POS tagging

POS Taggers

Mining Unstructured Data
3. POS tagging

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Outline

POS tagging

POS Taggers

1. POS tagging
   - Goal and motivation
   - Part of Speech categories

2. POS Taggers
   - Stochastic taggers
   - Hidden Markov Model
   - Viterbi algorithm

Outline

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

Goal

- Morphological analysis provides lexical information related to forms (POS, num, gen, tense, . . .)
- Multiple analyses can result (POS tags from Penn Treebank tagset)

form	analyses	example of use
fish	NNS	'Cats eat fish'
	VB	'I want to fish'
bass	NN	'I saw you play the bass'
	JJ	'Bass clarinets sound good'

- **Goal:** disambiguate POS of word forms occurring in text

Motivation

Examples of applications of POS tagging:

- Syntactic parsing: words with the same POS tag play a similar syntactic role

  Ex: a determiner followed by a common noun is a noun phrase

- Machine translation

  Ex: (POS tags from Penn Treebank tagset)

'El hombre	**bajo**	toca el	**bajo**	**bajo**	el puente'
POS	NN		NN	NN	
tagging	JJ		JJ	JJ	
	IN		IN	IN	
	VB		VB	VB	
possible	low		bass	under	
English	small			below	
words	short				
	poor				
'The	**small**	man plays the	**bass**	**under**	the bridge'

Outline

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

Open class vs. Closed class

- General classes:
  - **Closed class**: never invent new closed items (functional words)
    Usual subclasses for indo-european languages:
    prepositions, conjunctions, determiners, pronouns, auxiliary verbs or particles (prepositions or adverbs in phrasal verbs)
  - **Open class**: new open items can be invented
    Usual subclasses for indo-european languages:
    nouns, non-auxiliary verbs, adjectives and adverbs
- Each language defines its particular set of subclasses
- Subclasses can be represented with a particular granularity by a set of categories
  Ex: Brown corpus: annotated with 87 different POS tags
  Ex: Penn Treebank corpus: with 48 different POS tags

Penn Treebank tagset

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

12 categories more related to punctuation marks

Ex: to/TO give/VB priority/NN to/IN teacher/NN pay/NN rises/NNS

Outline

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

POS tagging methods

Frequently used methods:

- Rule-based methods:
  - Rules built manually are not frequently used. High production cost
  - Rules learnt automatically from training corpus.

    Ex: Brill's tagger.
- Stochastic methods:
  - Based on Hidden Markov Models learnt automatically from training corpus.

Outline

POS tagging

POS Taggers

Stochastic taggers

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

Stochastic taggers

Goal: Assign the most likely POS-tag sequence to a word sequence.

$W = w_1 \ldots w_n$ (a word sequence)

$T = t_1 \ldots t_n$ (a POS-tag sequence)

Tagger result: $\hat{T} = \underset{T}{\operatorname{argmax}}\, P(T|W)$

1. How is $P(T|W)$ computed?
   Apply a Hidden Markov Model
2. How is $\hat{T}$ found?
   Apply Viterbi algorithm

Outline

POS tagging

POS Taggers

Hidden Markov Model

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

Preliminaries: Markov model

POS tagging
POS Taggers
Hidden Markov Model

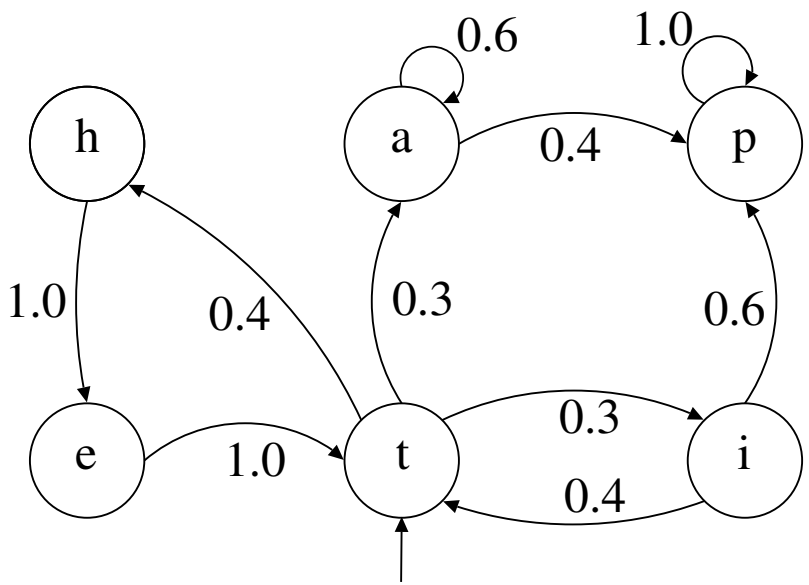
- $X = (X_1, \ldots, X_T)$ sequence of random variables taking values in observed states $S = \{s_1, \ldots, s_N\}$
- Inference: Sequence probability $P(X)$?
- Markov Properties
  - Limited Horizon:
    $P(X_{t+1} = s_k \mid X_1, \ldots, X_t) = P(X_{t+1} = s_k \mid X_t)$
  - Time Invariant (Stationary):
    $P(X_{t+1} = s_i \mid X_t = s_j) = P(X_2 = s_i \mid X_1 = s_j)$
- Transition matrix:
  $a_{ij} = P(X_{t+1} = s_j \mid X_t = s_i); \quad \forall i, j \; a_{ij} \geqslant 0; \quad \forall i \; \sum_{j=1}^{N} a_{ij} = 1$
- Initial probabilities (or extra state s_0):
  $\pi_i = P(X_1 = s_i); \quad \sum_{i=1}^{N} \pi_i = 1$

Preliminaries: Markov model

Sequence probability: (Bayesian rule+limited horizon)

$$P(X_1, .., X_T) =$$
$$= P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1X_2) \ldots P(X_T \mid X_1..X_{T-1})$$
$$= P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2) \ldots P(X_T \mid X_{T-1})$$
$$= \pi_{X_1} \prod_{t=2}^{T} a_{X_{t-1}X_t}$$

Example:



$$P(t, h, e, t, i, p, p) = 1 \cdot (0.4 \cdot 1 \cdot 1 \cdot 0.3 \cdot 0.6 \cdot 1) = 0.072$$

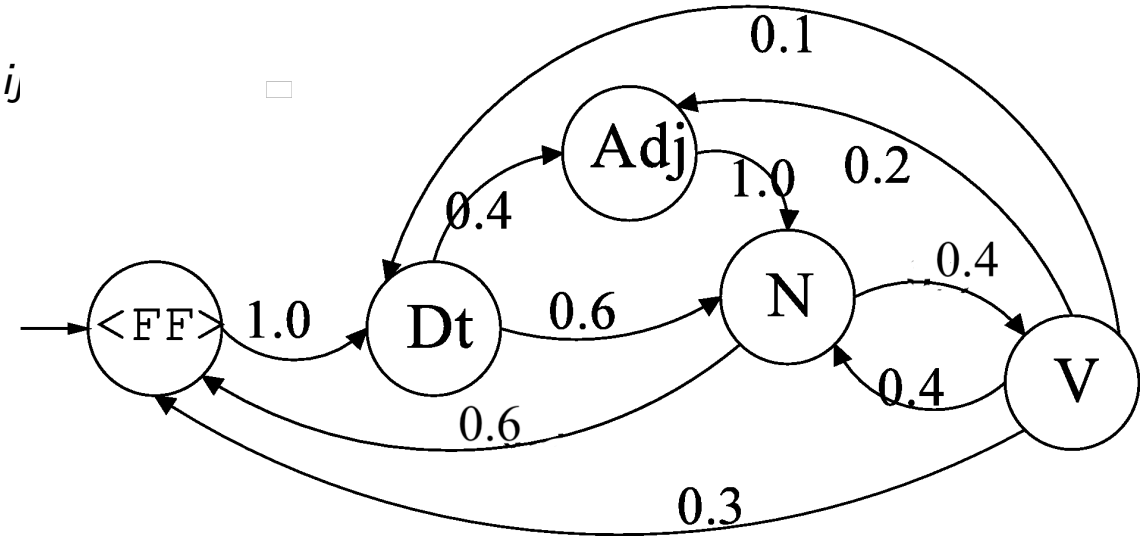
Hidden Markov model

- $X = (X_1, \ldots, X_T)$ sequence of random variables taking values in unobserved [hidden] states $S = \{s_1, \ldots, s_N\}$ given a sequence of observations $O = (O_1, \ldots, O_T)$
- Inference: Probability of ...
  - a process: $P(O)$?
  - the state of a process at the end: $P(X_T \mid O)$?
  - the explanation of a process: $P(X_1, \ldots, X_T \mid O)$?
    POS tagging: X = POS tags; O = words
- Transition matrix:
  $a_{ij} = P(X_{t+1} = s_j \mid X_t = s_i); \quad \forall i,j \; a_{ij} \geqslant 0; \quad \forall i \; \sum_{j=1}^{N} a_{ij} = 1$
- Initial probabilities (or extra state s_0):
  $\pi_i = P(X_1 = s_i); \quad \sum_{i=1}^{N} \pi_i = 1$
- Emission Probability:
  $b_{ik} = P(O_t = k \mid X_t = s_i) \quad \forall i,k \; b_{ik} \geqslant 0; \quad \forall i \; \sum_{k=1}^{N} b_{ik} = 1$

Hidden Markov model

Example (horizon=1; bigrams)

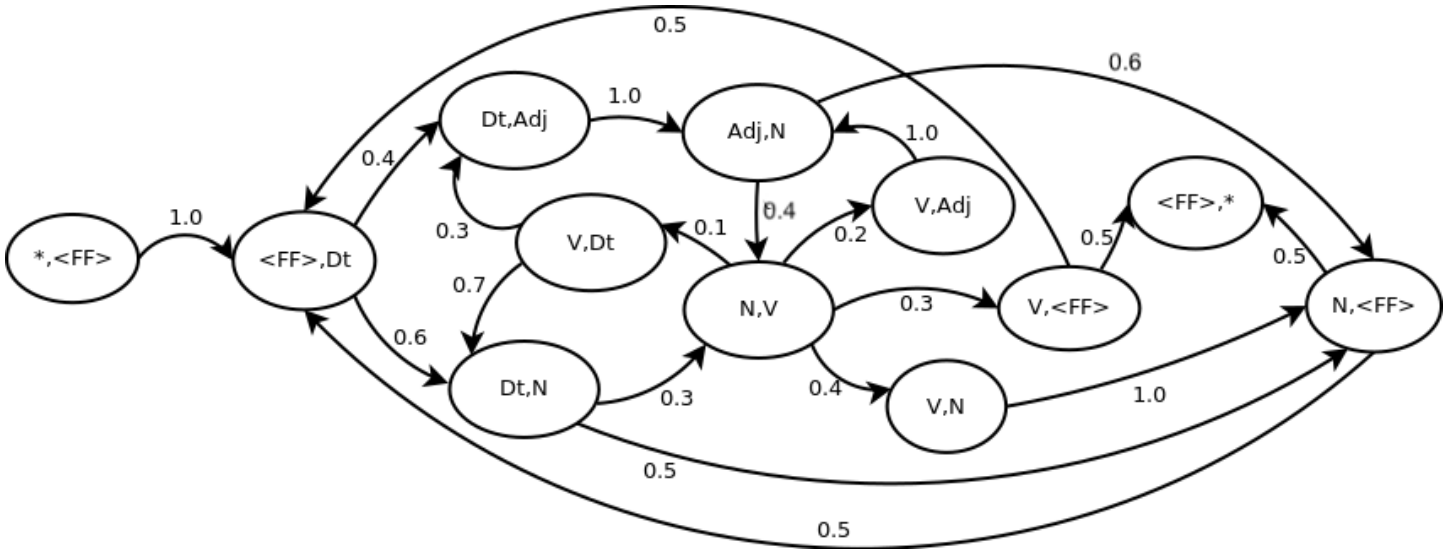
a_{ij}



b_{ik}	.	the	this	cat	kid	eats	runs	fish	fresh	little	big
<FF>	1.0										
Dt		0.6	0.4								
N				0.3	0.1		0.1	0.3	0.2		
V					0.1	0.5	0.3	0.1			
Adj					0.1				0.2	0.3	0.4

Hidden Markov model

Example (horizon=2; trigrams)



b_{ik}	.	the	this	cat	kid	eats	runs	fish	fresh	little	big
?,<FF>	1.0										
?,Dt		0.6	0.4								
?,N				0.3	0.1		0.1	0.3	0.2		
?,V					0.1	0.5	0.3	0.1			
?,Adj					0.1				0.2	0.3	0.4

Learning of parameters

- Parameters a_{ij}, b_{ik} and π_i can be estimated over a training corpus C
- Use smoothing techniques
- Use Baum-Welch algorithm
- **learning of parameters is out of this course**

Learning of parameters

Example: MLE estimator; assume u, v, w are different POS tags in the training corpus

- bigram-based HMM

$$a(u,v) \approx P_{MLE}(v \mid u) = \frac{c(u,v)}{c(u)}$$

$$b(O_i,u) \approx P_{MLE}(O_i \mid u) = \frac{c(u,O_i)}{c(u)}$$

$$\pi(u) \approx P_{MLE}(u \mid *) = \frac{c(*,u)}{c(*)}$$

- trigram-based HMM

$$a(uv,vw) \approx P_{MLE}(vw \mid uv) = \frac{c(u,v,w)}{c(u,v)}$$

$$b(O_i,uv) = b(O_i,v) \approx P_{MLE}(O_i \mid v) = \frac{c(v,O_i)}{c(v)}$$

$$\pi(*u) \approx P_{MLE}(*u \mid **) = \frac{c(*,*,u)}{c(**)} \quad \pi(uv) \approx P_{MLE}(uv \mid *u) = \frac{c(*,u,v)}{c(*u)}$$

Exercise

Given the following corpus,

horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN goes/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

apply MLE to estimate the non-zero parameters for the POS-tags involved in the sentence using bigrams:

"time flies like horse flies ."

Exercise

horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*						
NN						
NNS						
VBP						
VBZ						
IN						

Exercise

* horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
* eat/VB breakfast/NN at/IN morning/NN time/NN ./.
* take/VB time/NN with/IN arrow/NN projects/NNS ./.
* dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
* flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
* bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*						
NN						
NNS						
VBP						
VBZ						
IN						

Exercise

* horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
  eat/VB breakfast/NN at/IN morning/NN time/NN ./.
  take/VB time/NN with/IN arrow/NN projects/NNS ./.
* dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
  flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
  bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2					
NN						
NNS						
VBP						
VBZ						
IN						

Exercise

horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

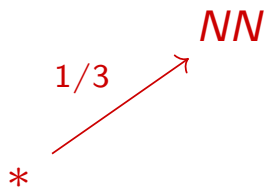
"time flies like horse flies ."

POS tagging
POS Taggers
Hidden Markov Model

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2					
NN						
NNS						
VBP						
VBZ						
IN						



Exercise

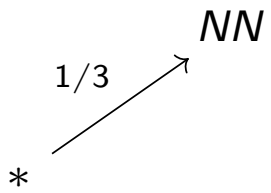
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
* flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
* bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN						
NNS						
VBP						
VBZ						
IN						



Exercise

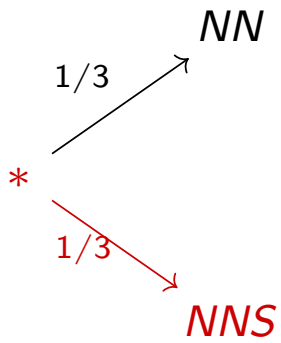
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN						
NNS						
VBP						
VBZ						
IN						



Exercise

horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

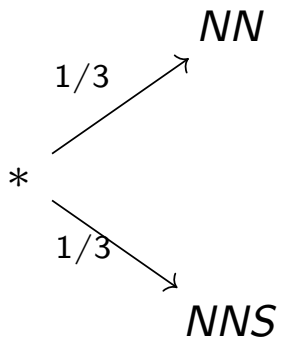
"time flies like horse flies ."

POS tagging
POS Taggers
Hidden Markov Model

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN						
NNS						
VBP						
VBZ						
IN						



Exercise

horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

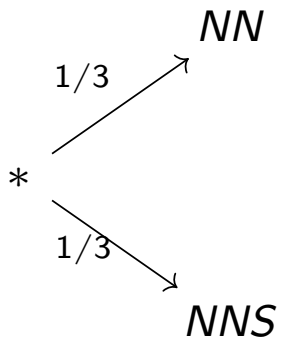
"time flies like horse flies ."

POS tagging
POS Taggers
Hidden Markov Model

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3					
NNS						
VBP						
VBZ						
IN						



Exercise

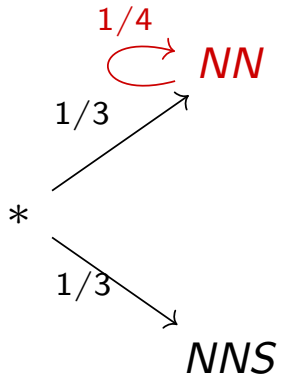
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2		2			
NN	3					
NNS						
VBP						
VBZ						
IN						



Exercise

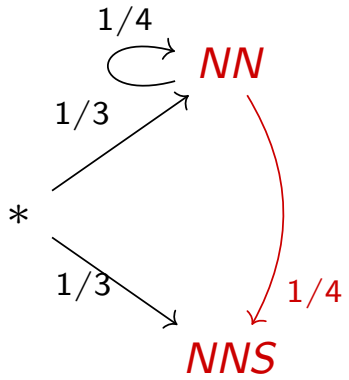
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3				
NNS						
VBP						
VBZ						
IN						



Exercise

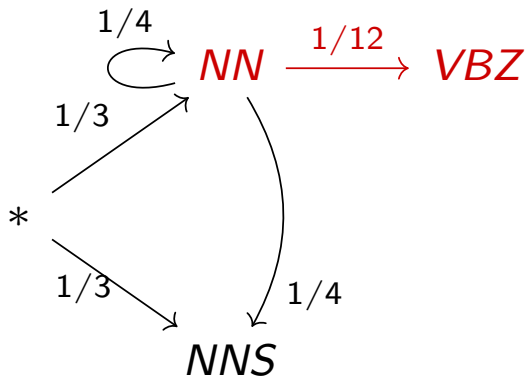
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3		1		
NNS						
VBP						
VBZ						
IN						



Exercise

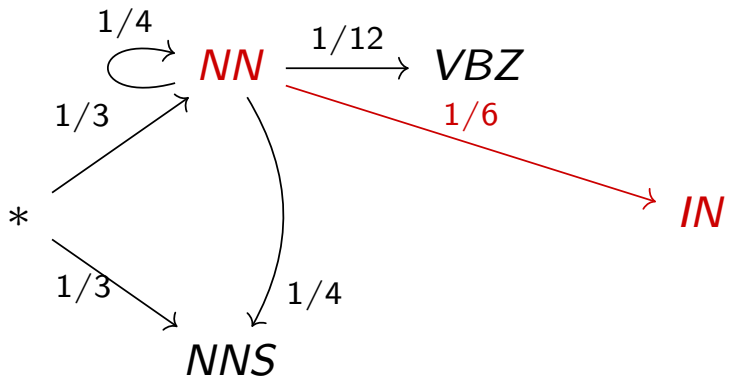
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3		1	2	
NNS						
VBP						
VBZ						
IN						



Exercise

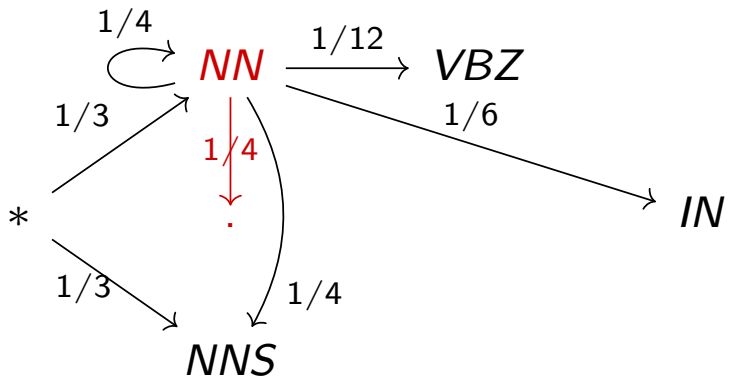
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	
VBP	
VBZ	
IN	
.	

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3		1	2	3
NNS						
VBP						
VBZ						
IN						



Exercise

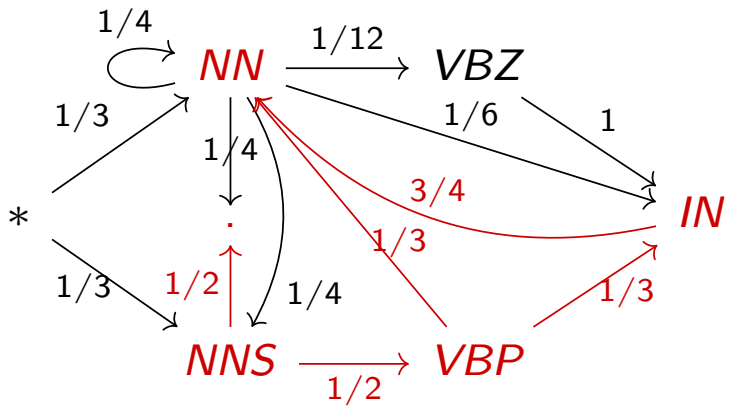
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	6
VBP	3
VBZ	1
IN	4
.	6

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3		1	2	3
NNS			3			3
VBP	1				1	
VBZ					1	
IN	3					



Exercise

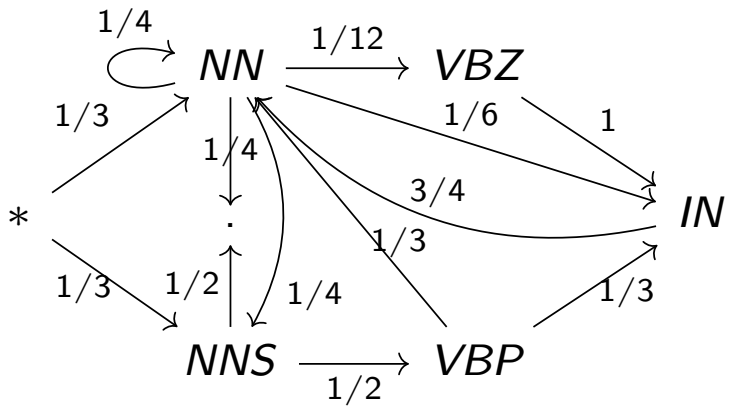
horse/NN flies/NNS time/VBP morning/NN rays/NNS ./.
eat/VB breakfast/NN at/IN morning/NN time/NN ./.
take/VB time/NN with/IN arrow/NN projects/NNS ./.
dinner/NN time/NN flies/VBZ before/IN sleep/NN ./.
flies/NNS smell/VBP an/DT arrow/NN drink/NN ./.
bees/NNS sting/VBP like/IN some/DT flies/NNS ./.

"time flies like horse flies ."

Joint transition matrix A and initial probabilities π:

c(u)	
*	6
NN	12
NNS	6
VBP	3
VBZ	1
IN	4
.	6

c(u,v)	NN	NNS	VBP	VBZ	IN	.
*	2	2				
NN	3	3		1	2	3
NNS			3			3
VBP	1				1	
VBZ					1	
IN	3					



Emission matrix B:

c(u)	
*	6
NN	12
NNS	6
VBP	3
VBZ	1
IN	4
.	6

c(u,O_i)	time	flies	like	horse	.
NN	3			1	
NNS		3			
VBP	1				
VBZ		1			
IN			1		
.					6

$b_{i,k}$	time	flies	like	horse.	.
NN	1/4			1/12	
NNS		1/2			
VBP	1/3				
VBZ		1			
IN			1/4		
.					1

How is the prob. of a POS-tag sequence computed?

