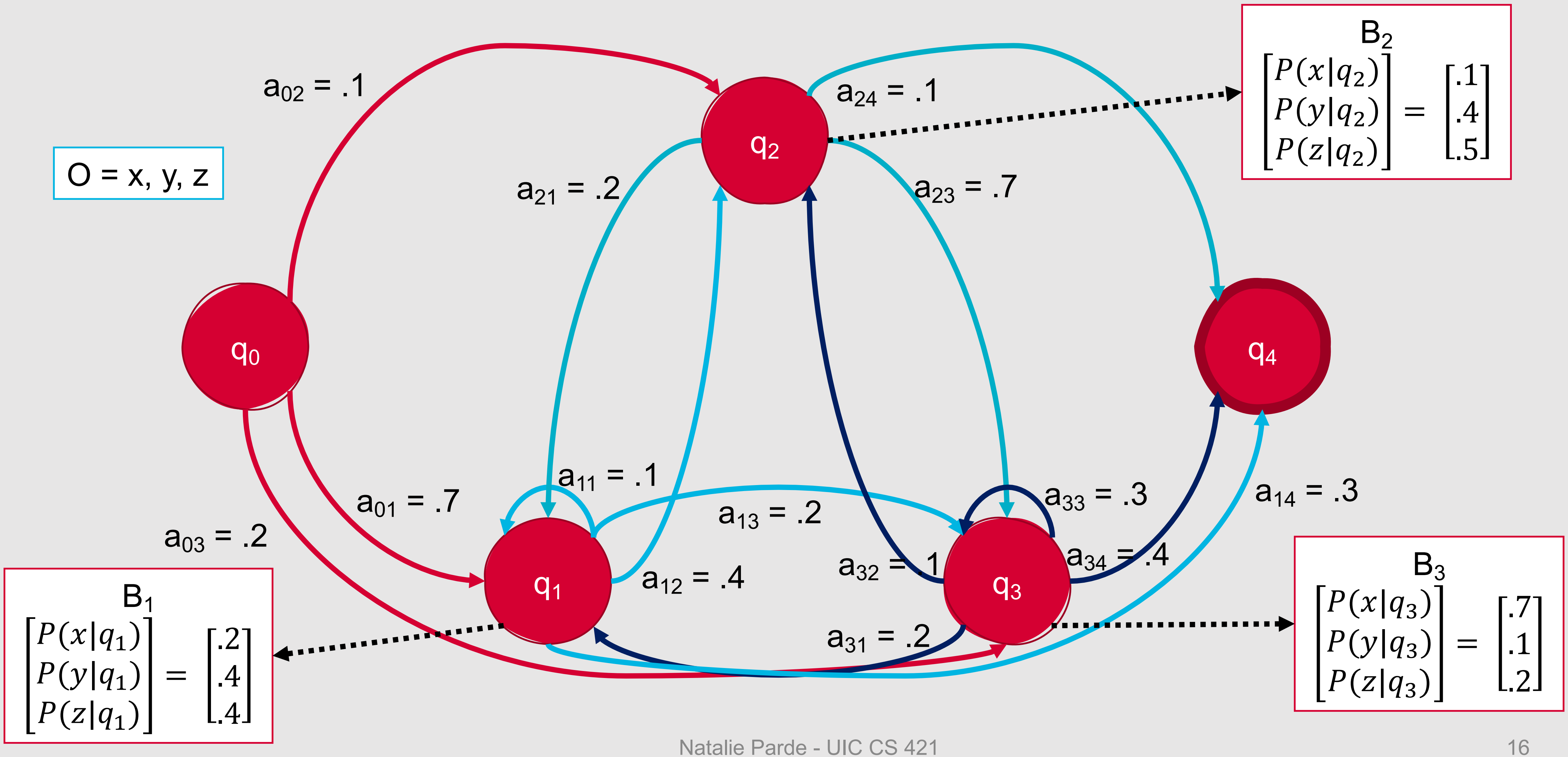
Sample Hidden Markov Model



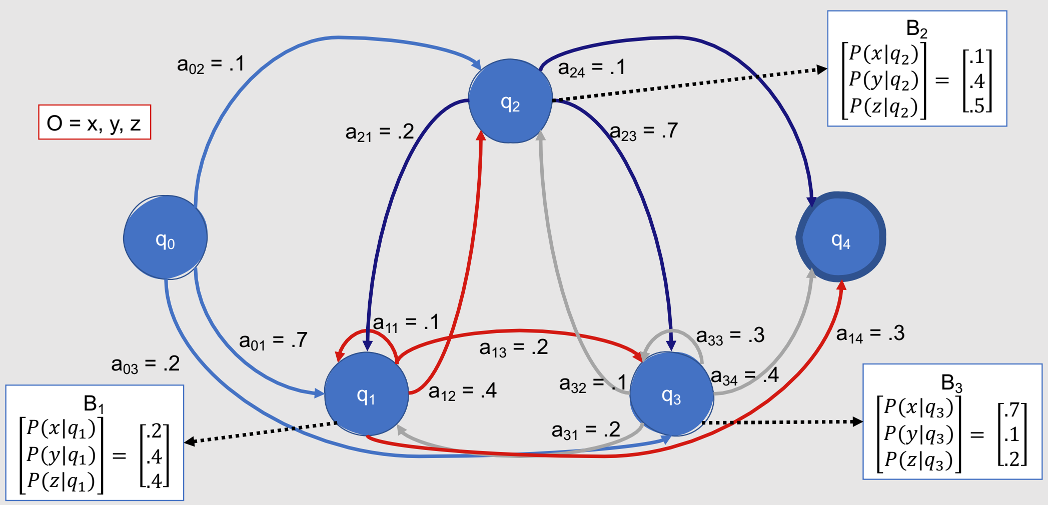
Formal Definition

- A Hidden Markov Model can be specified by enumerating the following properties:
  - The set of states, ***Q***
  - A sequence of observation likelihoods, ***B***, also called emission probabilities, each expressing the probability of an observation o_t being generated from a state *i*
  - A start state, $\boldsymbol{q_0}$, and final state, $\boldsymbol{q_F}$, that are not associated with observations, together with transition probabilities out of $\boldsymbol{q_0}$ and into $\boldsymbol{q_F}$
  - A transition probability matrix, ***A***, where each a_{ij} represents the probability of moving from state *i* to state *j*, such that $\sum_{j=1}^{n} a_{ij} = 1 \; \forall i$
  - A sequence of T observations, ***O***, each drawn from a vocabulary V = $v_1, v_2, \ldots, v_V$

Sample Hidden Markov Model

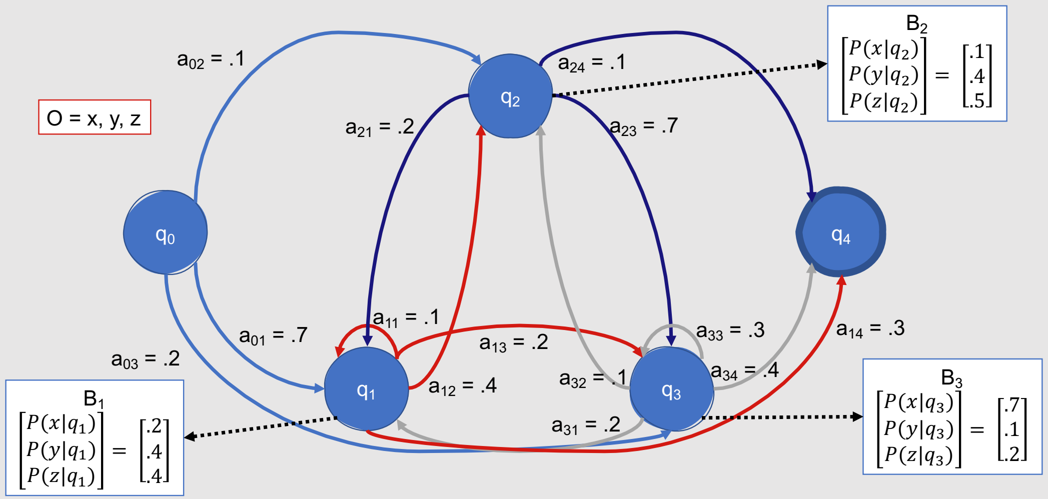


Corresponding Transition Matrix



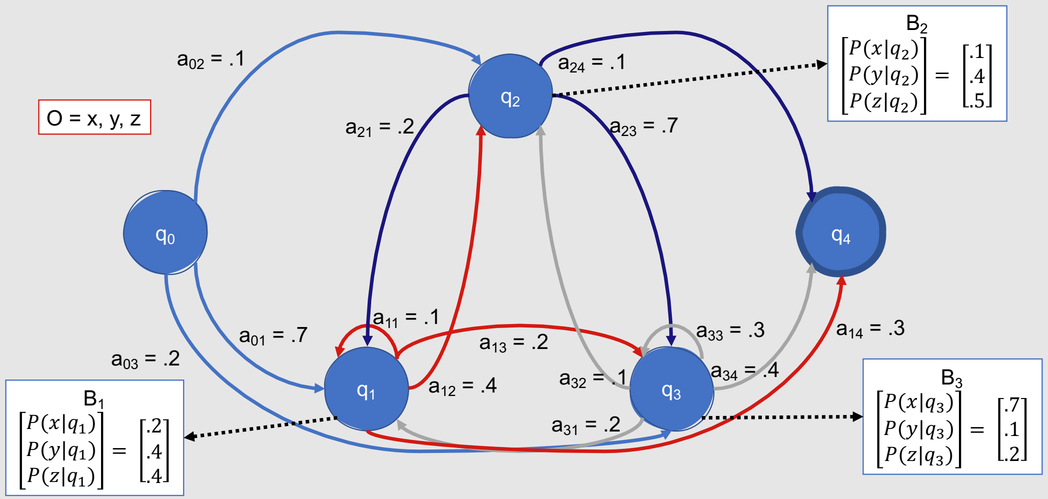
	q0	q1	q2	q3	q4
q0	N/A	.7	.1	.2	N/A
q1					
q2					
q3					
q4					

Corresponding Transition Matrix



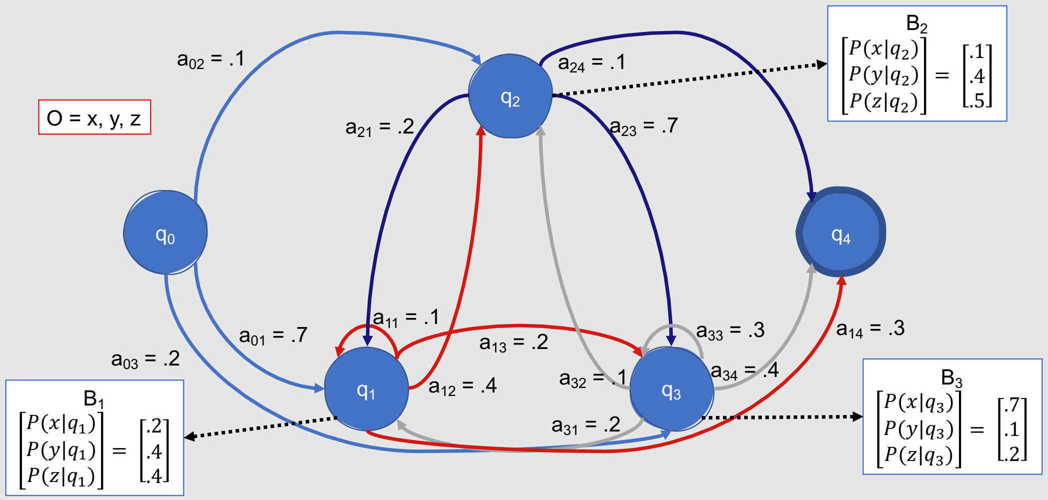
	q0	q1	q2	q3	q4
q0	N/A	.7	.1	.2	N/A
q1	N/A	.1	.4	.2	.3
q2					
q3					
q4					

Corresponding Transition Matrix



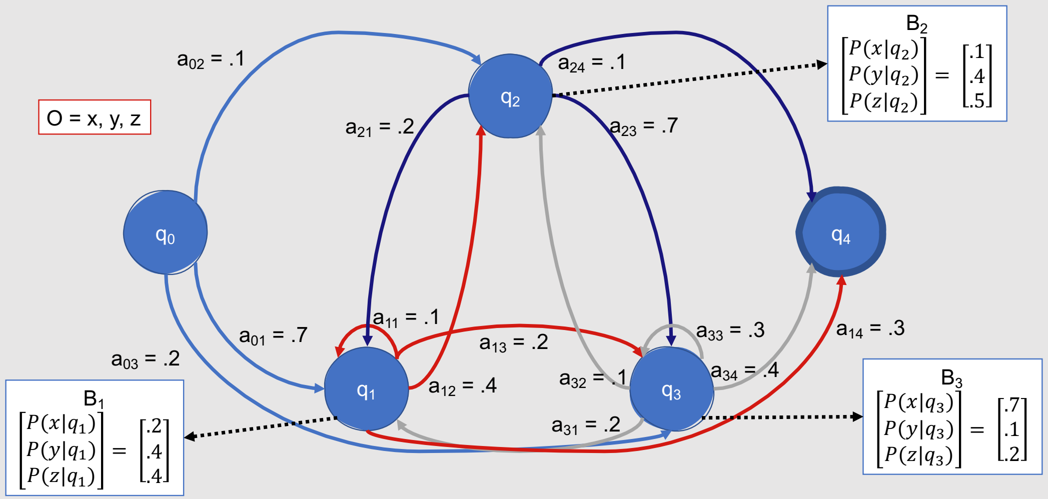
	q0	q1	q2	q3	q4
q0	N/A	.7	.1	.2	N/A
q1	N/A	.1	.4	.2	.3
q2	N/A	.2	N/A	.7	.1
q3					
q4					

Corresponding Transition Matrix



	q0	q1	q2	q3	q4
q0	N/A	.7	.1	.2	N/A
q1	N/A	.1	.4	.2	.3
q2	N/A	.2	N/A	.7	.1
q3	N/A	.2	.1	.3	.4
q4					

Corresponding Transition Matrix



	q0	q1	q2	q3	q4
q0	N/A	.7	.1	.2	N/A
q1	N/A	.1	.4	.2	.3
q2	N/A	.2	N/A	.7	.1
q3	N/A	.2	.1	.3	.4
q4	N/A	N/A	N/A	N/A	N/A

HMMs can also be used for probabilistic text generation!

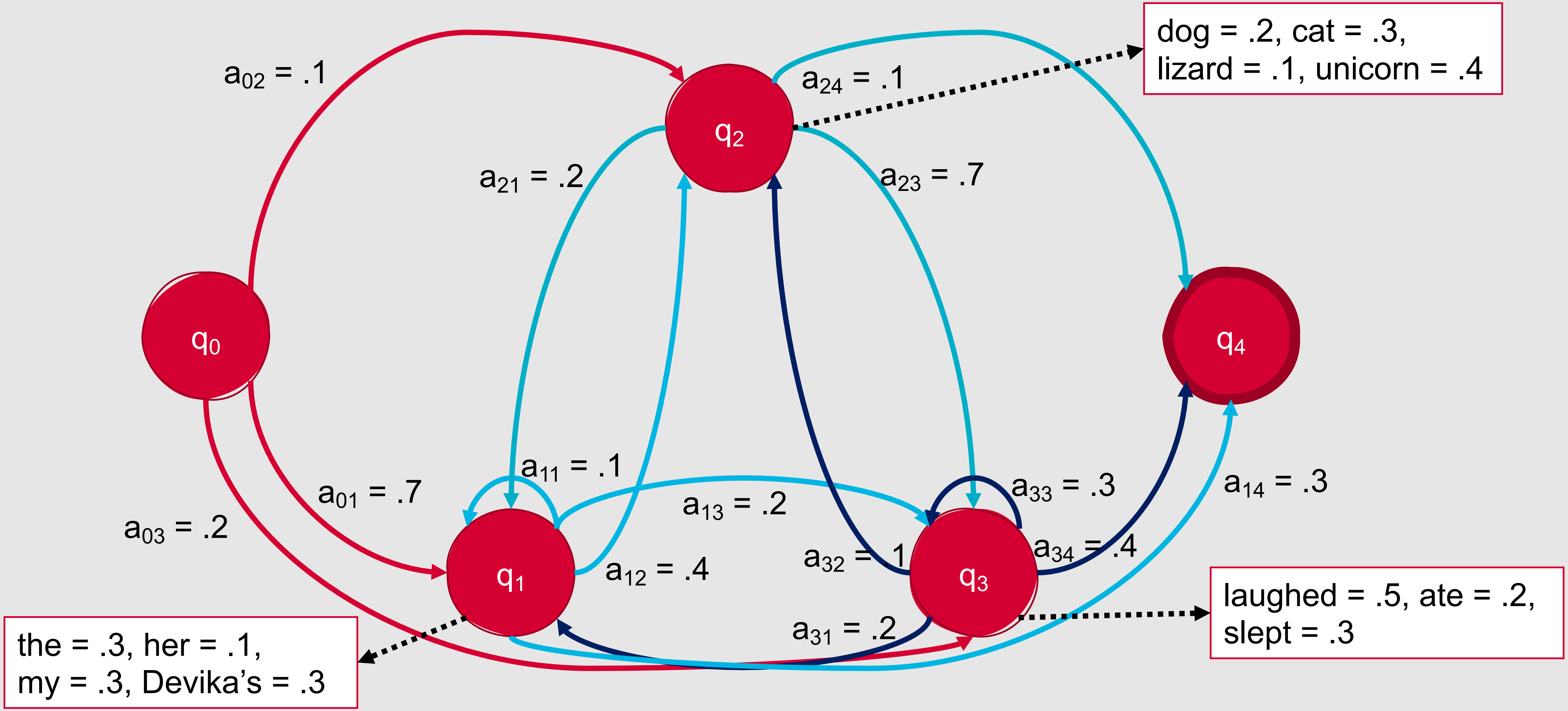
- More generally, you can use an HMM to generate a sequence of T observations: $O = o_1, o_2, \ldots, o_T$

Begin in the start state

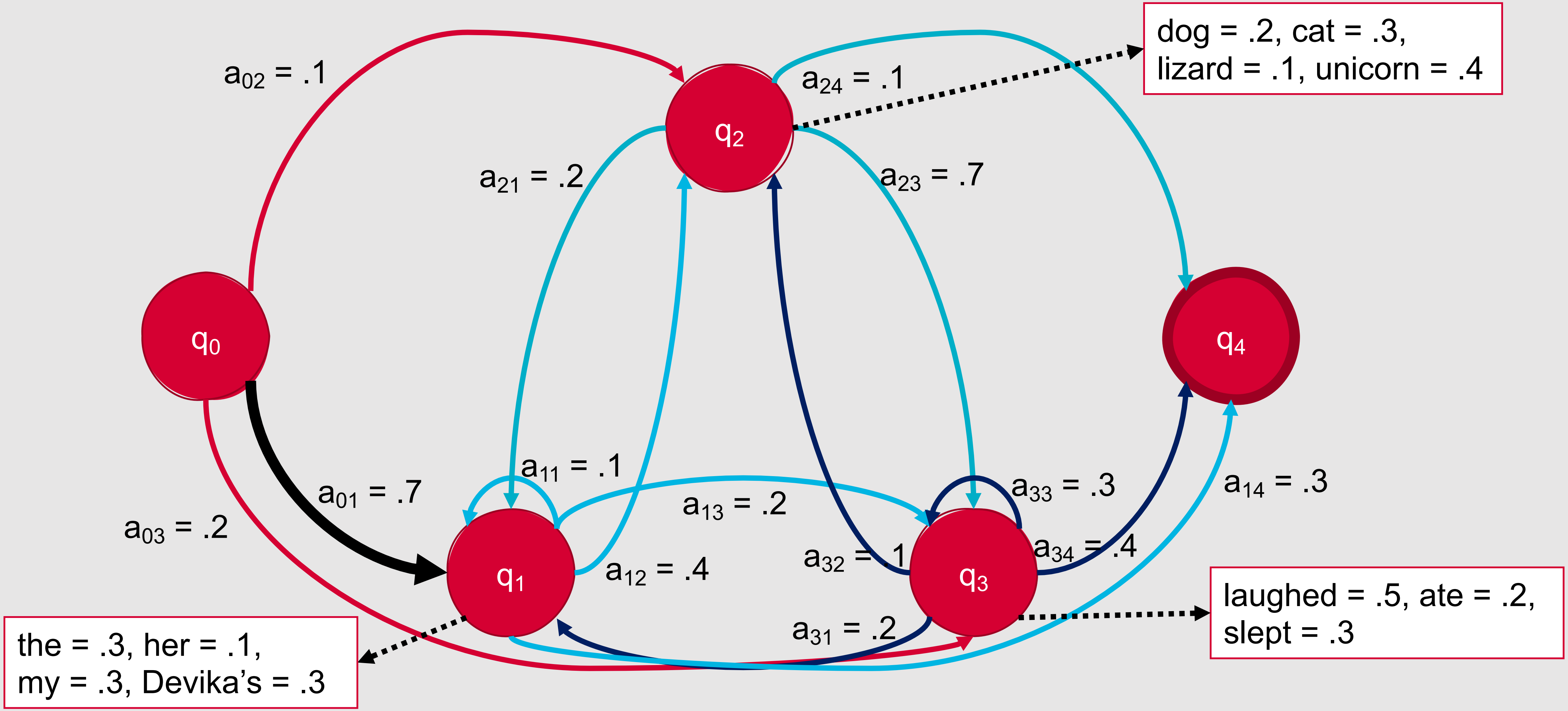
For t in [0, …, T]:

> *Randomly select a new state based on the transition distribution for the current state*
>
> *Randomly select an observation from the new state based on the observation distribution for that state*

