



# Why Language is Hard: Structure and Predictions

Advanced Machine Learning for NLP

Jordan Boyd-Graber

SLIDES ADAPTED FROM LIANG HUANG

## POS Tagging: Task Definition

---

- Annotate each word in a sentence with a part-of-speech marker.
- Lowest level of syntactic analysis.

John	saw	the	saw	and	decided	to	take	it	to	the	table
NNP	VBD	DT	NN	CC	VBD	TO	VB	PRP	IN	DT	NN

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Total score of hypothesis  $z$  given input  $x$

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Feature weight

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Feature present (binary)

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation) How do we move from a model and unlabeled data to labeled data? (Inference)



## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation:  
**HMM**) How do we move from a model and unlabeled data to labeled data?  
(Inference)

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation:  
**HMM**) How do we move from a model and unlabeled data to labeled data?  
(Inference)

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i = \log p(z_1 = i)$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j}$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation: HMM) How do we move from a model and unlabeled data to labeled data? (Inference)

## Typical Features ( $\phi$ )

---

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i = \log p(z_1 = i)$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j} = \log p(z_n = j | z_{n-1} = i)$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w}$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation: HMM) How do we move from a model and unlabeled data to labeled data? (Inference)

## Typical Features ( $\phi$ )

Assume  $K$  parts of speech, a lexicon size of  $V$ , a series of observations  $\{x_1, \dots, x_N\}$ , and a series of unobserved states  $\{z_1, \dots, z_N\}$ .

$\pi$  Start state scores (vector of length  $K$ ):  $\pi_i = \log p(z_1 = i)$

$\theta$  Transition matrix (matrix of size  $K$  by  $K$ ):  $\theta_{i,j} = \log p(z_n = j | z_{n-1} = i)$

$\beta$  An emission matrix (matrix of size  $K$  by  $V$ ):  $\beta_{j,w} = \log p(x_n = w | z_n = j)$

### Score

$$f(x, z) \equiv \sum_i w_i \phi_i(x, z) \quad (1)$$

Two problems: How do we move from data to algorithm? (Estimation: HMM) How do we move from a model and unlabeled data to labeled data? (Inference)

## Viterbi Algorithm

---

- Given an unobserved sequence of length  $L$ ,  $\{x_1, \dots, x_L\}$ , we want to find a sequence  $\{z_1 \dots z_L\}$  with the highest score.

## Viterbi Algorithm

---

- Given an unobserved sequence of length  $L$ ,  $\{x_1, \dots, x_L\}$ , we want to find a sequence  $\{z_1 \dots z_L\}$  with the highest score.
- It's impossible to compute  $K^L$  possibilities.
- So, we use dynamic programming to compute most likely tags for each token subsequence from 0 to  $t$  that ends in state  $k$ .
- Memoization: fill a table of solutions of sub-problems
- Solve larger problems by composing sub-solutions
- Base case:

$$f_1(k) = \pi_k + \beta_{k,x_i} \quad (2)$$

- Recursion:

$$f_n(k) = \max_j (f_{n-1}(j) + \theta_{j,k}) + \beta_{k,x_n} \quad (3)$$

## Viterbi Algorithm

---

- Given an unobserved sequence of length  $L$ ,  $\{x_1, \dots, x_L\}$ , we want to find a sequence  $\{z_1 \dots z_L\}$  with the highest score.
- It's impossible to compute  $K^L$  possibilities.
- So, we use dynamic programming to compute most likely tags for each token subsequence from 0 to  $t$  that ends in state  $k$ .
- Memoization: fill a table of solutions of sub-problems
- Solve larger problems by composing sub-solutions
- Base case:

$$f_1(k) = \pi_k + \beta_{k,x_i} \quad (2)$$

- Recursion:

$$f_n(k) = \max_j (f_{n-1}(j) + \theta_{j,k}) + \beta_{k,x_n} \quad (3)$$



## Viterbi Algorithm

---

- Given an unobserved sequence of length  $L$ ,  $\{x_1, \dots, x_L\}$ , we want to find a sequence  $\{z_1 \dots z_L\}$  with the highest score.
- It's impossible to compute  $K^L$  possibilities.
- So, we use dynamic programming to compute most likely tags for each token subsequence from 0 to  $t$  that ends in state  $k$ .
- Memoization: fill a table of solutions of sub-problems
- Solve larger problems by composing sub-solutions
- Base case:

$$f_1(k) = \pi_k + \beta_{k,x_i} \quad (2)$$

- Recursion:

$$f_n(k) = \max_j (f_{n-1}(j) + \theta_{j,k}) + \beta_{k,x_n} \quad (3)$$

## Viterbi Algorithm

---

- Given an unobserved sequence of length  $L$ ,  $\{x_1, \dots, x_L\}$ , we want to find a sequence  $\{z_1 \dots z_L\}$  with the highest score.
- It's impossible to compute  $K^L$  possibilities.
- So, we use dynamic programming to compute most likely tags for each token subsequence from 0 to  $t$  that ends in state  $k$ .
- Memoization: fill a table of solutions of sub-problems
- Solve larger problems by composing sub-solutions
- Base case:

$$f_1(k) = \pi_k + \beta_{k,x_i} \quad (2)$$

- Recursion:

$$f_n(k) = \max_j (f_{n-1}(j) + \theta_{j,k}) + \beta_{k,x_n} \quad (3)$$

- The complexity of this is now  $K^2L$ .
- In class: example that shows why you need all  $O(KL)$  table cells (garden pathing)
- But just computing the max isn't enough. We also have to remember where we came from. (Breadcrumbs from best previous state.)

$$\Psi_n = \operatorname{argmax}_j f_{n-1}(j) + \theta_{j,k} \quad (4)$$

- The complexity of this is now  $K^2L$ .
- In class: example that shows why you need all  $O(KL)$  table cells (garden pathing)
- But just computing the max isn't enough. We also have to remember where we came from. (Breadcrumbs from best previous state.)

$$\Psi_n = \operatorname{argmax}_j f_{n-1}(j) + \theta_{j,k} \quad (4)$$

- Let's do that for the sentence "come and get it"

POS	$\pi_k$	$\beta_{k,x_1}$	$f_1(k)$
MOD	log 0.234	log 0.024	-5.18
DET	log 0.234	log 0.032	-4.89
CONJ	log 0.234	log 0.024	-5.18
N	log 0.021	log 0.016	-7.99
PREP	log 0.021	log 0.024	-7.59
PRO	log 0.021	log 0.016	-7.99
V	log 0.234	log 0.121	-3.56

**come** and get it (with HMM probabilities)

Why logarithms?