Explanation probability:

Generative model (joint probabilities) instead of conditional model

$$P(X \mid O) = \frac{P(X,O)}{P(O)} \approx P(X,O)$$

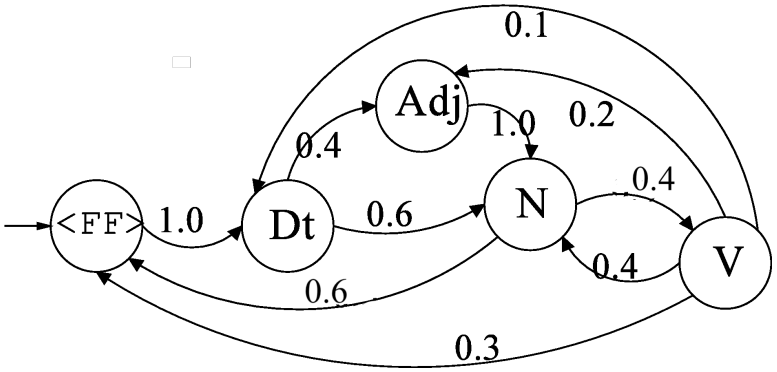
$$P(X_1,..,X_T,O) = P(X_1,..,X_T) \cdot P(O \mid X_1 \ldots X_T)$$

$$P(X_1,..,X_T) = \pi_{X_1} \prod_{t=2}^{T} a_{X_{t-1}X_t}$$
$$P(O \mid X_1 \ldots X_T) = \prod_{t=1}^{T} b_{O_t X_t}$$

$$P(X_1,..,X_T,O) = \pi_{X_1} \cdot b_{O_1 X_1} \cdot \prod_{t=2}^{T} a_{X_{t-1}X_t} \cdot b_{O_t X_t}$$

How is the prob. of a POS-tag sequence computed?

Following the previous example



b_{ik}	.	this	cat	eats	fish	...
<FF>	1.0					
Dt		0.4				
N			0.3		0.3	
V				0.5	0.1	
Adj						

$P(X, O) = P(X, ., this, cat, eats, fish)$? 7 possible X sequences

X =<FF>,Dt,Adj,N,<FF>
X =<FF>,Dt,Adj,N,V
X =<FF>,Dt,N,<FF>,Dt
X =<FF>,Dt,N,V,<FF>
X =<FF>,Dt,N,V,N

$P(X,O) = (1 \cdot 1) \cdot (1 \cdot 0.4) \cdot (0.6 \cdot 0.3) \cdot (0.4 \cdot 0.5) \cdot (0.4 \cdot 0.3) = 0.001728$

X =<FF>,Dt,N,V,Adj
X =<FF>,Dt,N,V,Dt

How is the best POS-tag sequence found?

We want to find

$$\hat{X} = \underset{X}{\operatorname{argmax}} P(X \mid O) \approx \underset{X}{\operatorname{argmax}} P(X, O)$$

- Brute force, $O(N^T)$

  N states (POS tags) and T observations (word sequence length)
- Viterbi algorithm, dinamic programming, $O(T * N^2)$

Outline

POS tagging

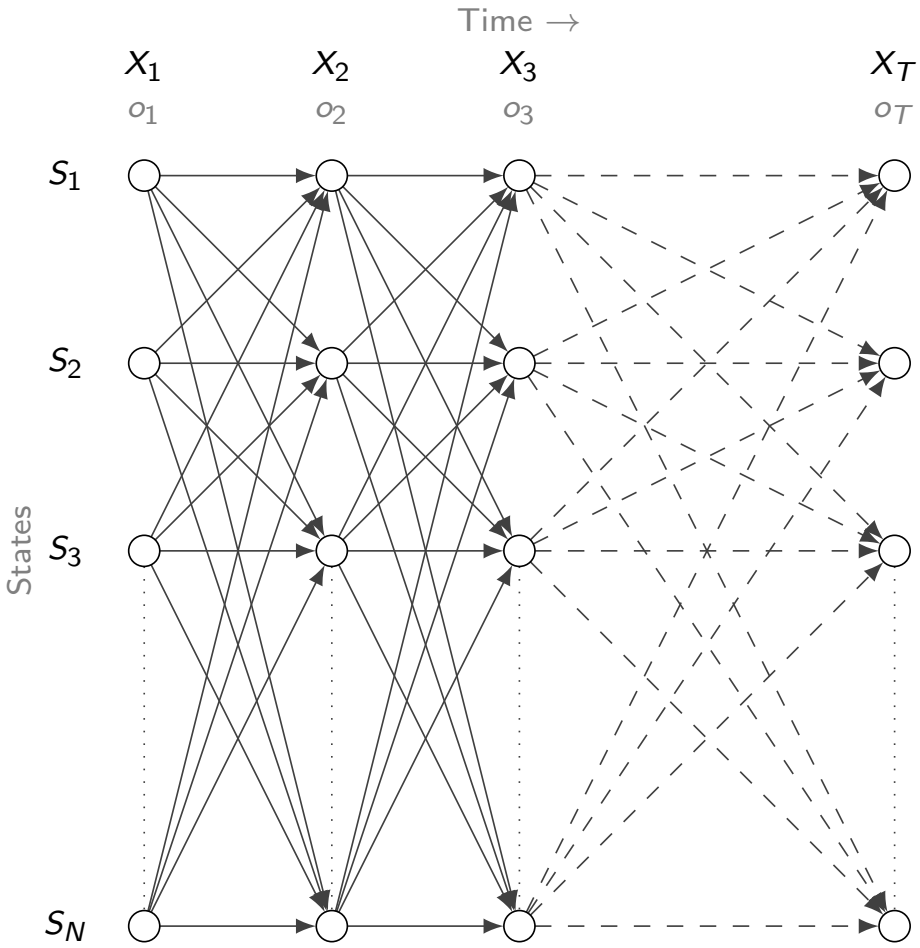
POS Taggers

Viterbi algorithm

1 POS tagging
- Goal and motivation
- Part of Speech categories

2 POS Taggers
- Stochastic taggers
- Hidden Markov Model
- Viterbi algorithm

Viterbi algorithm

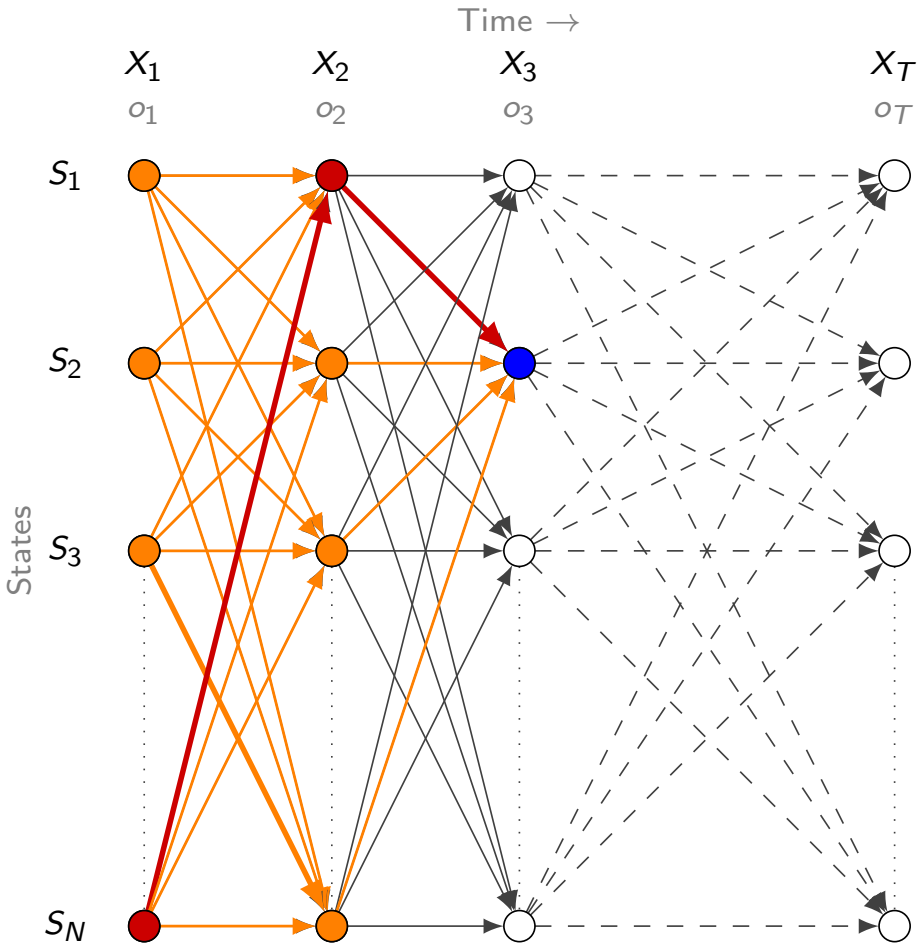


Auxiliary structure:

TRELLIS of a fully connected HMM

Node (S_i, X_t) stores information related to state S_i about the partial sequence from X_1 to X_t: $\delta_t(i)$ and $\varphi_t(i)$

Viterbi algorithm



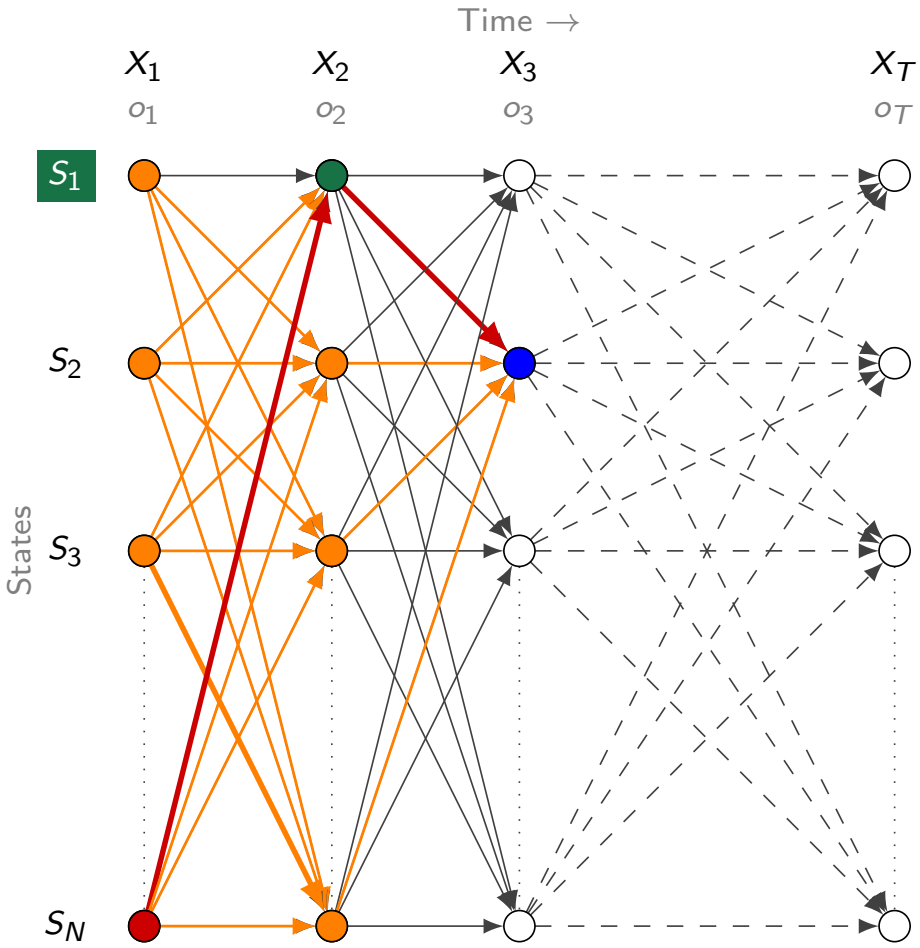
Auxiliary structure:

TRELLIS of a fully connected HMM

Node (S_i, X_t) stores information related to state S_i about the partial sequence from X_1 to X_t: $\delta_t(i)$ and $\varphi_t(i)$

$$\delta_t(i) = \max_{X_1,\ldots,X_{t-1}} P(X_1, \ldots, X_{t-1}, X_t = S_i, O)$$

Viterbi algorithm



Auxiliary structure:

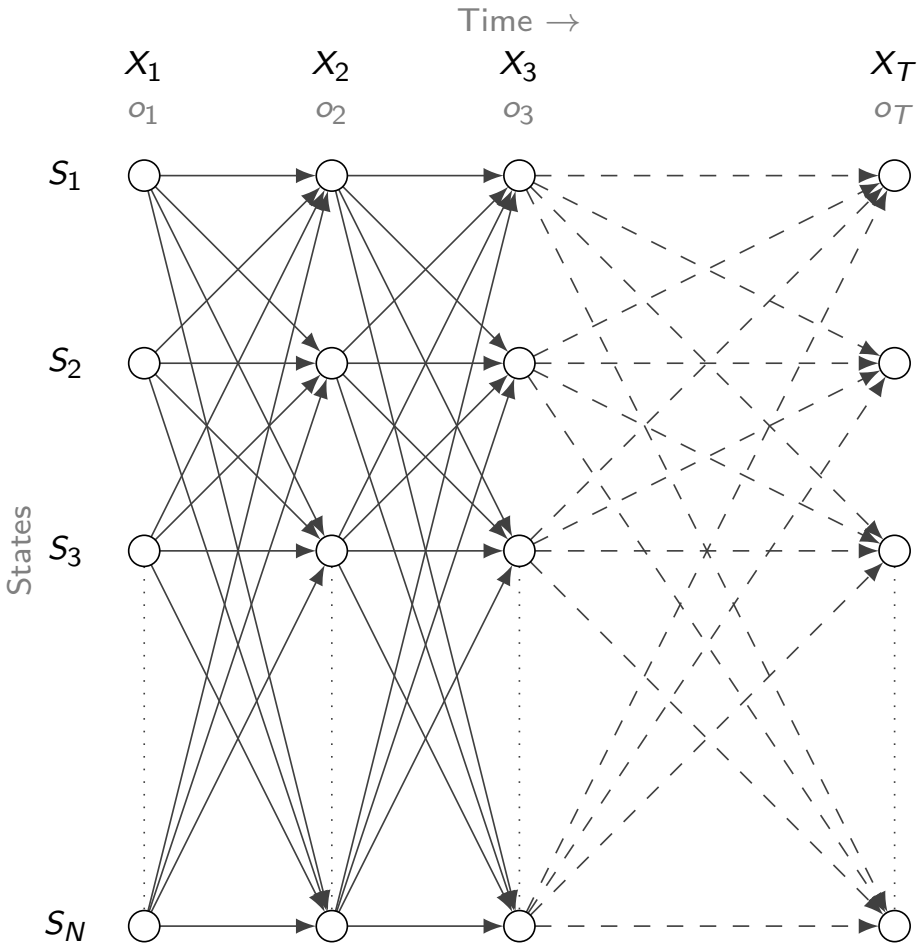
TRELLIS of a fully connected HMM

Node (S_i, X_t) stores information related to state S_i about the partial sequence from X_1 to X_t: $\delta_t(i)$ and $\varphi_t(i)$

$$\delta_t(i) = \max_{X_1,\ldots,X_{t-1}} P(X_1,\ldots,X_{t-1}, X_t = S_i, O)$$

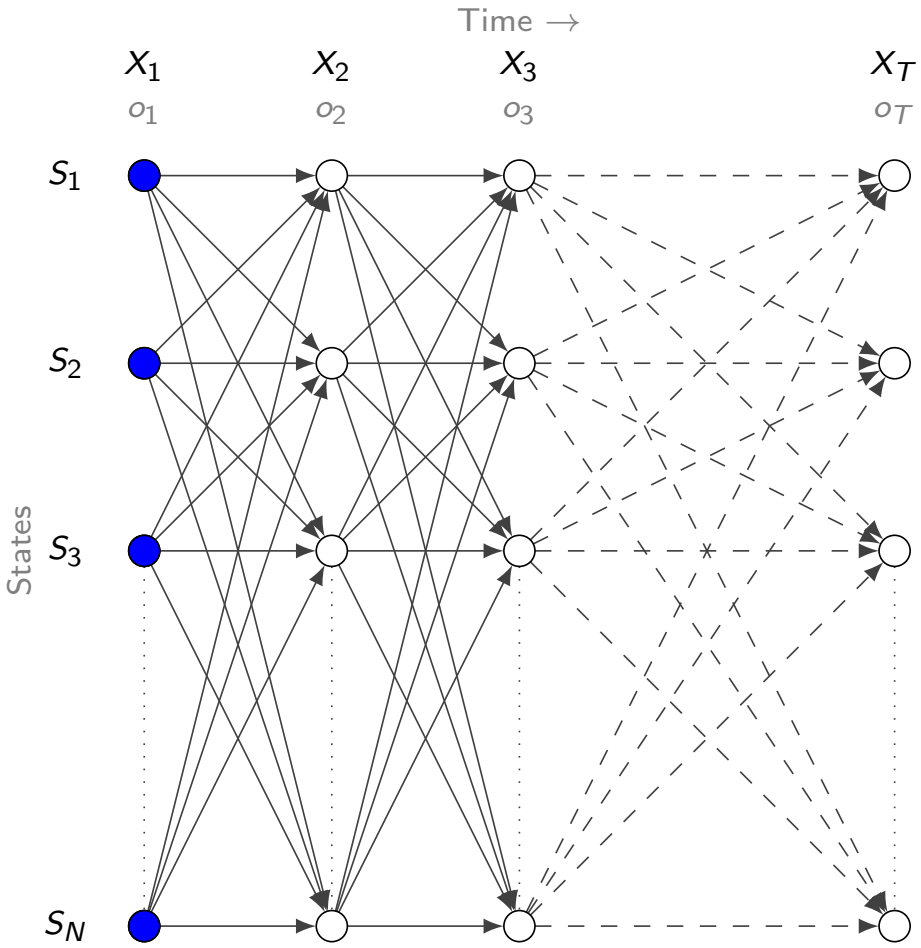
$$\varphi_t(i) = \mathit{last}\left(\operatorname*{argmax}_{X_1,\ldots,X_{t-1}} P(X_1,\ldots,X_{t-1}, X_t = S_i, O)\right)$$

Viterbi algorithm



Algorithm:

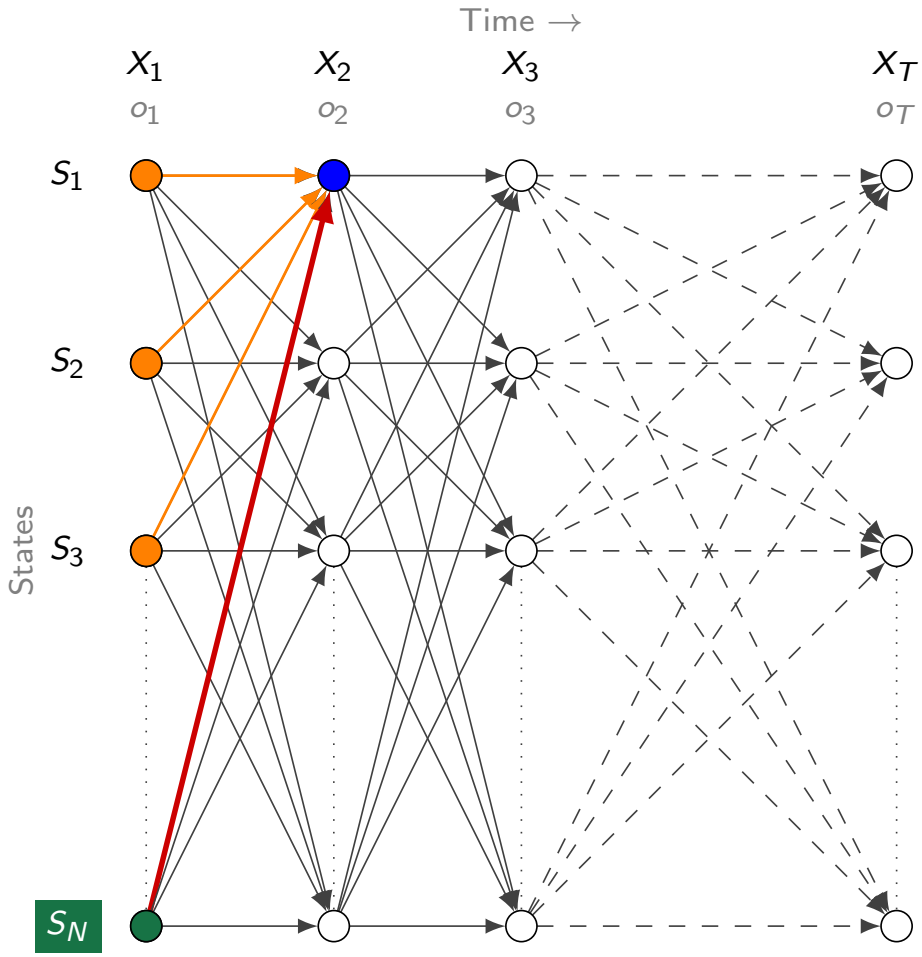
Viterbi algorithm



Algorithm:

1. Initialization step:
   $\forall j = 1 \ldots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

Viterbi algorithm



Algorithm:

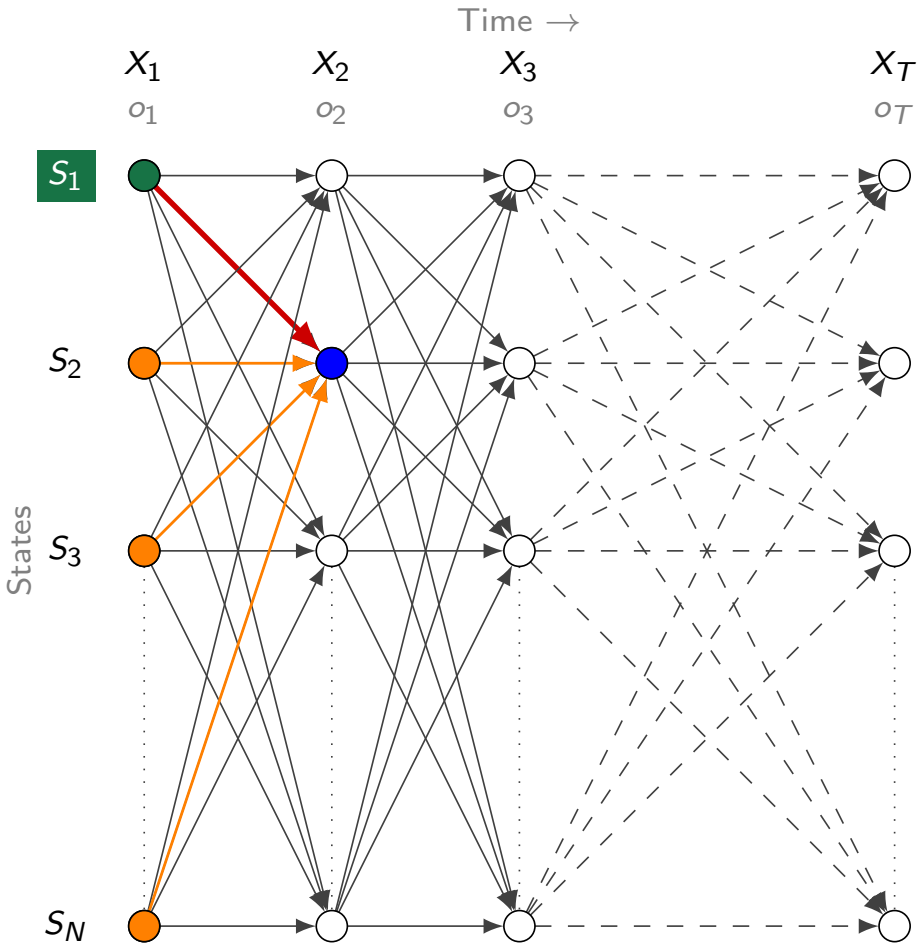
1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

