COMP5046
Natural Language Processing

Lecture 6: Part-of-Speech (POS) Tagging

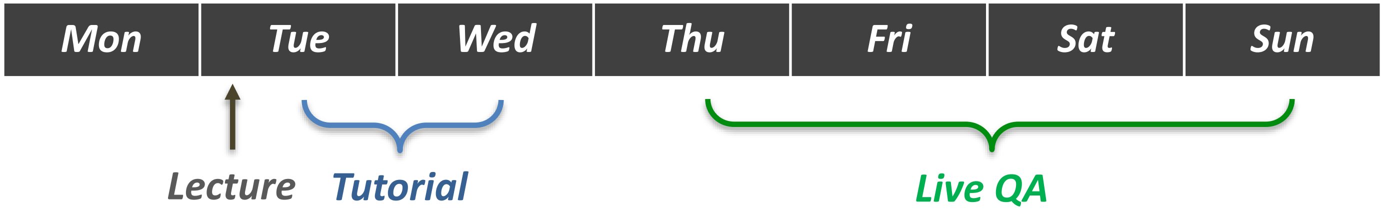
Semester 1, 2020
School of Computer Science
The University of Sydney, Australia

THE UNIVERSITY OF SYDNEY

Dr Caren Han
Caren.Han@sydney.edu.au

1 The course update

Online Teaching Update



Mon | Tue | Wed | Thu | Fri | Sat | Sun

Lecture *Tutorial* *Live QA*

Lecture:

- *Tue 2-4pm every week*
- *Live Streaming via Twitch*
- *Recording file via canvas*

Tutorial:

- *Tue 4pm, 5pm, 6pm*
- *Wed 4pm, 5pm*
- *Live Streaming via Zoom*

Live QA:

- *9-10pm from Thu, Fri, Sat, Sun*
- *Live Streaming via Zoom*

If you have any other questions, please post it to ***EdStem***

More detailed information can be found from https://edstem.org/courses/3933/discussion/188336

1 The course update

Assessment Update

[Original Assessment Plan]

Assessment	Weight	Due	Length
Lab exercises	10%	Multiple weeks (By Monday 6pm or Tuesday 6pm)	n/a
Assignment 1	20%	Week 8	n/a
Assignment 2	20%	Week 14	n/a
Final exam	50%	Formal Exam Period	2 hours

To pass UoS, achieve at least 40% (20 out of 50)

To pass UoS, achieve at least 40% (20 out of 50)

[Updated Assessment Plan]

Assessment	Weight	Due	Length
Lab exercises	10%	**Multiple weeks** **(All by Tuesday 6pm)**	n/a
Assignment 1	20%	Week 8	n/a
Assignment 2	30%	Week 14	n/a
Oral Quiz – Case Study (5Qs interview style)	15%	1 day after the formal exam period via Zoom [the exact date will be announced soon]	15 mins
Take-home Exam (Open Book)	25%	Formal exam period	1 hours

To pass UoS, achieve at least 40% (24 out of 60)

To pass UoS, achieve at least 40% (16 out of 40)

Sample Q would be demonstrated in Week 13.

1 The course topics

What will you learn in this course?

- Week 1: Introduction to Natural Language Processing (NLP)
- Week 2: Word Embeddings (Word Vector for Meaning)
- Week 3: Word Classification with Machine Learning I
- Week 4: Word Classification with Machine Learning II

NLP and Machine Learning

- Week 5: Language Fundamental
- Week 6: Part of Speech Tagging
- Week 7: Dependency Parsing
- Week 8: Language Model

NLP Techniques

- Week 9: Information Extraction I: Named Entity Recognition
- Week 10: Information Extraction II: Relation Extraction
- Week 11: Application I: Question and Answering
- Week 12: Application II: Machine Translation

Advanced Topic

- Week 13: Future of NLP and Exam Review

0 LECTURE PLAN

Lecture 6: Part of Speech Tagging

1. **Part-of-Speech Tagging**
2. Baseline Approaches
   1. Rule-based Model
   2. Look-up Table Model
   3. N-Gram Model
3. Probabilistic Approaches
   1. Hidden Markov Model
   2. Conditional Random Field
4. Deep Learning Approaches

1 Part-of-Speech Tagging

Parts of Speech (or word classes)

A class of words based on the word's function, the way it works in a sentence

8 parts of speech are commonly listed

2000 years ago (starting with Aristotle)

Nouns	*Verbs*	*Pronouns*	*Prepositions*
Adverbs	*Conjunctions*	*Participles*	*Articles*

Dionysius Thrax of Alexandria (c. 100 BCE)