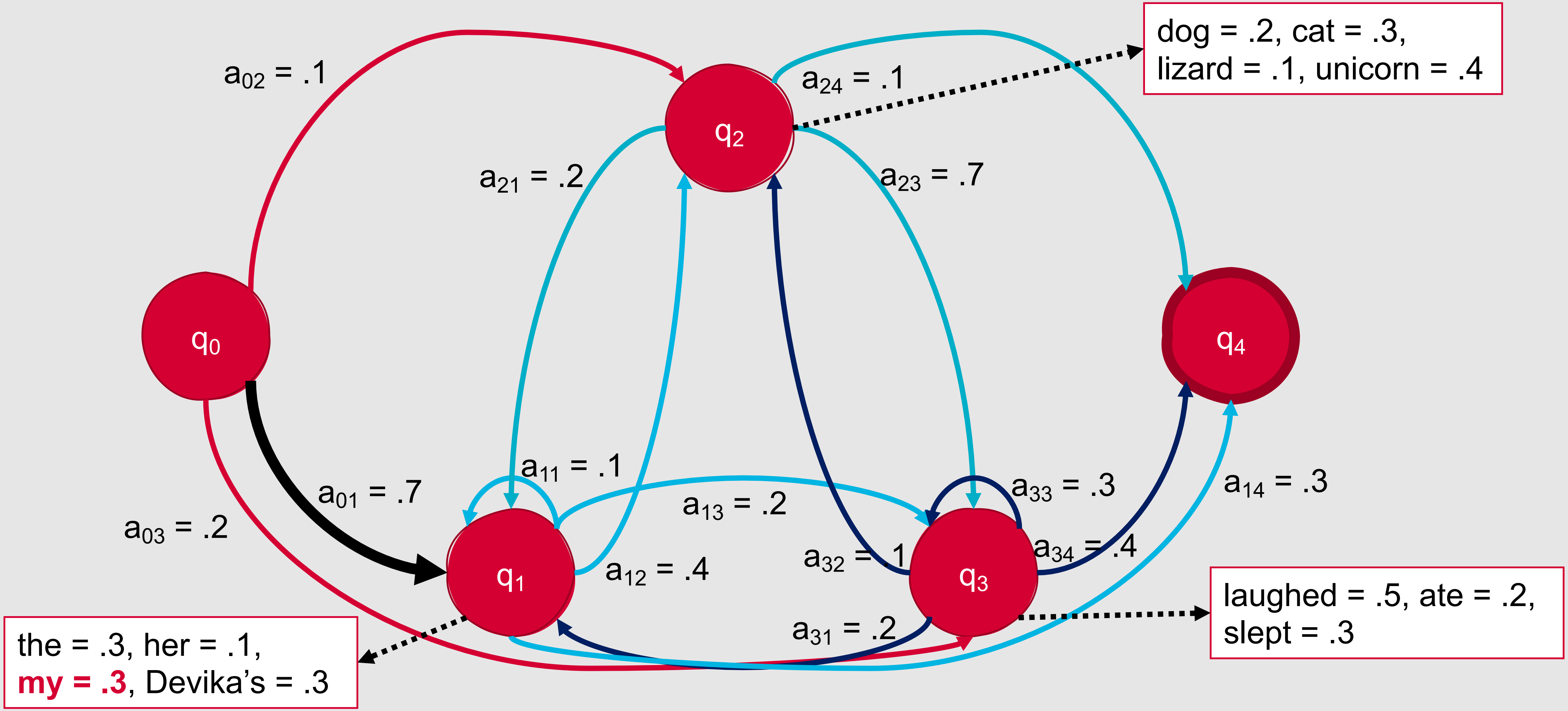
Sample Text Generation



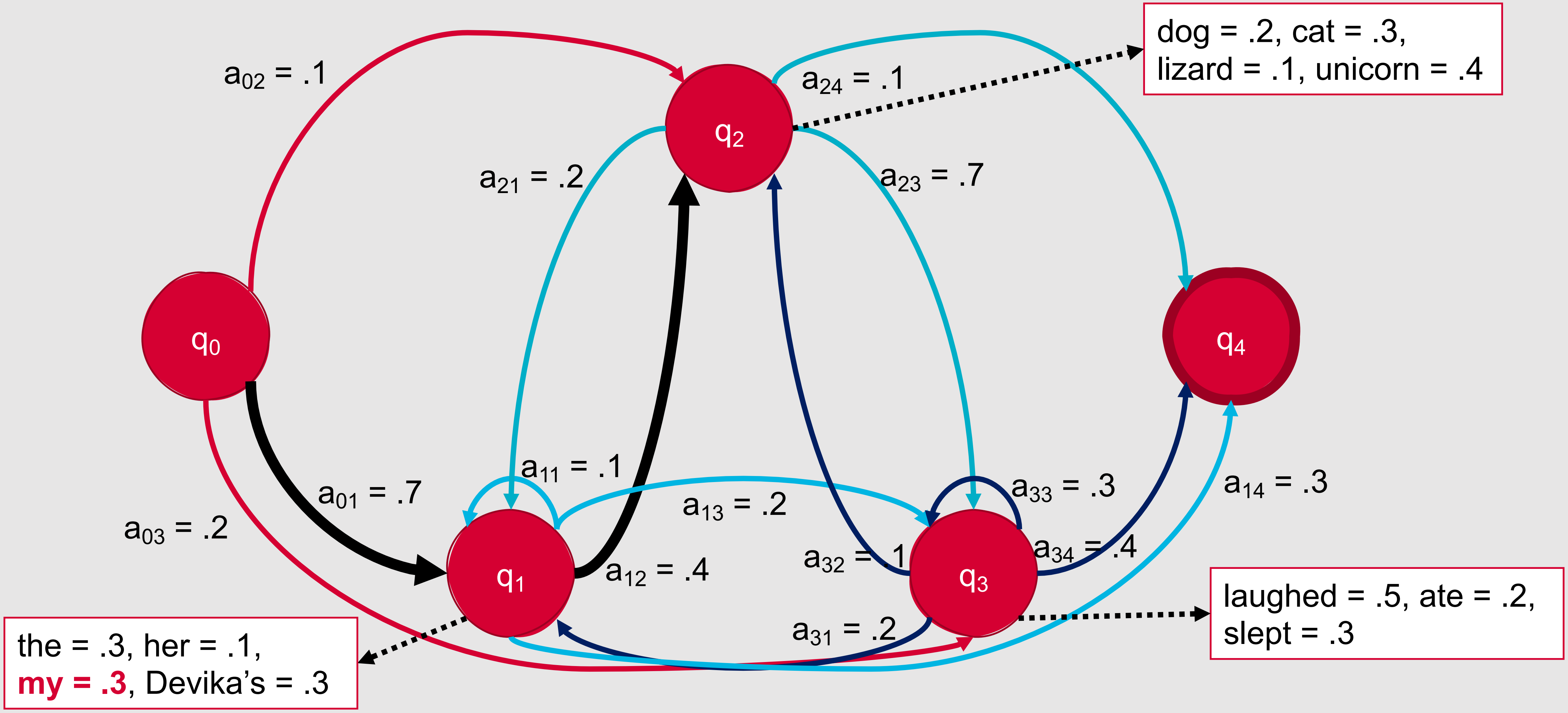
Sample Text Generation



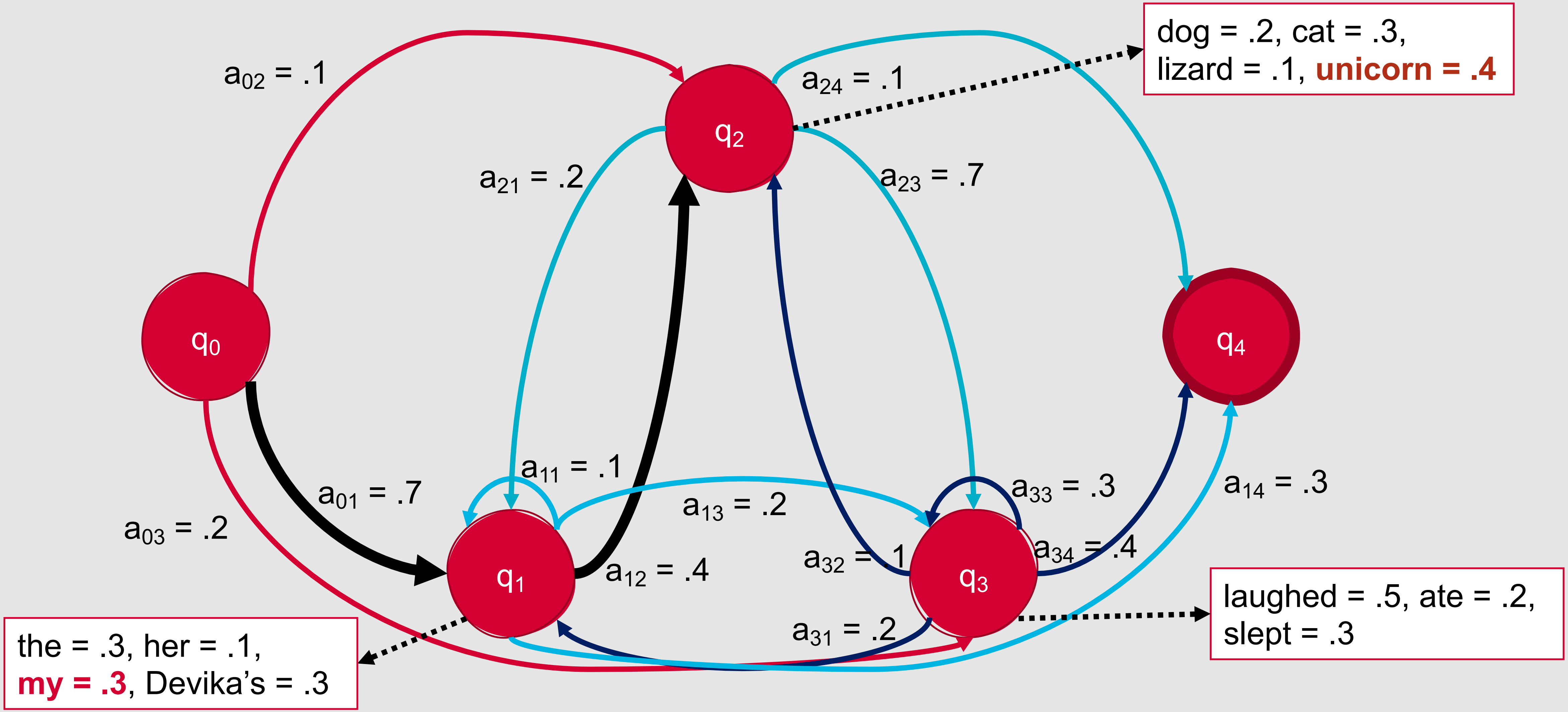
Sample Text Generation



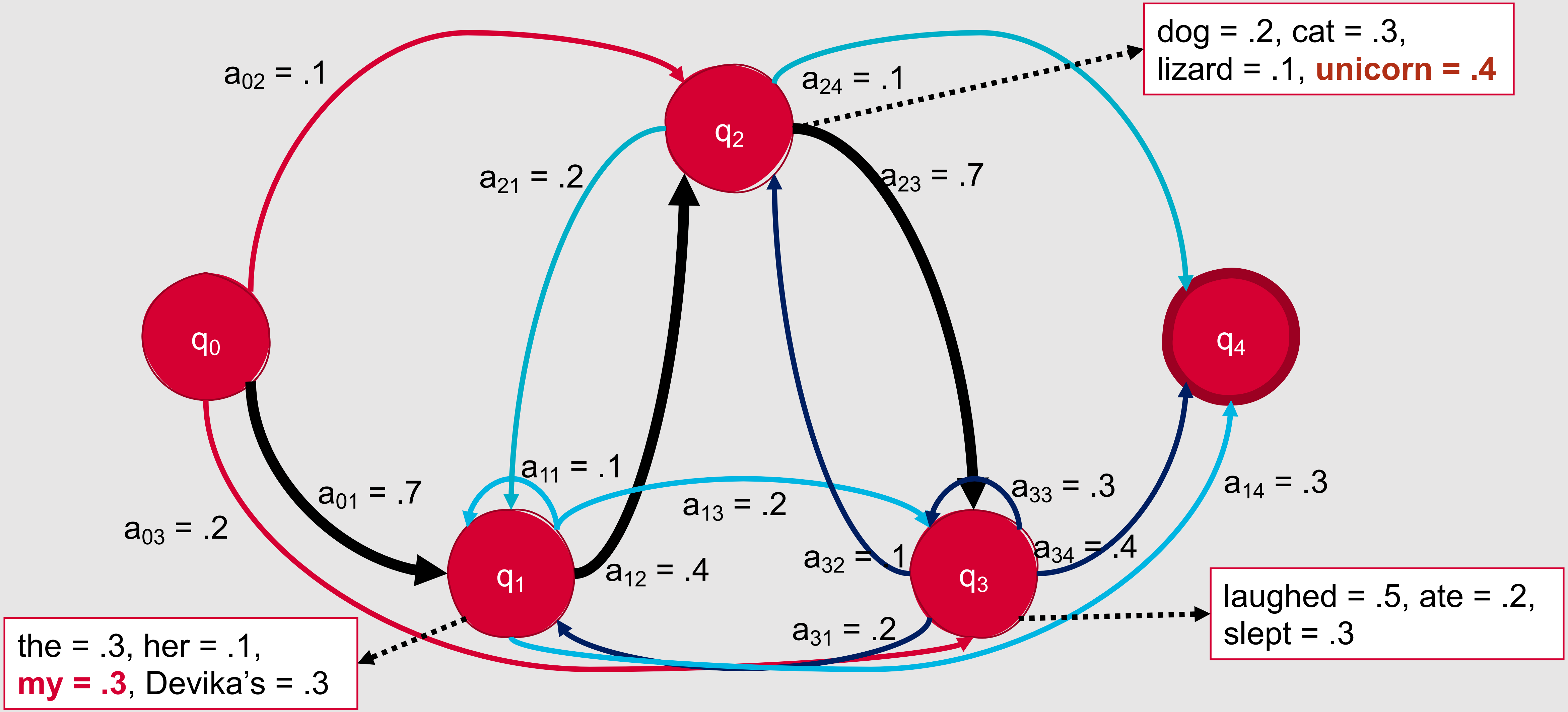
Sample Text Generation



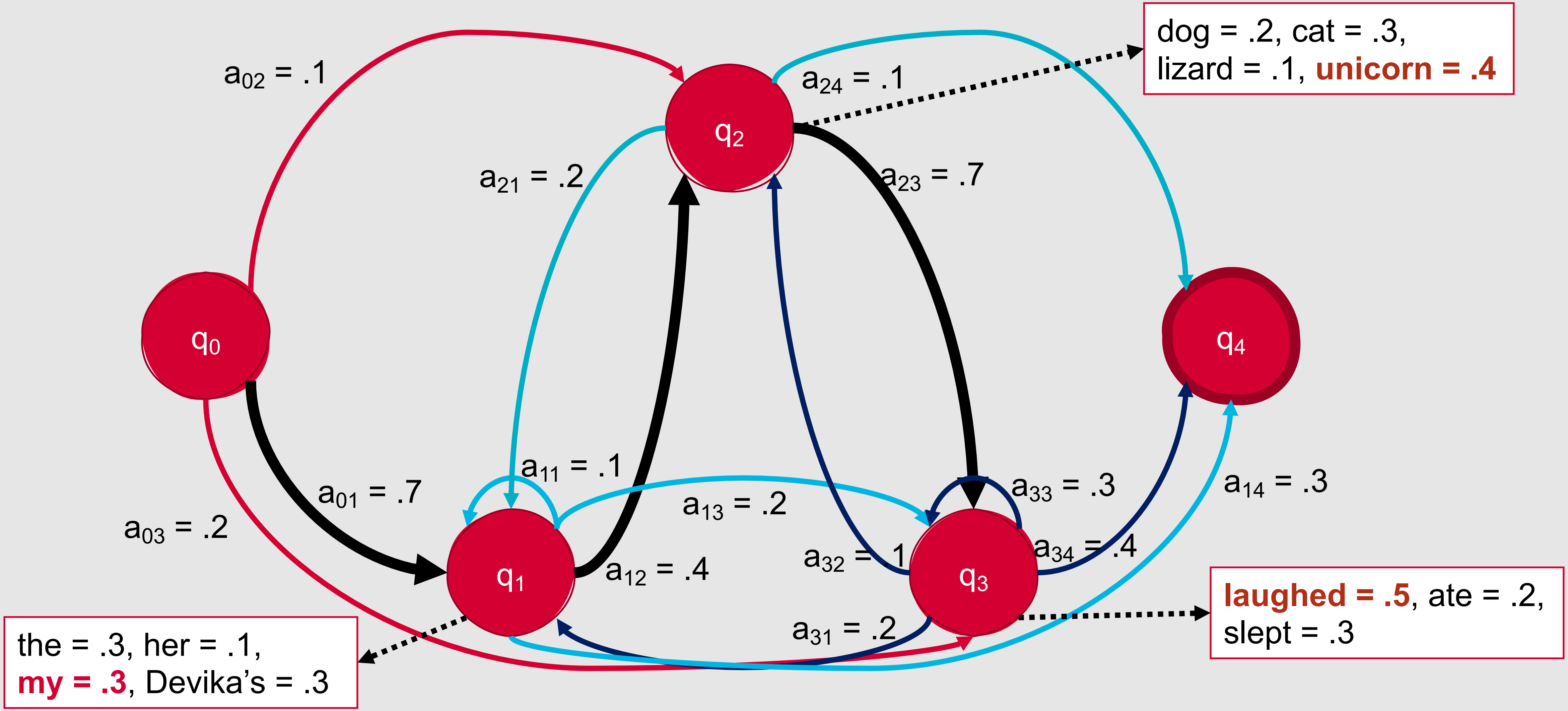
Sample Text Generation



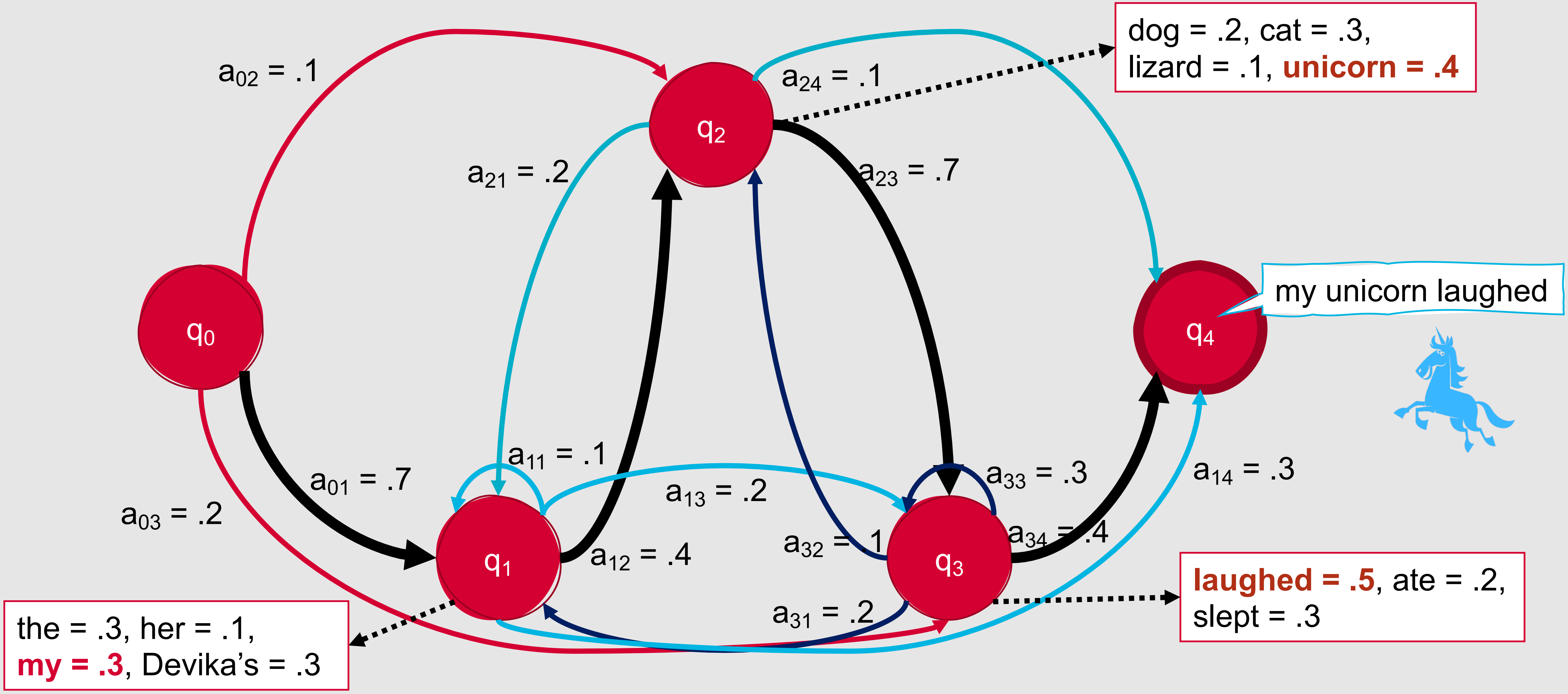
Sample Text Generation



Sample Text Generation



Sample Text Generation



Three Fundamental HMM Problems

- **Observation Likelihood:** How likely is a particular observation sequence to occur?
- **Decoding:** What is the best sequence of hidden states for an observed sequence?
  - What is the best sequence of labels for our test data?
- **Learning:** What are the transition probabilities and observation likelihoods that best fit the observation sequence and HMM states?
  - How do we empirically fit our training data?

This Week's Topics

Tuesday

- Hidden Markov Models
- Forward Algorithm
- Viterbi Algorithm
- Forward-Backward Algorithm

Thursday

- Parts of Speech
- POS Tagsets
- POS Tagging

Observation Likelihood

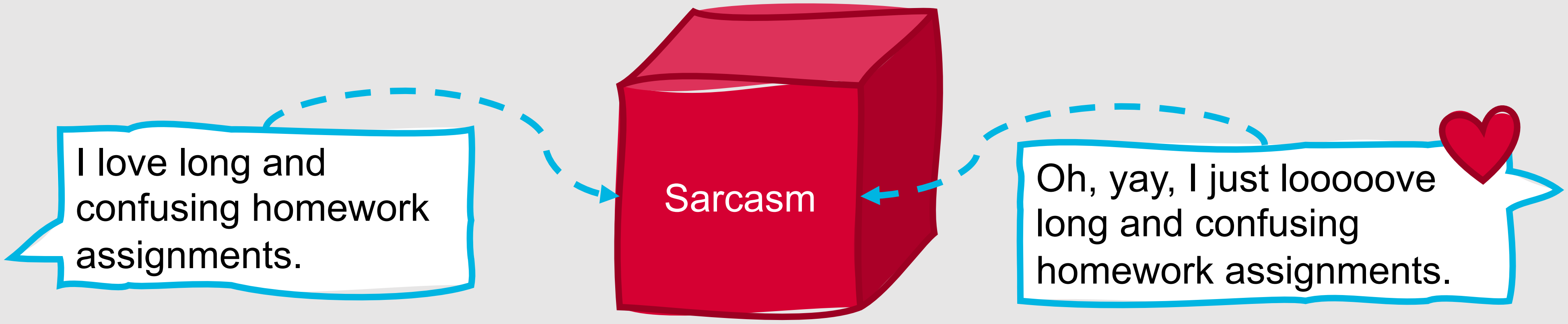
- Given a sequence of observations and an HMM, what is the probability that this sequence was generated by the model?
- Useful for two tasks:
  - Sequence classification
  - Selecting the most likely sequence

Sequence Classification

- Assuming an HMM is available for every possible class, what is the most likely class for a given observation sequence?
  - Which HMM is most likely to have generated the sequence?

Most Likely Sequence

- Of two or more possible sequences, which one was most likely generated by a given HMM?



How can we compute the observation likelihood?

- Naïve Solution:
  - Consider all possible state sequences, Q, of length T that the model, λ, could have traversed in generating the given observation sequence, O
  - Compute the probability of a given state sequence from A, and multiply it by the probability of generating the given observation sequence for that state sequence
    - $P(O,Q \mid \lambda) = P(O \mid Q, \lambda) * P(Q \mid \lambda)$
  - Repeat for all possible state sequences, and sum over all to get $P(O \mid \lambda)$
- But, this is computationally complex!
  - $O(TN^T)$

How can we compute the observation likelihood?

- Efficient Solution:
  - **Forward Algorithm:** Dynamic programming algorithm that computes the observation probability by summing over the probabilities of all possible hidden state paths that could generate the observation sequence.
  - Implicitly folds each of these paths into a single **forward trellis**
- Why does this work?
  - Markov assumption (the probability of being in any state at a given time *t* only relies on the probability of being in each possible state at time *t*-1)
- Works in $O(TN^2)$ time!

How does the forward algorithm work?

- Let $\alpha_i(j)$ be the probability of being in state *j* after seeing the first *t* observations, given your HMM λ
- $\alpha_i(j)$ is computed by summing over the probabilities of every path that could lead you to this cell
  - $\alpha_i(j) = P(o_1, o_2 \ldots o_t, q_t = j|\lambda) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t)$
  - $\alpha_{t-1}(i)$: The previous forward path probability from the previous time step
  - a_{ij}: The transition probability from previous state q_i to current state q_j
  - $b_j(o_t)$: The state observation likelihood of the observed item o_t given the current state *j*

Formal Algorithm

```
create a probability matrix forward[N+2,T]

for each state q in [1, …, N] do:
     forward[q,1] ← a0,q * bq(o1)
for each time step t from 2 to T do:
     for each state q in [1, …, N] do:
          forward[q,t] ←∑_{q'=1}^{N} forward[q',t−1] ∗ a_{q',q} ∗ b_q(o_t)
forwardprob ←∑_{q=1}^{N} forward[q,T]
```

$\text{forward}[q,1] \leftarrow a_{0,q} * b_q(o_1)$

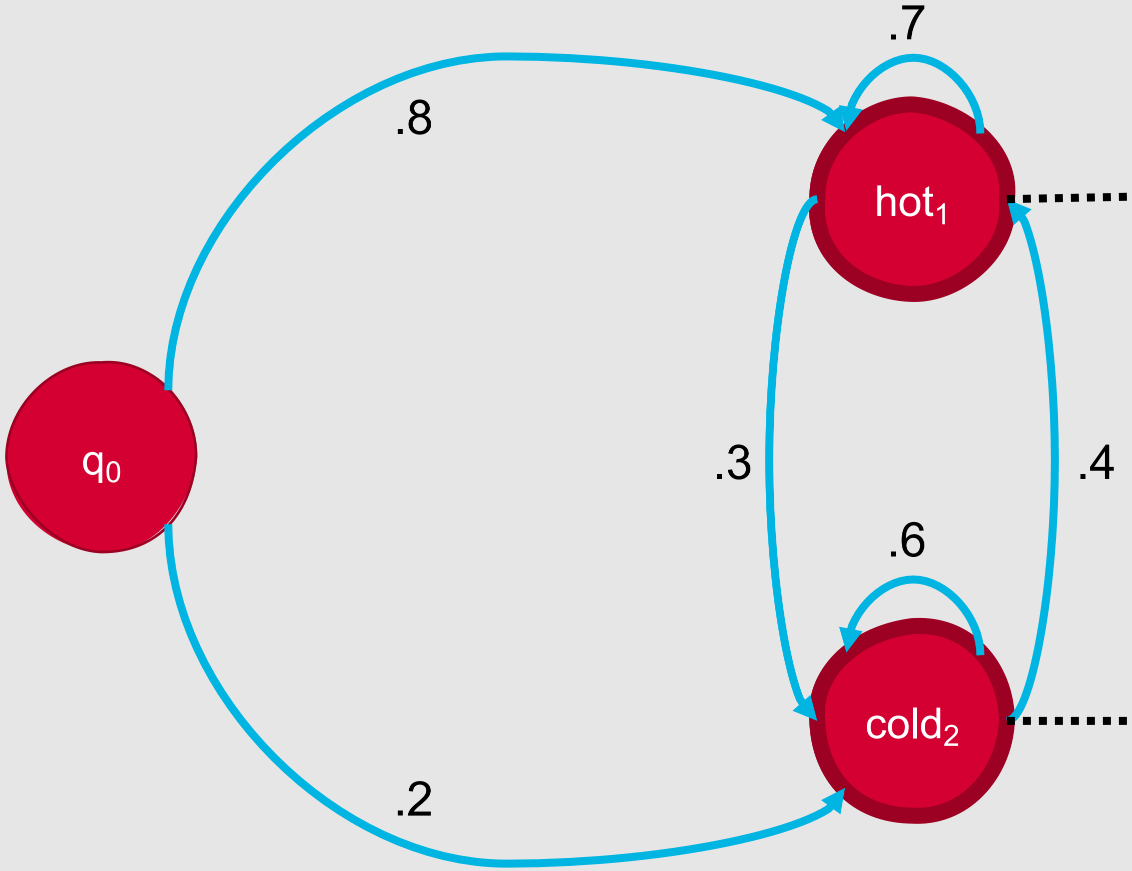
$\text{forward}[q,t] \leftarrow \sum_{q'=1}^{N} forward[q',t-1] * a_{q',q} * b_q(o_t)$

$\text{forwardprob} \leftarrow \sum_{q=1}^{N} forward[q,T]$

Sample Problem

- You’re trying to solve a problem that relies on you knowing which days it was hot and cold in Chicago during the summer of 1923
- Unfortunately, you have no official records of the weather in Chicago for that summer, although you’re trying to model some key weather patterns from that year using an HMM
- You do have one promising lead: You find a detailed diary tracking how many ice cream cones the author of that diary ate on each day
- You decide to focus on a three-day sequence:
  - Day 1: 3 ice cream cones
  - Day 2: 1 ice cream cone
  - Day 3: 3 ice cream cones
- Your first task is to determine whether this HMM does a good job at modeling your sequence

Your HMM



B1

$$\begin{bmatrix} P(1|hot_1) \\ P(2|hot_1) \\ P(3|hot_1) \end{bmatrix} = \begin{bmatrix} .2 \\ .4 \\ .4 \end{bmatrix}$$