Viterbi algorithm



Algorithm:

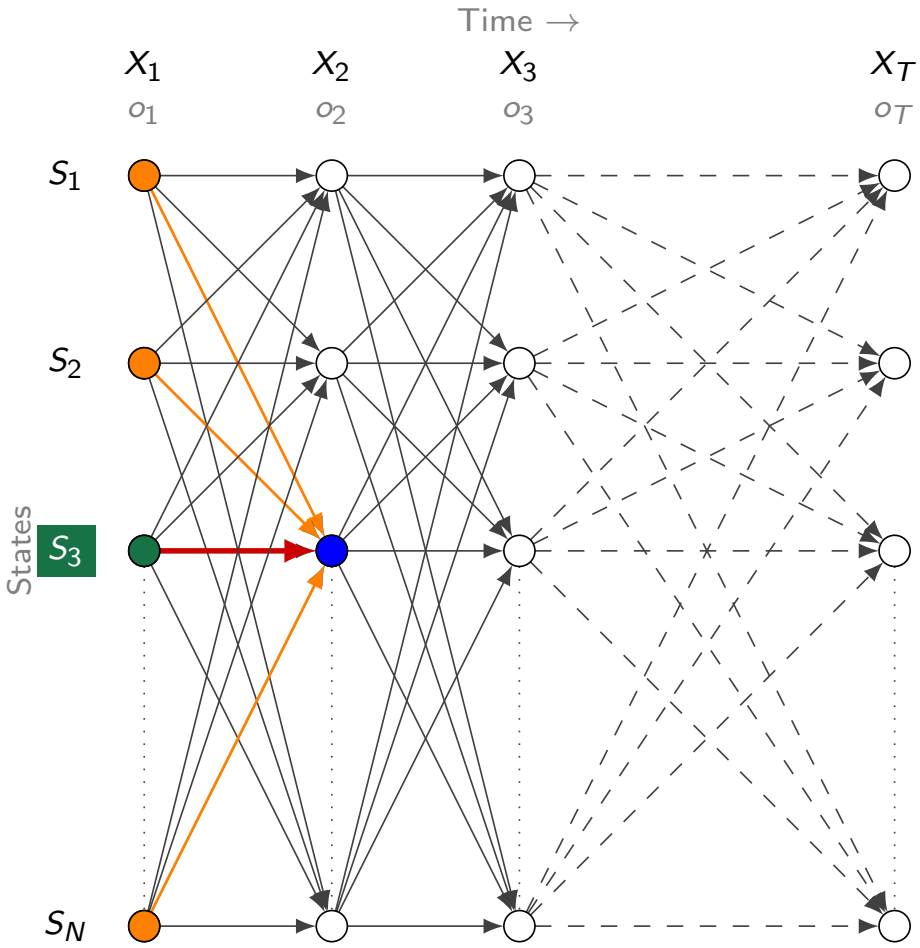
1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname*{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

Viterbi algorithm



Algorithm:

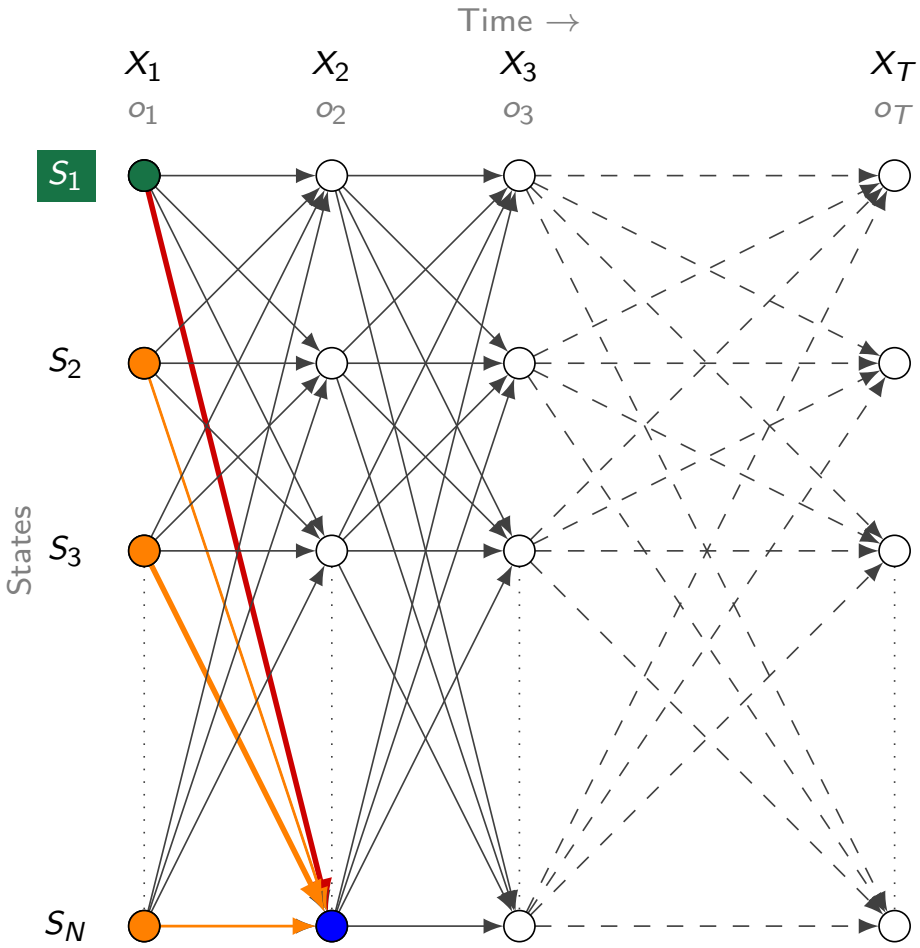
1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname*{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

Viterbi algorithm



Algorithm:

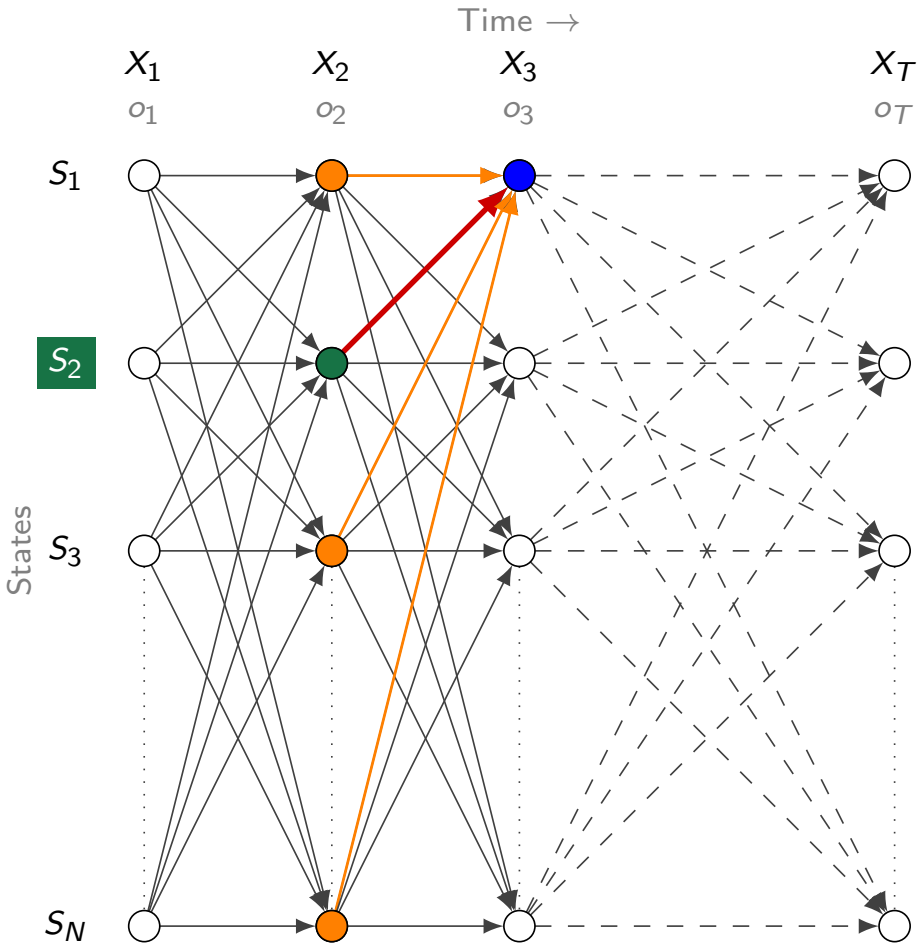
1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname*{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

Viterbi algorithm



Algorithm:

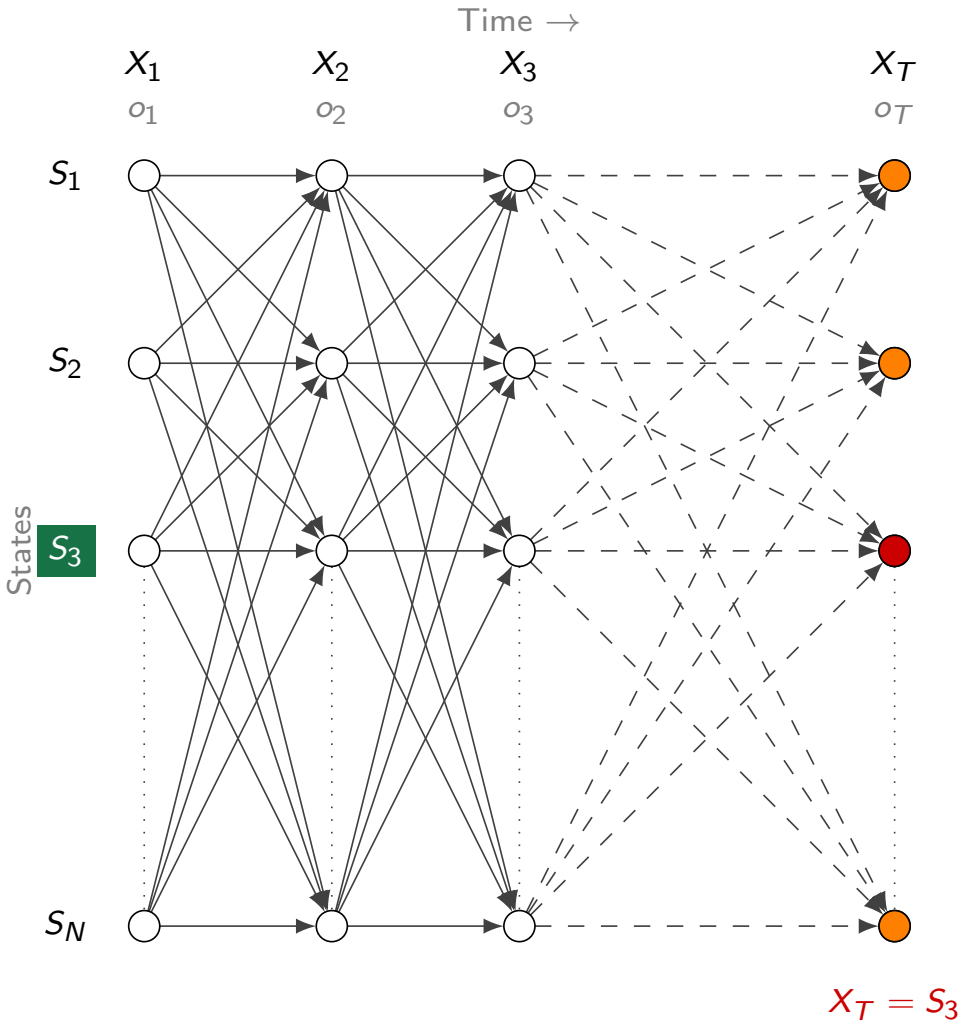
1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname*{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

Viterbi algorithm



Algorithm:

1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

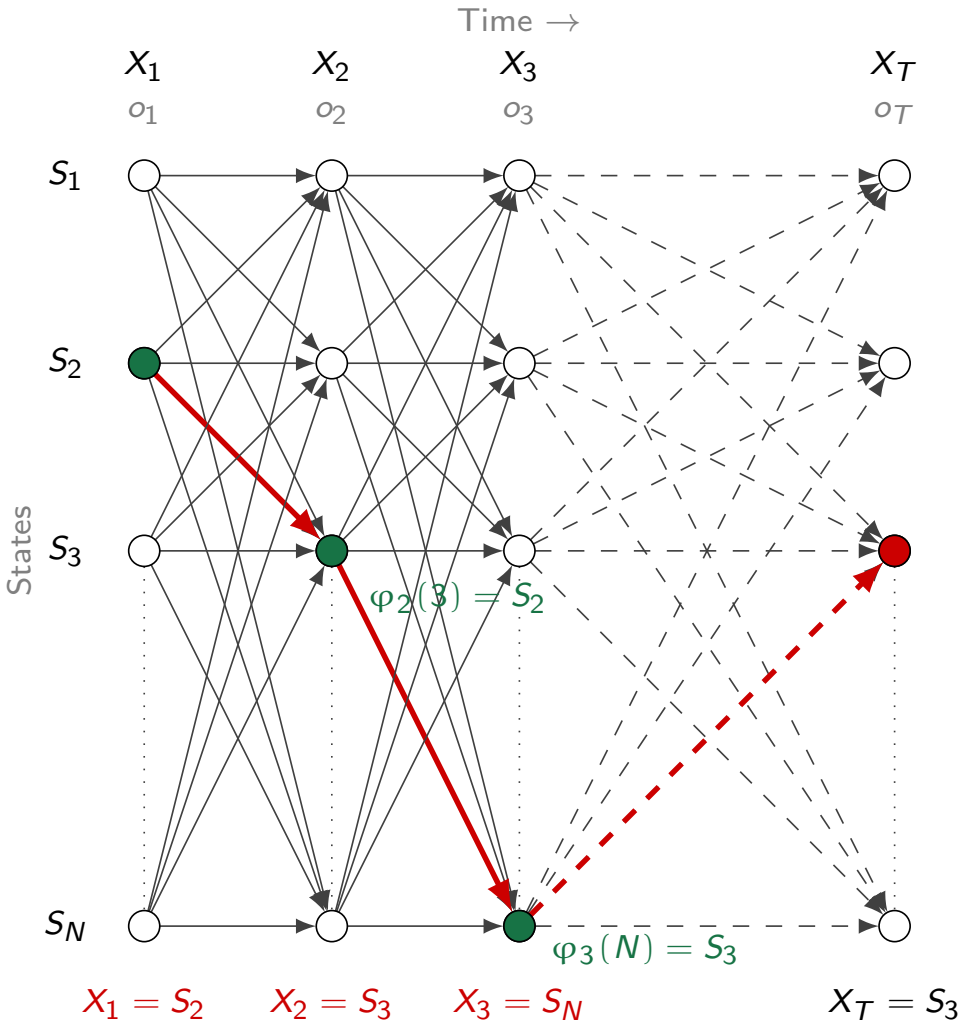
2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

3. Termination step:
   $X_T = \operatorname{argmax}_i(\delta_T(i))$

Viterbi algorithm



Algorithm:

1. Initialization step:
   $\forall j = 1 \dots N :$
   $\delta_1(j) = \pi_j * b_{j,o_1}$

2. Inference step:
   $\forall t = 2 \dots T : \quad \forall j = 1 \dots N :$

$$\delta_t(j) = \max_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$$

$$\varphi_t(j) = \operatorname{argmax}_{1 \leqslant i \leqslant N} (\delta_{t-1}(i) * a_{i,j})$$

3. Termination step:
   $X_T = \operatorname{argmax}_i(\delta_T(i))$

4. Backward path readout step:
   $\forall t = 1 \dots T - 1 :$
   $X_t = \varphi_{t+1}(X_{t+1})$

Exercise

Apply Viterbi algorithm using the following HMM to

The	kid	fishes	fish
DT	NN	NNS	NN
			NNS
	JJ	VBZ	VBP

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

B	the	big	kid	fish	time	fishes	times
DT	1						
JJ		0.8	0.2				
NN			0.3	0.4	0.3		
NNS				0.3		0.4	0.3
VBZ						0.6	0.4
VBP				0.7	0.3		

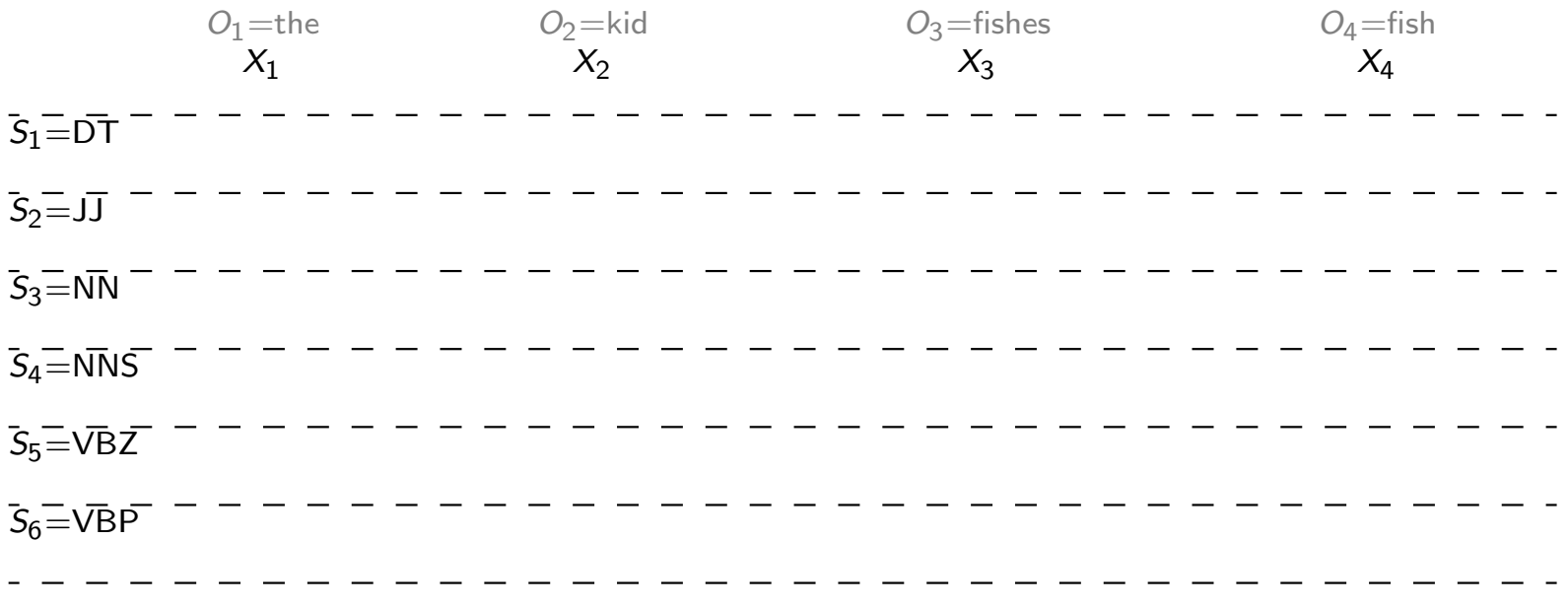
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi:



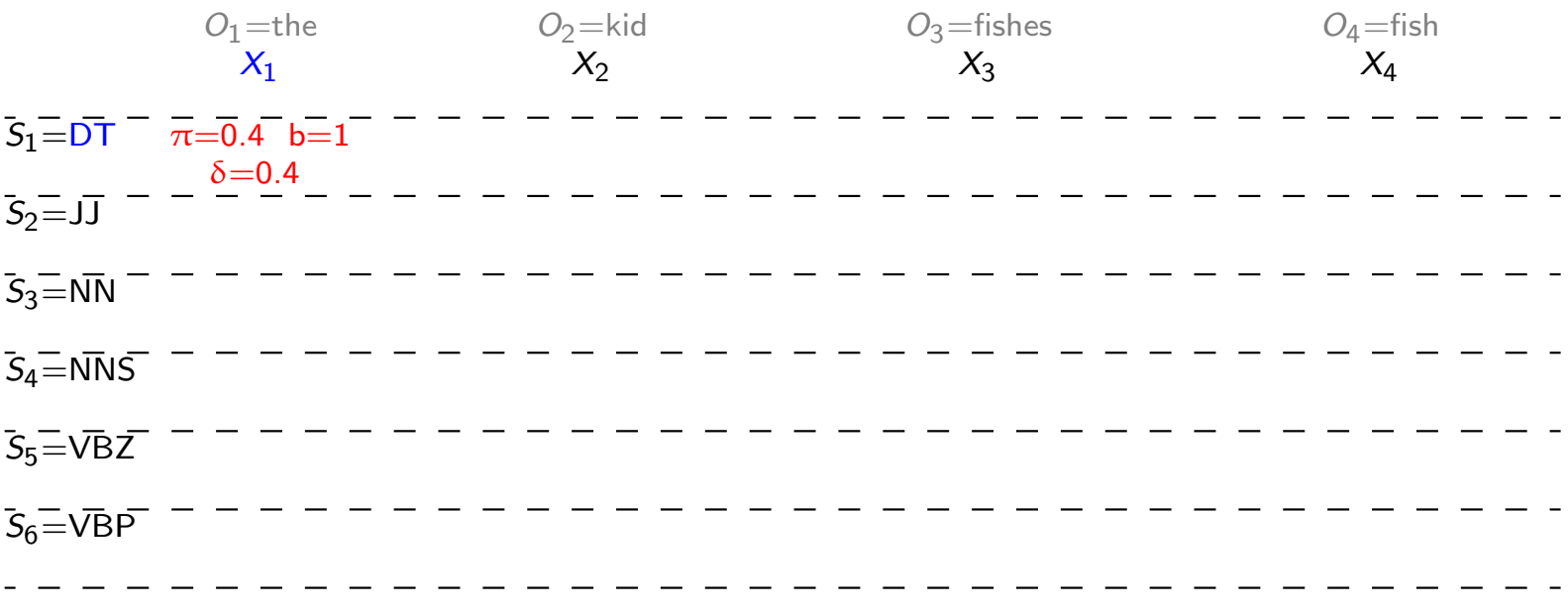
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