- ① More interpretable than a float with lots of zeros.
- ② Underflow is less of an issue
- ③ Generalizes to linear models (next!)
- ④ Addition is cheaper than multiplication

$$\log(ab) = \log(a) + \log(b) \quad (5)$$

POS	$f_1(j)$		$f_2(\text{CONJ})$
MOD	-5.18		
DET	-4.89		
CONJ	-5.18		
N	-7.99		
PREP	-7.59		
PRO	-7.99		
V	-3.56		

come **and** get it

POS	$f_1(j)$		$f_2(\text{CONJ})$
MOD	-5.18		
DET	-4.89		
CONJ	-5.18		???
N	-7.99		
PREP	-7.59		
PRO	-7.99		
V	-3.56		

come **and** get it

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18		???
DET	-4.89		
CONJ	-5.18		
N	-7.99		
PREP	-7.59		
PRO	-7.99		
V	-3.56		

come **and** get it



POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18		???
DET	-4.89		
CONJ	-5.18		
N	-7.99		
PREP	-7.59		
PRO	-7.99		
V	-3.56		

come **and** get it

$$f_0(V) + \theta_{V, \text{CONJ}} = f_0(k) + \theta_{V, \text{CONJ}} = -3.56 + -1.65$$

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18		???
DET	-4.89		
CONJ	-5.18		
N	-7.99		
PREP	-7.59		
PRO	-7.99		
V	-3.56	-5.21	

come **and** get it

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18		
DET	-4.89		
CONJ	-5.18		???
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	-5.21	

come **and** get it

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18	-8.48	???
DET	-4.89	-7.72	
CONJ	-5.18	-8.47	
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	-5.21	

come **and** get it

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18	-8.48	???
DET	-4.89	-7.72	
CONJ	-5.18	-8.47	
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	-5.21	

come **and** get it

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18	-8.48	
DET	-4.89	-7.72	
CONJ	-5.18	-8.47	
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	-5.21	

come **and** get it

$$\log f_1(k) = -5.21 + \beta_{\text{CONJ}}, \text{ and } =$$

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18	-8.48	
DET	-4.89	-7.72	
CONJ	-5.18	-8.47	
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	-5.21	

come **and** get it

$$\log f_1(k) = -5.21 + \beta_{\text{CONJ}}, \text{ and } = -5.21 - 0.64$$

POS	$f_1(j)$	$f_1(j) + \theta_{j,\text{CONJ}}$	$f_2(\text{CONJ})$
MOD	-5.18	-8.48	<b>-6.02</b>
DET	-4.89	-7.72	
CONJ	-5.18	-8.47	
N	-7.99	$\leq -7.99$	
PREP	-7.59	$\leq -7.59$	
PRO	-7.99	$\leq -7.99$	
V	-3.56	<b>-5.21</b>	

come **and** get it



POS	$f_1(k)$	$f_2(k)$	$b_2$	$f_3(k)$	$b_3$	$f_4(k)$	$b_4$
MOD	-5.18	-6.02	V				
DET	-4.89						
CONJ	-5.18						
N	-7.99						
PREP	-7.59						
PRO	-7.99						
V	-3.56						
WORD	come	and		get		it	

POS	$f_1(k)$	$f_2(k)$	$b_2$	$f_3(k)$	$b_3$	$f_4(k)$	$b_4$
MOD	-5.18	-0.00	X				
DET	-4.89	-0.00	X				
CONJ	-5.18	-6.02	V				
N	-7.99	-0.00	X				
PREP	-7.59	-0.00	X				
PRO	-7.99	-0.00	X				
V	-3.56	-0.00	X				
WORD	come	and		get		it	

POS	$f_1(k)$	$f_2(k)$	$b_2$	$f_3(k)$	$b_3$	$f_4(k)$	$b_4$
MOD	-5.18	-0.00	X	-0.00	X		
DET	-4.89	-0.00	X	-0.00	X		
CONJ	-5.18	-6.02	V	-0.00	X		
N	-7.99	-0.00	X	-0.00	X		
PREP	-7.59	-0.00	X	-0.00	X		
PRO	-7.99	-0.00	X	-0.00	X		
V	-3.56	-0.00	X	-9.03	CONJ		
WORD	come	and		get		it	

POS	$f_1(k)$	$f_2(k)$	$b_2$	$f_3(k)$	$b_3$	$f_4(k)$	$b_4$
MOD	-5.18	-0.00	X	-0.00	X	-0.00	X
DET	-4.89	-0.00	X	-0.00	X	-0.00	X
CONJ	-5.18	-6.02	V	-0.00	X	-0.00	X
N	-7.99	-0.00	X	-0.00	X	-0.00	X
PREP	-7.59	-0.00	X	-0.00	X	-0.00	X
PRO	-7.99	-0.00	X	-0.00	X	-14.6	V
V	-3.56	-0.00	X	-9.03	CONJ	-0.00	X
WORD	come	and		get		it	