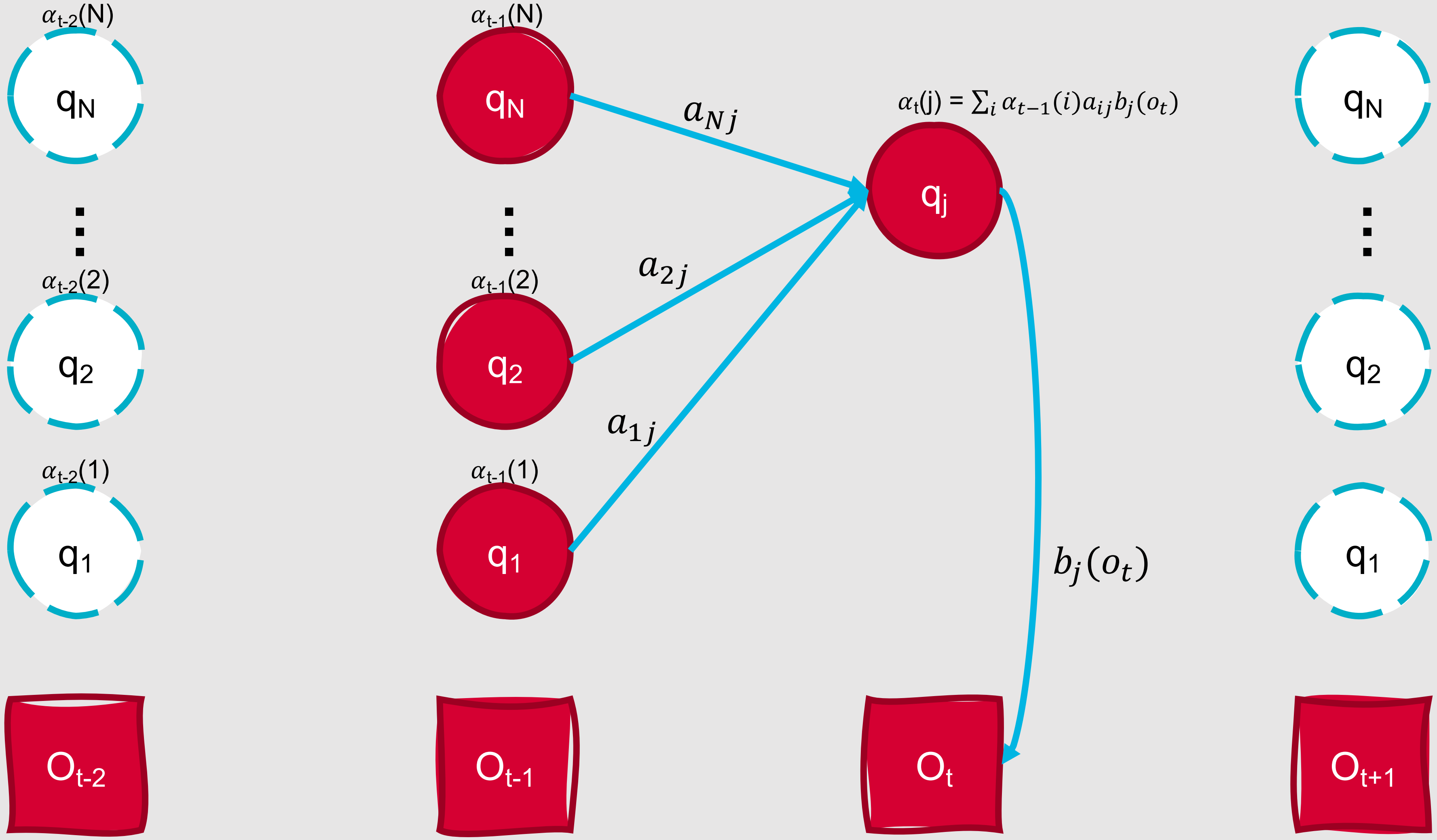
B2

$$\begin{bmatrix} P(1|cold_1) \\ P(2|cold_1) \\ P(3|cold_1) \end{bmatrix} = \begin{bmatrix} .5 \\ .4 \\ .1 \end{bmatrix}$$

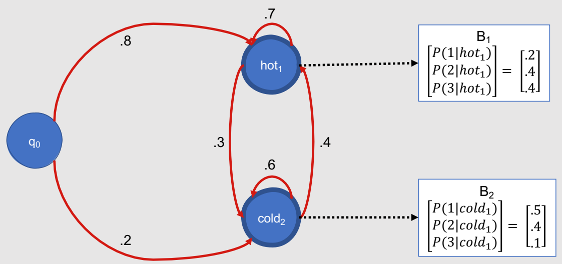
Forward Trellis

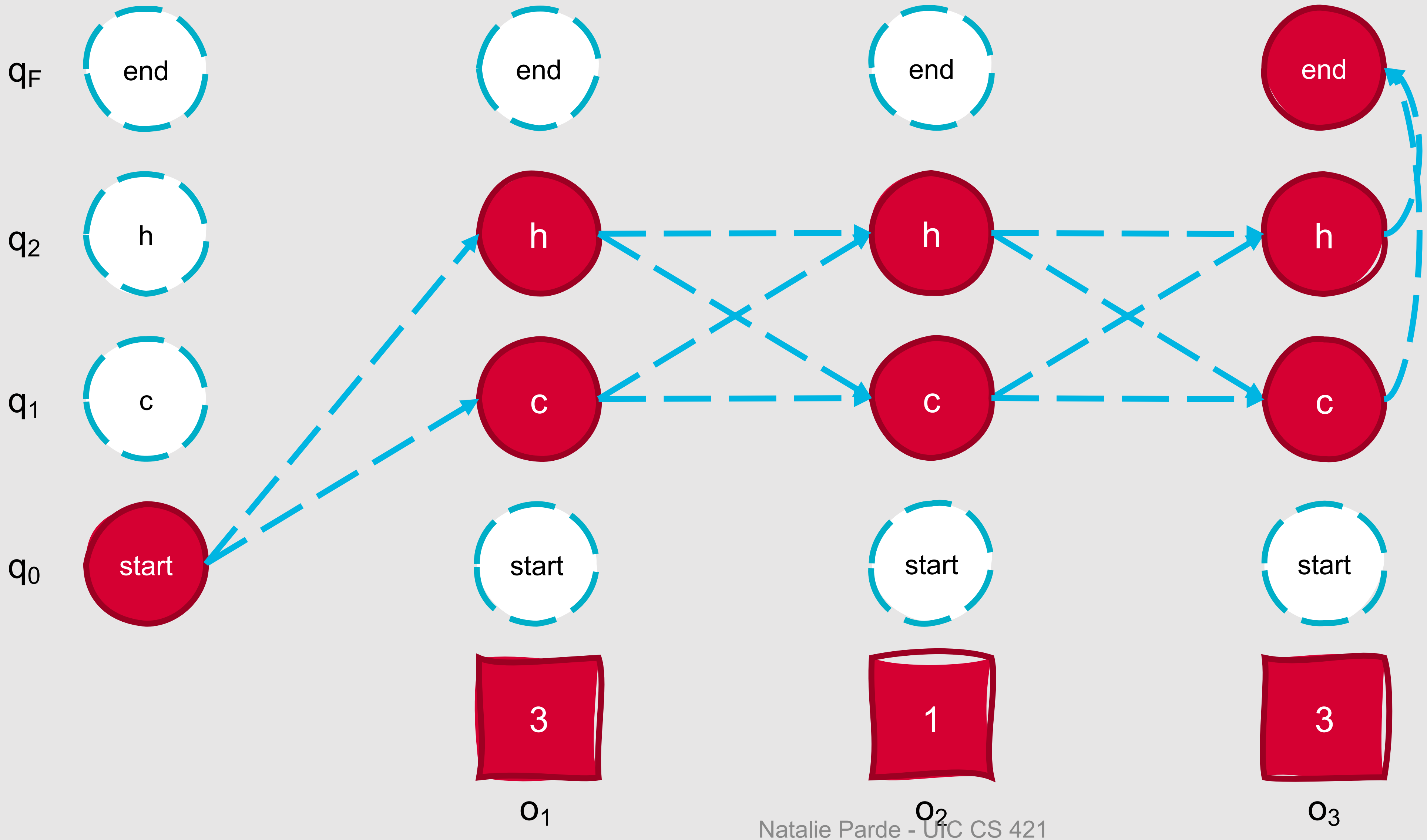
- Incorporates all the information you'll need to implement the forward algorithm
  - Observations
  - Transition probabilities
  - State observation likelihoods
  - Forward probabilities from earlier observations