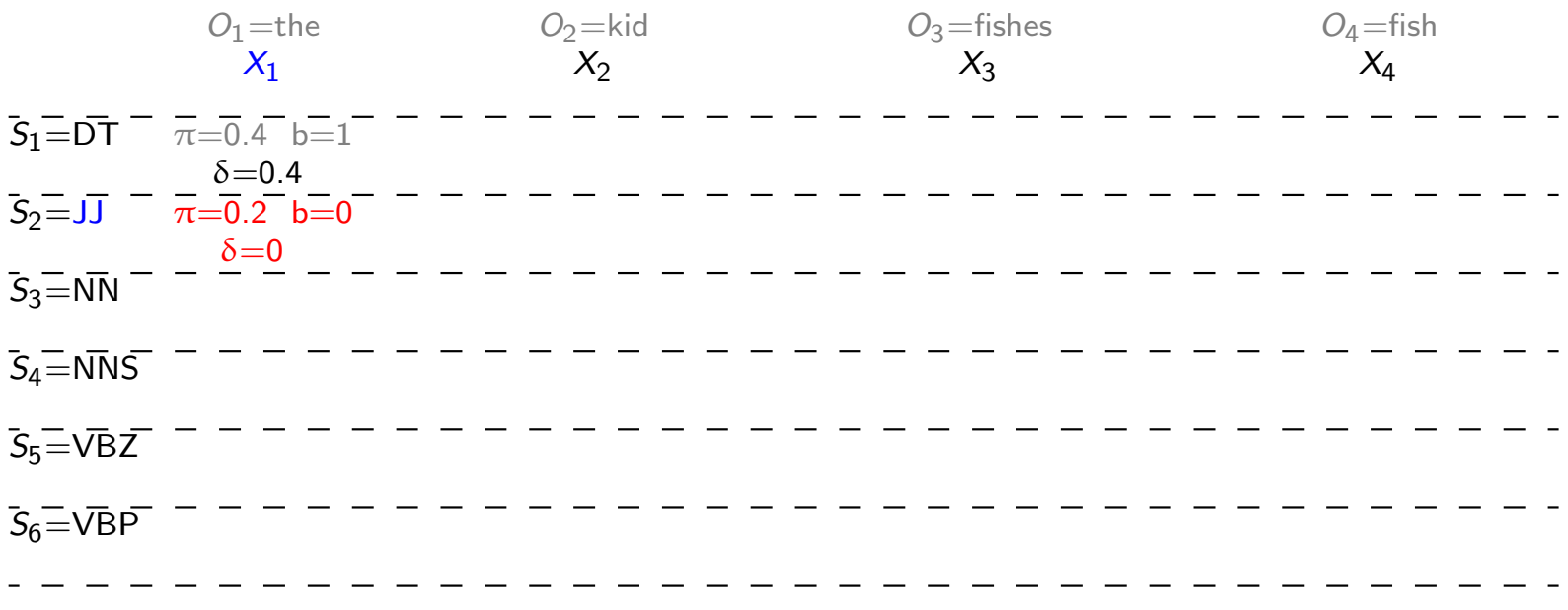
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ	0	0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



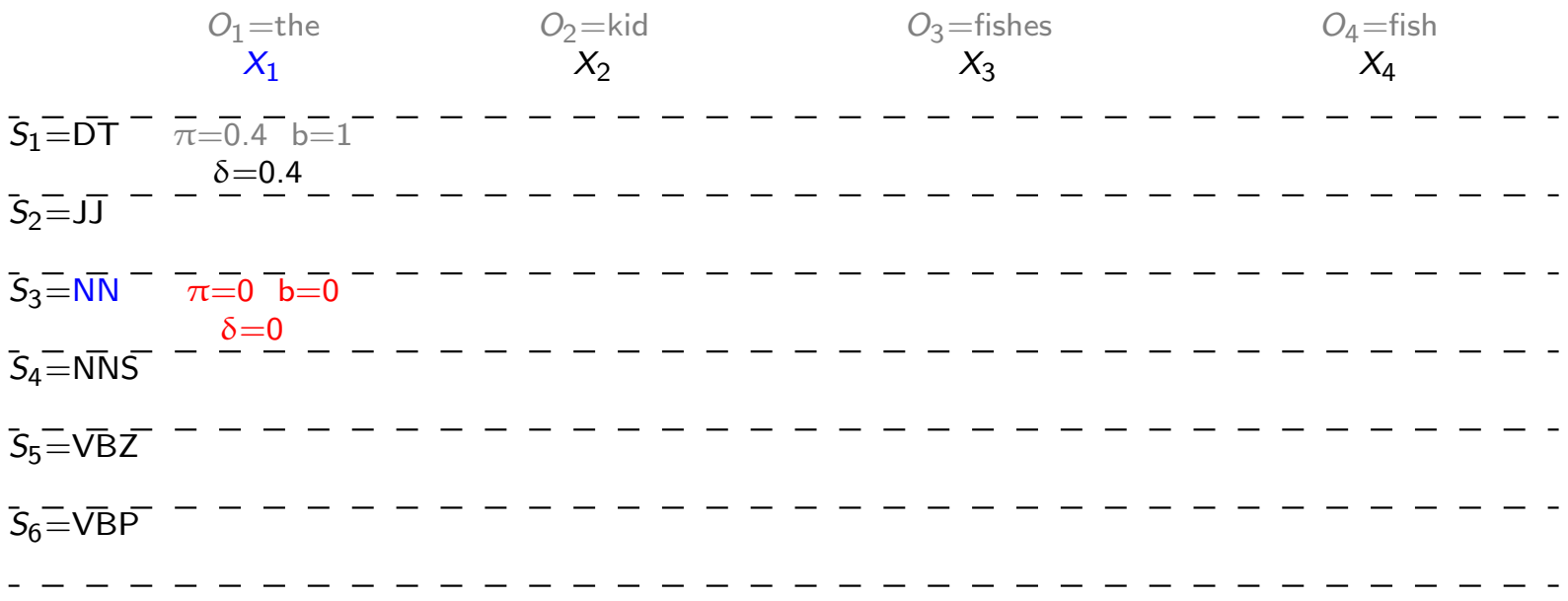
Exercise

π	
DT	0.4
JJ	0.2
NN	0
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN	0	0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



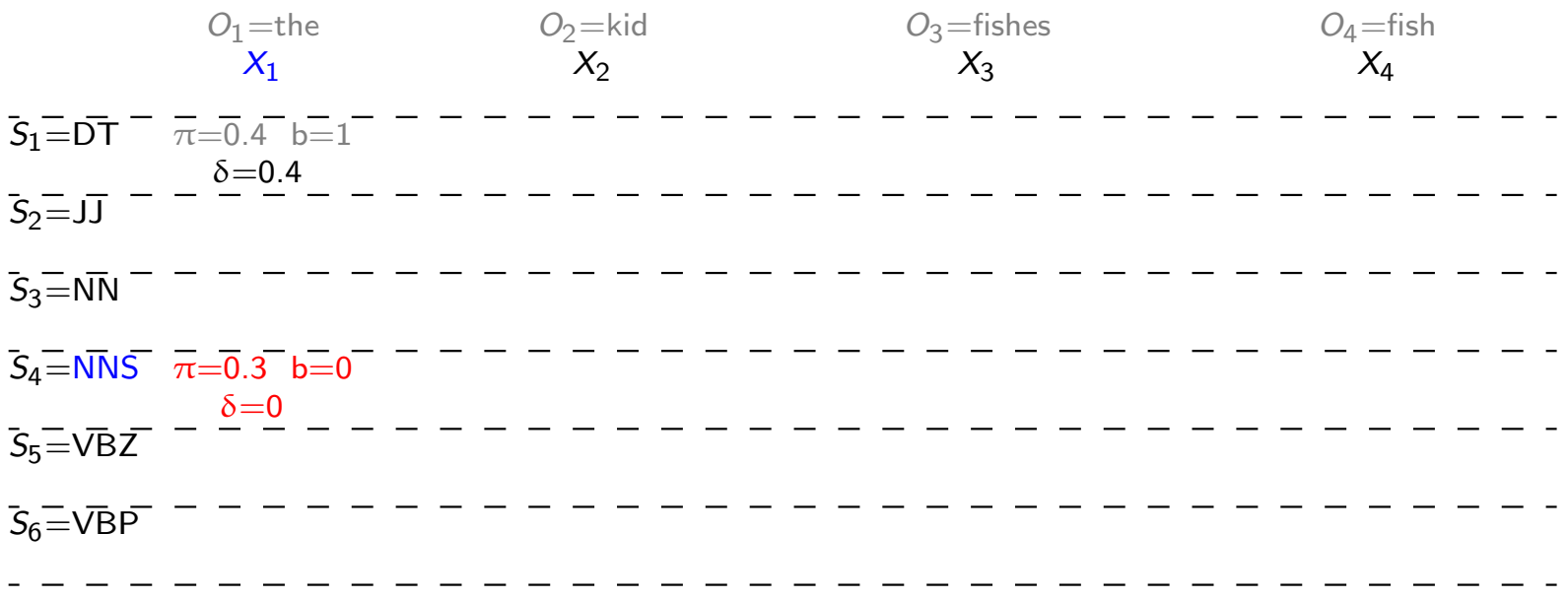
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS	0		0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



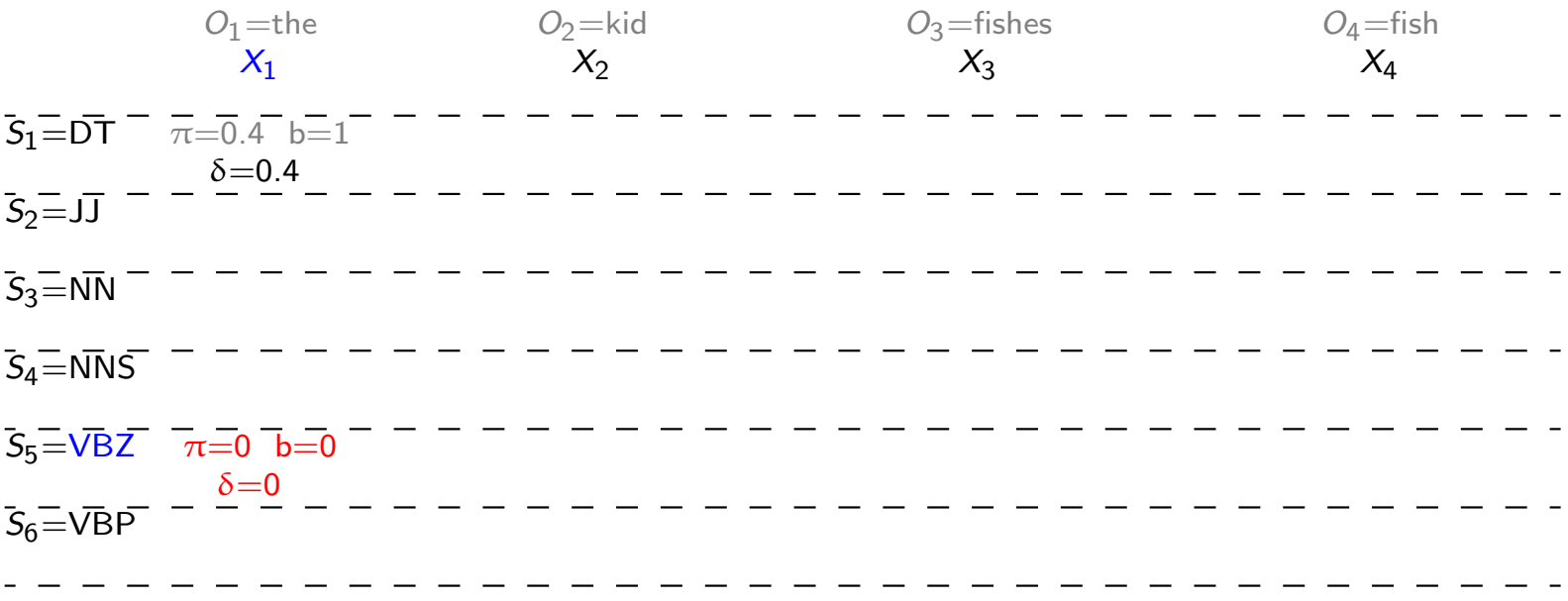
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	0
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ	0			0.6
VBP			0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



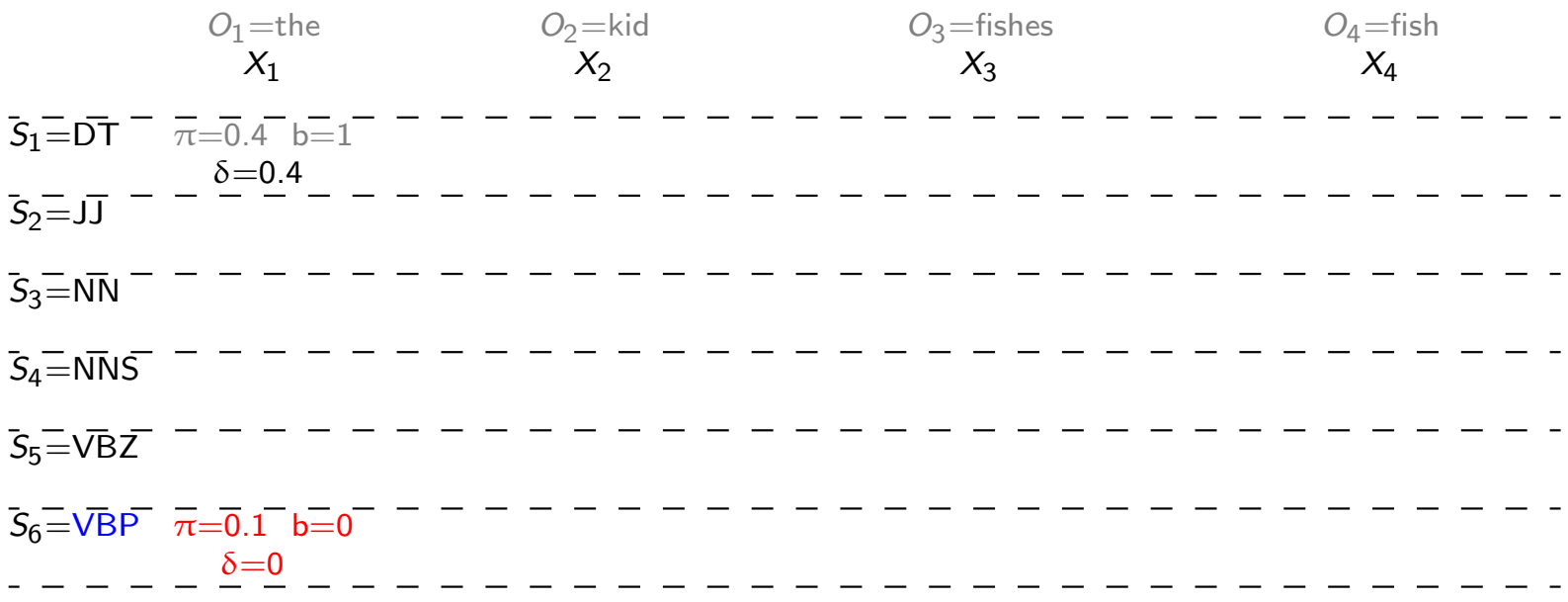
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP	0		0.7	

Trellis from Viterbi: Initialization step: $\delta_1(i) = \pi_i * b_{i,o_1}$



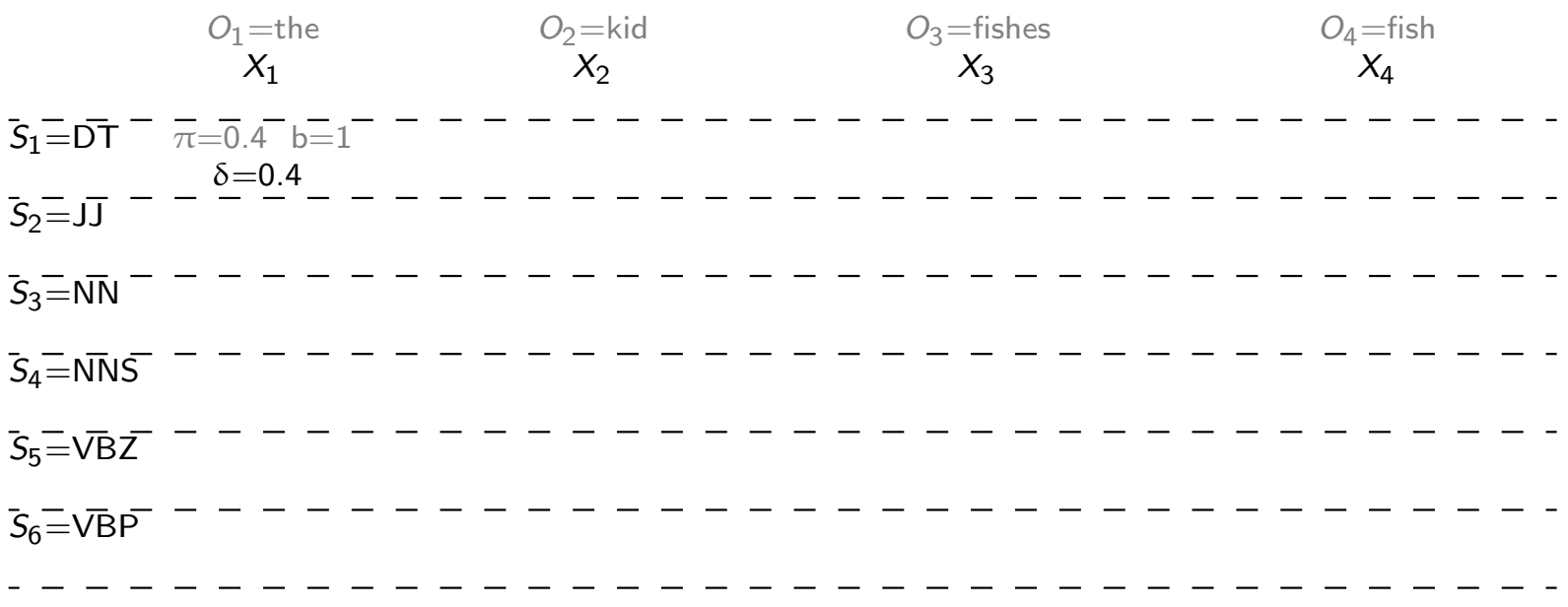
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



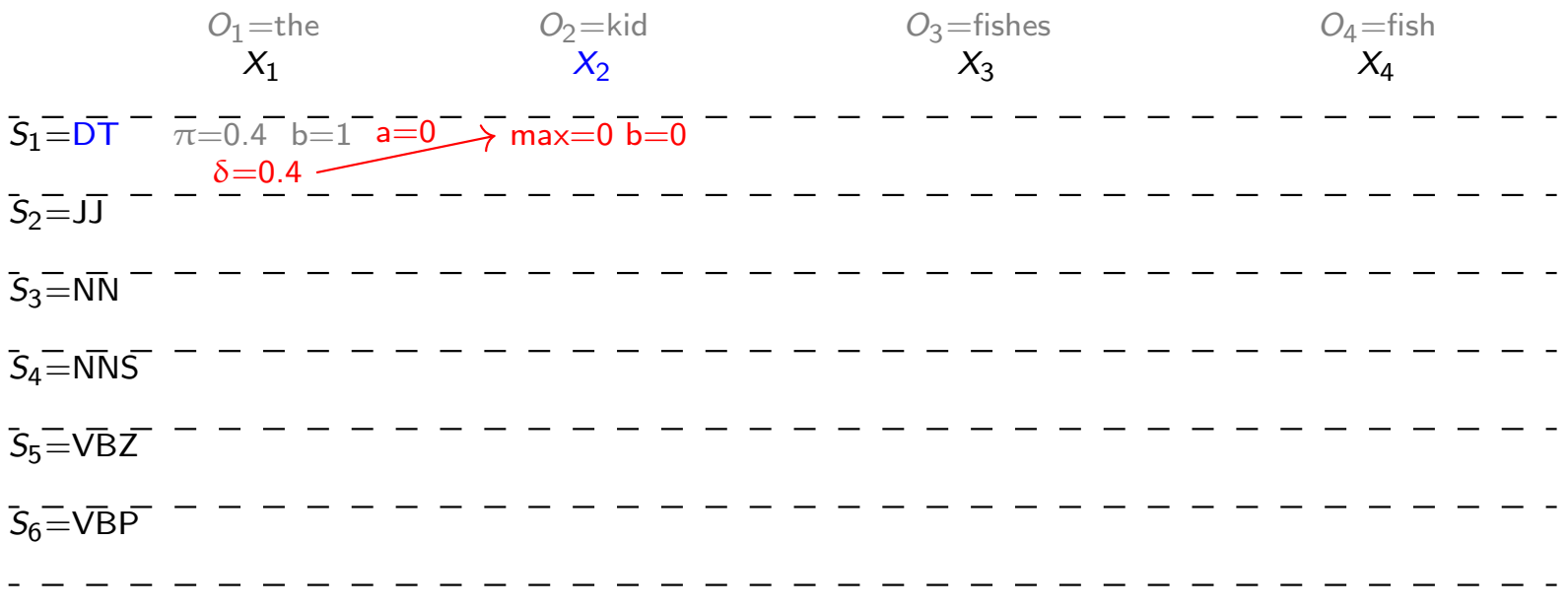
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT	0	0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



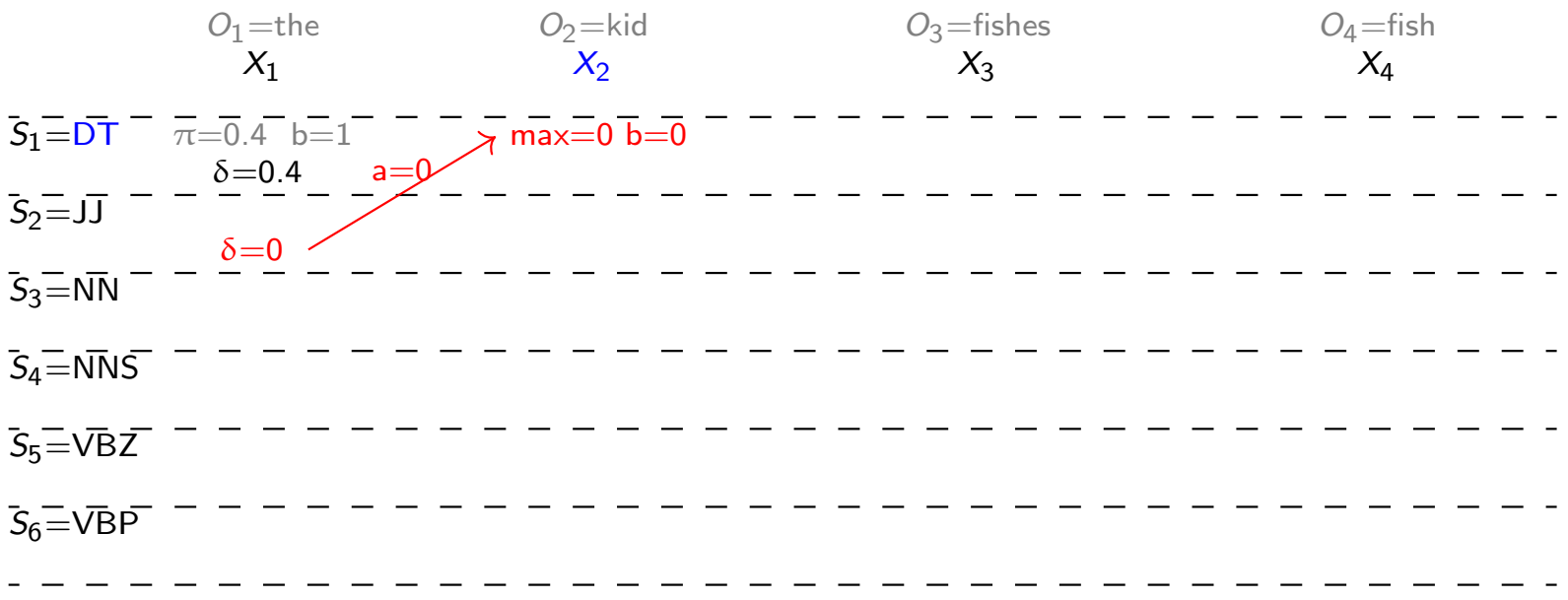
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ	0		0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



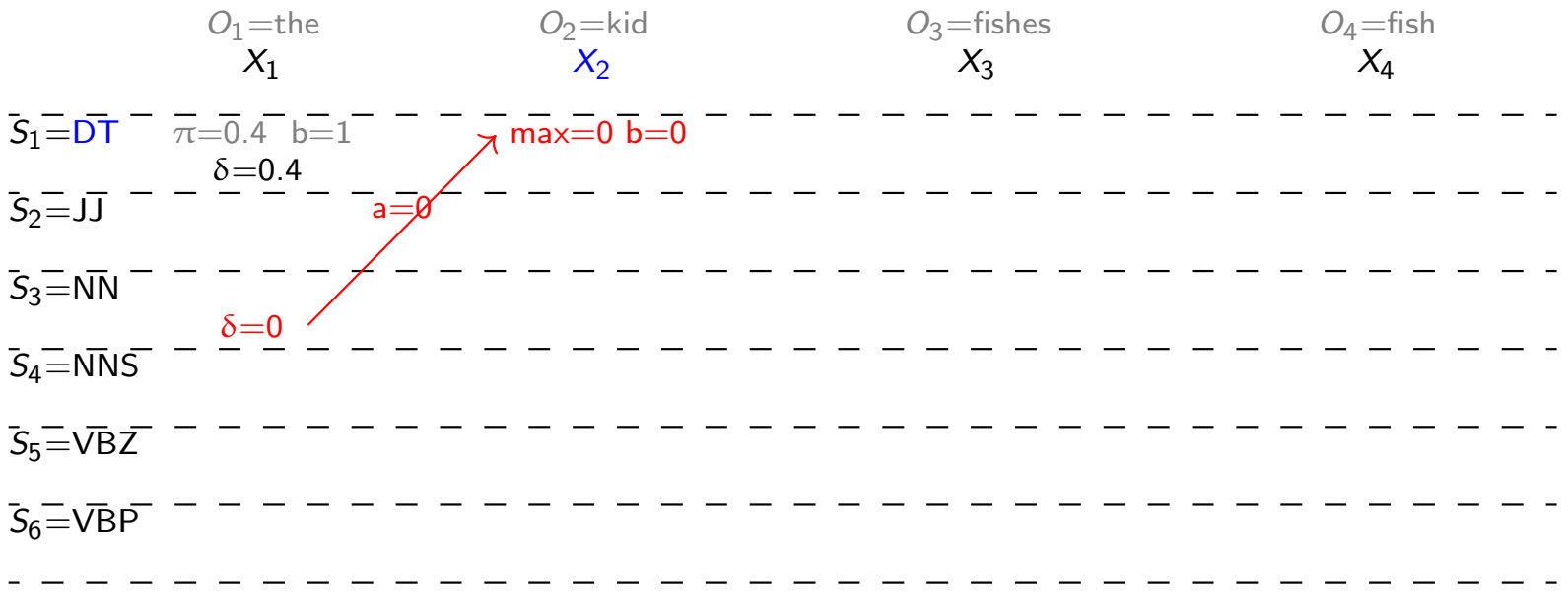
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN	0			0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



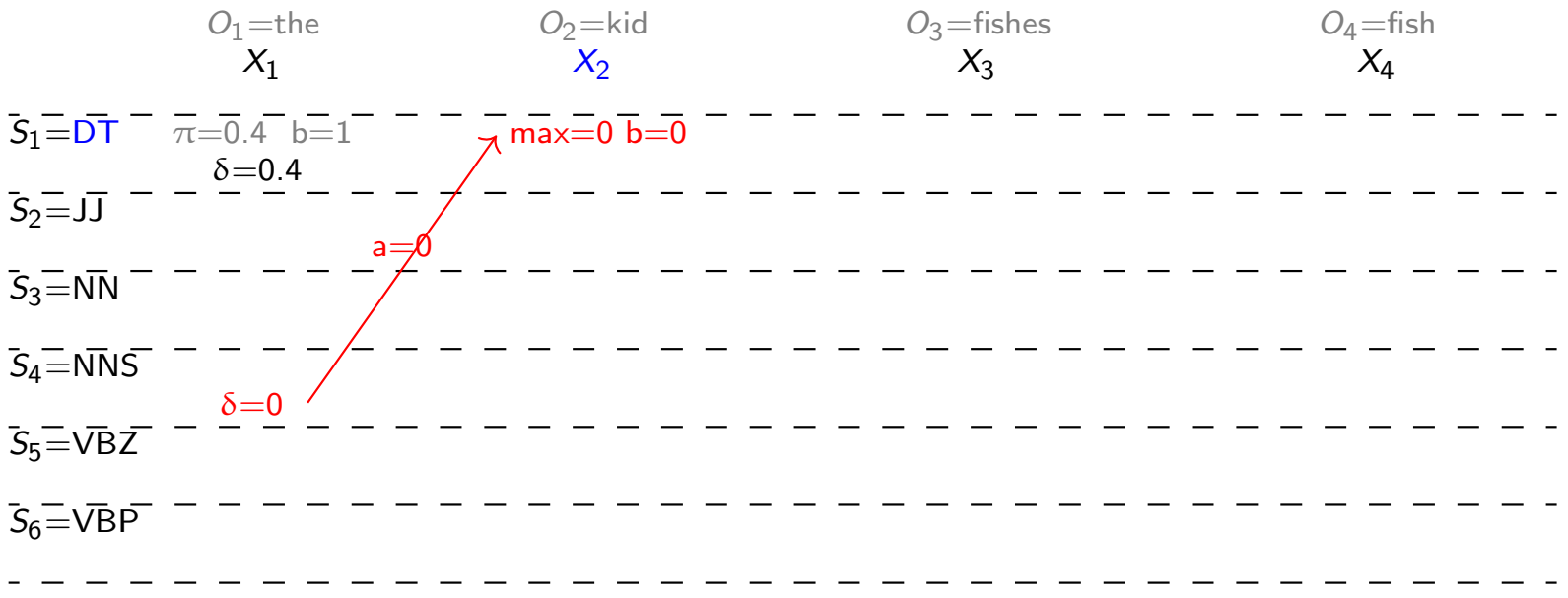
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS	0					1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



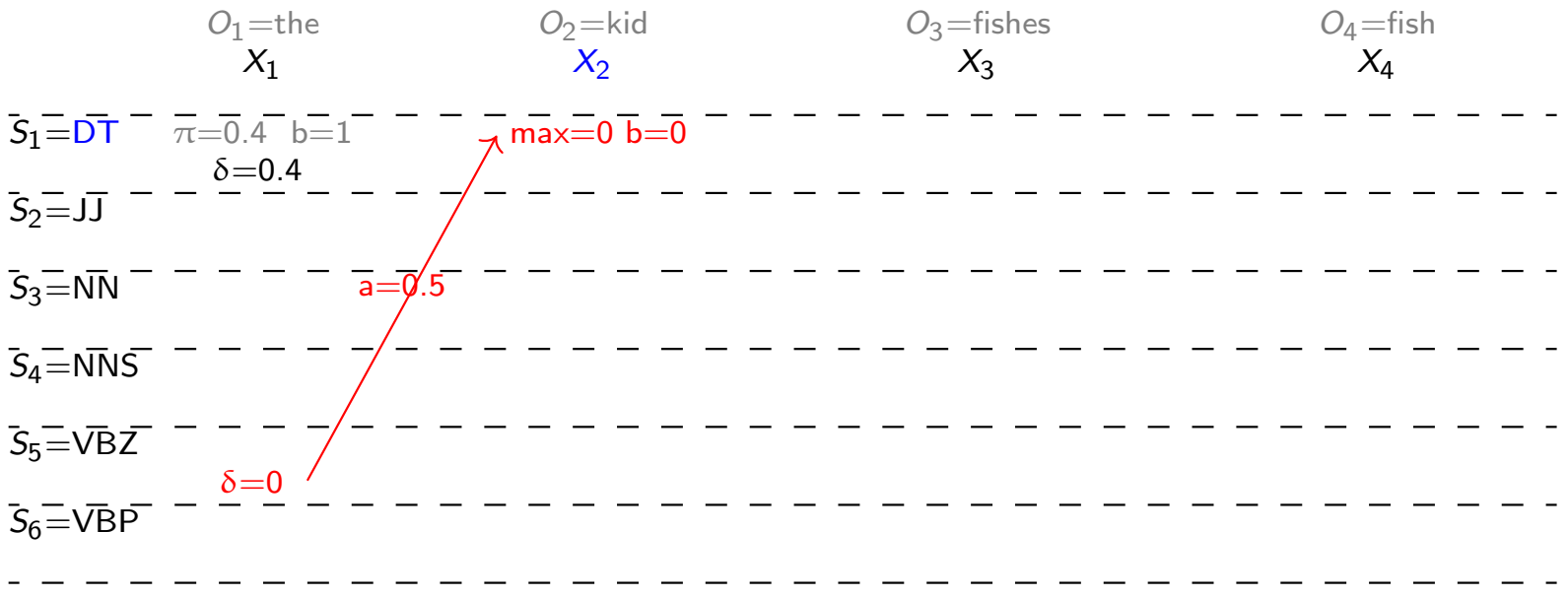
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



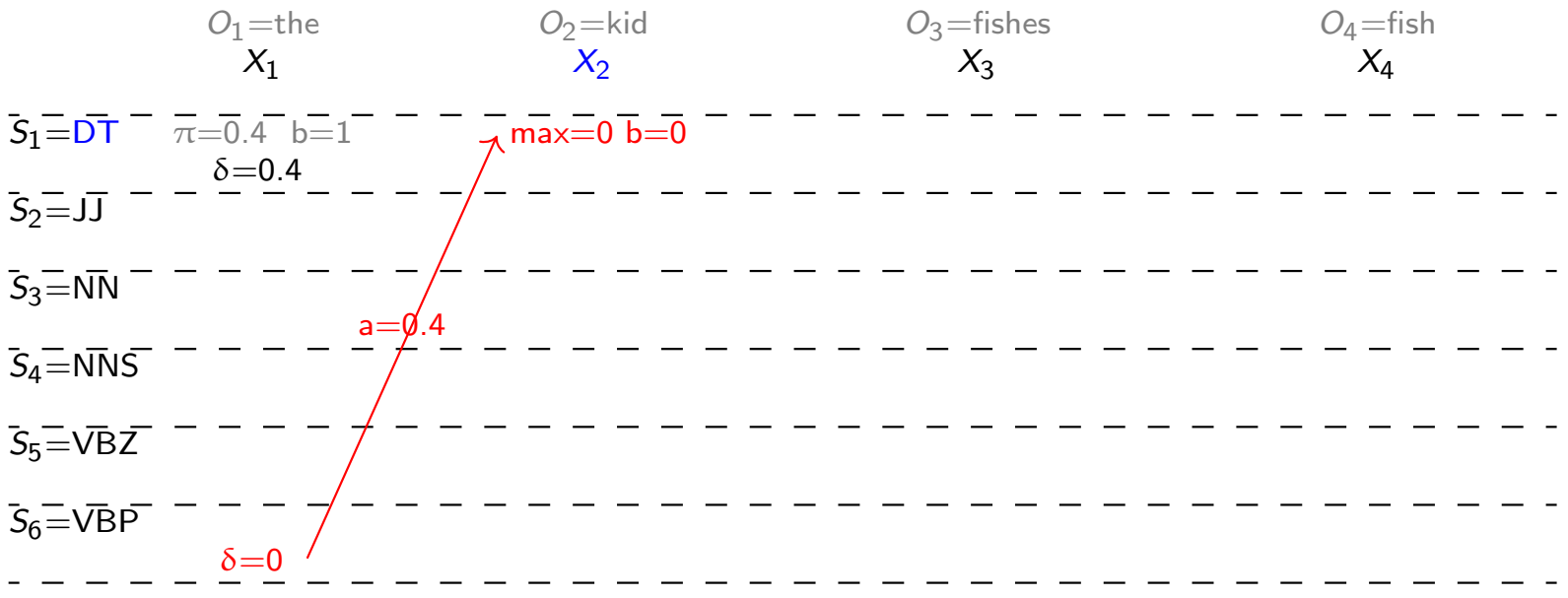
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1	0		
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



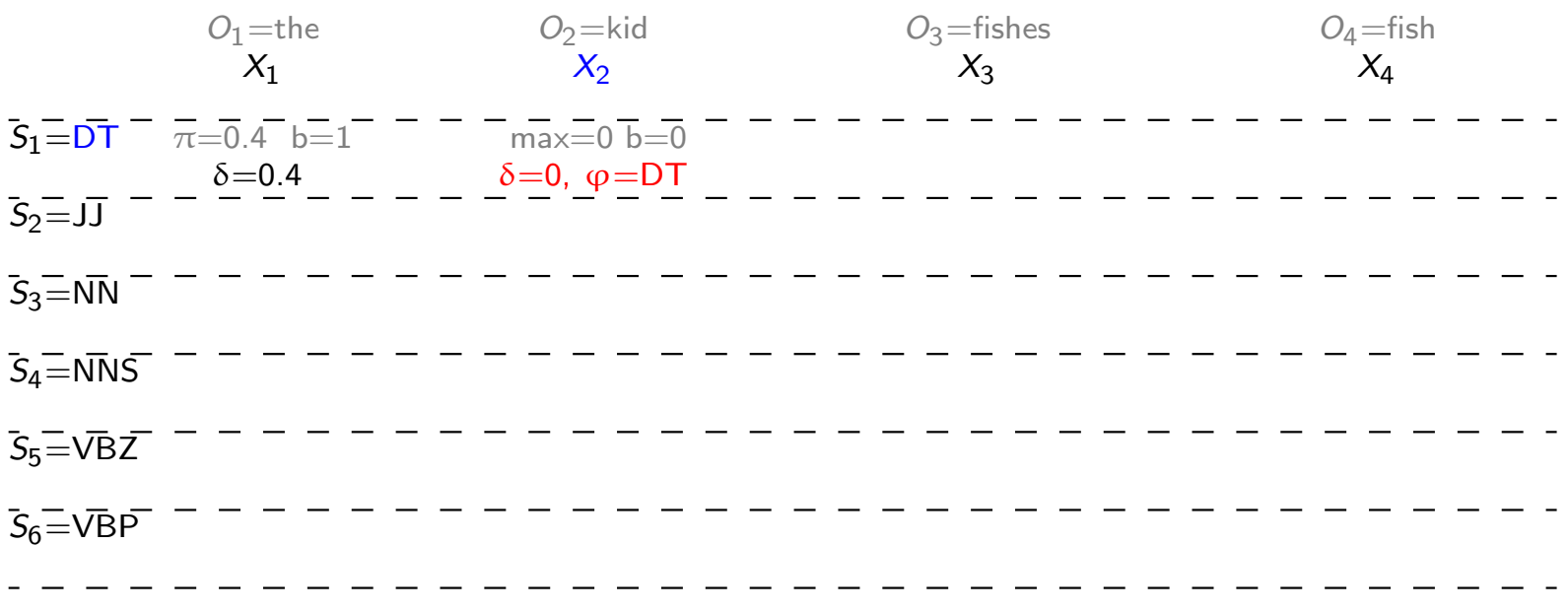
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



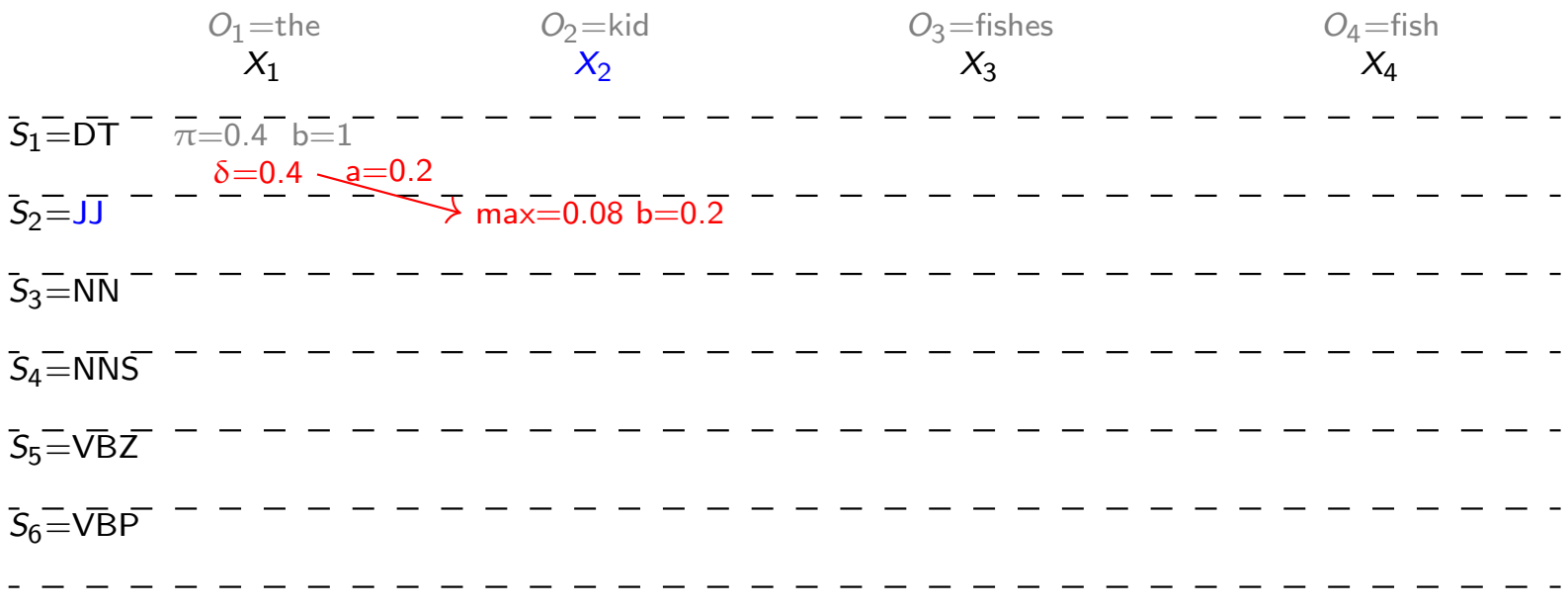
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



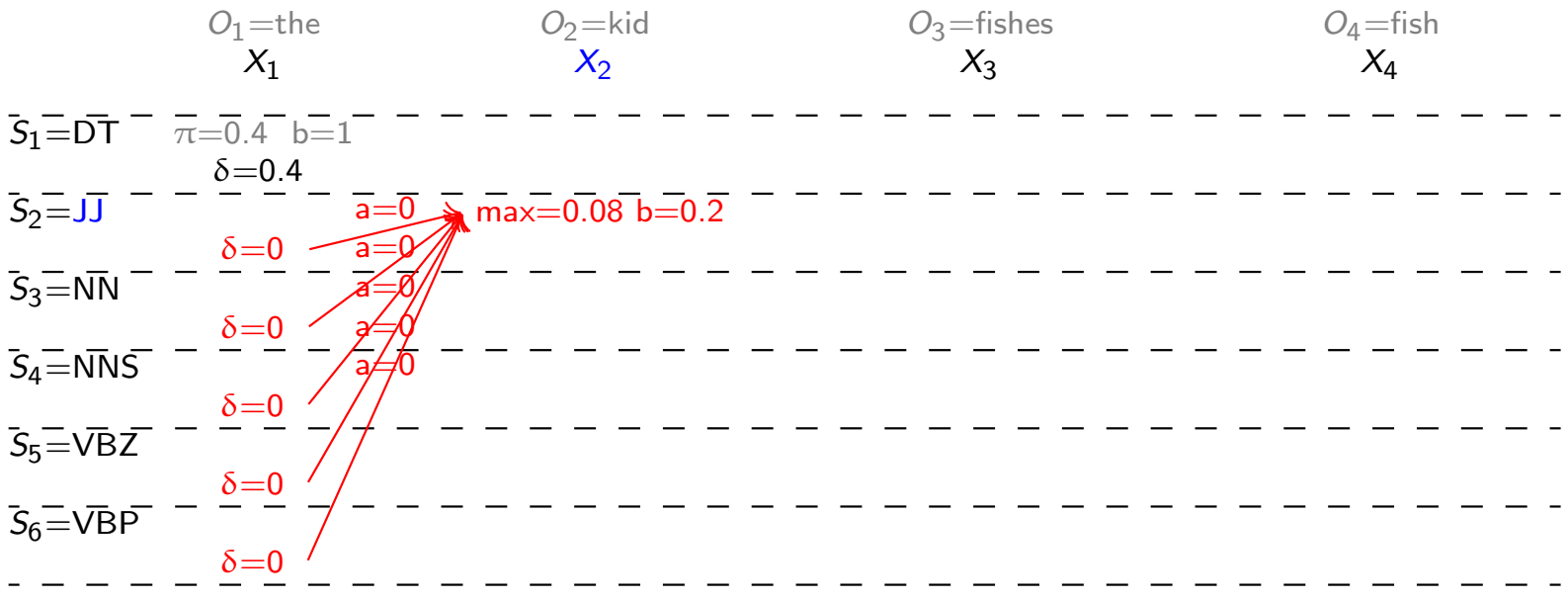
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ		0	0.8	0.2		
NN		0		0.1	0.9	
NNS		0				1
VBZ	0.5	0	0.2	0.3		
VBP	0.4	0	0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



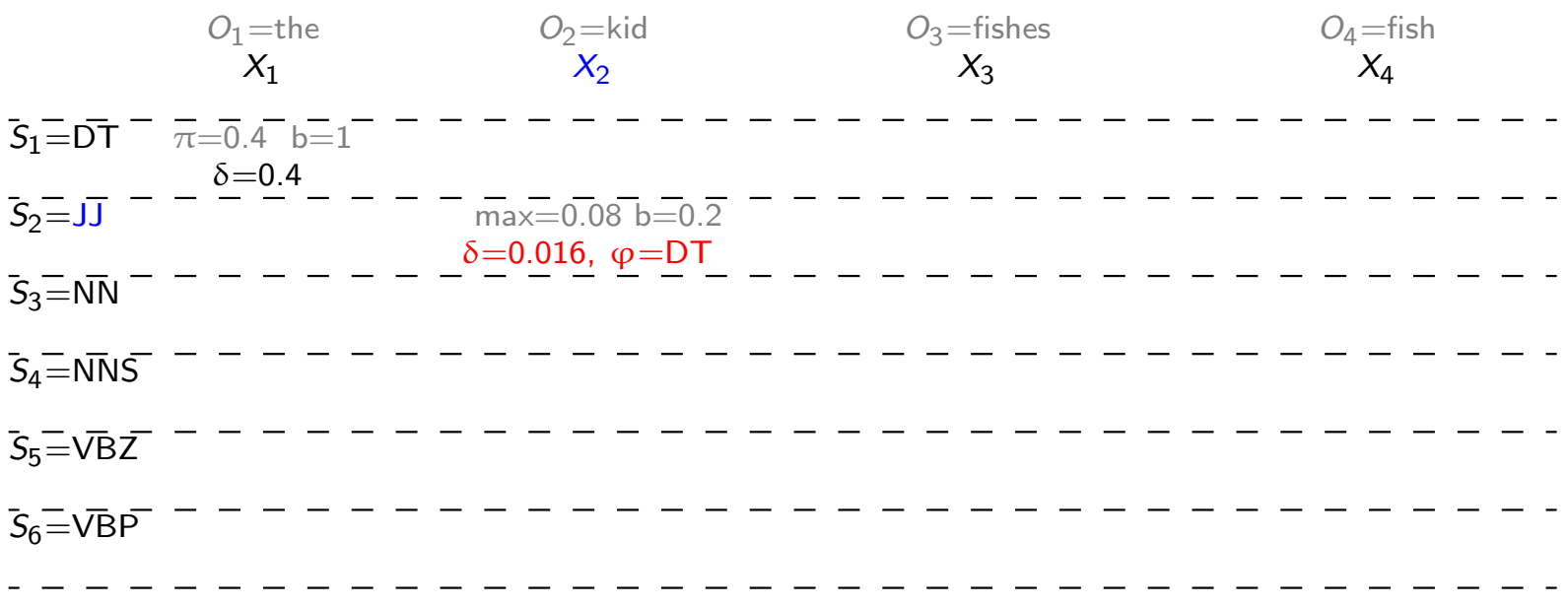
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