Forward Step

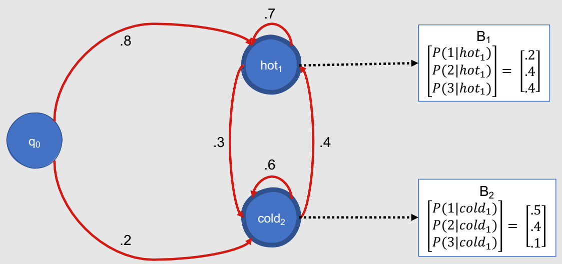


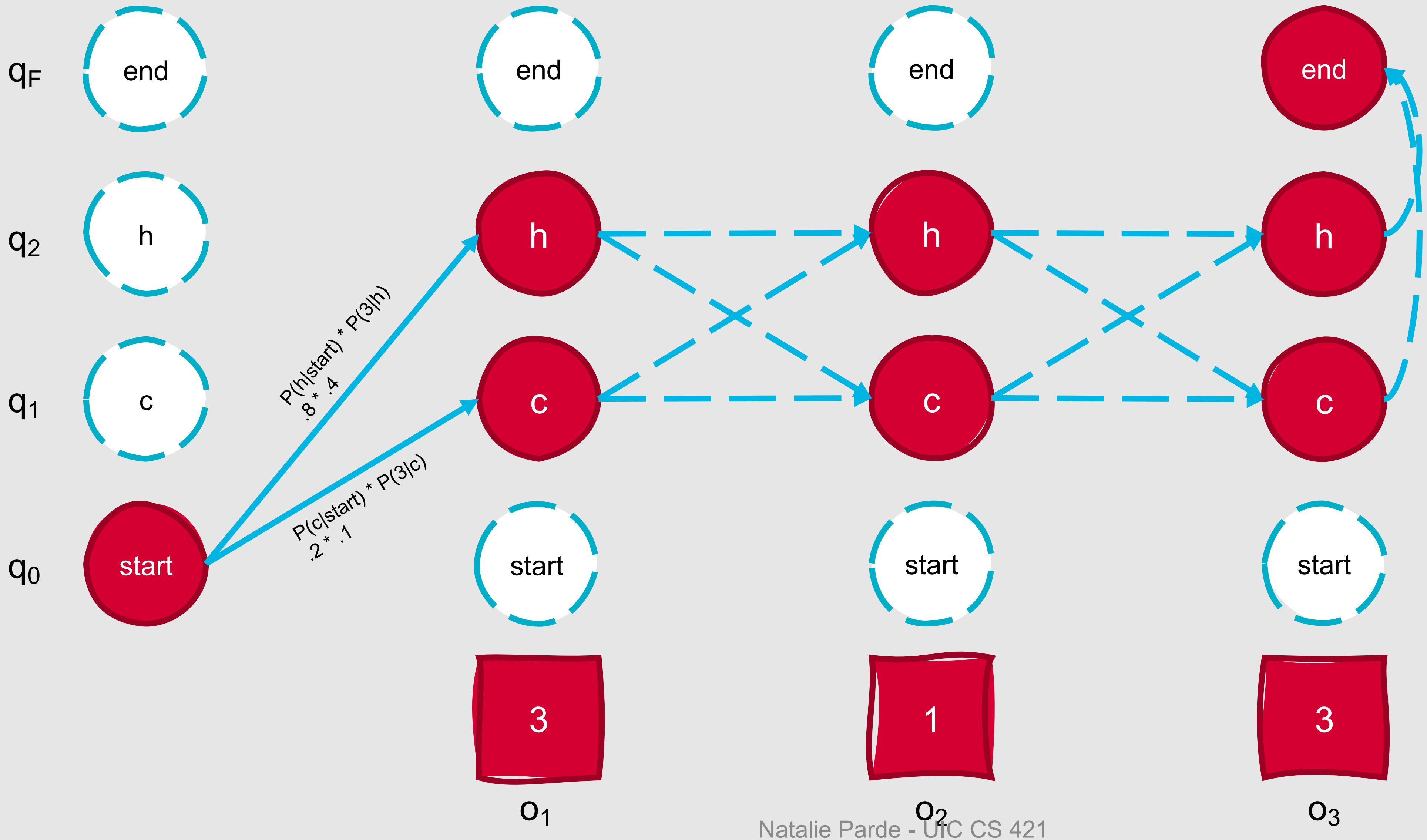
Forward Trellis



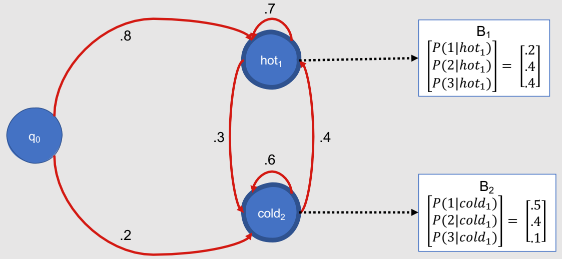


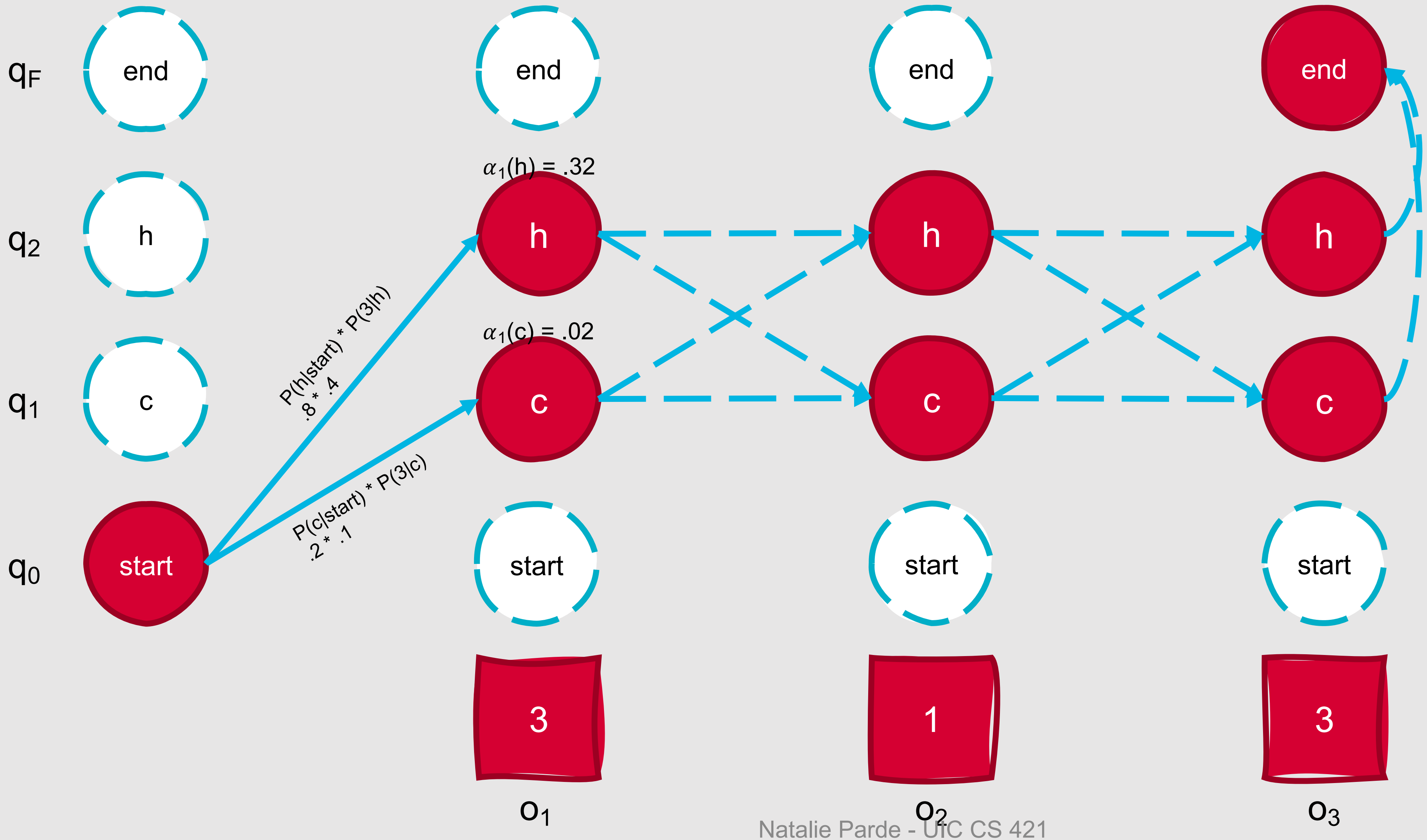
Forward Trellis



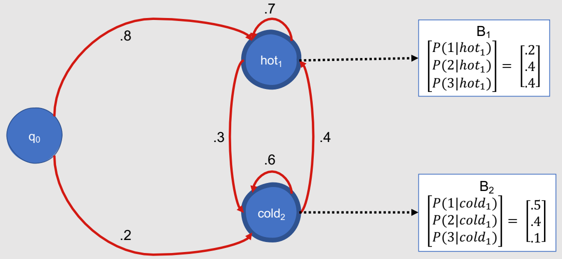


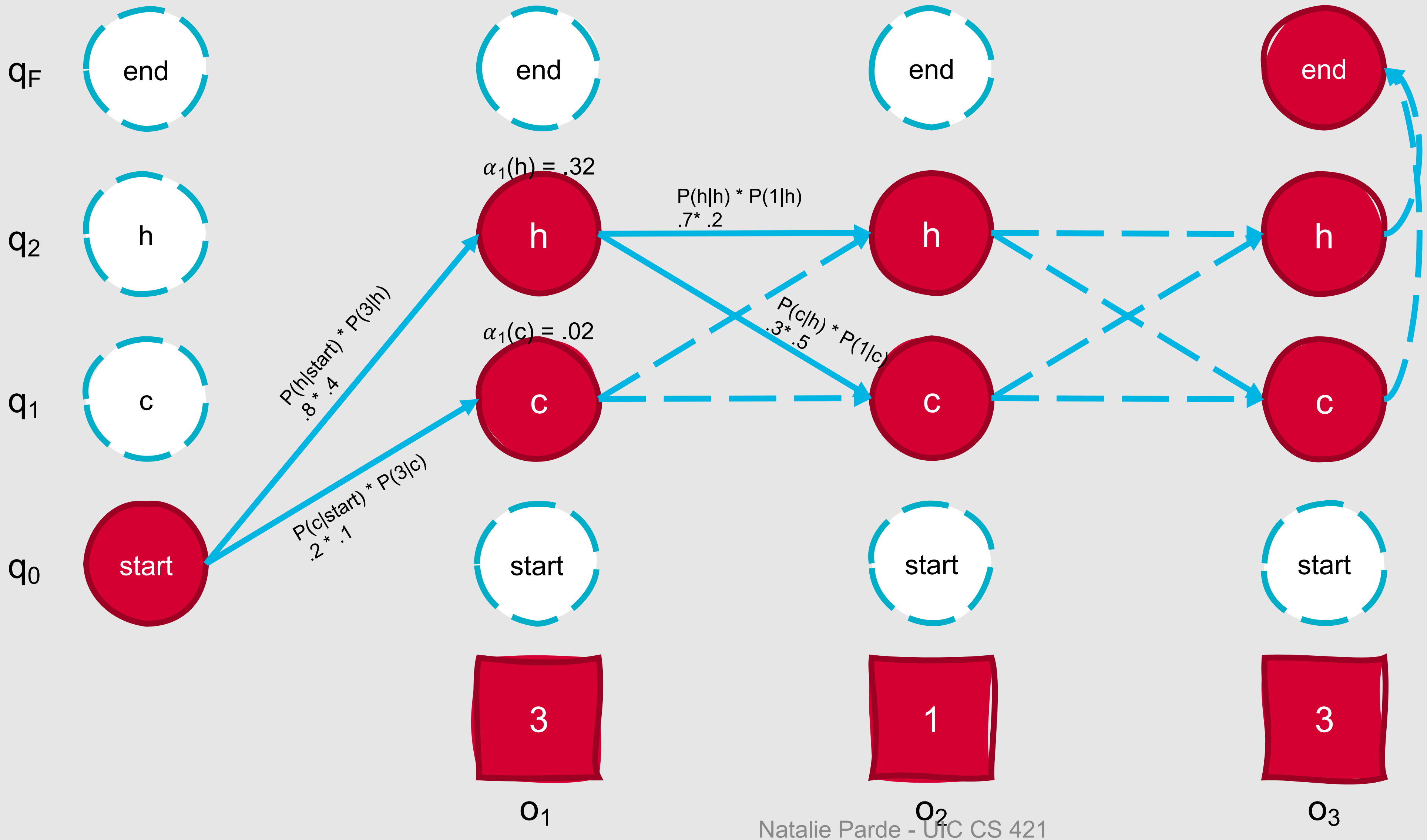
Forward Trellis



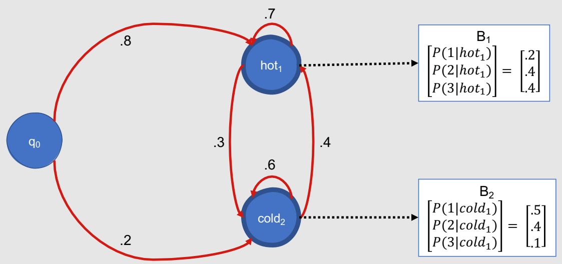


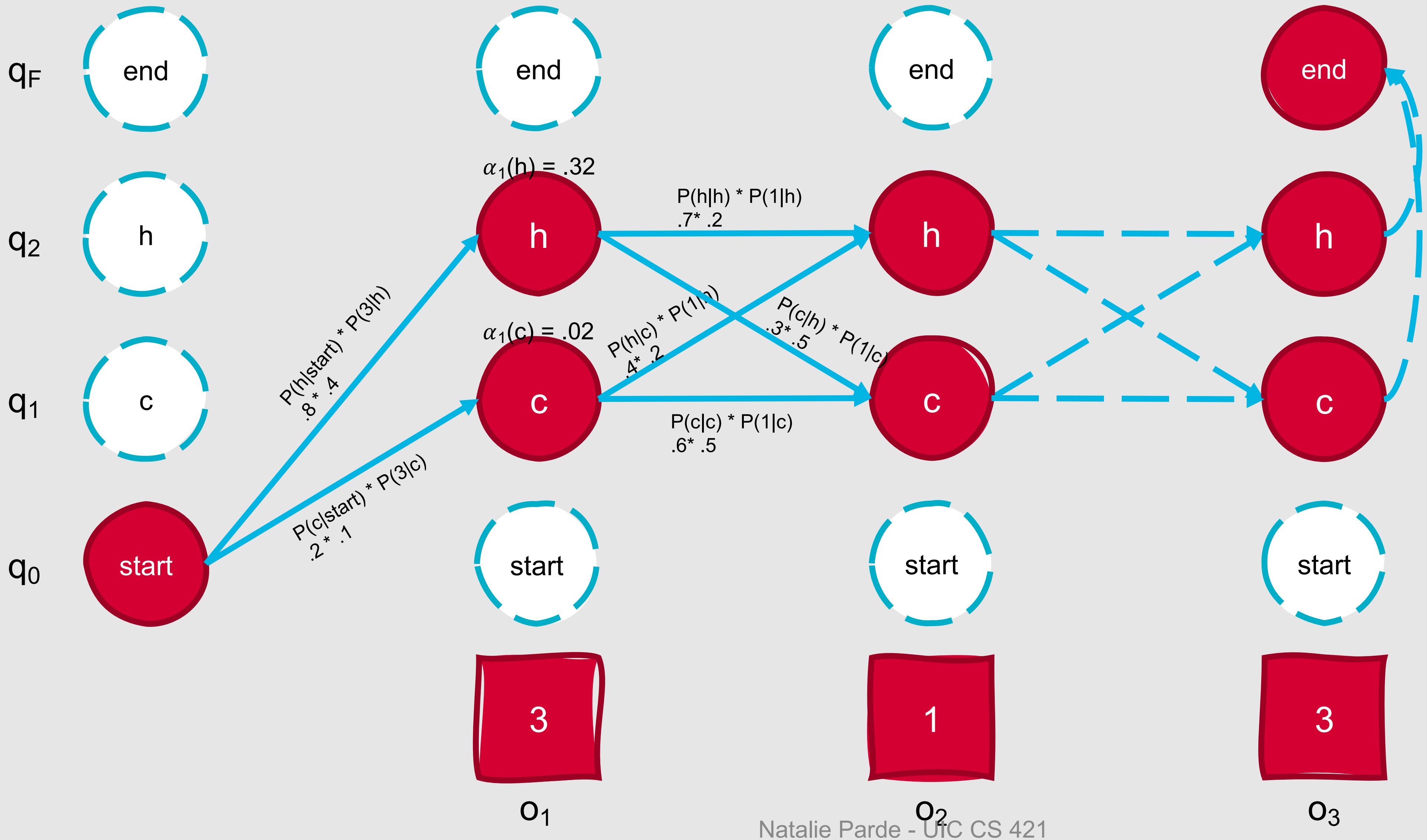
Forward Trellis



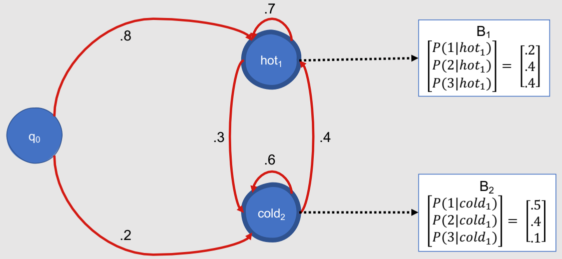


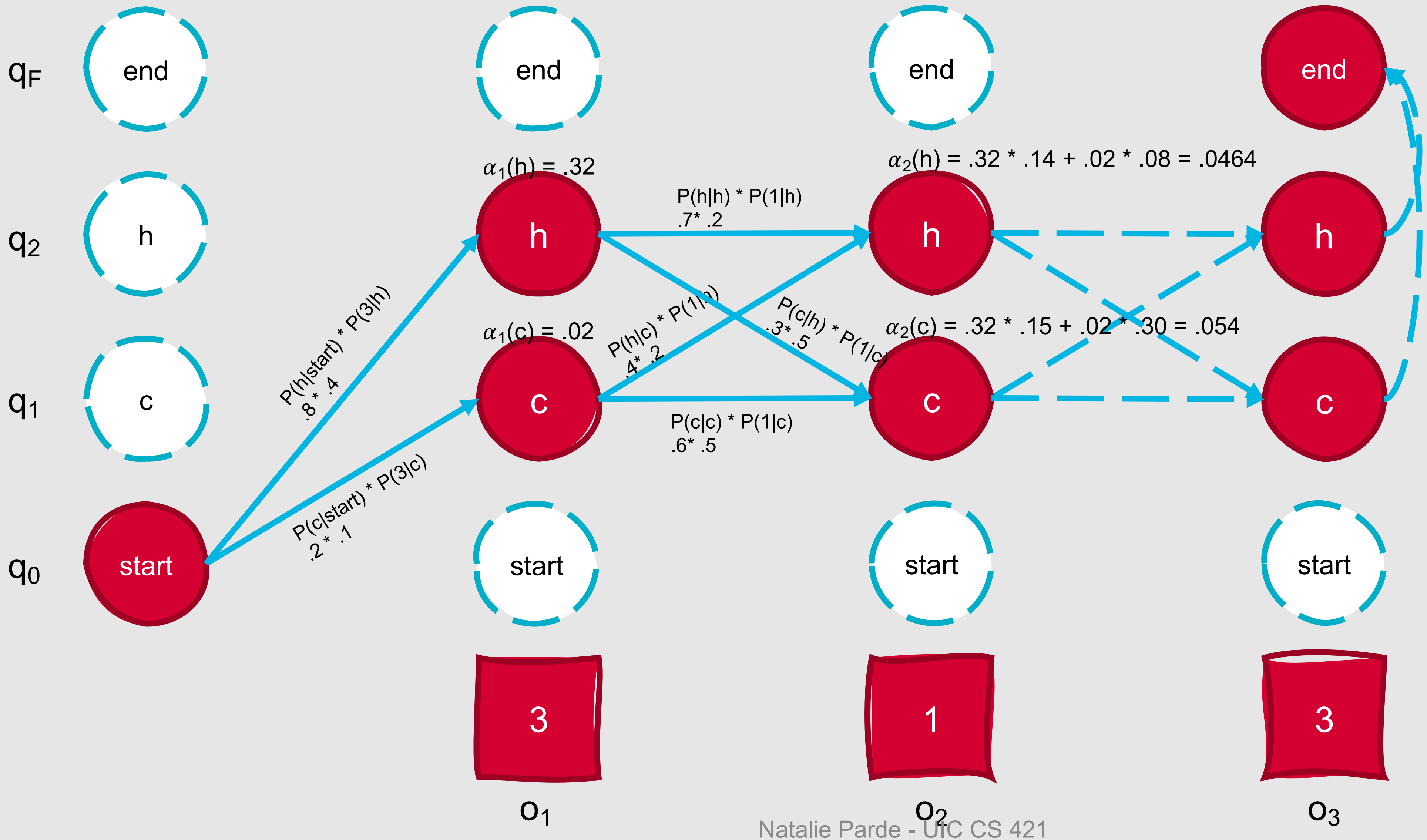
Forward Trellis



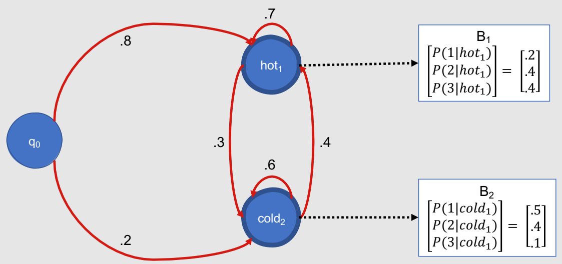


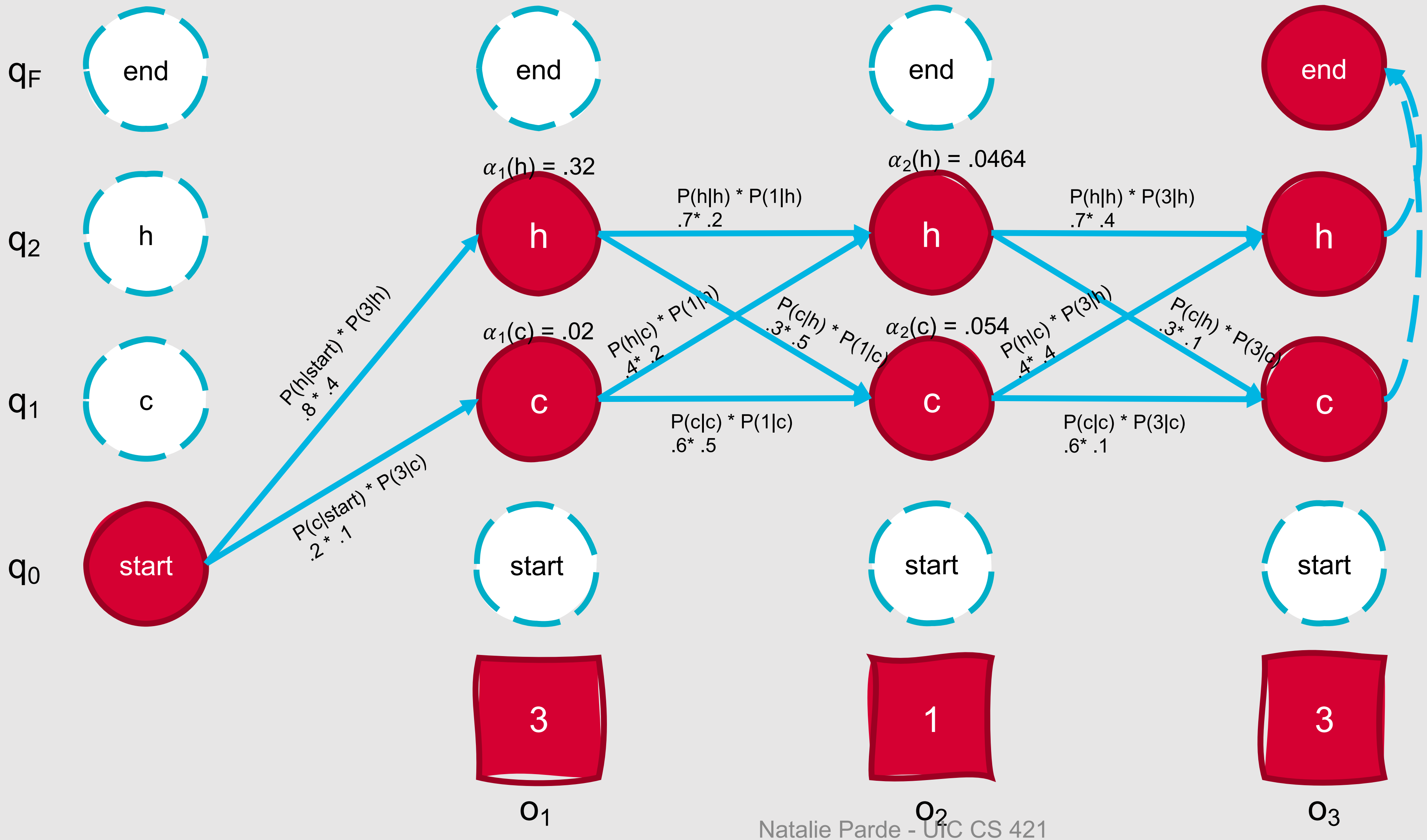
Forward Trellis



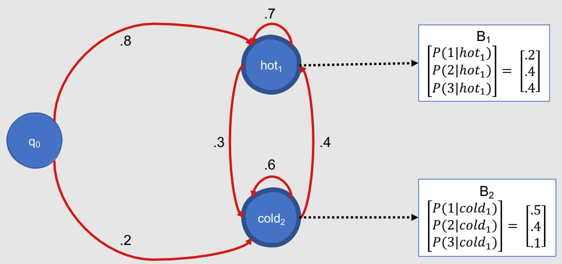


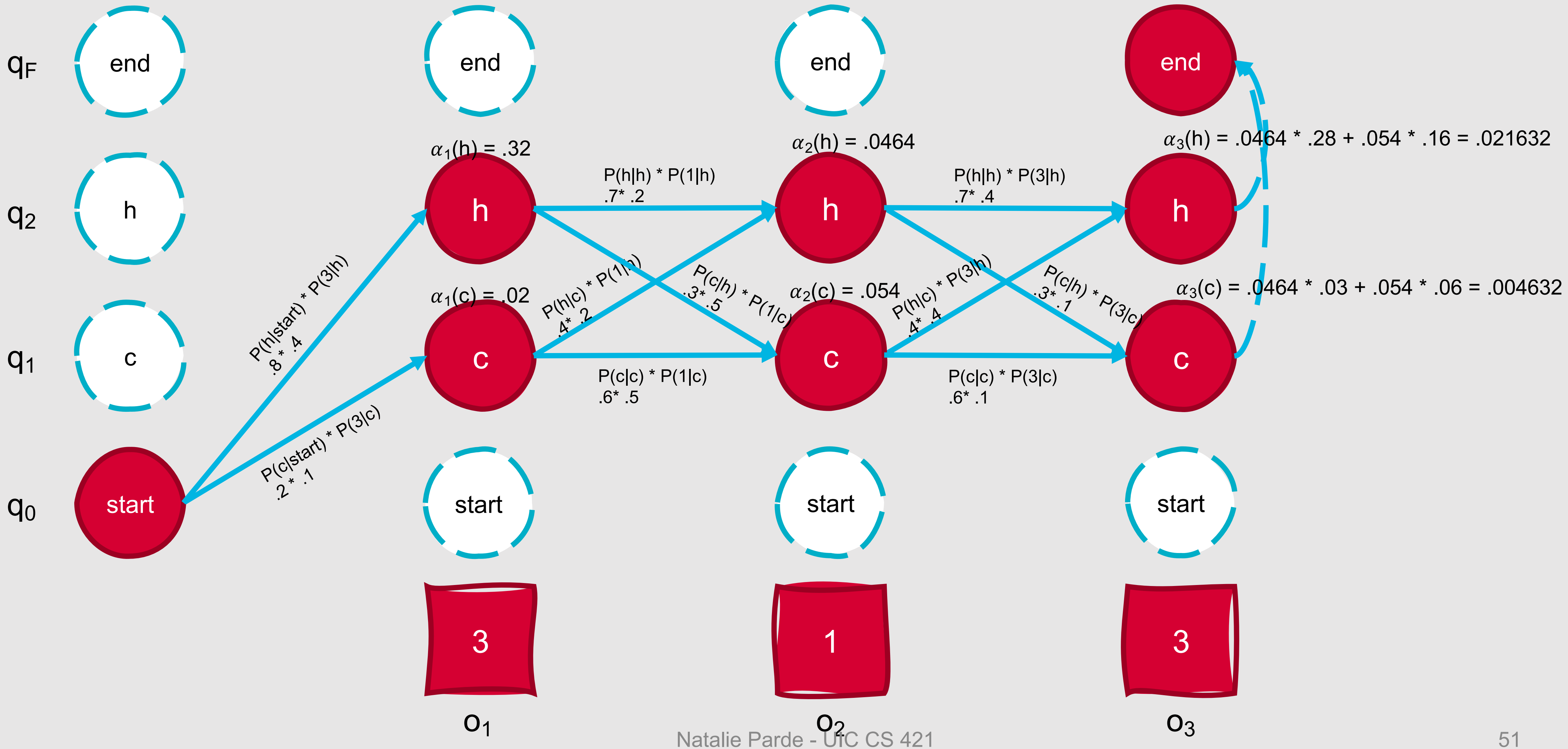
Forward Trellis



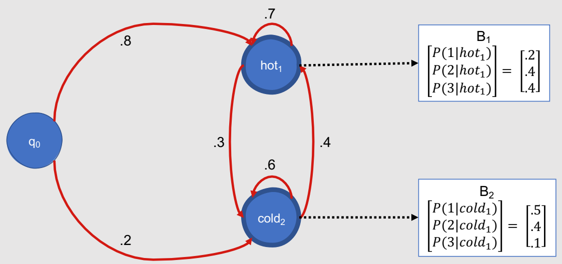


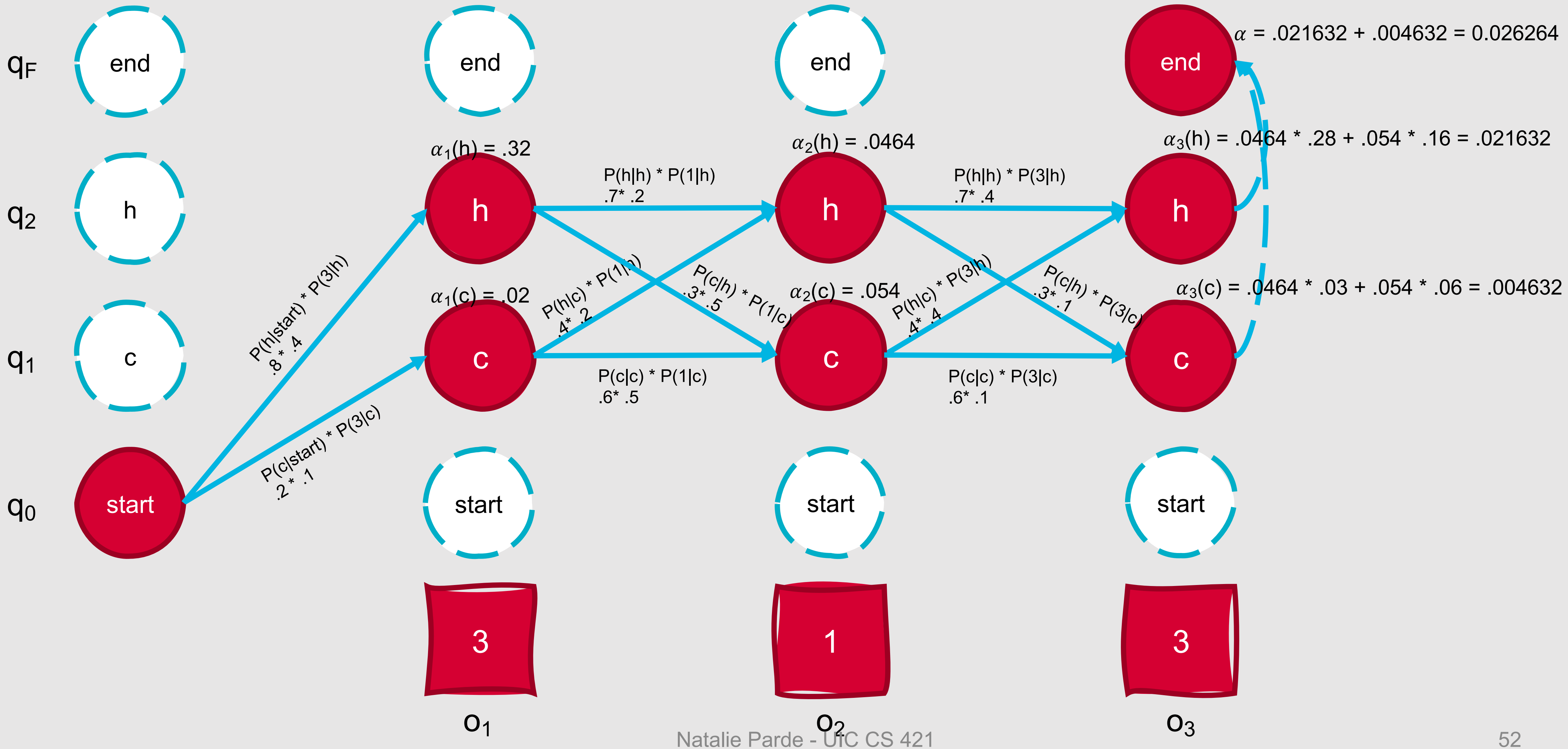
Forward Trellis





Forward Trellis





We’ve so far tackled one of the fundamental HMM tasks.

- What is the probability that a sequence of observations fits a given HMM?
  - Calculate using forward probabilities!
- However, there are still two remaining tasks to explore….

This Week's Topics

Hidden Markov Models
Forward Algorithm
Viterbi Algorithm
Forward-Backward Algorithm

Thursday

Tuesday

Parts of Speech
POS Tagsets
POS Tagging

Decoding

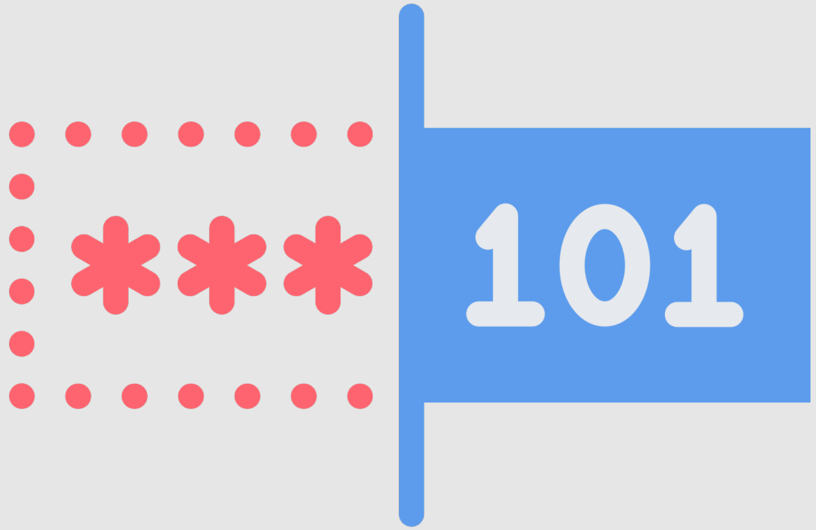
- Given an observation sequence and an HMM, what is the best hidden state sequence?
  - How do we choose a state sequence that is optimal in some sense (e.g., best explains the observations)?
- Very useful for sequence labeling!

Decoding

- Naïve Approach:
  - For each hidden state sequence Q, compute P(O|Q)
  - Pick the sequence with the highest probability
- However, this is computationally inefficient!
  - $O(N^T)$

How can we decode sequences more efficiently?

- **Viterbi Algorithm**
  - Another dynamic programming algorithm
  - Uses a similar trellis to the Forward algorithm
- Viterbi time complexity: $O(N^2T)$



Viterbi Intuition

- **Goal:** Compute the joint probability of the observation sequence together with the best state sequence
- So, **recursively compute the probability of the most likely subsequence of states** that accounts for the first *t* observations and ends in state q_j.
  - $v_t(j) = \max_{q_0, q_1, \ldots, q_{t-1}} P(q_0, q_1, \ldots, q_{t-1}, o_1, \ldots, o_t, q_t = q_j | \lambda)$
- Also **record backpointers** that subsequently allow you to backtrace the most probable state sequence
  - $bt_t(j)$ stores the state at time *t*-1 that maximizes the probability that the system was in state q_j at time *t*, given the observed sequence

Formal Algorithm

```
create a path probability matrix Viterbi[N+2,T]

for each state q in [1,…,N] do:
      Viterbi[q,1] ← a_{0,q} * b_q(o_1)
      backpointer[q,1] ← 0
for each time step t in [2,…,T] do:
      for each state q in [1,…,N] do:
```

$$viterbi[q,t] \leftarrow \max_{q' \in [1,\ldots,N]} viterbi[q',t-1] * a_{q',q} * b_q(o_t)$$

$$backpointer[q,t] \leftarrow \operatorname{argmax}_{q' \in [1,\ldots,N]} viterbi[q',t-1] * a_{q',q} * b_q(o_t)$$

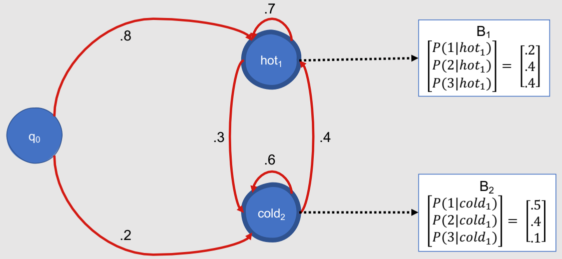
bestpathprob $\leftarrow \max_{q' \in [1,\ldots,N]} viterbi[q',T]$

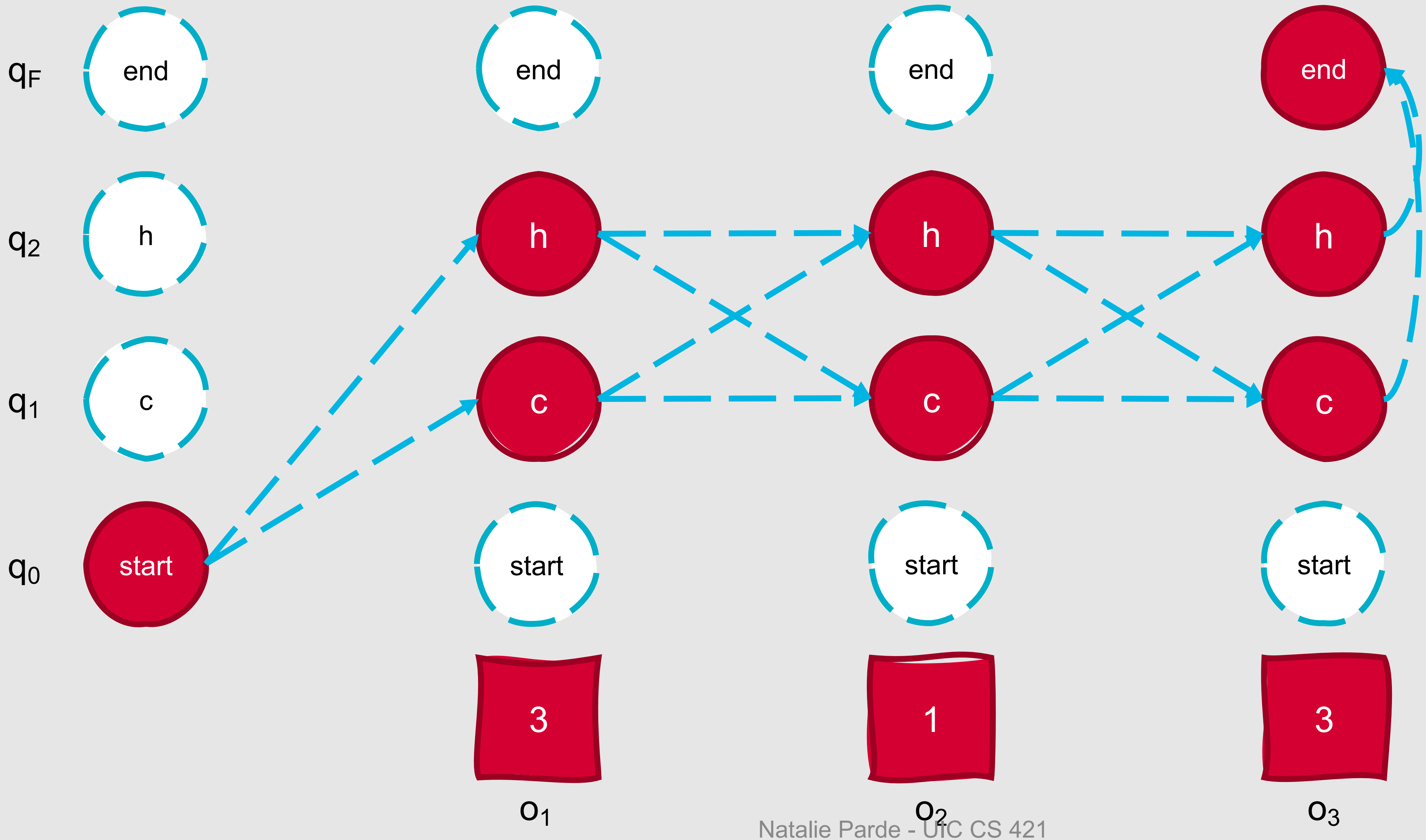
bestpathpointer $\leftarrow \operatorname{argmax}_{q' \in [1,\ldots,N]} viterbi[q',T]$

Seem familiar?