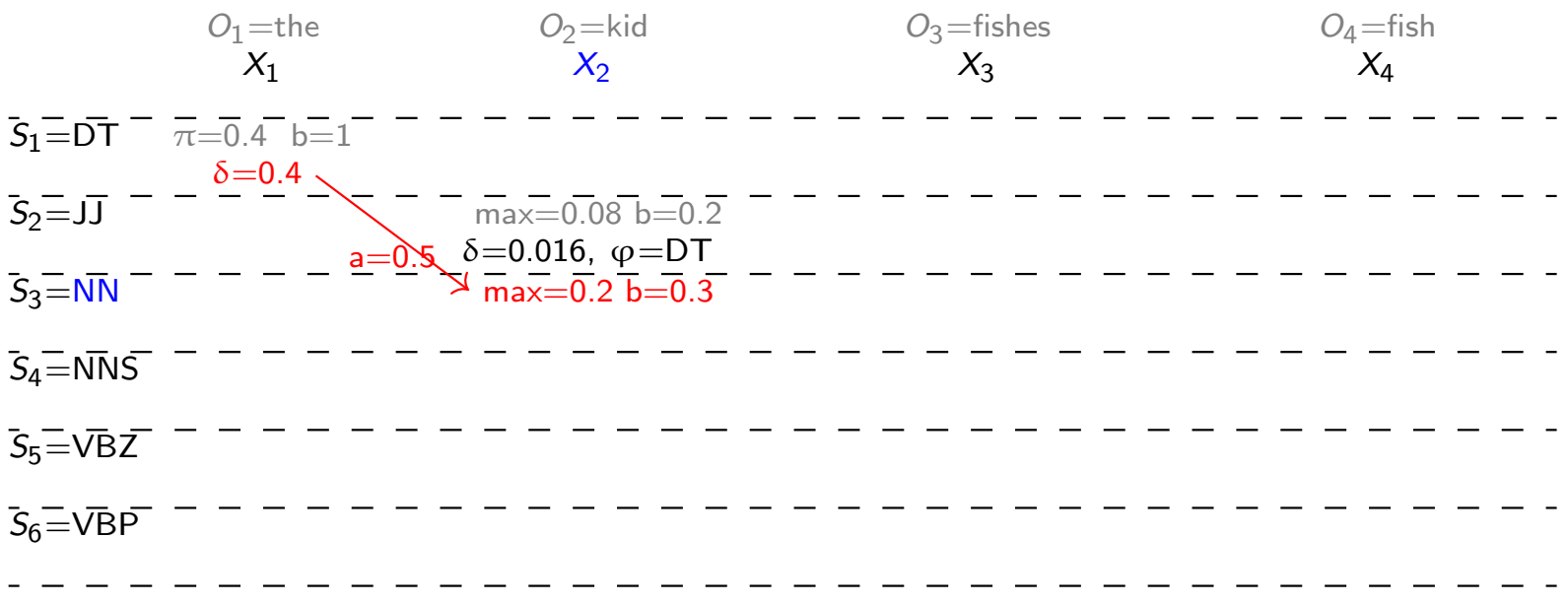
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



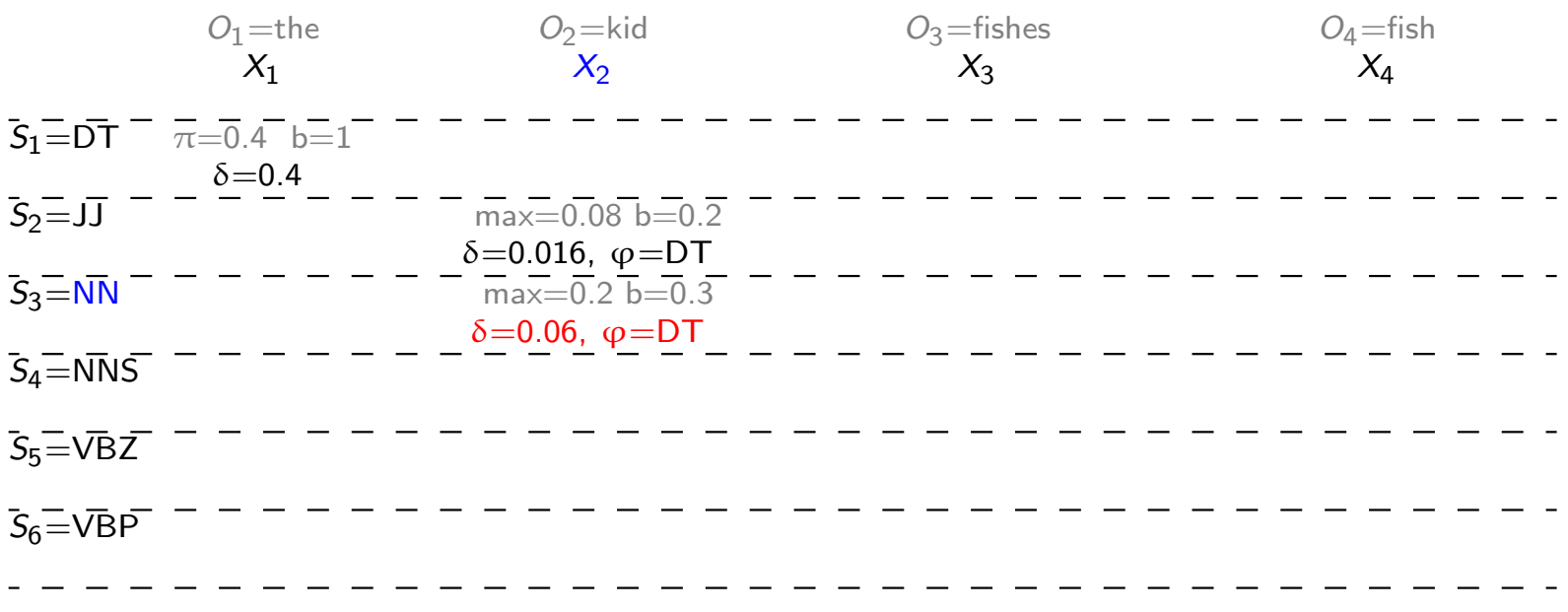
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



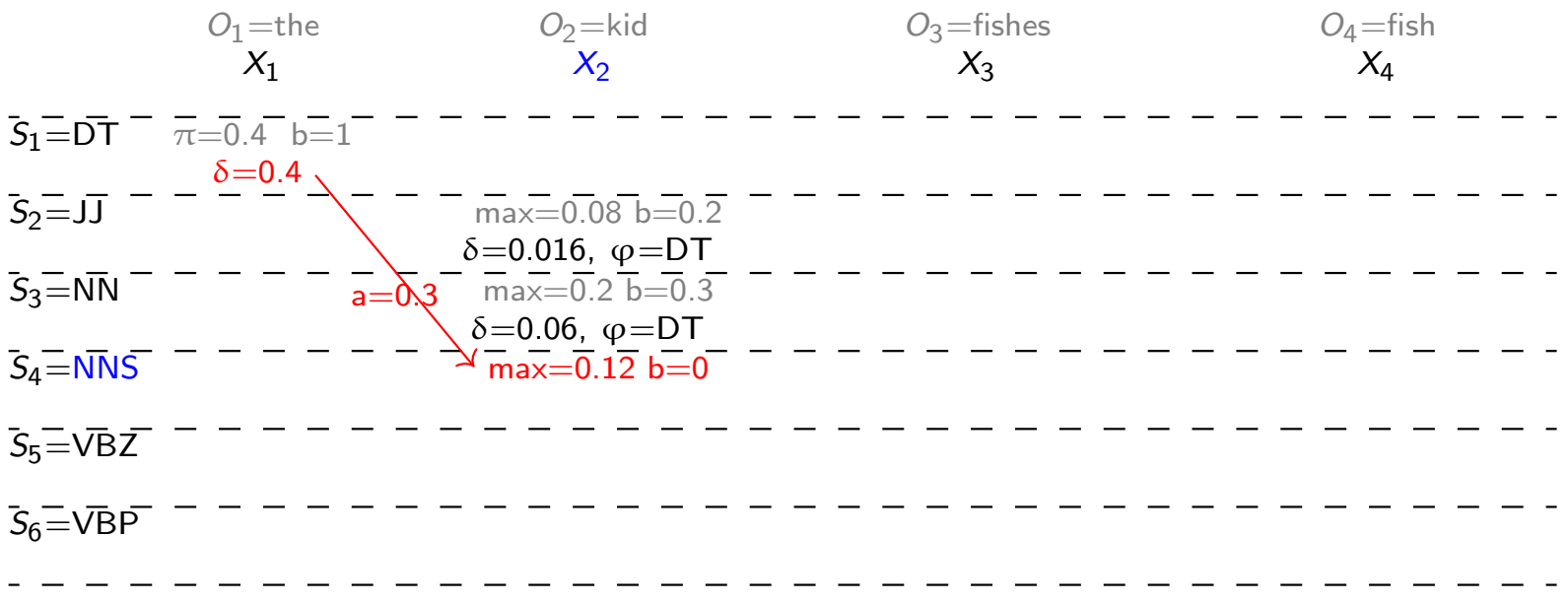
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS		0	0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



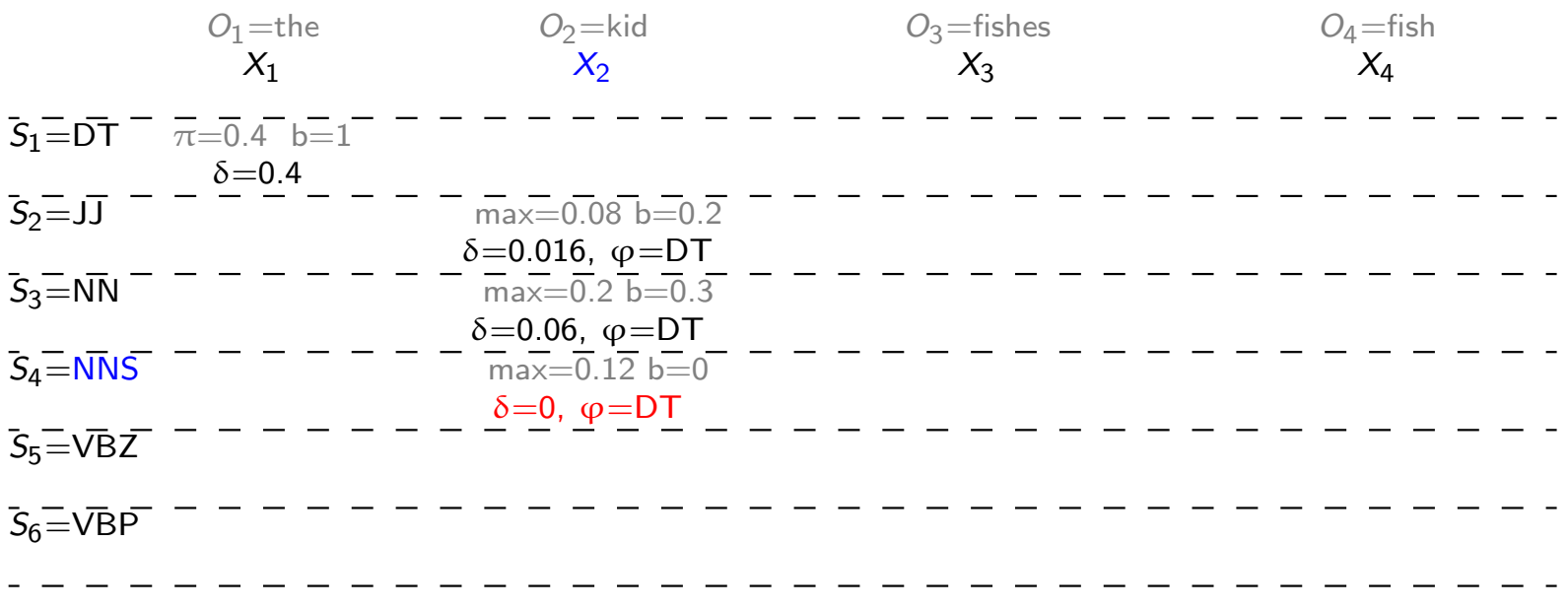
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



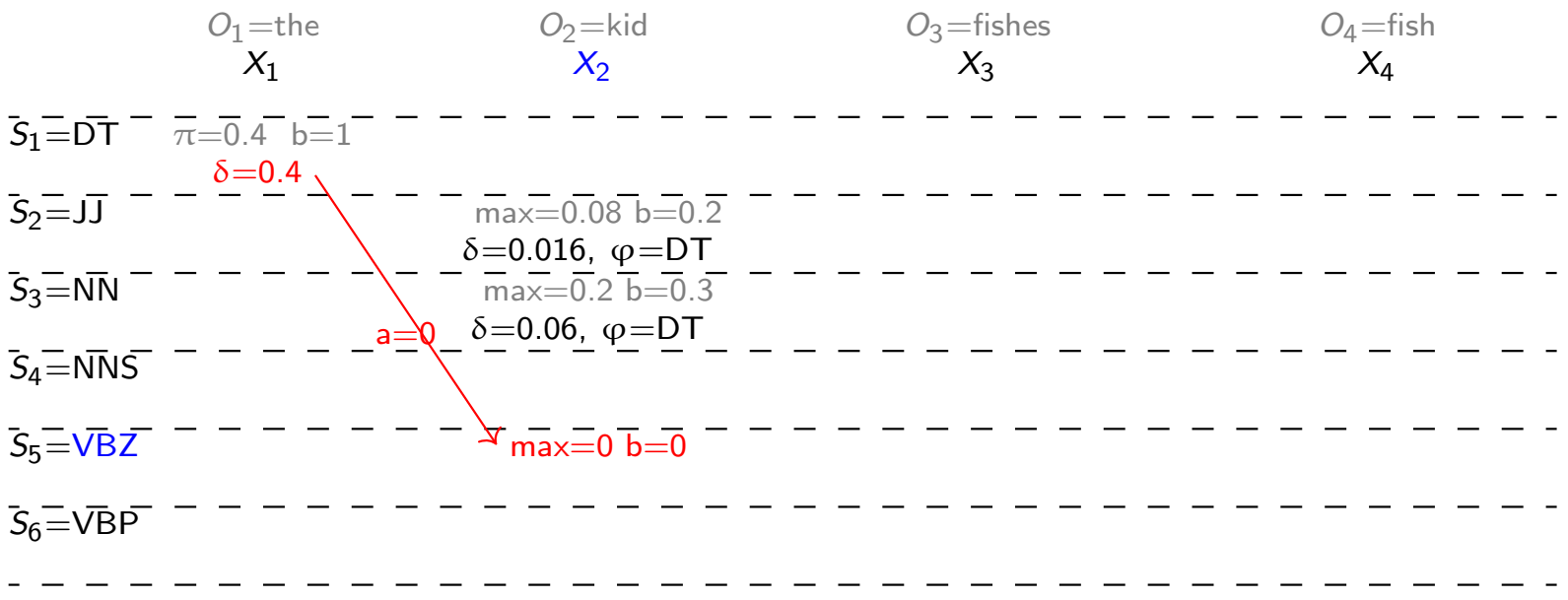
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3	0	
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ		0		0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



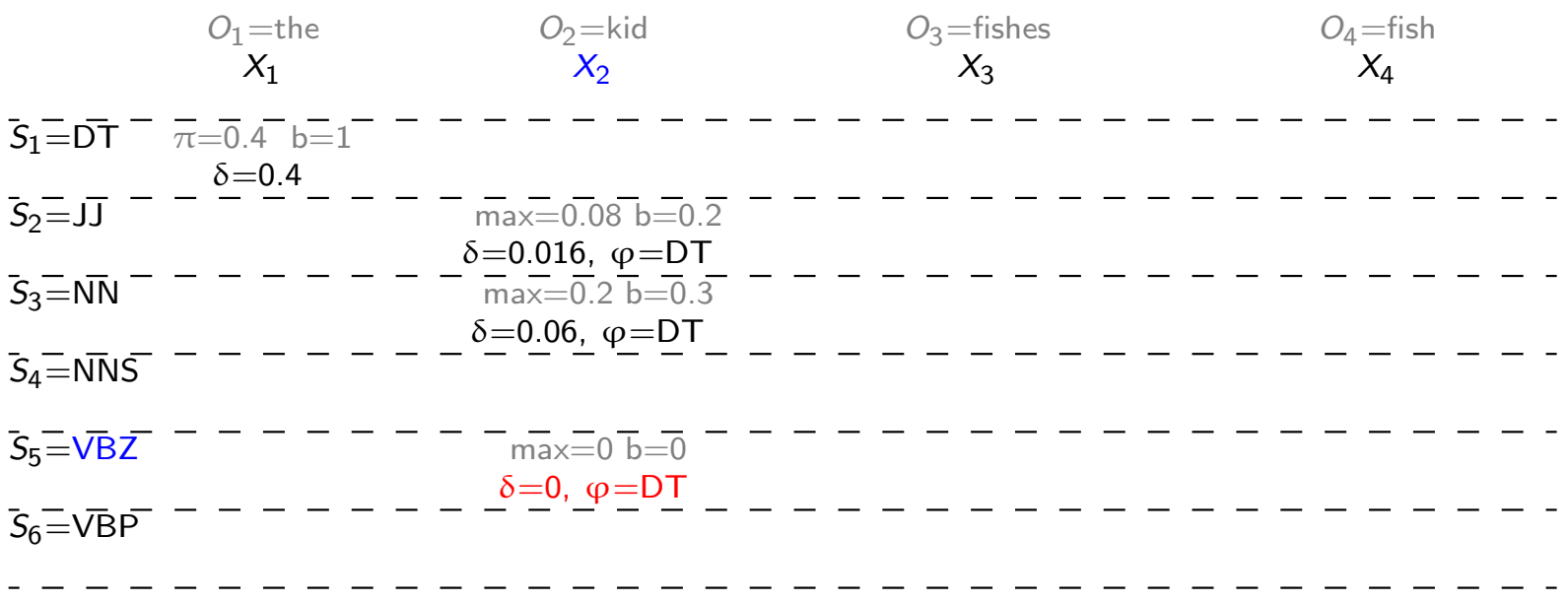
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



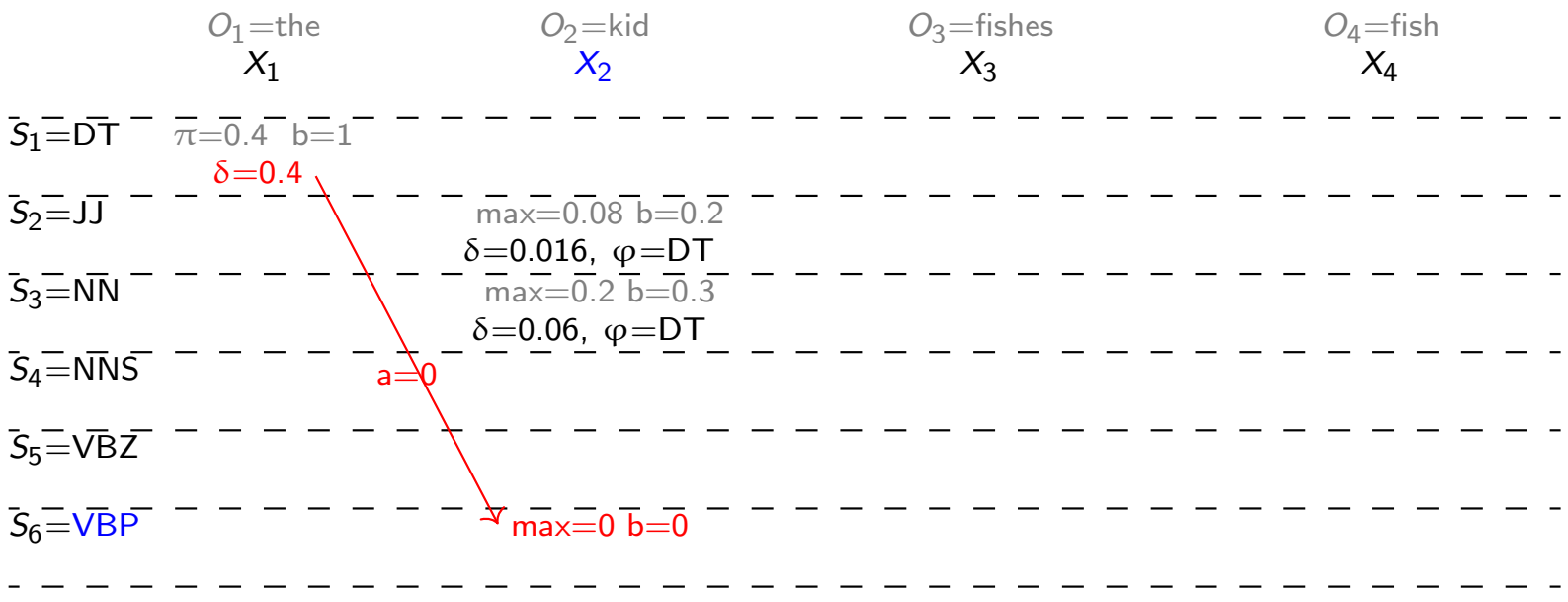
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		0
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP		0	0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



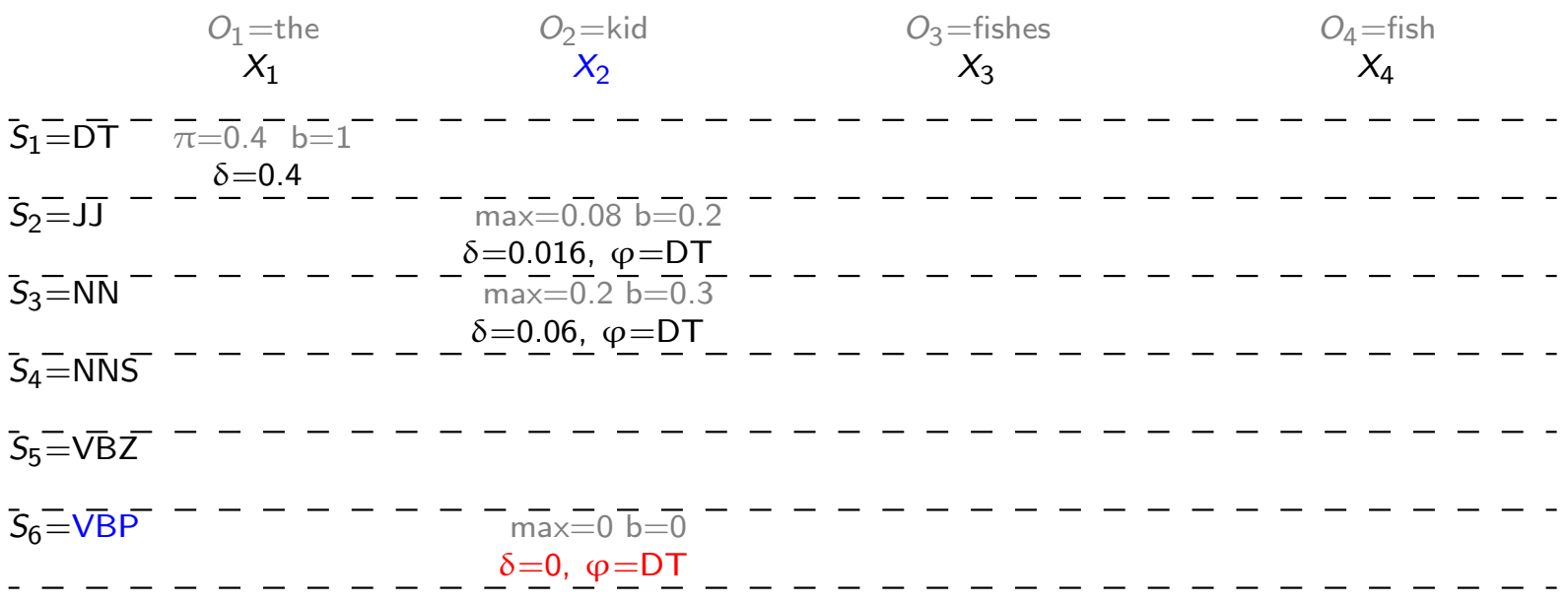
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