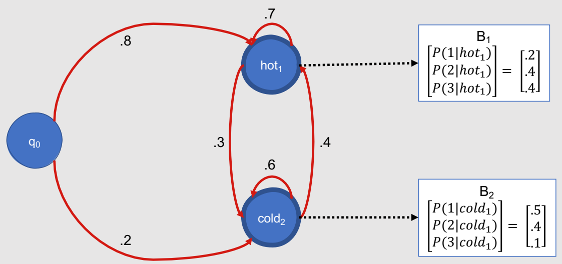
- Viterbi is basically the forward algorithm + backpointers!
- Instead of summing across prior forward probabilities, we use a max function

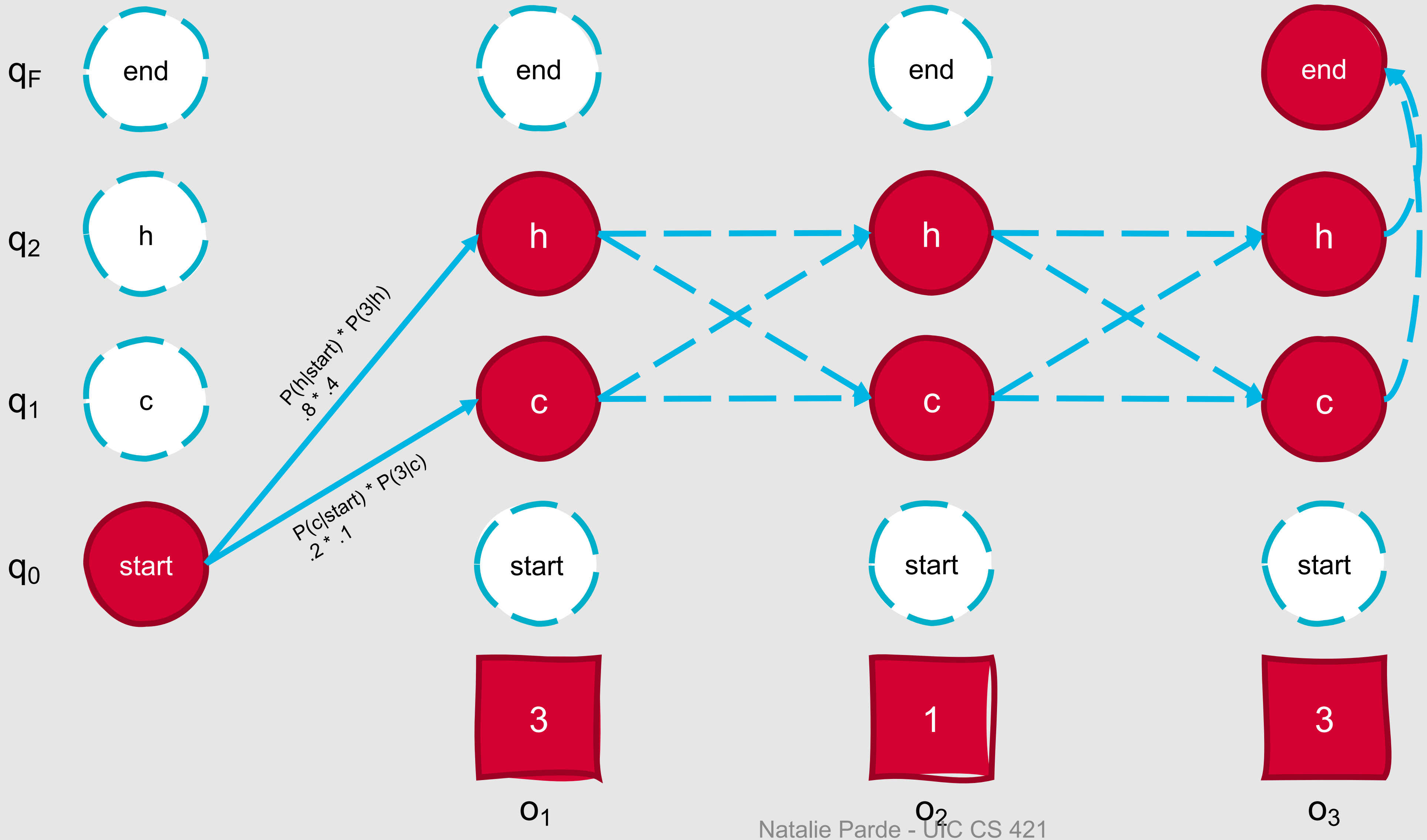
Viterbi Trellis



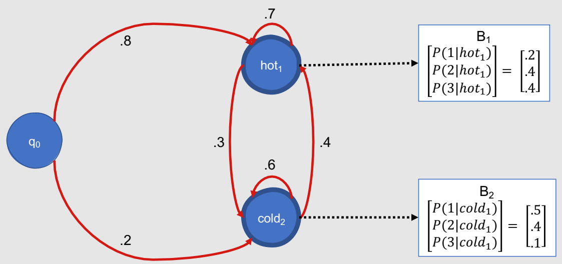


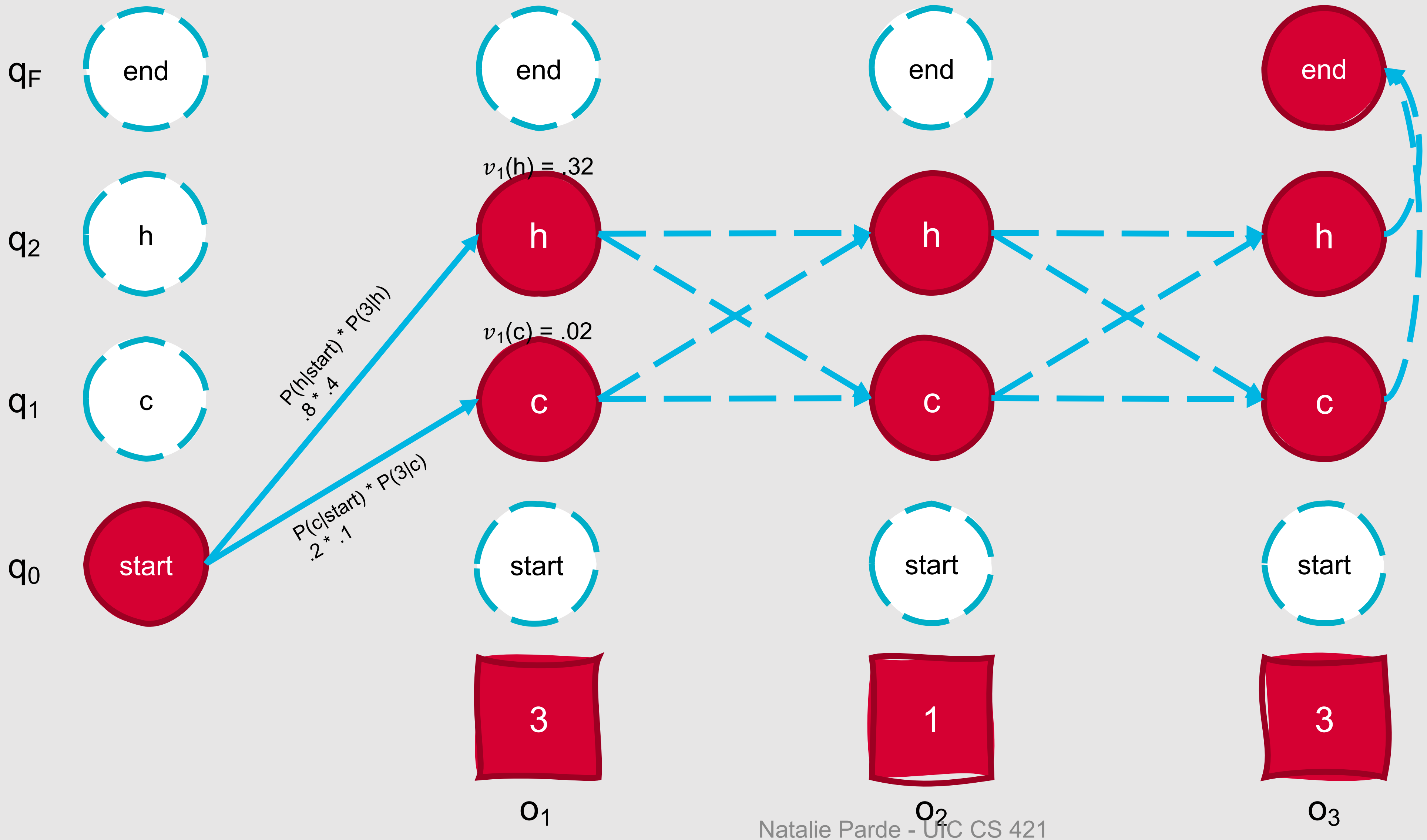
Viterbi Trellis



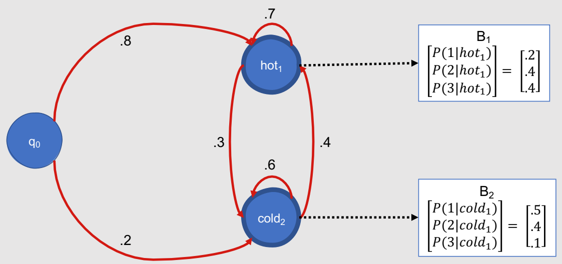


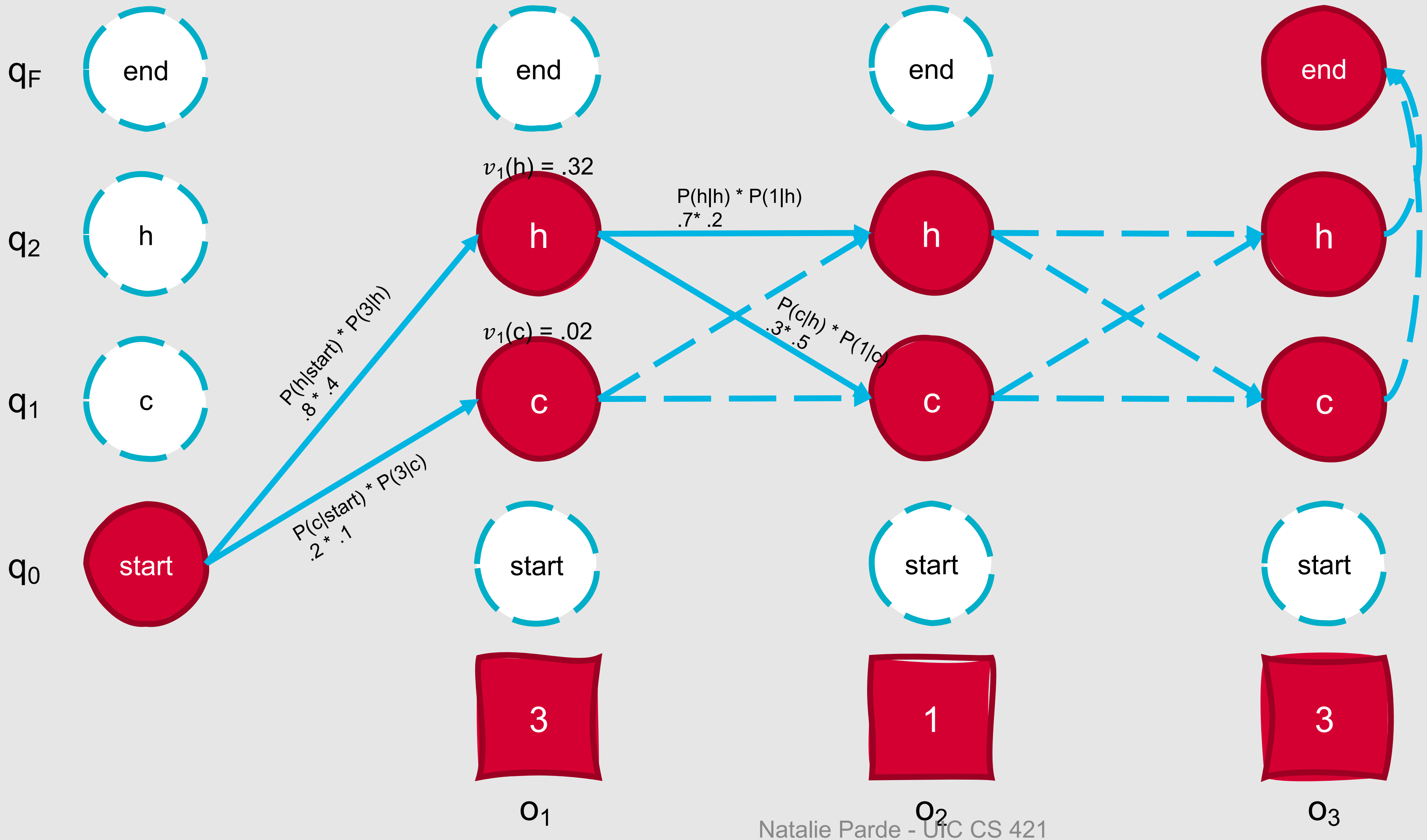
Viterbi Trellis



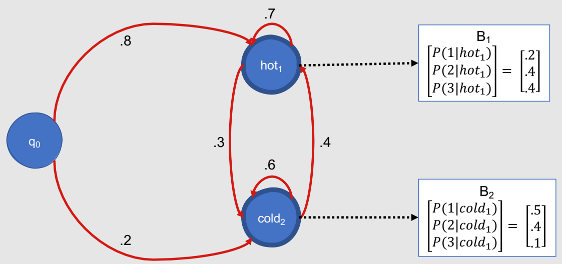


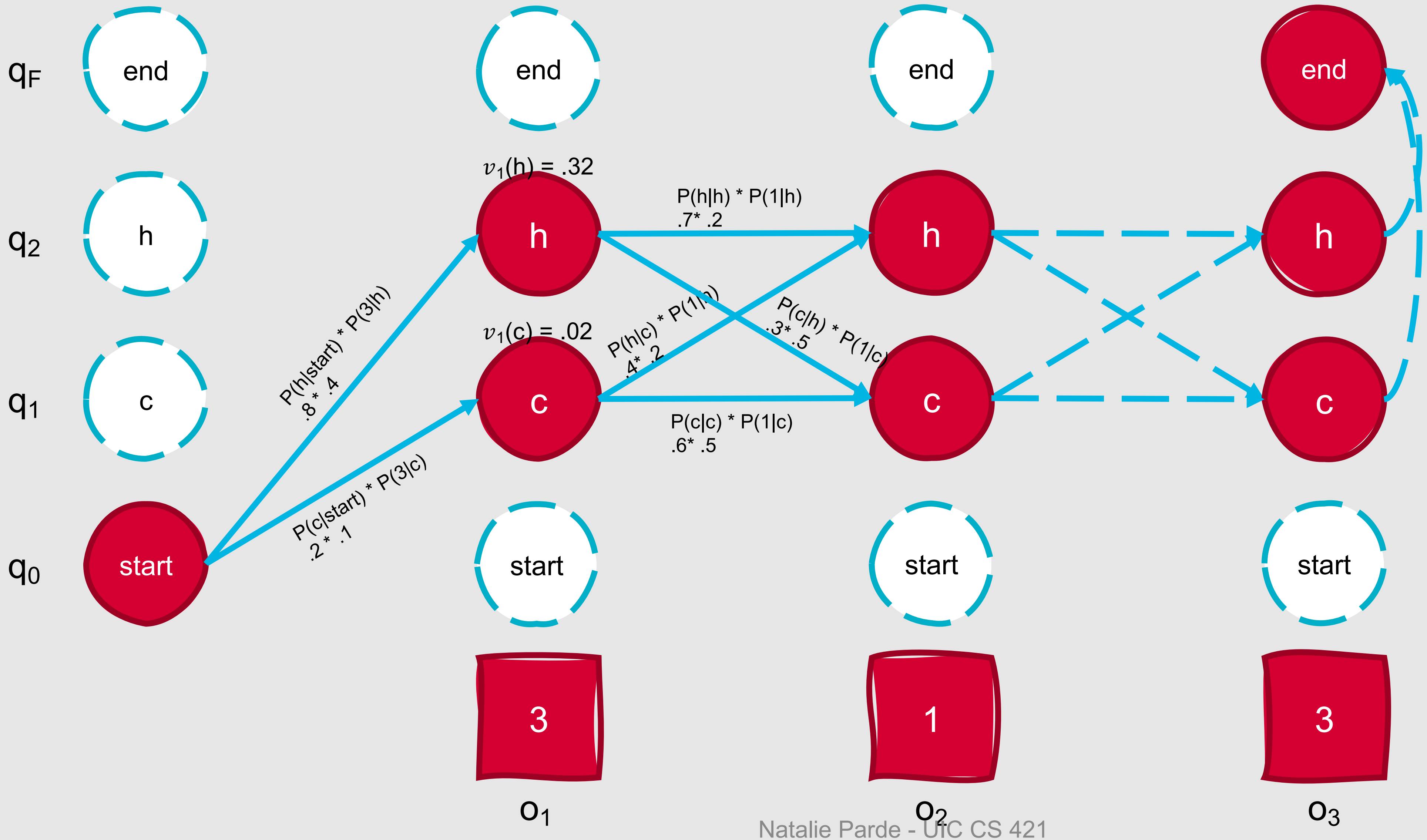
Viterbi Trellis



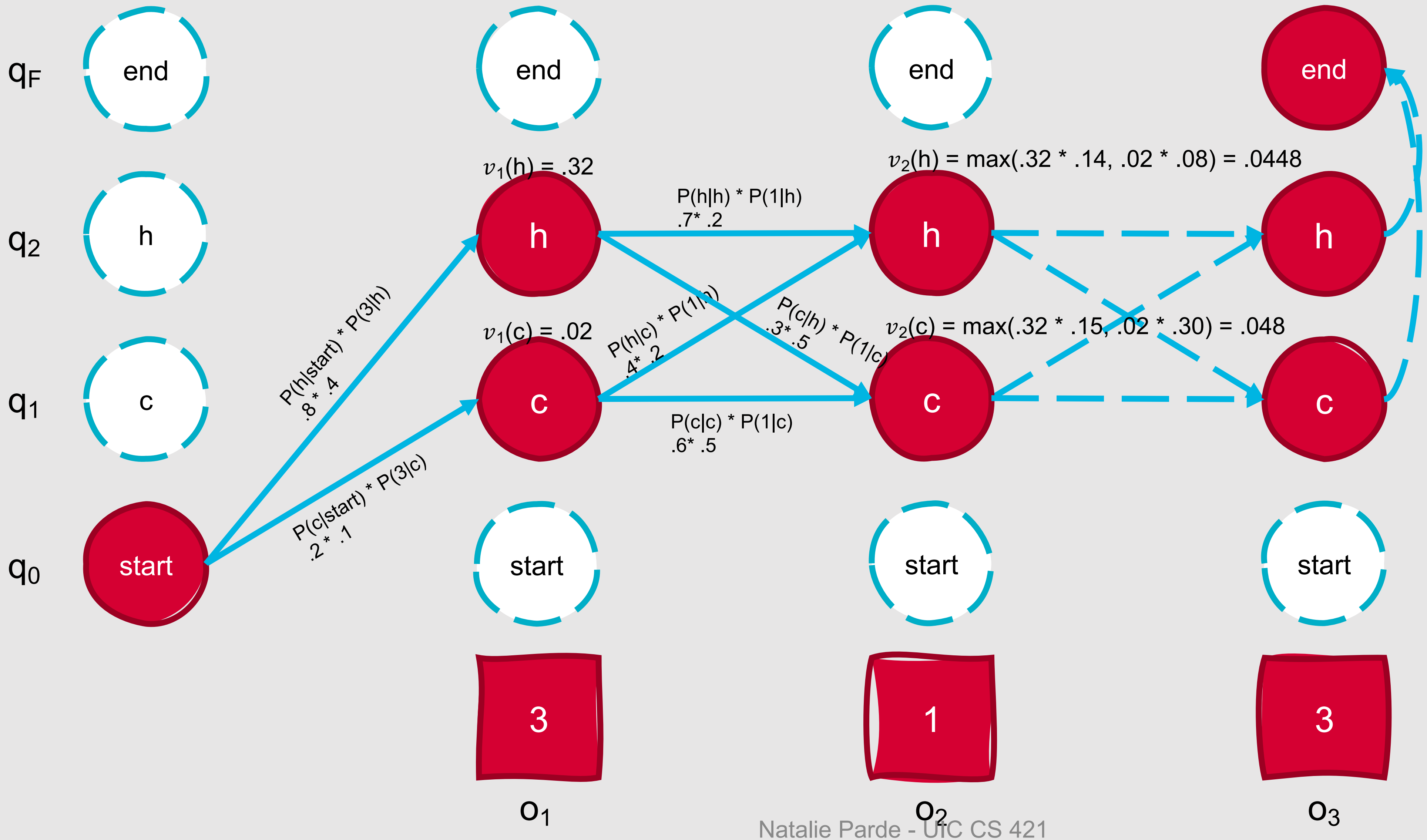


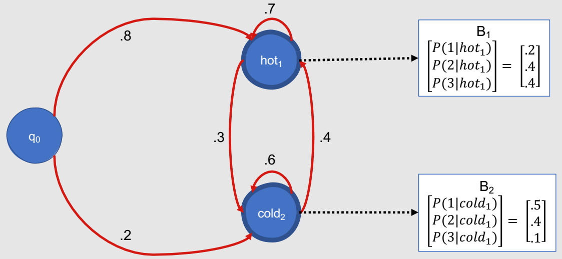
Viterbi Trellis



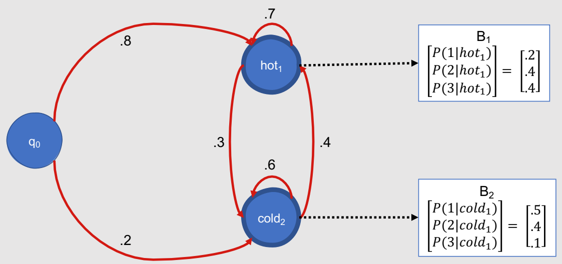


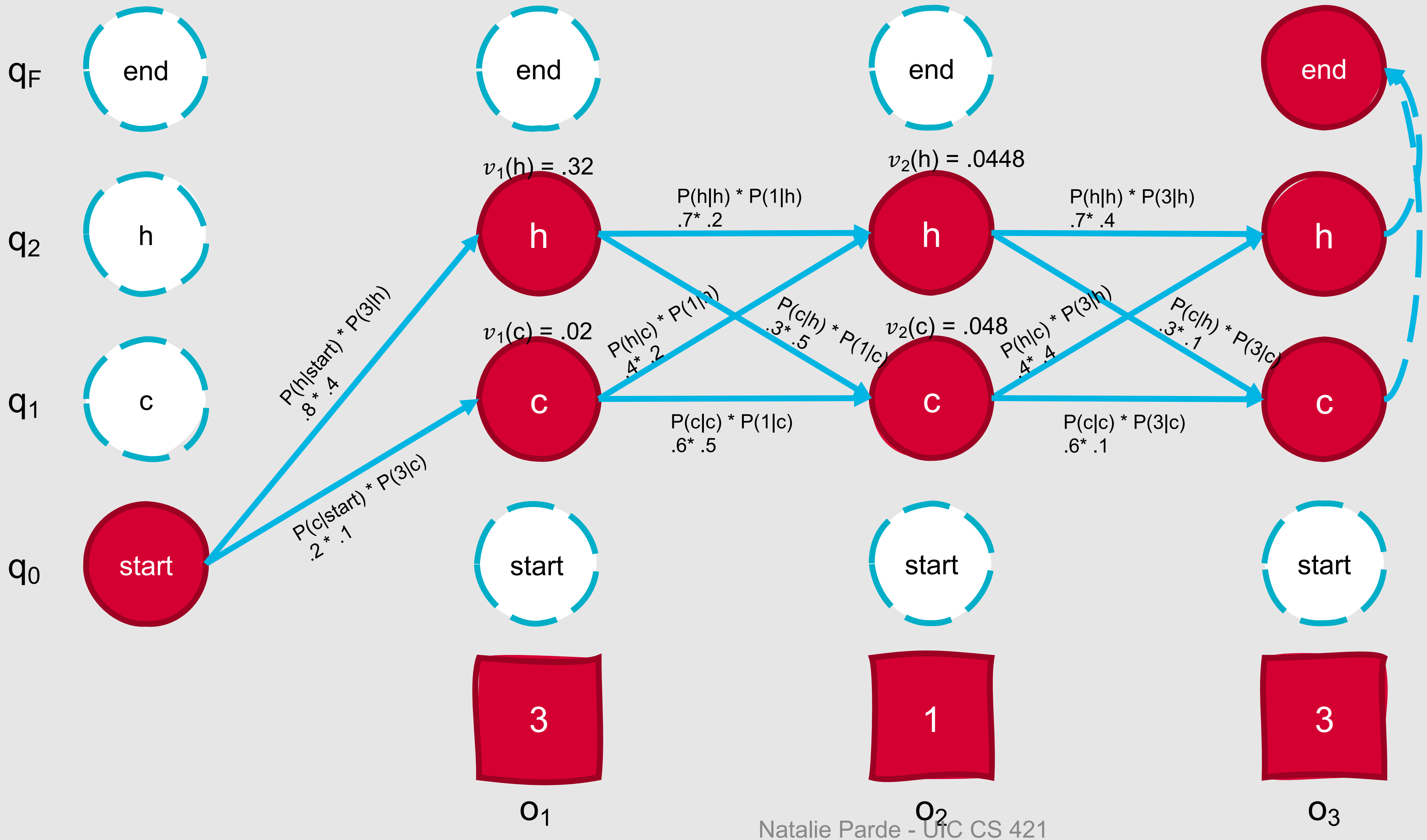
Viterbi Trellis



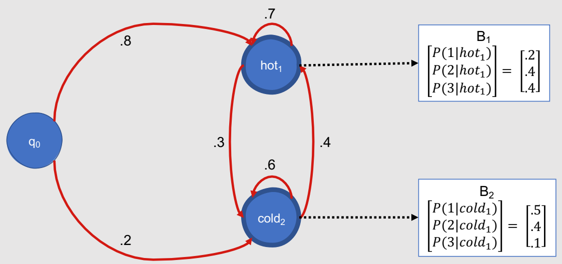


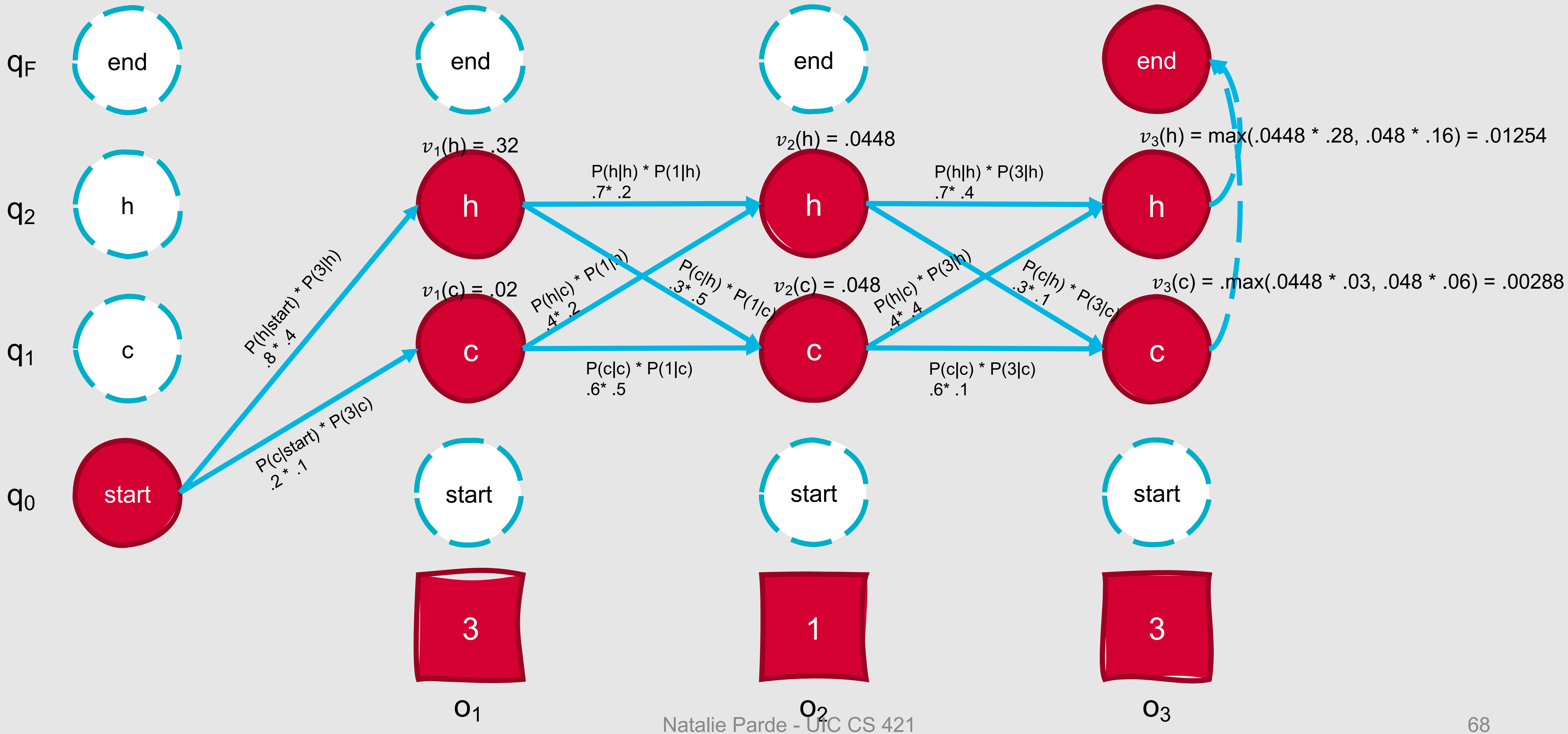
Viterbi Trellis



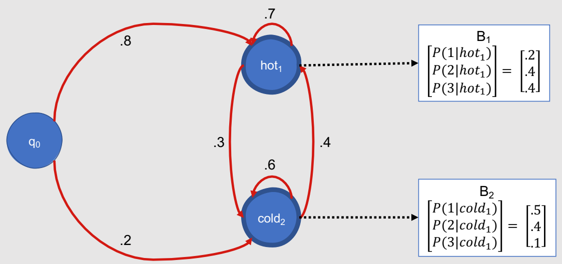


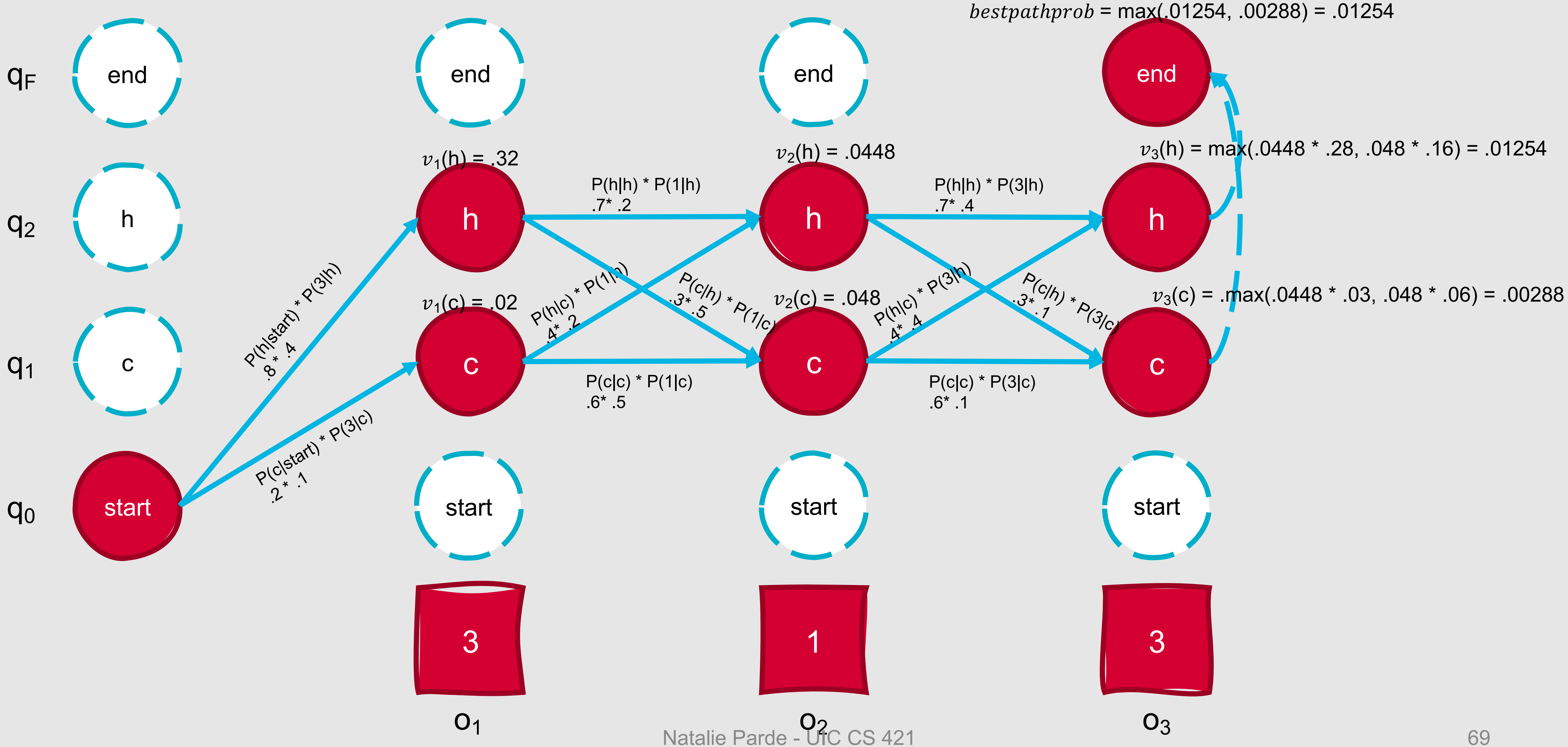
Viterbi Trellis



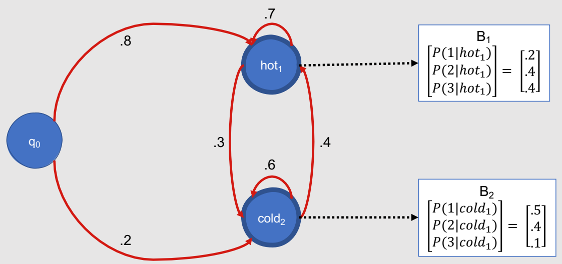


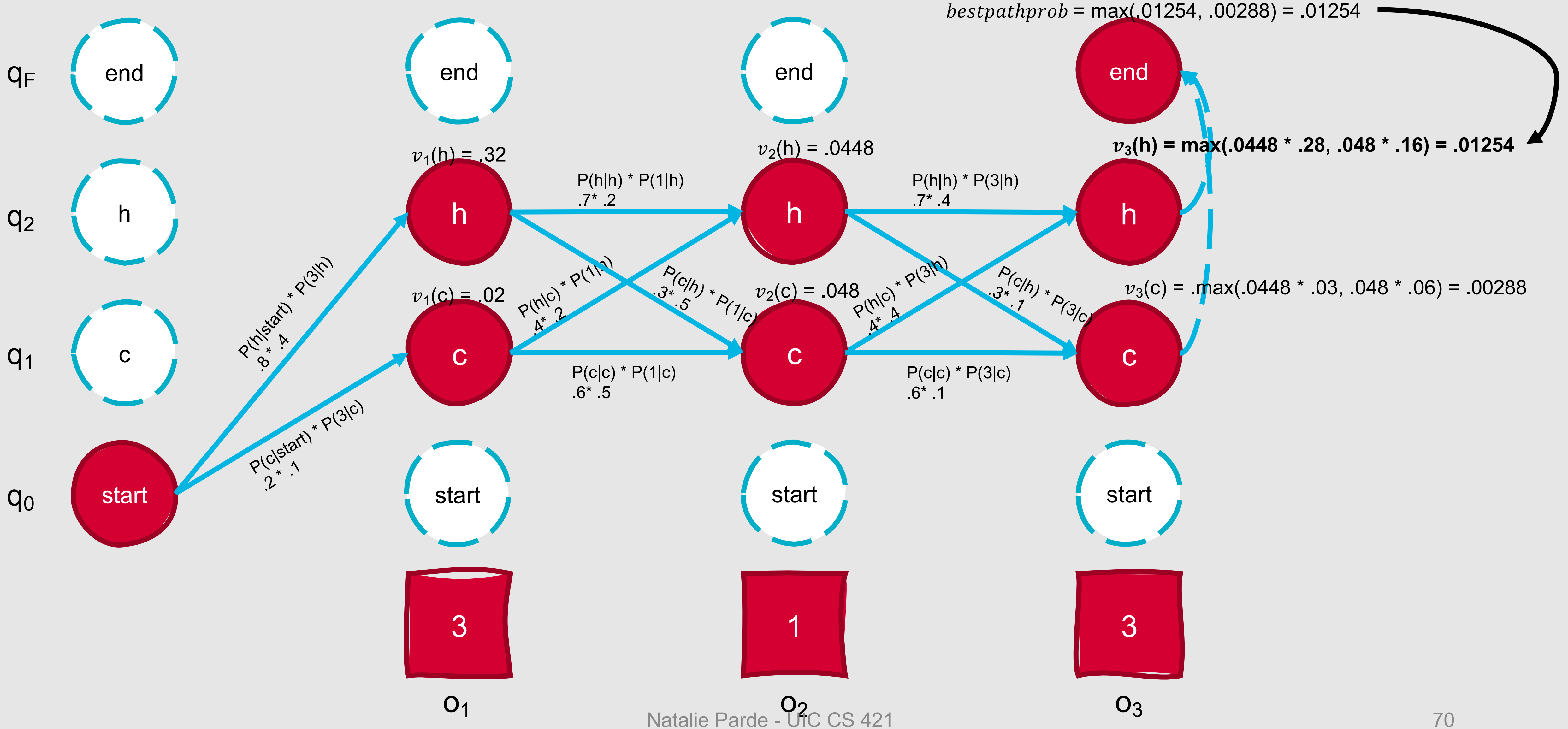
Viterbi Trellis



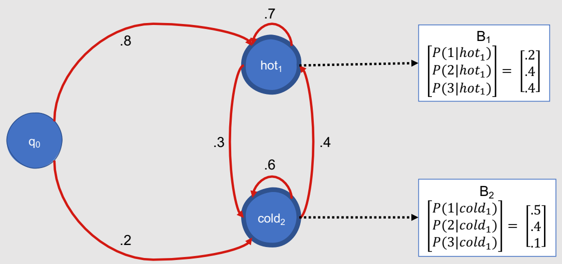


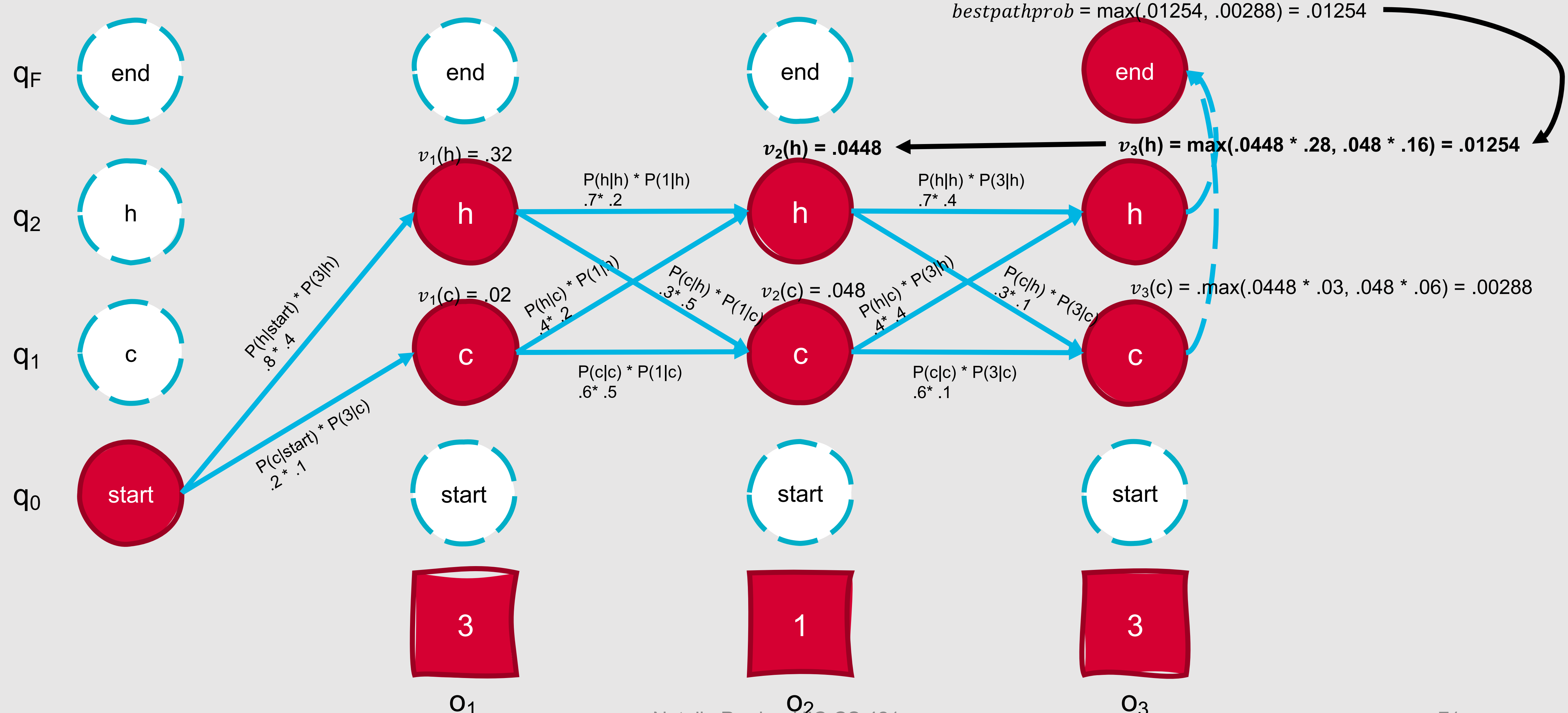
Viterbi Backtrace



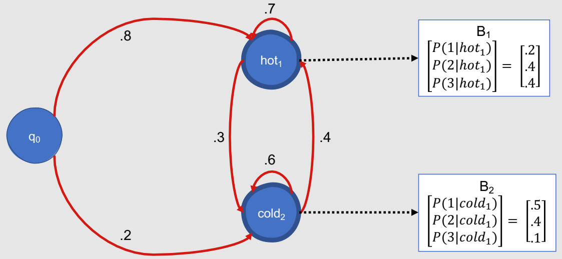


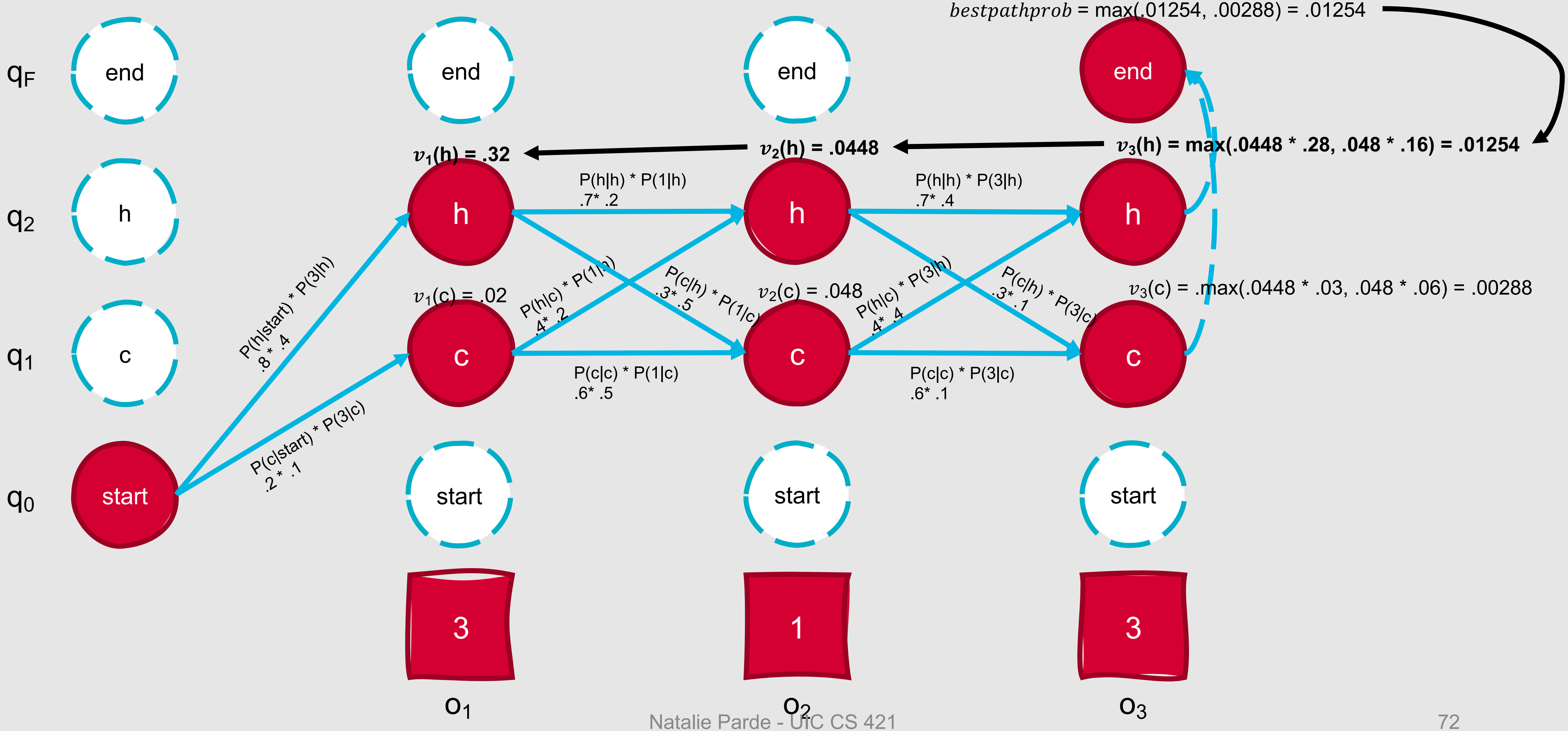
Viterbi Backtrace



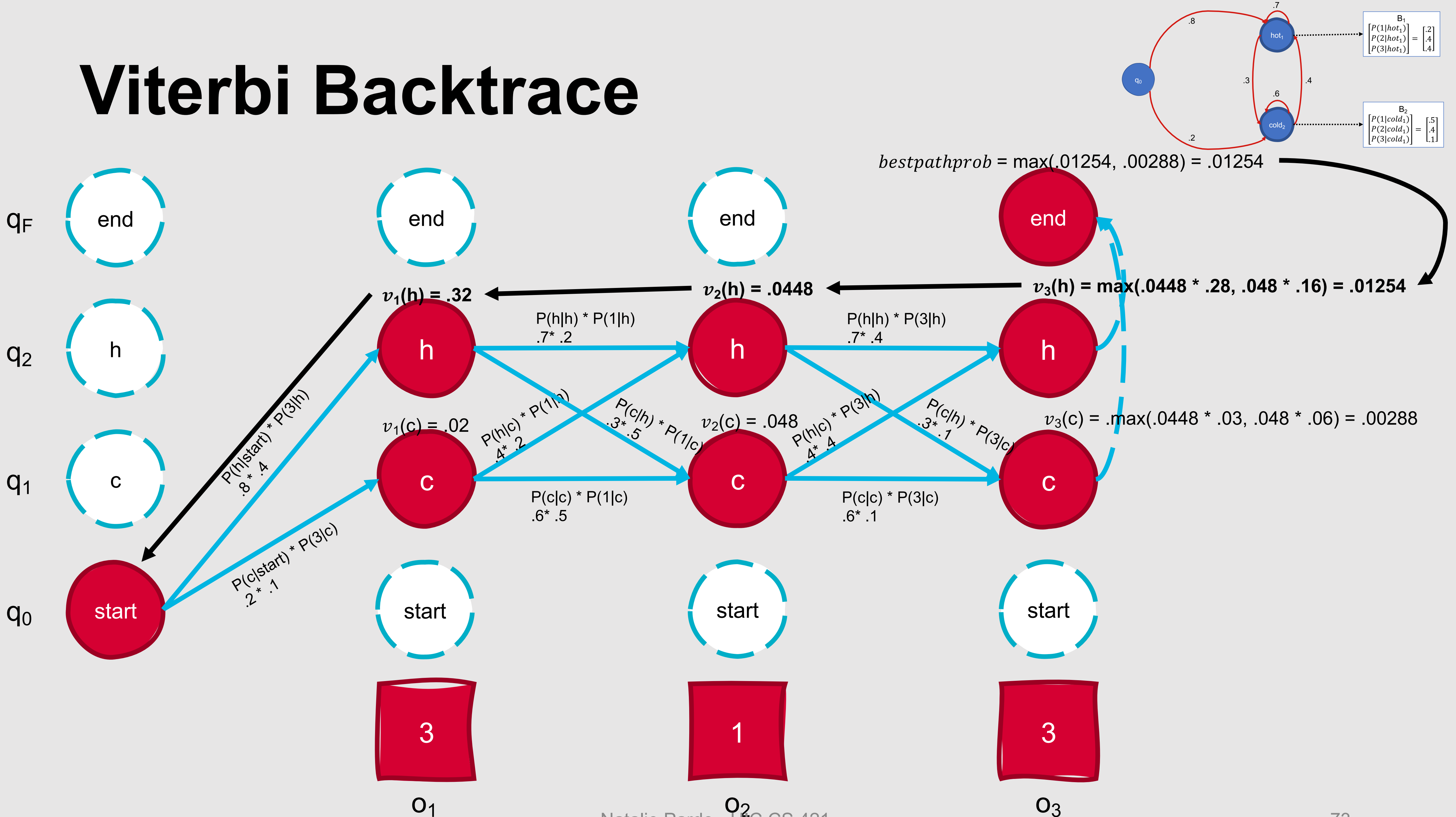


Viterbi Backtrace





Viterbi Backtrace



The Viterbi algorithm is used in many domains, even beyond text processing!

- **Speech recognition**
  - Given an input acoustic signal, find the most likely sequence of words or phonemes
- **Digital error correction**
  - Given a received, potentially noisy signal, determine the most likely transmitted message
- **Computer vision**
  - Given noisy measurements in video sequences, estimate the most likely trajectory of an object over time
- **Economics**
  - Given historical data, predict financial market states at certain timepoints

This Week's Topics

Tuesday

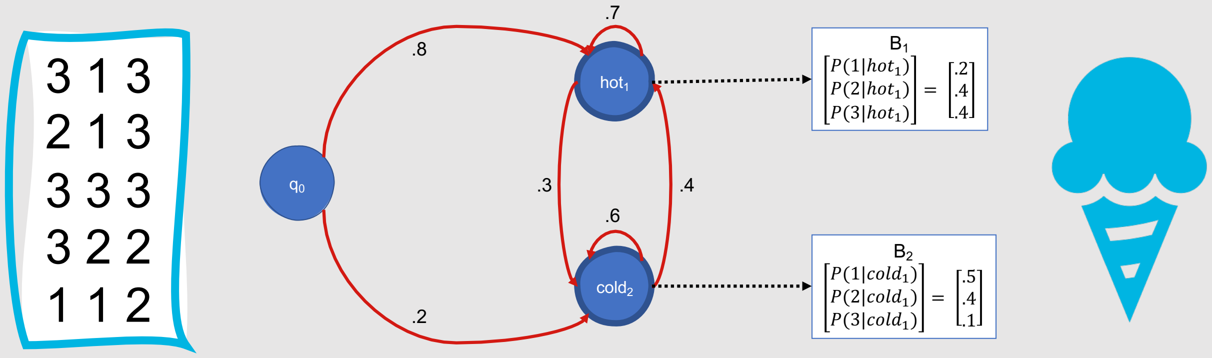
- Hidden Markov Models
- Forward Algorithm
- Viterbi Algorithm
- Forward-Backward Algorithm

Thursday

- Parts of Speech
- POS Tagsets
- POS Tagging

Finally …how do we train HMMs?

- If we have a set of observations, can we learn the parameters (transition probabilities and observation likelihoods) directly?