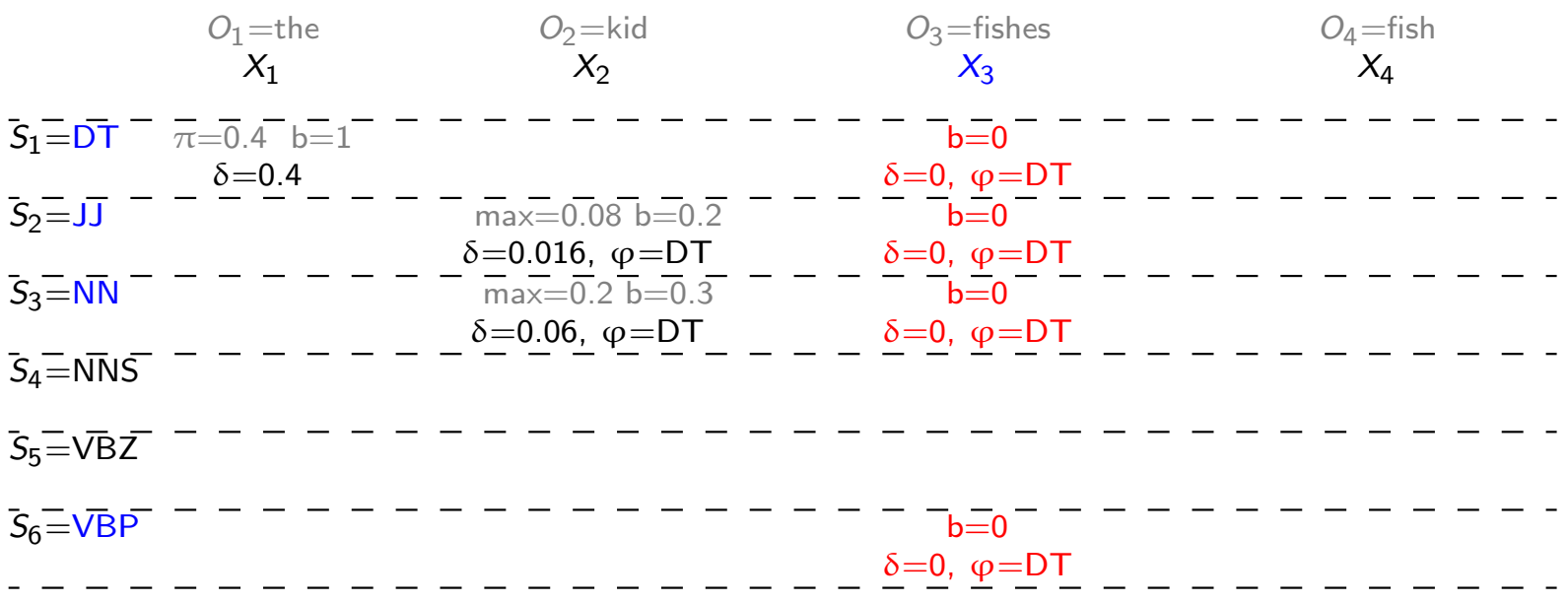
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			0
JJ		0.2		0
NN		0.3	0.4	0
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	0

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



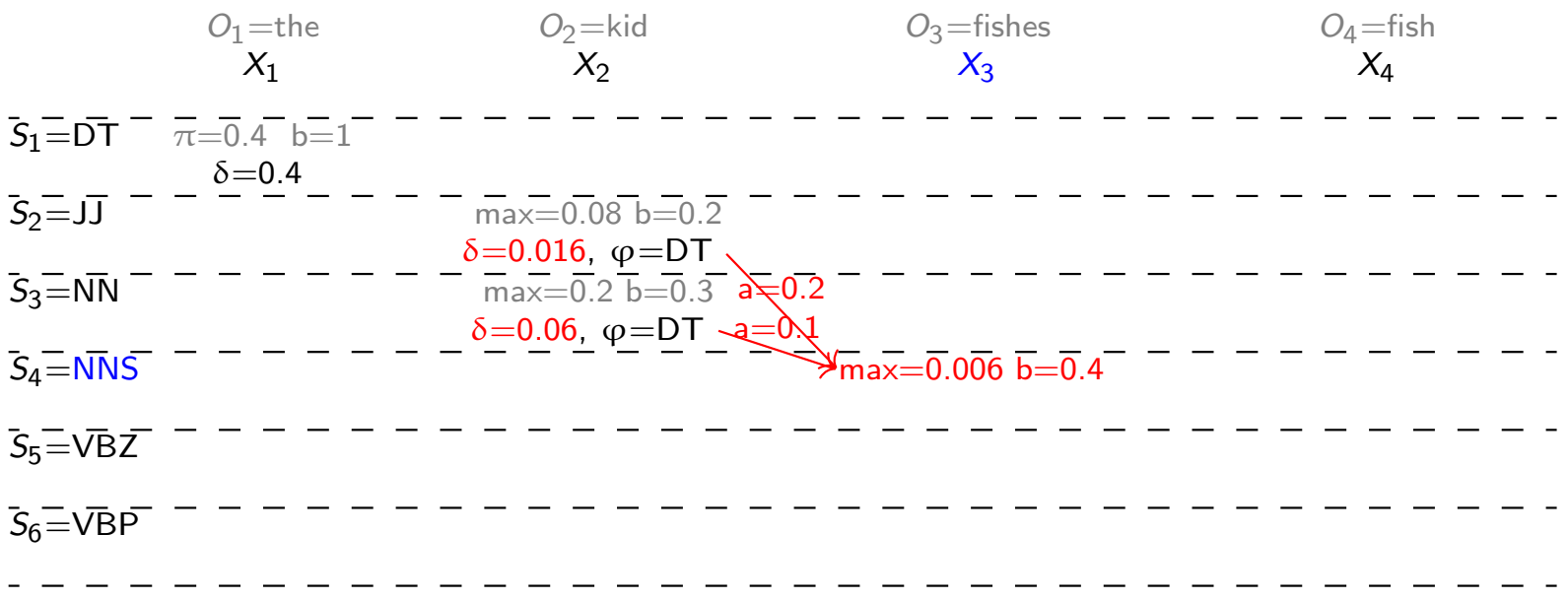
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



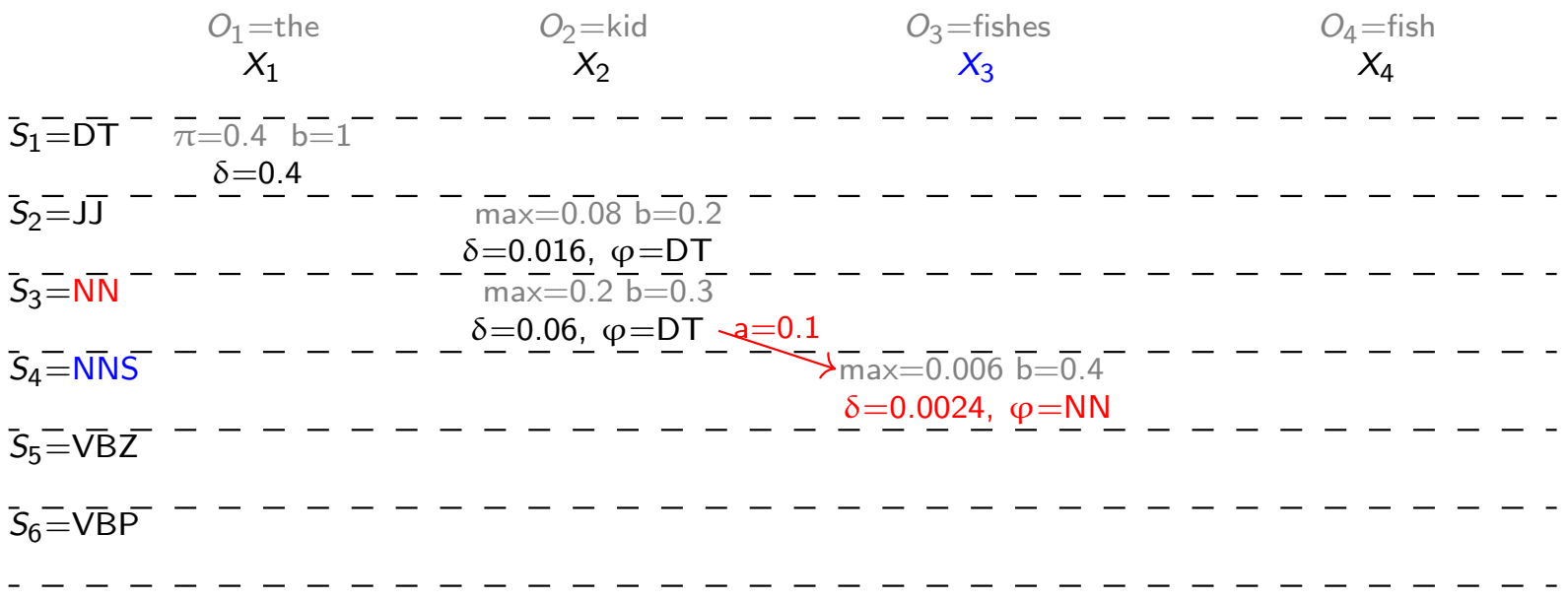
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



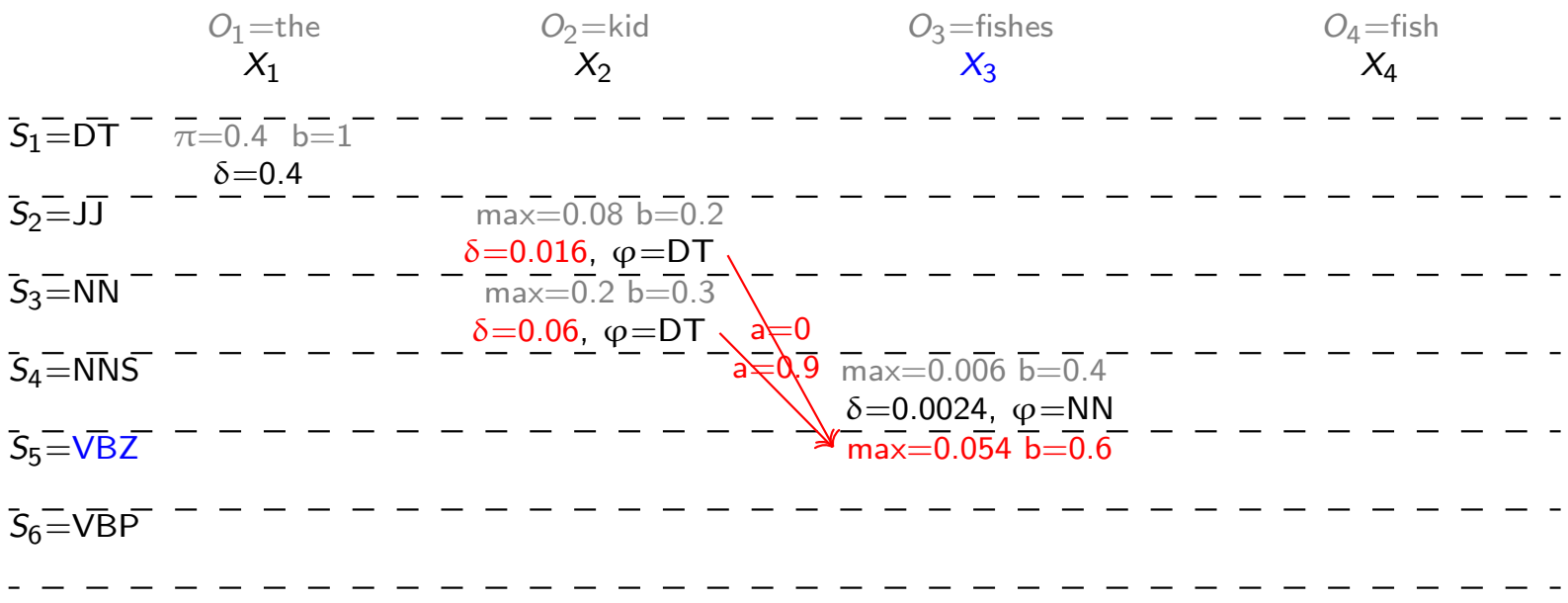
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2	0	
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



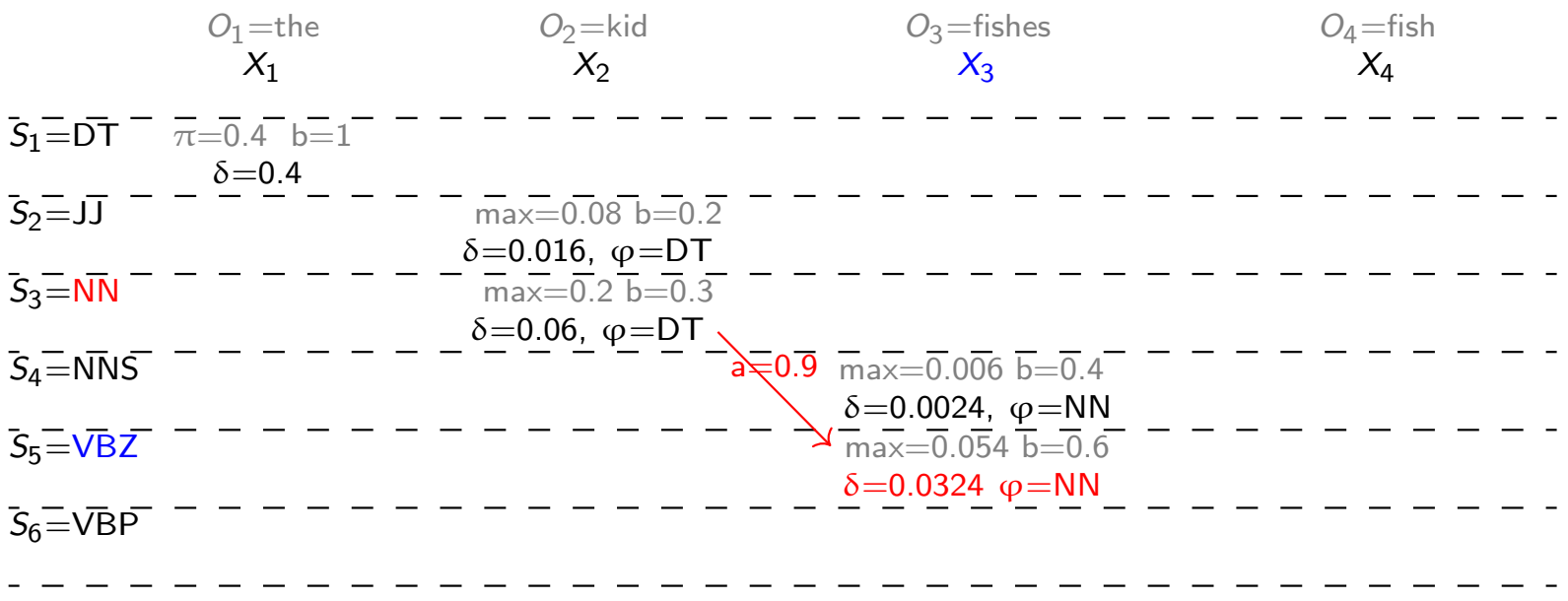
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



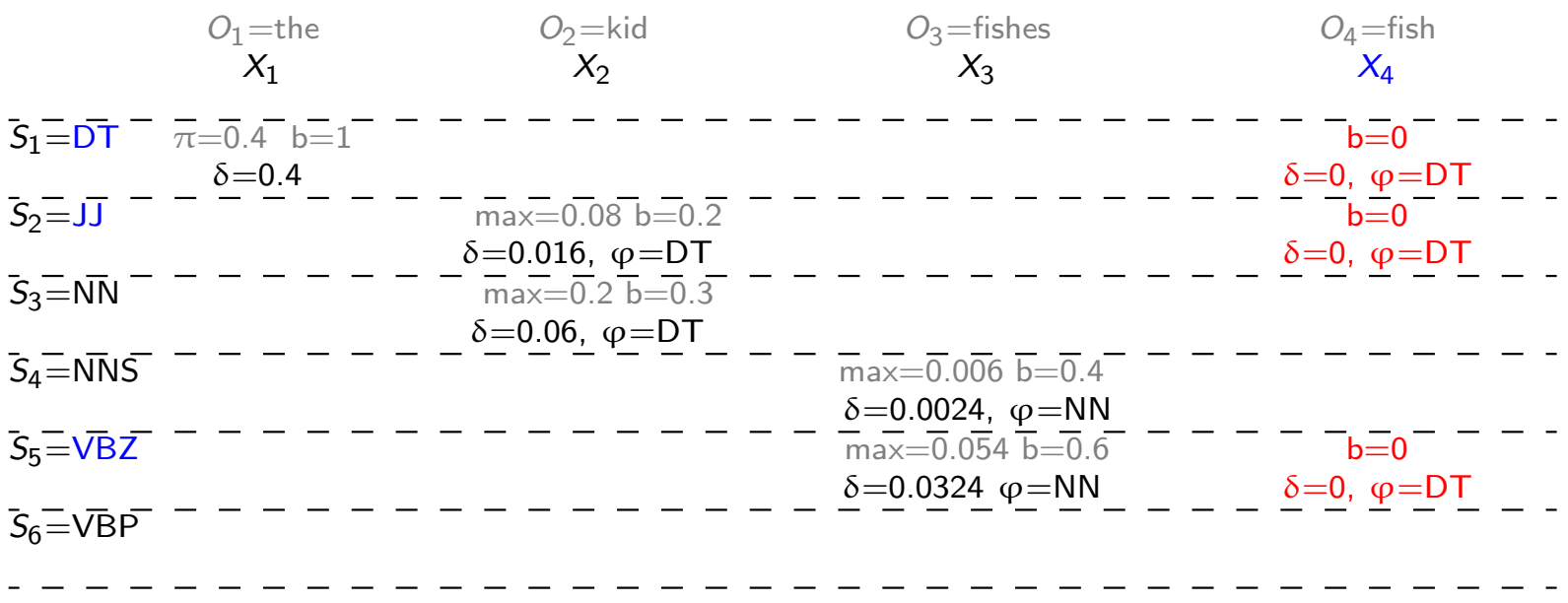
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1		0	
JJ		0.2	0	
NN		0.3	0.4	
NNS			0.3	0.4
VBZ			0	0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



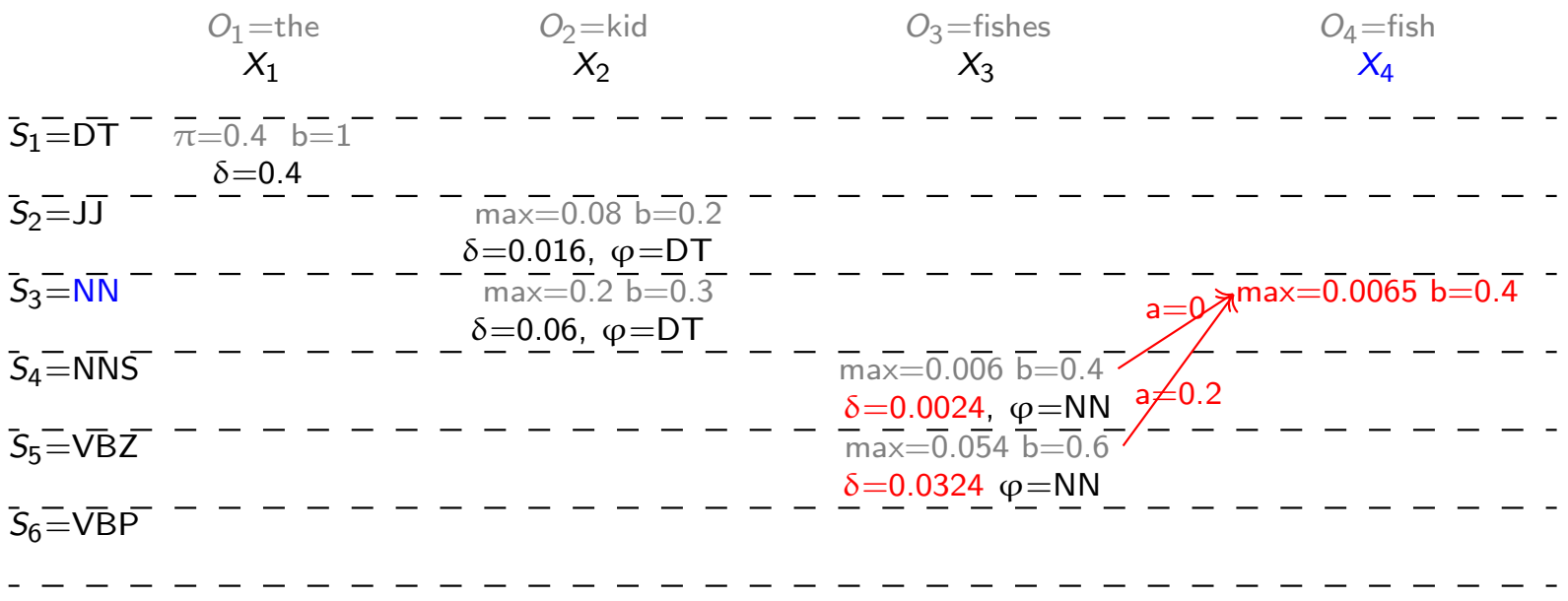
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS			0			1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