Forward-Backward Algorithm

- Special case of expectation-maximization (EM) algorithm
- Input:
  - Unlabeled sequence of observations, O
  - Vocabulary of hidden states, Q
- Output: Transition probabilities and observation likelihoods

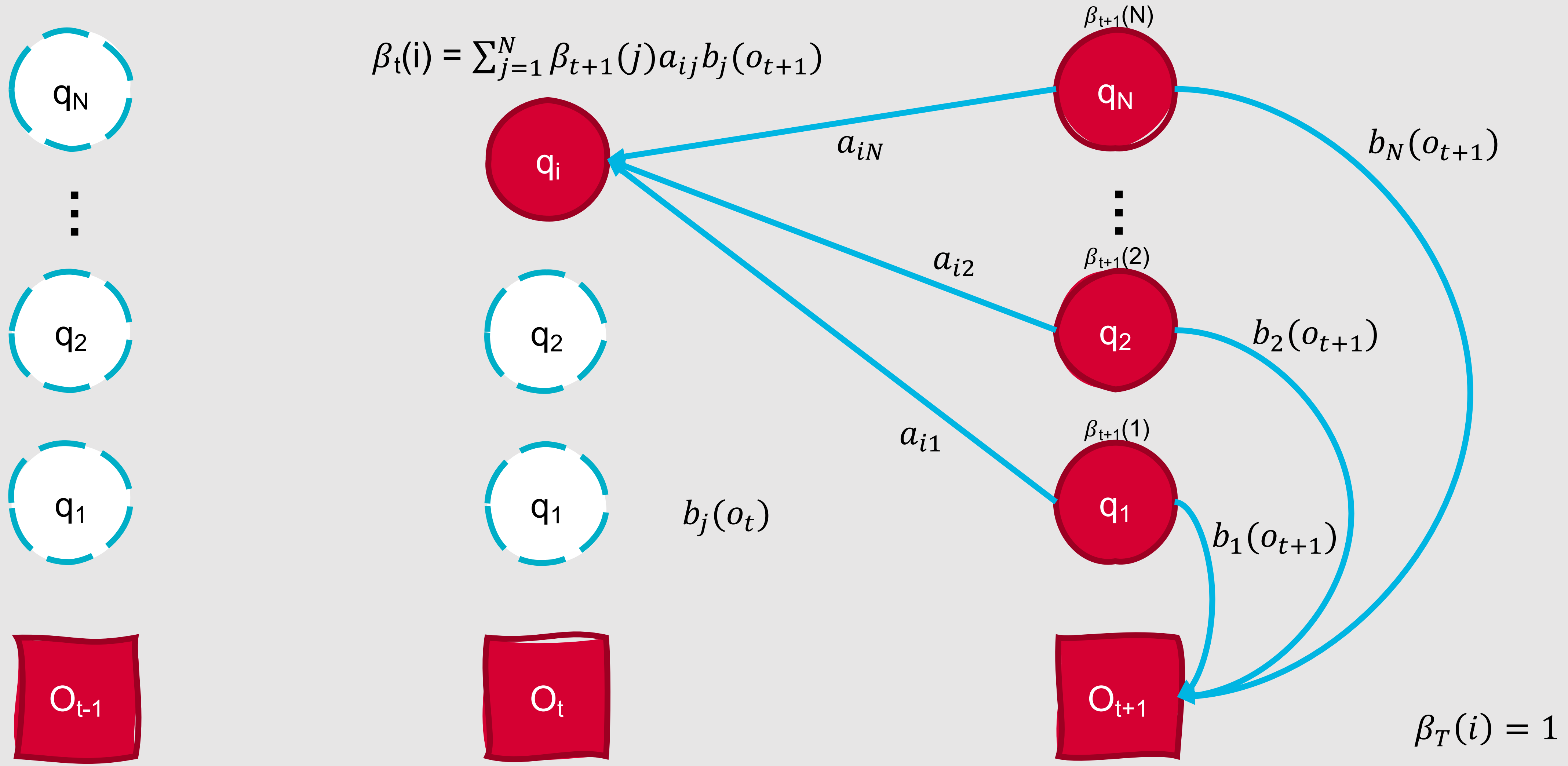
How does the algorithm compute these outputs?

- **Iteratively estimate the counts for transitions from one state to another**
  - Start with base estimates for a_{ij} and b_j, and iteratively improve those estimates
- **Get estimated probabilities** by:
  - Computing the forward probability for an observation
  - Dividing that probability mass among all the different paths that contributed to this forward probability (**backward probability**)

Backward Algorithm

- We define the backward probability as follows:
  - $\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T | q_t = i, \lambda)$
  - Probability of generating partial observations from time *t*+1 until the end of the sequence, given that the HMM λ is in state *i* at time *t*
- Also computed using a trellis, but moves backwards instead

Backward Step



For the expectation step of the forward-backward algorithm, we re-estimate transition probabilities and observation likelihoods.

- We re-estimate transition probabilities, a_{ij}, as follows:
  - Let $\zeta_t(i,j) = \frac{a_t(i) a_{aij} b_j \beta_{t+1}(j)}{a_T(q_F)}$
  - Then, $\widehat{a_{ij}} = \frac{\text{expected \# transitions from state } i \text{ to state } j}{\text{expected \# transitions from state } i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \sum_{j=1}^{N} \xi_t(i,j)}$
  - Check out the course textbook (Appendix A) for an in-depth discussion of how the numerator and denominator above are derived!

Re-Estimating Observation Likelihood

- We re-estimate b_j as follows:
  - Let $\gamma_t(j) = \frac{a_t(j)\beta_t(j)}{a_T(q_F)}$
  - Then, $\widehat{b_j}(v_k) = \frac{\text{expected \# of times in state } j \text{ and observing symbol } v_k}{\text{expected number of times in state } j} = \frac{\sum_{t=1 \text{ s.t. } o_t = v_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$

Putting it all together, we have the forward-backward algorithm!

```
initialize A and B
iterate until convergence:

    # Expectation Step
    compute γ_t(j) for all t and j
    compute ζ_t(i,j) for all t, i, and j

    # Maximization Step
    α_ij = â_ij for all i and j
    b_j(v_k) = b̂_j(v_k) for all j, and all v_k in the output vocab V
```

Summary: Hidden Markov Models

- **HMMs** are probabilistic generative models for sequences
- They make predictions based on underlying hidden states
- Three fundamental HMM problems include:
  - Computing the likelihood of a sequence of observations
  - Determining the best sequence of hidden states for an observed sequence
  - Learning HMM parameters given an observation sequence and a set of hidden states
- Observation likelihood can be computed using the **forward algorithm**
- Sequences of hidden states can be decoded using the **Viterbi algorithm**
- HMM parameters can be learned using the **forward-backward algorithm**

This Week's Topics

Tuesday

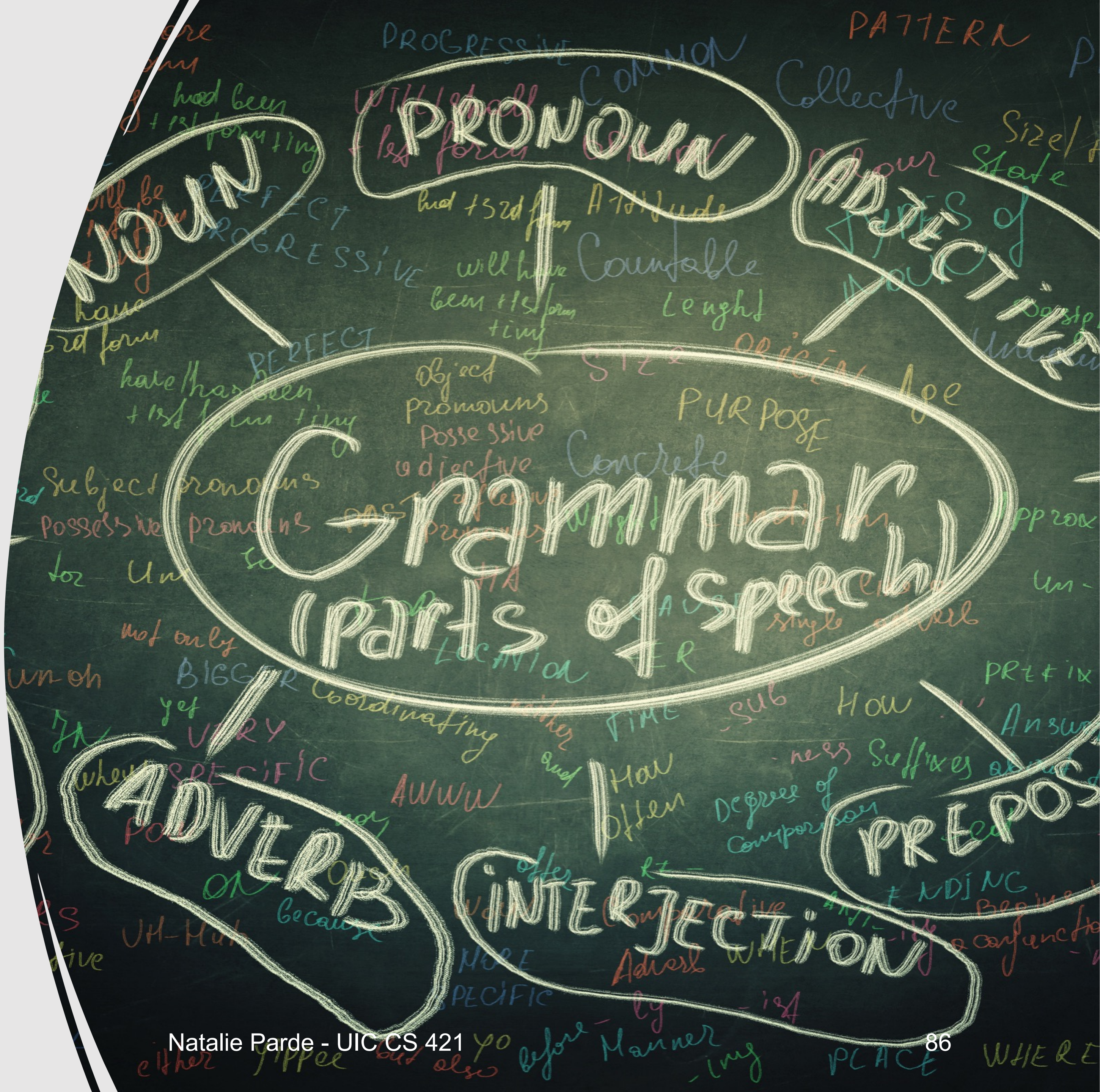
- Hidden Markov Models
- Forward Algorithm
- Viterbi Algorithm
- Forward-Backward Algorithm

Thursday

- Parts of Speech
- POS Tagsets
- POS Tagging

What are parts of speech?

- Traditional (broad) categories:
    - noun
    - verb
    - adjective
    - adverb
    - preposition
    - article
    - interjection
    - pronoun
    - conjunction
- Sometimes also referred to as **lexical categories**, **word classes**, or **morphological classes**



Parts of Speech

Noun

- People, places, or things
- Doctor, mountain, cellphone….

Verb

- Actions or states
- Eat, sleep, be….

Adjective

- Descriptive attributes
- Purple, triangular, windy….

Adverb

- Modifies other words by answering *how*, *in what way*, *when*, *where*, and *to what extent* questions
- Gently, quite, quickly….

Parts of Speech

- **Pronoun**
  - Refers to nouns mentioned elsewhere
  - he, she, you….
- **Preposition**
  - Describes relationship between noun/pronoun and other word in clause
  - on, above, to….
- **Article**
  - Indicates specificity
  - a, an, the….
- **Interjection**
  - Exclamations
  - oh, yikes, ah….
- **Conjunction**
  - Coordinates words in the same clause or connects multiple clauses/sentences
  - and, but, if….

What is part-of-speech (POS) tagging?

The process of automatically assigning grammatical word classes to individual tokens in text.

Why is POS tagging useful?

- First step of many pipelined NLP tasks:
  - Speech synthesis
  - Constituency parsing
  - Dependency parsing
  - Information extraction
  - And many more!

Even when using end-to-end approaches or pretrained LLMs, POS tagging is useful.

Offers an avenue for interpretable linguistic analysis!

POS Tag Categories

Each POS type falls into one of two larger classes:

- Open
- Closed

Open class:

- New members can be created at any time
- In English:
  - Nouns, verbs, adjectives, and adverbs
- Many (but not all!) languages have these four classes

Closed class:

- A small, fixed membership …new members cannot be created spontaneously
- Usually function words
- In English:
  - Prepositions and auxiliaries (may, can, been, etc.)

Finer-Grained POS Classes

- Broader POS classes often have smaller subclasses
  - Noun:
    - Proper (Illinois)
    - Common (state)
  - Verb:
    - Main (tweet)
    - Modal (had)
- Some subclasses of a part of speech might be open, while others are closed

Open Class

Nouns

Proper

IBM
Italy

Common

cat / cats
snow

Verbs

Main

see
registered

Modal

can
had

Adjectives *old older oldest*

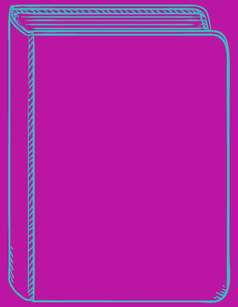
Adverbs *slowly*

Closed Class

Determiners *the some*

Conjunctions *and or*

Prepositions *to with*



POS Tagging

- Can be very challenging!
- Words often have more than one valid part of speech tag
  - Today's faculty meeting went really **well**! = adverb
  - Do you think the undergrads are **well**? = adjective
  - **Well**, did you see the latest response to your email? = interjection
  - Jurafsky and Martin's book is a **well** of information. = noun
  - Laughter began to **well** up inside her at, as always, a highly inconvenient time. = verb

POS Tagging

- Goal: Determine the *best* POS tag for a particular instance of a word.

verb determiner

Give me a **break**!

pronoun noun

verb noun

Did the window **break**?

determiner verb

This Week's Topics

Tuesday

- Hidden Markov Models
- Forward Algorithm
- Viterbi Algorithm
- Forward-Backward Algorithm

Thursday

- Parts of Speech
- POS Tagsets
- POS Tagging

POS Tagsets

In order to determine which POS tag to assign to a word, we first need to decide which **tagset** we will use

Tagset: A finite set of POS tags, where each tag defines a distinct grammatical role

Can range from very coarse to very fine

Penn Treebank Tagset

- **Most common POS tagset**
- 36 POS tags + 12 other tags (punctuation and currency)
- Used when developing the Penn Treebank, a corpus created at the University of Pennsylvania containing more than 4.5 million words of American English
- Link to documentation: https://catalog.ldc.upenn.edu/docs/LDC95T7/cl93.html

Penn Treebank Tagset

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

What do some of these distinctions mean?

cities

Chicago

Chicagos

city

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

What do some of these distinctions mean?

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

eat

ate

eating

eaten

eat

eats

should

What do some of these distinctions mean?

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
F weird	oreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB** weirder	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	M weirdest	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

What do some of these distinctions mean?

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	calmly	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, calmer ve	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singul calmest	**SYM**	Symbol	**WRB**	Wh-adverb

As a general (but not perfect!) rule….

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3^{rd} person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3^{rd} person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

Closed Class

As a general (but not perfect!) rule….

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

Open Class

Other Popular POS Tagsets

- **Brown Corpus**
  - ~1 million words of American English text
  - 82 (!) POS tags
- **C5 Tagset**
  - Text from the British National Corpus
  - 61 POS tags
- **C7 Tagset**
  - Text from the British National Corpus
  - 146 (!!) POS tags

This Week's Topics

Hidden Markov Models
Forward Algorithm
Viterbi Algorithm
Forward-Backward Algorithm

Tuesday

Thursday

Parts of Speech
POS Tagsets
POS Tagging

So …how can we assign POS tags?

So …how can we assign POS tags?

Time	flies	like	an	arrow;	fruit	flies	like	a	banana

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN									

CC	Coordinating Conjunction	**NNS**	Noun, plural	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ								

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN							

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner	**NNPS**	Proper noun, plural	**VB**	Verb, base form ? ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT						

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner :)	**NNPS**	Proper noun, plural	**VB**	Verb, base form ? ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN				

CC	Coordinating Conjunction	**NNS**	Noun, plural ?	**TO**	to
CD	Cardinal Number	**NNP**	Proper noun, singular	**UH**	Interjection
DT	Determiner :)	**NNPS**	Proper noun, plural	**VB**	Verb, base form ? ?
EX	Existential *there*	**PDT**	Predeterminer	**VBD**	Verb, past tense
FW	Foreign word	**POS**	Possessive ending	**VBG**	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	**PRP**	Personal pronoun	**VBN**	Verb, past participle
JJ	Adjective	**PRP$**	Possessive pronoun	**VBP**	Verb, non-3rd person singular present
JJR	Adjective, comparative	**RB**	Adverb	**VBZ**	Verb, 3rd person singular present ?
JJS	Adjective, superlative	**RBR**	Adverb, comparative	**WDT**	Wh-determiner
LS	List item marker	**RBS**	Adverb, superlative	**WP**	Wh-pronoun
MD	Modal	**RP**	Particle	**WP$**	Possessive wh-pronoun
NN	Noun, singular or mass ?	**SYM**	Symbol	**WRB**	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS			

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ?	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ??
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ		

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ??	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ???
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ	DT	

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ??	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ???
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

So …how can we assign POS tags?

Time	flies	like	an	arrow	fruit	flies	like	a	banana
NN	VBZ	IN	DT	NN	NN	NNS	VBZ	DT	NN

CC	Coordinating Conjunction	NNS	Noun, plural ??	TO	to
CD	Cardinal Number	NNP	Proper noun, singular	UH	Interjection
DT	Determiner	NNPS	Proper noun, plural	VB	Verb, base form ??
EX	Existential *there*	PDT	Predeterminer	VBD	Verb, past tense
FW	Foreign word	POS	Possessive ending	VBG	Verb, gerund or present participle
IN	Preposition or subordinating conjunction ??	PRP	Personal pronoun	VBN	Verb, past participle
JJ	Adjective	PRP$	Possessive pronoun	VBP	Verb, non-3rd person singular present
JJR	Adjective, comparative	RB	Adverb	VBZ	Verb, 3rd person singular present ???
JJS	Adjective, superlative	RBR	Adverb, comparative	WDT	Wh-determiner
LS	List item marker	RBS	Adverb, superlative	WP	Wh-pronoun
MD	Modal	RP	Particle	WP$	Possessive wh-pronoun
NN	Noun, singular or mass ?	SYM	Symbol	WRB	Wh-adverb

Ambiguity is a big issue for POS taggers!

- Many words have multiple senses
  - **time** = noun, verb
  - **flies** = noun, verb
  - **like** = verb, preposition

Just how ambiguous is natural language?

- Brown Corpus: Approximately 11% of word types have multiple valid part of speech labels
- These tend to be very common words!
    - We think **that** the meeting will only last two more hours. = IN
    - Was **that** the 32nd Piazza post today? = DT
    - You can’t eat **that** many donuts every time the clock strikes midnight! = RB
- Overall, ~40% of word *tokens* are instances of ambiguous word *types*

Despite this, modern POS taggers still work quite well.

- Accuracy > 97%
- Even a simple baseline can achieve ~90% accuracy
  - Tag every word with its most frequent tag
  - Tag unknown words as nouns

How do POS taggers work?

- Numerous ways to predict POS tags:
  - Rule-based
  - Statistical
    - HMMs
    - Neural sequence modeling

Rule-Based POS Tagging

- Start with a dictionary, and assign all relevant tags to the words in that dictionary
- Manually design rules to selectively remove invalid tags for test instances in context
- Keep the remaining correct tag for each word

Example Rule-Based Approach

- Start with a dictionary that specifies permissible tags for our small vocabulary:
  - she
    - PRP
  - promised
    - VBN, VBD
  - to
    - TO
  - back
    - VB, JJ, RB, NN
  - the
    - DT
  - bill
    - NN, VB

Example Rule-Based Approach

she	promised	to	back	the	bill
PRP	VBN	TO	VB	DT	NN
	VBD		JJ		VB
			RB		
			NN		

Example Rule-Based Approach

Eliminate VBN if VBD is an option when VBN|VBD follows “<start> PRP”

she	promised	to	back	the	bill
PRP	~~VBN~~	TO	VB	DT	NN
	VBD		JJ		VB
			RB		
			NN		

Example Rule-Based Approach

she	promised	to	back	the	bill
PRP	~~VBN~~	TO	VB	DT	NN
	VBD		~~JJ~~		~~VB~~
			~~RB~~		
			~~NN~~		

Rule-based POS taggers are an adequate baseline, but….

- Like all rule-based methods, they carry important disadvantages:
  - Time-consuming to build
  - Difficult to update or generalize to new domains
  - Might miss important patterns latent in the specified text domain

Nice alternative to rule-based POS tagging?

- **Statistical POS Tagging:** POS taggers that make decisions based on learned knowledge of POS tag distribution in a training corpus
  - *the* is usually tagged as DT
  - Words with uppercase letters are more likely to be tagged NNP or NNPS
  - Words starting with the prefix *un-* may be tagged JJ
  - Words ending with the suffix *–ly* may be tagged RB

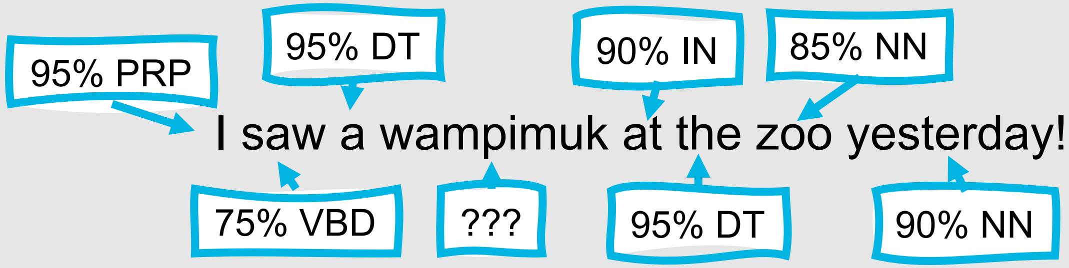
Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus

I saw a wampimuk at the zoo yesterday!

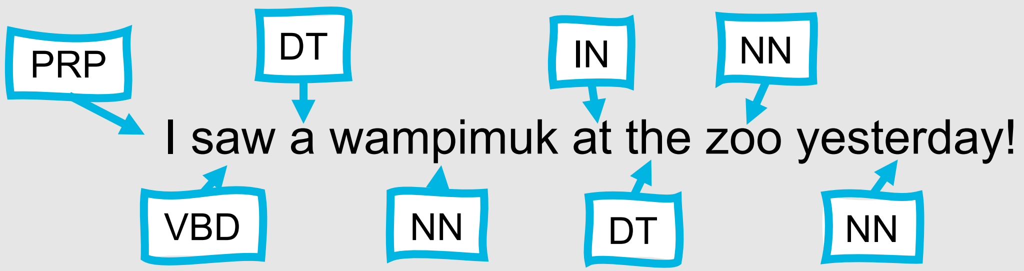
Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus



Simple Statistical POS Tagger

- Using a training corpus, determine the most frequent tag for each word
- Assign POS tags to new words based on those frequencies
- Assign NN to new words for which there is no information from the training corpus



Simple Statistical POS Tagger

- This approach works reasonably well
  - Approximately 90% accuracy
- However, we can do much better!
- One way to improve upon our results is to use **HMMs**

Bigram HMM POS Tagger

- To determine the tag t_i for a single word w_i:
  - $t_i = \underset{t_j \in \{t_0, t_1, \ldots, t_{t-1}\}}{\operatorname{argmax}} P(t_j | t_{i-1}) P(w_i | t_j)$
- This means we need to be able to compute two probabilities:
  - The probability that the tag is t_j given that the previous tag is t_{i-1}
    - $P(t_j | t_{i-1})$
  - The probability that the word is w_i given that the tag is t_j
    - $P(w_i | t_j)$
- We can compute both of these from corpora like the Penn Treebank or the Brown Corpus
- Then, we can find the most optimal sequence of tags using the Viterbi algorithm!

Example: Bigram HMM Tagger

Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

Proper noun, singular

Verb, 3rd person singular present

Verb, past participle

Infinitive to

Adverbial noun

- Given two possible sequences of tags from the Brown Corpus tagset for the following sentence, what is the best way to tag the word “fly”?