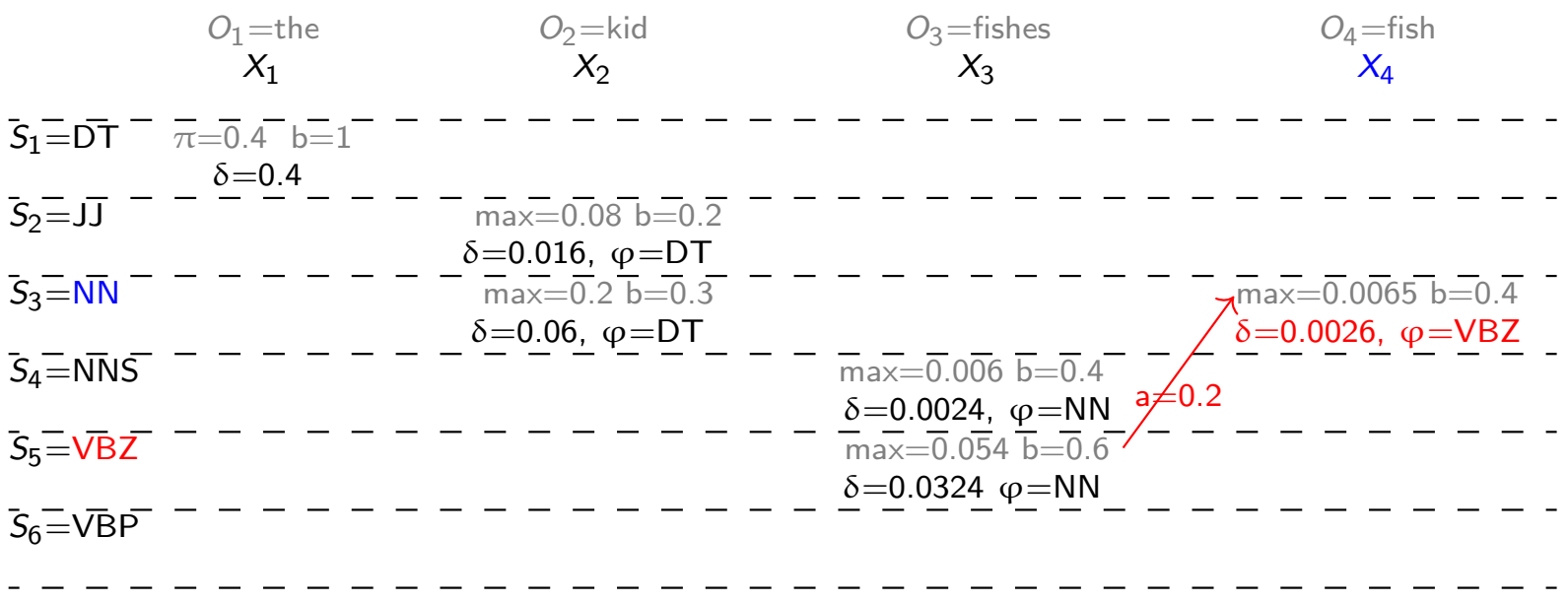
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



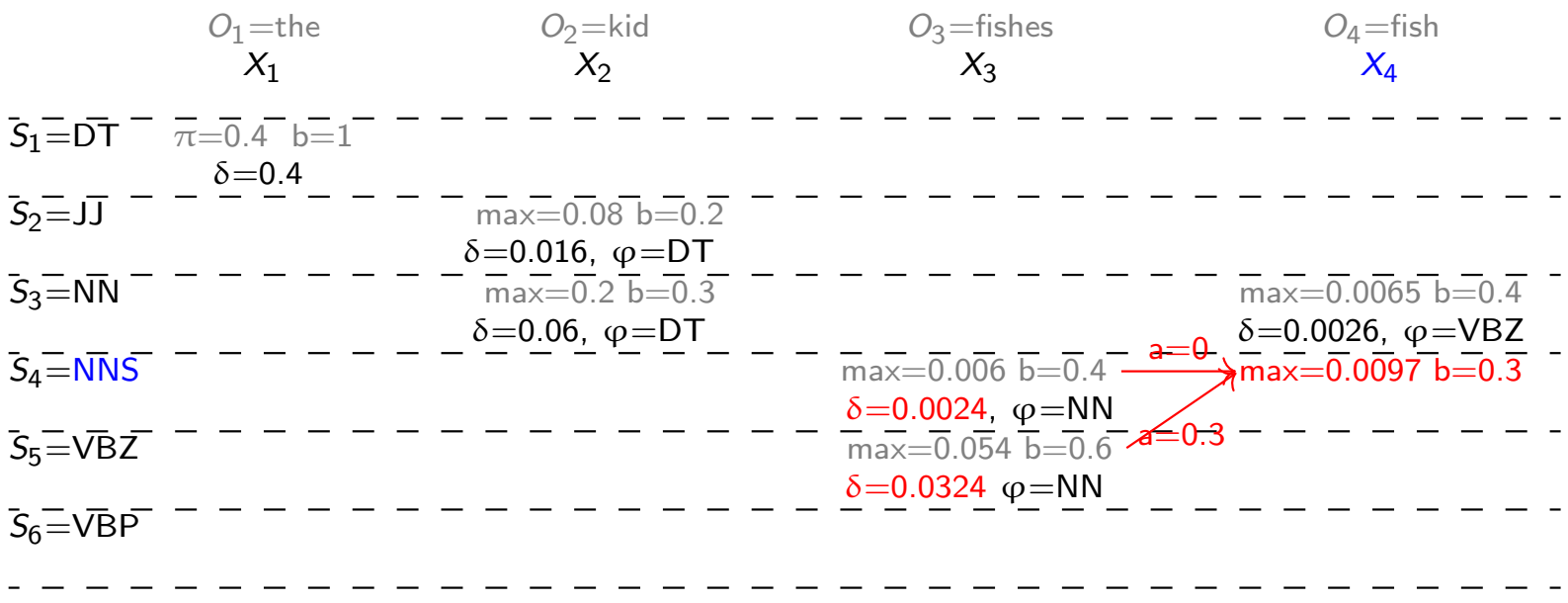
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS				0		1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



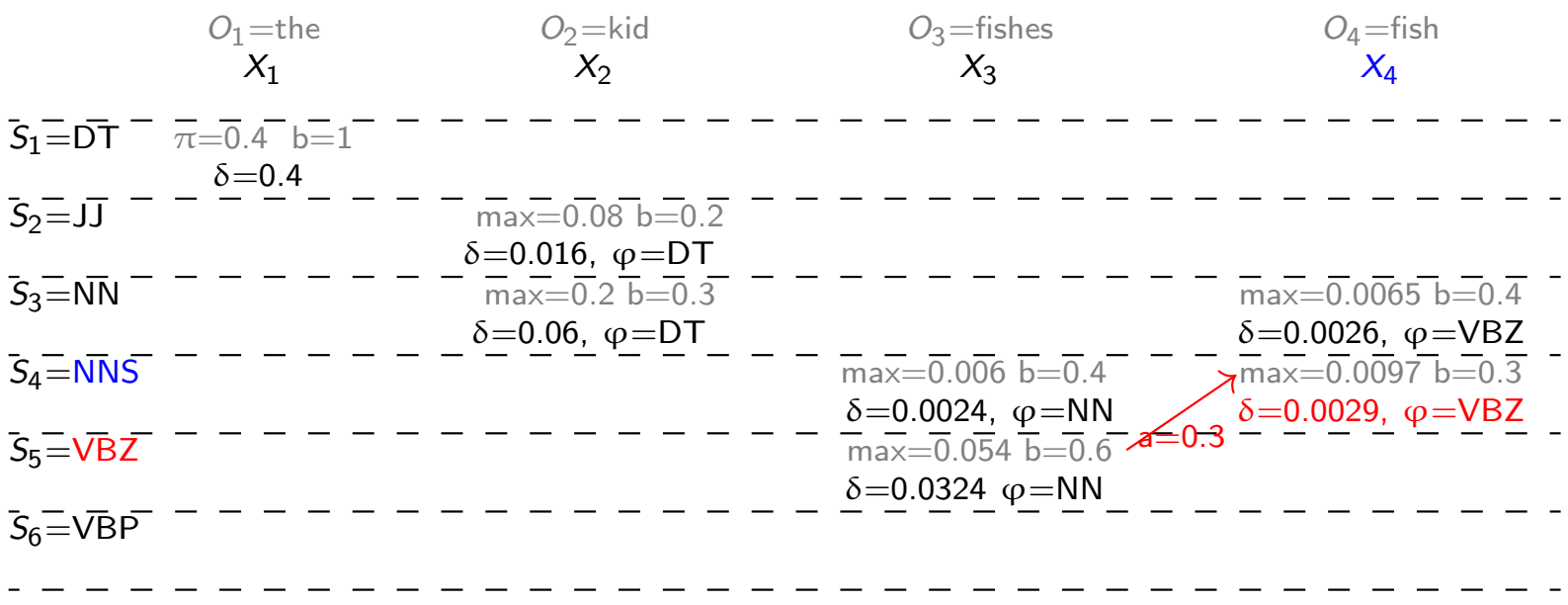
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



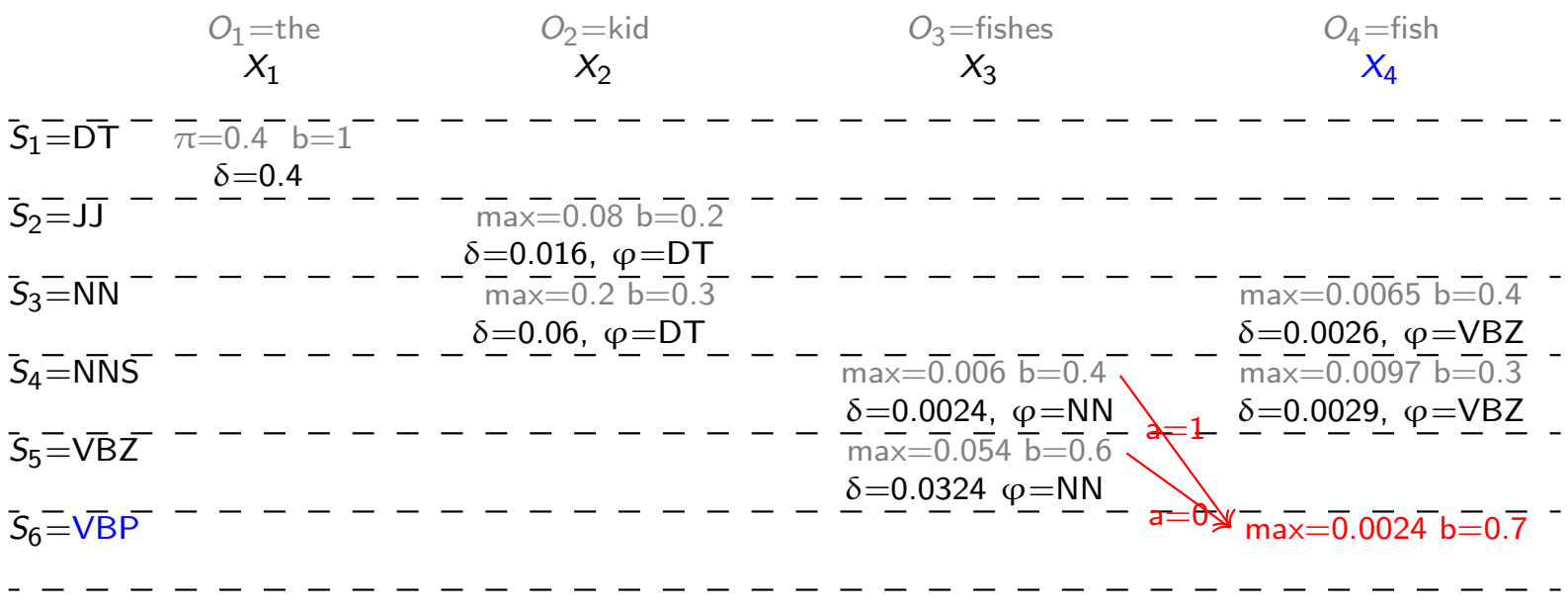
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		0
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



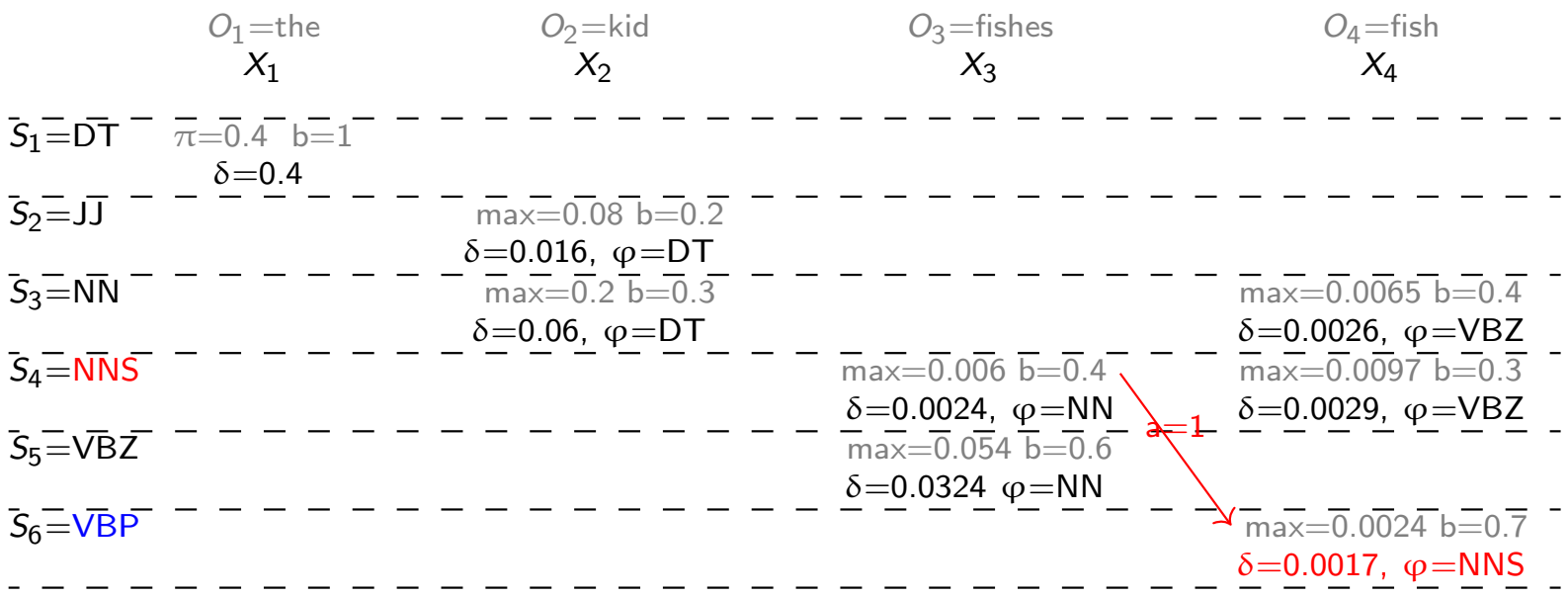
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Inference step: $\delta_t(j) = \max(\delta_{t-1}(i) * a_{i,j}) * b_{j,o_t}$ $\varphi_t(j) = \text{argmax}(\delta_{t-1}(i) * a_{i,j})$



Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Termination step: $X_4 = argmax_i(\delta_4(i))$

	O_1=the X_1	O_2=kid X_2	O_3=fishes X_3	O_4=fish X_4
S_1=DT	π=0.4 b=1 δ=0.4			
S_2=JJ		max=0.08 b=0.2 δ=0.016, φ=DT		
S_3=NN		max=0.2 b=0.3 δ=0.06, φ=DT		max=0.0065 b=0.4 δ=0.0026, φ=VBZ
S_4=NNS			max=0.006 b=0.4 δ=0.0024, φ=NN	max=0.0097 b=0.3 δ=0.0029, φ=VBZ
S_5=VBZ			max=0.054 b=0.6 δ=0.0324 φ=NN	
S_6=VBP				max=0.0024 b=0.7 δ=0.0017, φ=NNS

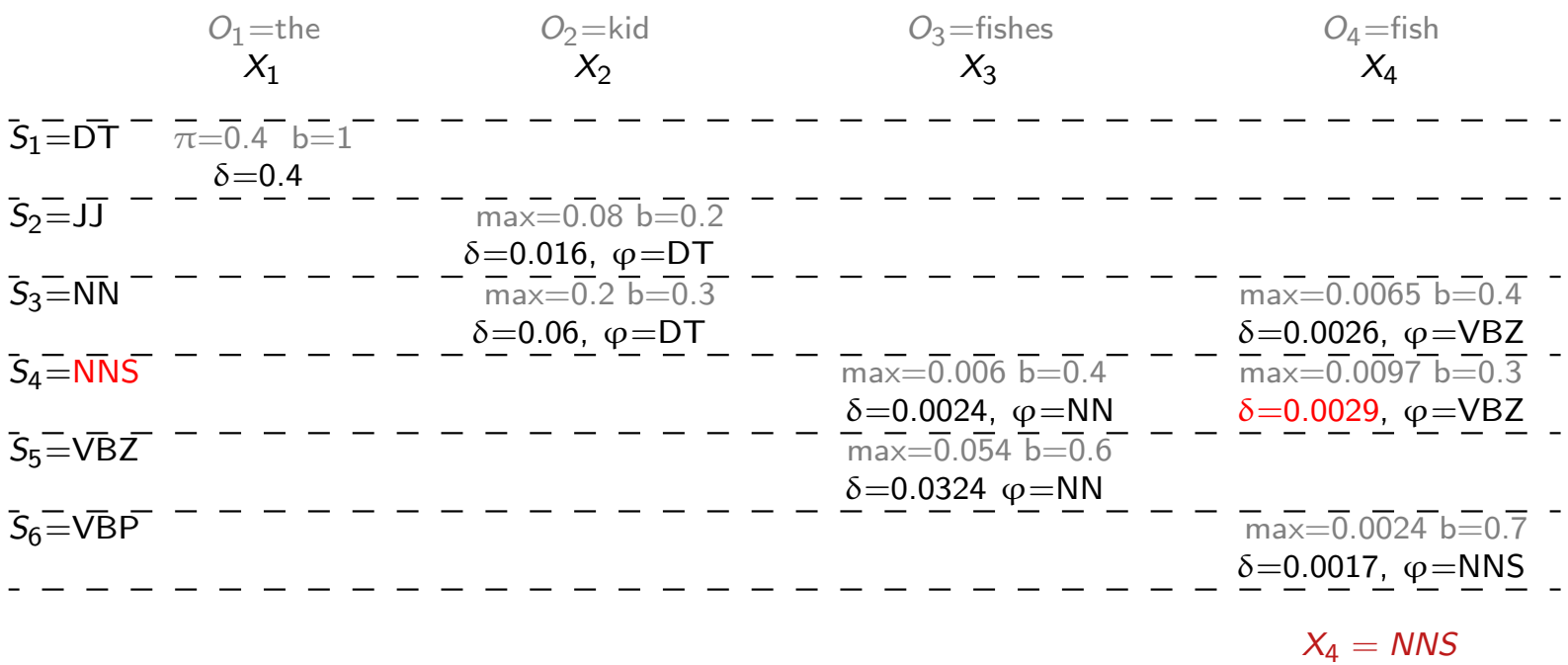
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Termination step: $X_4 = argmax_i(\delta_4(i))$



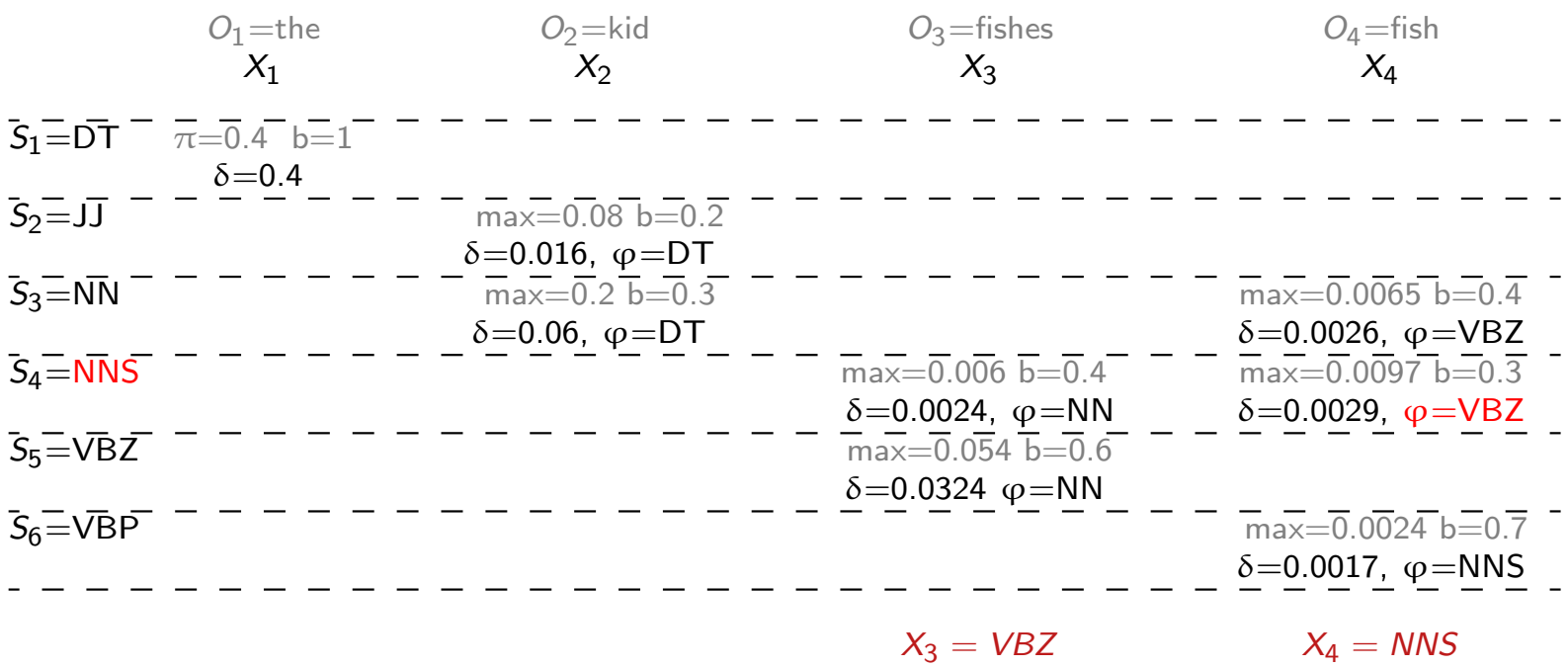
Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Backward path readout step: $X_t = \varphi(X_{t+1})$



Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Backward path readout step: $X_t = \varphi(X_{t+1})$

	O_1=the	O_2=kid	O_3=fishes	O_4=fish
	X_1	X_2	X_3	X_4
S_1=DT	π=0.4 b=1 δ=0.4			
S_2=JJ		max=0.08 b=0.2 δ=0.016, φ=DT		
S_3=NN		max=0.2 b=0.3 δ=0.06, φ=DT		max=0.0065 b=0.4 δ=0.0026, φ=VBZ
S_4=NNS			max=0.006 b=0.4 δ=0.0024, φ=NN	max=0.0097 b=0.3 δ=0.0029, φ=VBZ
S_5=VBZ			max=0.054 b=0.6 δ=0.0324 φ=NN	
S_6=VBP				max=0.0024 b=0.7 δ=0.0017, φ=NNS
		$X_2 = NN$	$X_3 = VBZ$	$X_4 = NNS$

Exercise

π	
DT	0.4
JJ	0.2
NN	
NNS	0.3
VBZ	
VBP	0.1

A	DT	JJ	NN	NNS	VBZ	VBP
DT		0.2	0.5	0.3		
JJ			0.8	0.2		
NN				0.1	0.9	
NNS						1
VBZ	0.5		0.2	0.3		
VBP	0.4		0.4	0.2		

B	the	kid	fish	fishes
DT	1			
JJ		0.2		
NN		0.3	0.4	
NNS			0.3	0.4
VBZ				0.6
VBP			0.7	

Trellis from Viterbi: Backward path readout step: $X_t = \varphi(X_{t+1})$