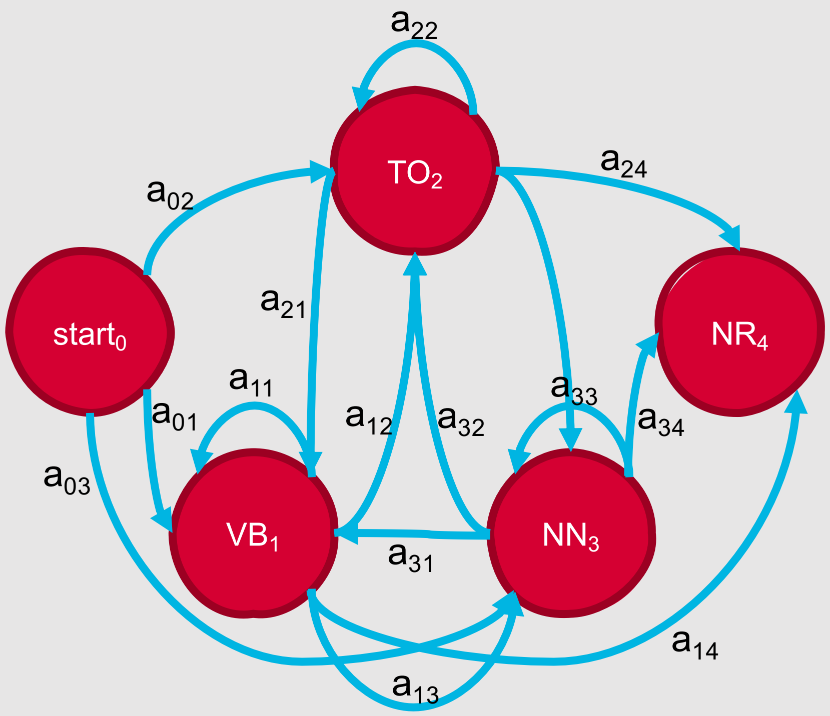
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

- Since we're creating a bigram HMM tagger and focusing on the word "fly," we only need to be concerned with the subsequence "to fly tomorrow"
  - For simplicity when decoding, we'll assume that:
    - The first word in the subsequence for sure has label TO (v_0(TO) = 1.0)
    - The word "tomorrow" for sure has label NR (P("tomorrow"|NR) = 1.0)

Example: Bigram HMM Tagger

We have the following HMM sample:

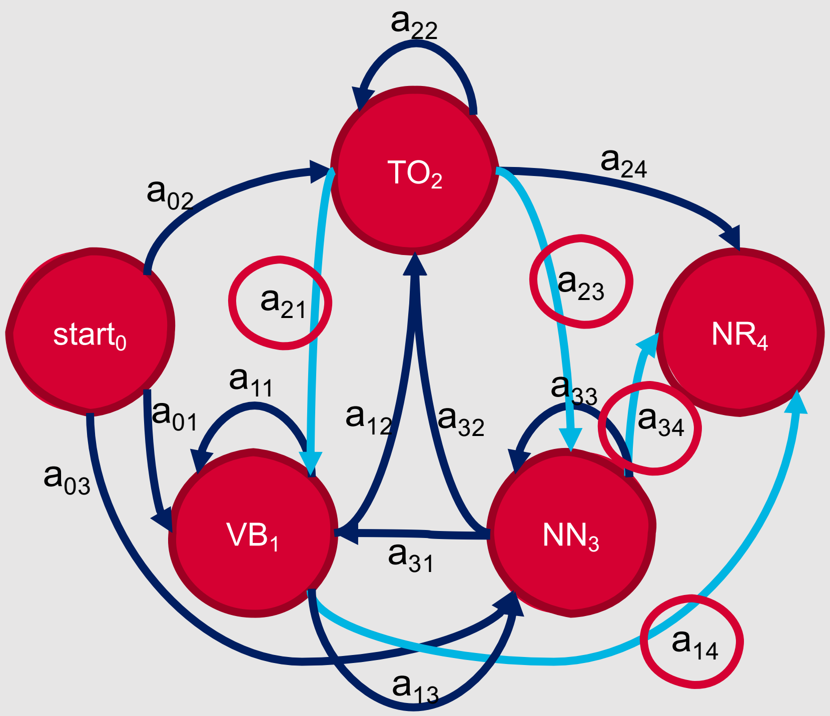
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



Example: Bigram HMM Tagger

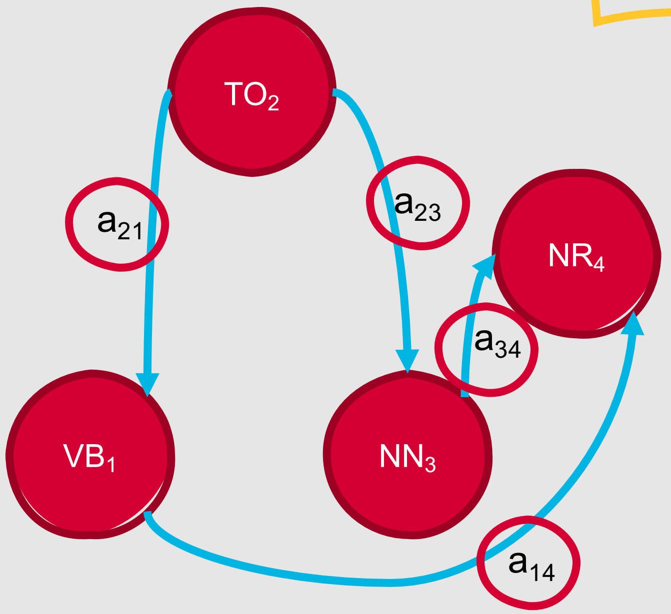
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

The specific transition probabilities we are interested in are:



Example: Bigram HMM Tagger

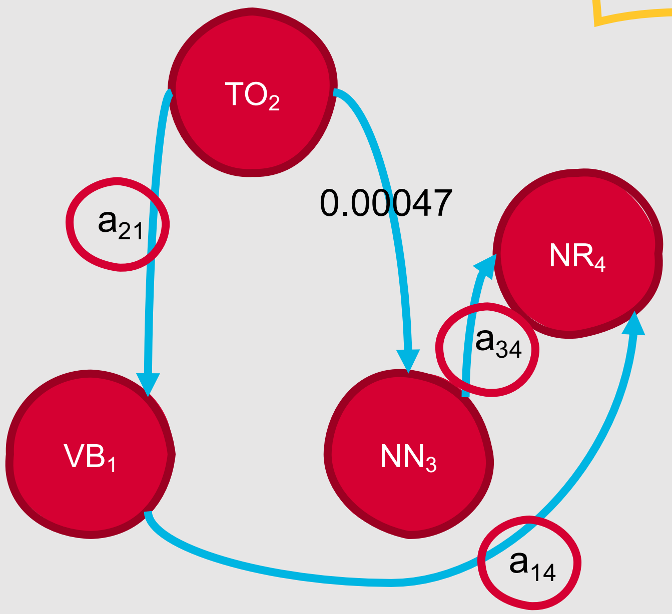
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can estimate the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$

Example: Bigram HMM Tagger

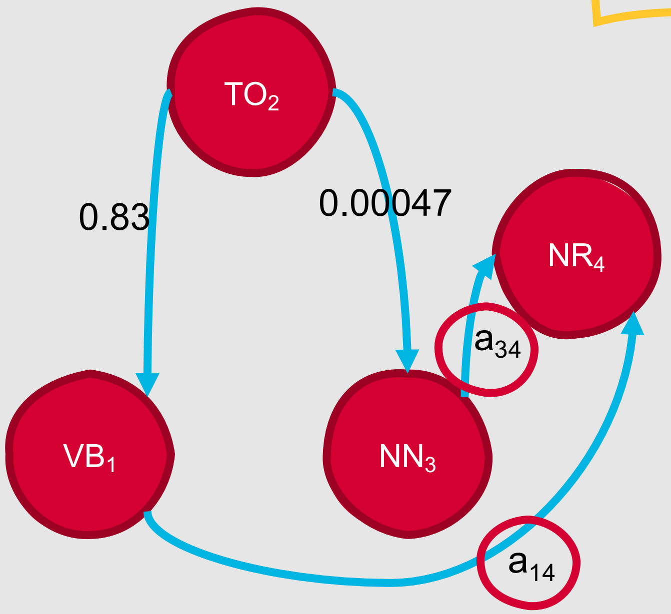
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can estimate the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047

Example: Bigram HMM Tagger

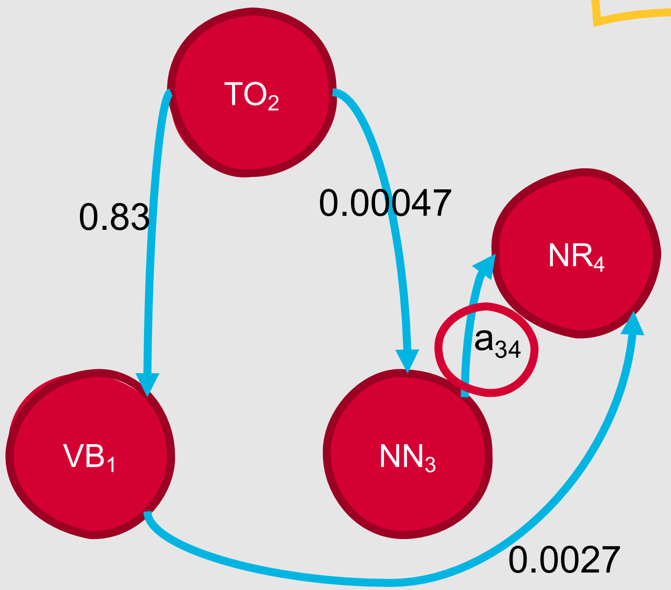
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can estimate the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83

Example: Bigram HMM Tagger

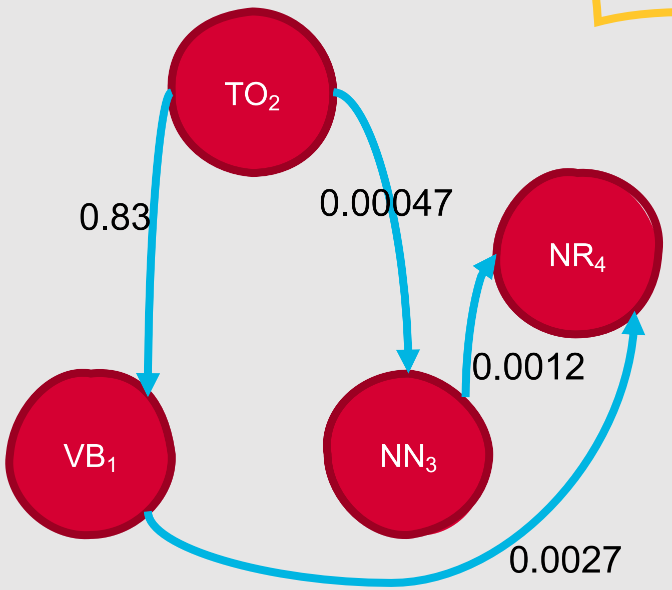
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



- We can estimate the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83
- P(NR|VB) = C(VB NR) / C(VB) = 0.0027

Example: Bigram HMM Tagger

Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR

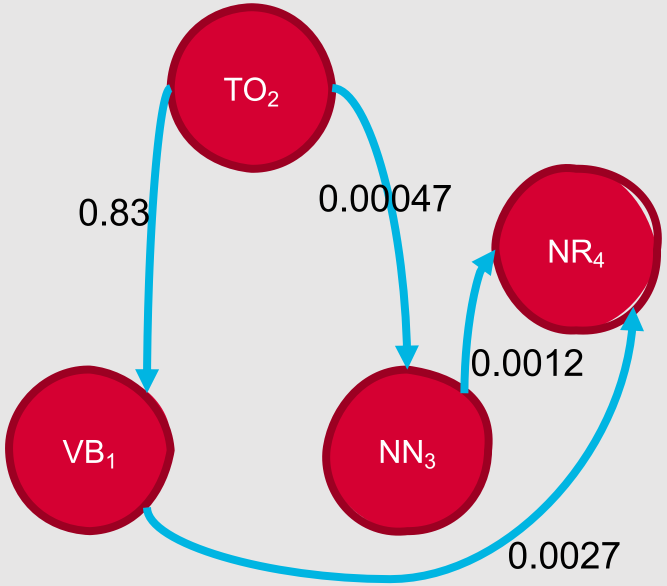


- We can estimate the transition probabilities for a_{21}, a_{23}, a_{34}, and a_{14} using frequency counts from the Brown Corpus
- $P(t_i|t_{i-1}) = \frac{c(t_{i-1}t_i)}{c(t_{i-1})}$
- So, P(NN|TO) = C(TO NN) / C(TO) = 0.00047
- Likewise, P(VB|TO) = C(TO VB) / C(TO) = 0.83
- P(NR|VB) = C(VB NR) / C(VB) = 0.0027
- Finally, P(NR|NN) = C(NN NR) / C(NN) = 0.0012

Example: Bigram HMM Tagger

Example: Bigram HMM Tagger

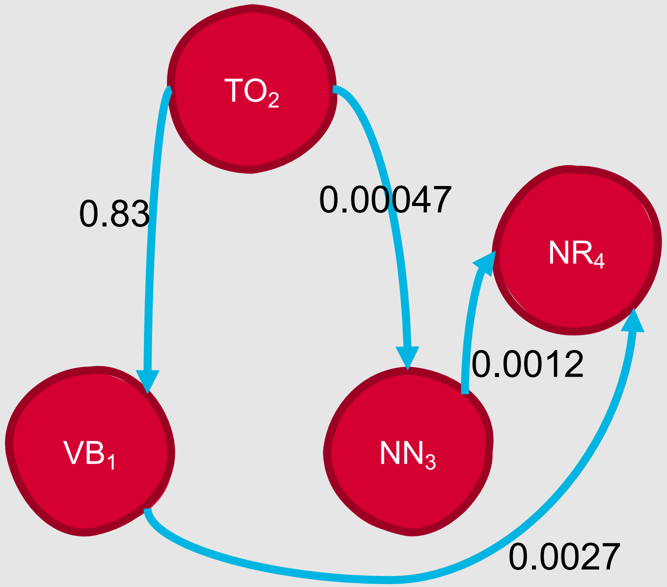
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



	fly
VB	
NN	

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also estimate these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "fly," we need to compute both P(fly|VB) and P(fly|NN)

Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



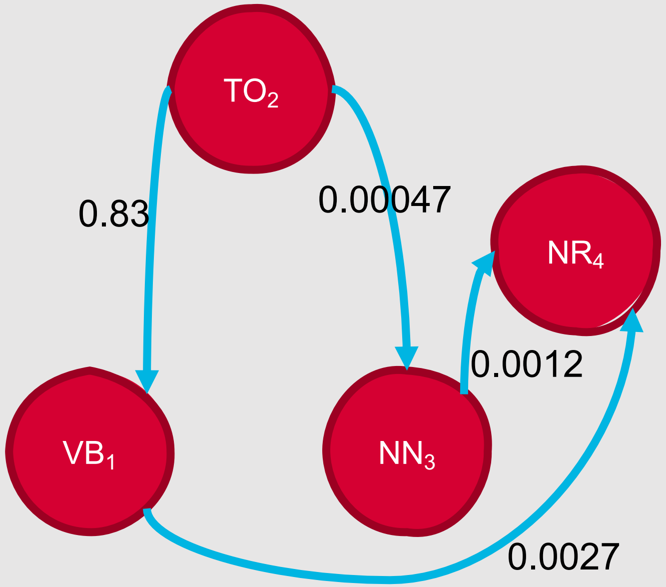
	fly
VB	0.00012
NN	

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also estimate these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "fly," we need to compute both P(fly|VB) and P(fly|NN)
- P(fly|VB) = C(fly, VB) / C(VB) = 0.00012

Example: Bigram HMM Tagger

Example: Bigram HMM Tagger

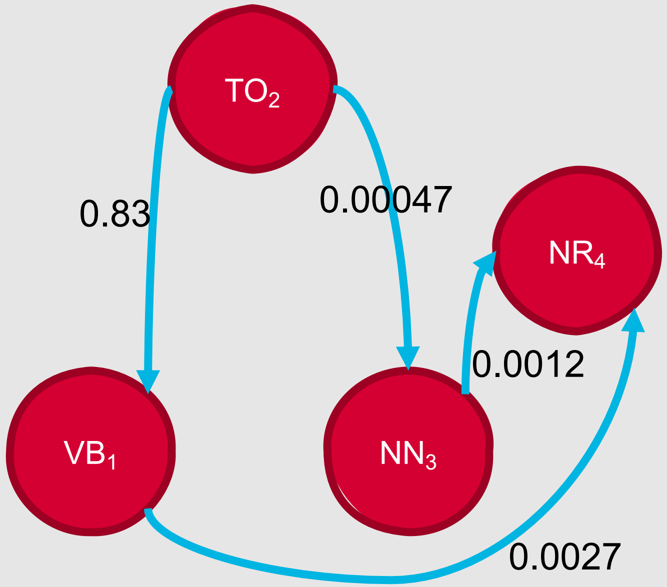
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



	fly
VB	0.00012
NN	0.00057

- We have our transition probabilities …what now?
- Observation likelihoods!
- We can also estimate these using frequency counts from the Brown Corpus
- $P(w_i|t_i) = \frac{c(w_i,t_i)}{c(t_i)}$
- Since we're trying to decide the best tag for "fly," we need to compute both P(fly|VB) and P(fly|NN)
- P(fly|VB) = C(fly, VB) / C(VB) = 0.00012
- P(fly|NN) = C(fly, NN) / C(NN) = 0.00057

Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	TO	VB	NR
NNP	VBZ	VBN	TO	NN	NR



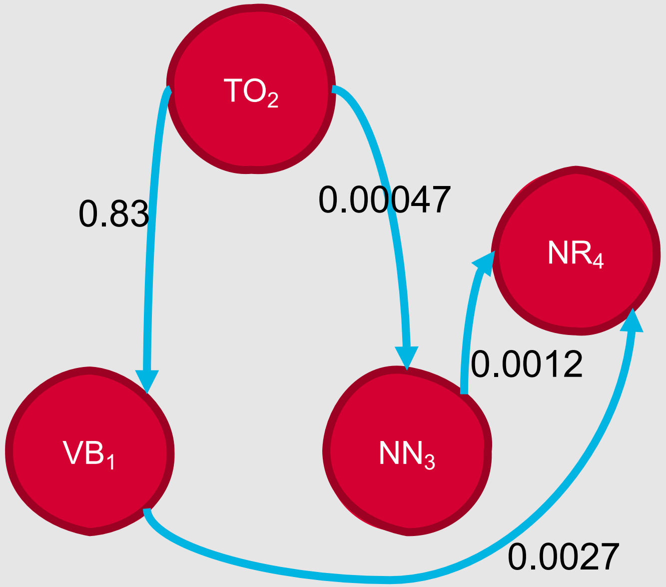
	fly
VB	0.00012
NN	0.00057

- Now, to decide how to tag “fly,” we can consider our two possible sequences:
  - to (TO) fly (VB) tomorrow (NR)
  - to (TO) fly (NN) tomorrow (NR)
- We will select the tag that maximizes the probability:
  - $P(t_i|TO)P(NR|t_i)P(fly|t_i)$
- We determine that:
  - P(VB|TO)P(NR|VB)P(fly|VB) = 0.83 * 0.0027 * 0.00012 = 0.00000027
  - P(NN|TO)P(NR|NN)P(fly|NN) = 0.00047 * 0.0012 * 0.00057 = 0.00000000032

Example: Bigram HMM Tagger

Example: Bigram HMM Tagger

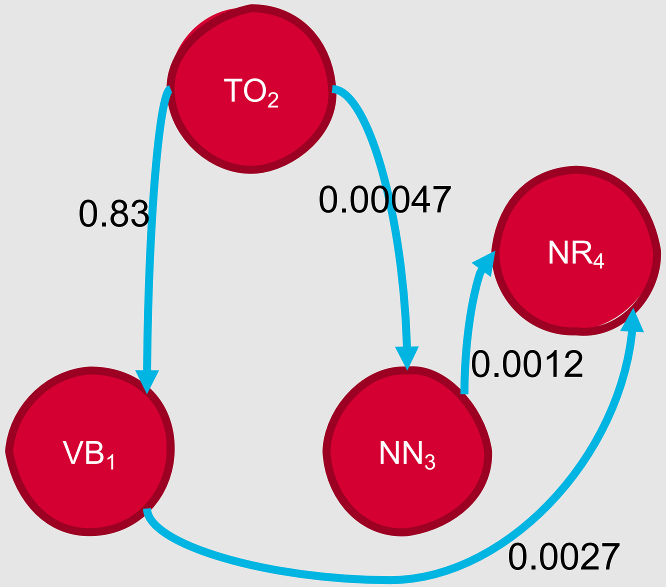
Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	**TO**	**VB**	**NR**
NNP	VBZ	VBN	TO	NN	NR



	fly
VB	0.00012
NN	0.00057

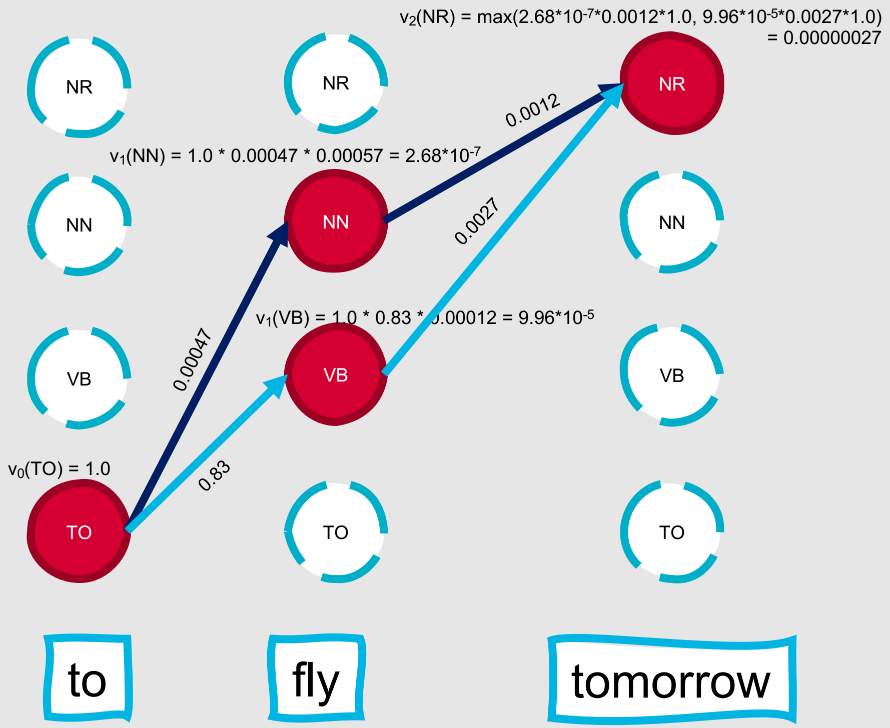
- Now, to decide how to tag "fly," we can consider our two possible sequences:
  - to (TO) fly (VB) tomorrow (NR)
  - to (TO) fly (NN) tomorrow (NR)
- We will select the tag that maximizes the probability:
  - $P(t_i|TO)P(NR|t_i)P(fly|t_i)$
- We determine that:
  - **P(VB|TO)P(NR|VB)P(fly|VB) = 0.83 * 0.0027 * 0.00012 = 0.00000027**
    - Optimal sequence!
  - P(NN|TO)P(NR|NN)P(fly|NN) = 0.00047 * 0.0012 * 0.00057 = 0.00000000032

Superman	is	expected	to	fly	tomorrow
NNP	VBZ	VBN	**TO**	**VB**	**NR**
NNP	VBZ	VBN	TO	NN	NR



	fly
VB	0.00012
NN	0.00057

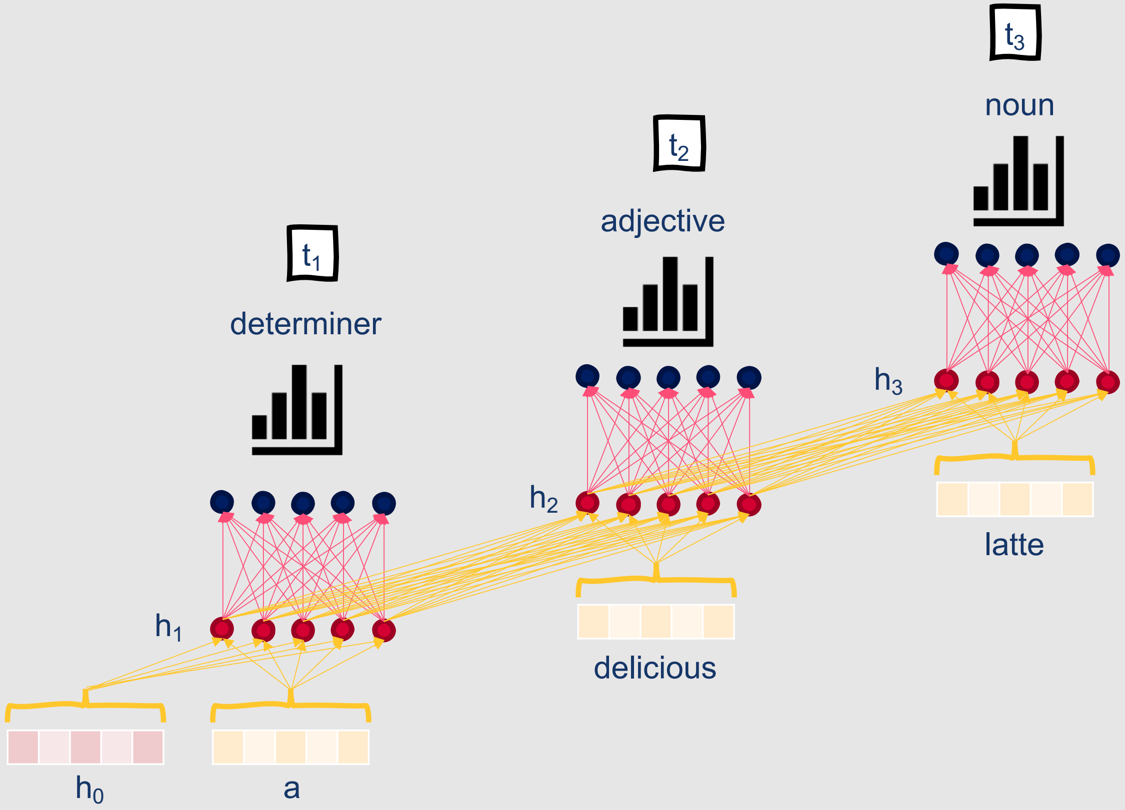
- Visualized in a Viterbi trellis, this would look like:



Example: Bigram HMM Tagger

Neural Sequence Modeling

- Use a sequential or pretrained neural network architecture
    - Recurrent neural networks
    - Transformers
- Predict a label for each item in the input sequence
    - If using a subword vocabulary, you will need to merge the labels predicted for all subwords in a word



How can POS taggers handle unknown words?

- New words are continually added to language, so it is likely that a POS tagger will encounter words not found in its training corpus
- Easy baseline approach: **Assume that unknown words are nouns**
- More sophisticated approach: **Assume that unknown words have a probability distribution similar to other words occurring only once in the training corpus**, and make an (informed) random choice
- Even more sophisticated approach: **Use morphological information** to choose the POS tag (for example, words ending with “ed” tend to be tagged VBN)

Evaluation Metrics for POS Taggers

- Common metrics for POS taggers are:
  - Accuracy
  - Precision
  - Recall
  - F1

Comparison

- The scores computed for these metrics should be compared to alternative POS tagging methods, to place the values in context
    - Is this a good accuracy, or just okay?
- It's good to compare to both a lower-bound baseline and an upper-bound ceiling
    - Baseline: What should your POS tagger definitely perform better than?
        - Most Frequent Class
    - Ceiling: What is the highest possible value for this task?
        - Human Agreement

What factors can impact performance?

- Many factors can lead to your results being higher or lower than expected!
- Some common factors:
  - The size of the training dataset
  - The specific characteristics of your tag set
  - The difference between your training and test corpora
  - The number of unknown words in your test corpus

Summary: Part-of-Speech Tagging

POS tagging is the process of automatically assigning grammatical word classes (parts of speech) to individual tokens

The most common POS tagset is the **Penn Treebank** tagset

Ambiguity is common in natural language, and is a major issue that POS taggers must address

Although POS taggers can be designed using many approaches, statistical (and neural) models are most common