Now (School Grammar) *+ articles or determiner*

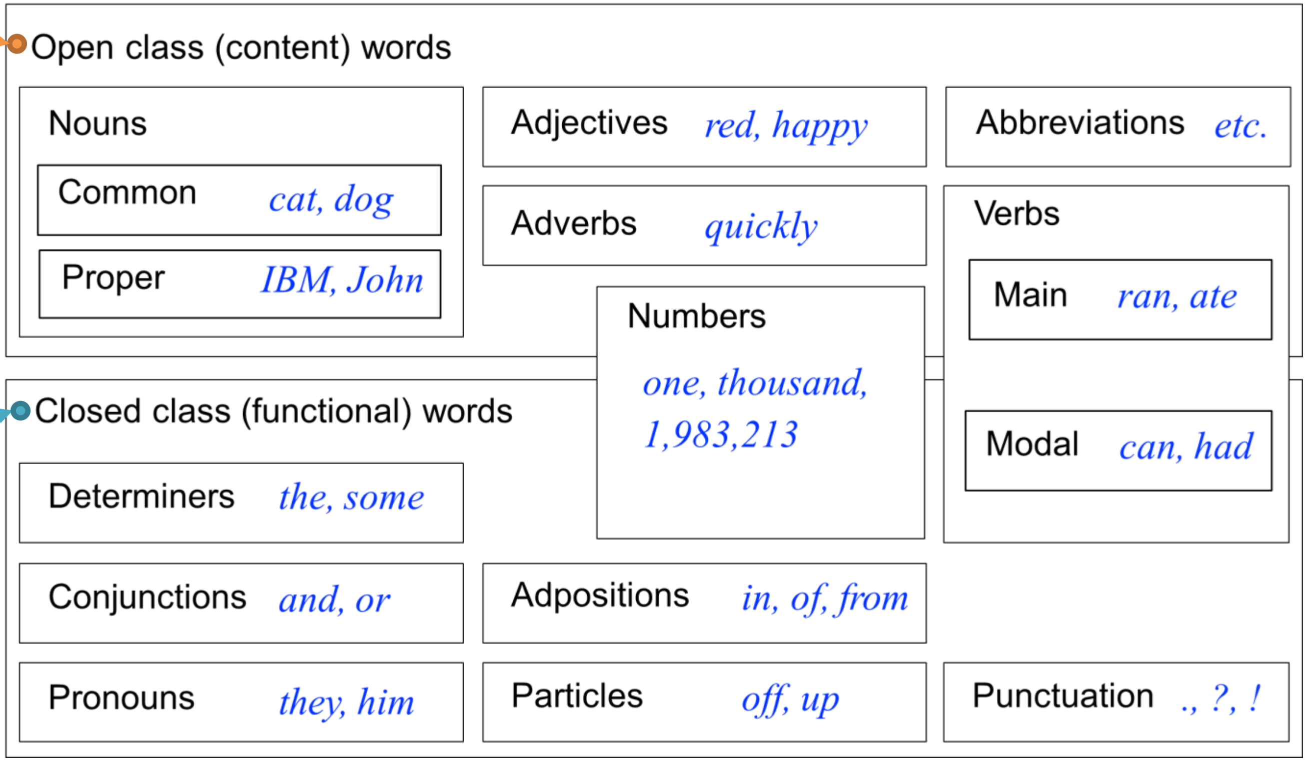
Nouns	*Verbs*	*Pronouns*	*Prepositions*
Adverbs	*Conjunctions*	*Adjectives*	*Interjections*

1 Part-of-Speech Tagging

Part-of-Speech (English)

One basic kind of linguistic structure: syntactic word classes

typically large, have fluid membership, and are often stable under translation



Relatively fixed membership, and the repertoire differs more from language to language

Part-of-Speech Tag sets – Modern English

In modern (English) NLP, larger and more fine-grained tag sets are preferred.

Example

Penn Treebank	*45 tags*	http://bit.ly/1gwbird
Brown Corpus	*87 tags*	https://bit.ly/2FGtdLd
C7 Tagset	*146 tags*	http://bit.ly/1Mh36KX

Trade-off between complexity and precision and whatever tag-set we use, there will be some words that are hard to classify.

1 Part-of-Speech Tagging

Criteria for part-of-speech tagging

Three different criteria might be considered.

- ***Distributional*** criteria: Where can the words occur?

- ***Morphological*** criteria: What form does the word have? (E.g. -tion, -ize). What affixes can it take? (E.g. -s, -ing, -est).

- ***Notional(or semantic)*** criteria: What sort of concept does the word refer to? (E.g. nouns often refer to ‘people, places or things’). More problematic: less useful for us

Part-of-Speech Tagging

Criteria for part-of-speech tagging: Nouns

Three different criteria might be considered.

- ***Distributional*** criteria: Where can the nouns appear?

  For example, nouns can appear with possession: “his car”, ”her idea”.

- ***Morphological*** criteria: What form does the word have? (E.g. -tion, -ize). What affixes can it take? (E.g. -s, -ing, -est).

  ness, -tion, -ity, and -ance tend to indicate nouns. (happiness, exertion, levity, significance).

- ***Notional(or semantic)*** criteria: What sort of concept does the word refer to?

  Nouns generally refer to living things (mouse), places (Sydney), non-living things (computer), or concepts (marriage).

1 Part-of-Speech Tagging

Criteria for part-of-speech tagging: Verbs

Three different criteria might be considered.

- ***Distributional*** criteria: Where can the verbs appear?

  Different types of verbs have different distributional properties. For example, base form verbs can appear as infinitives: "to jump", "to learn".

- ***Morphological*** criteria: What form does the word have? (E.g. -tion, -ize). What affixes can it take? (E.g. -s, -ing, -est).

  words that end in -ate or -ize tend to be verbs, and ones that end in -ing are often the present participle of a verb (automate, equalize; rising, washing)

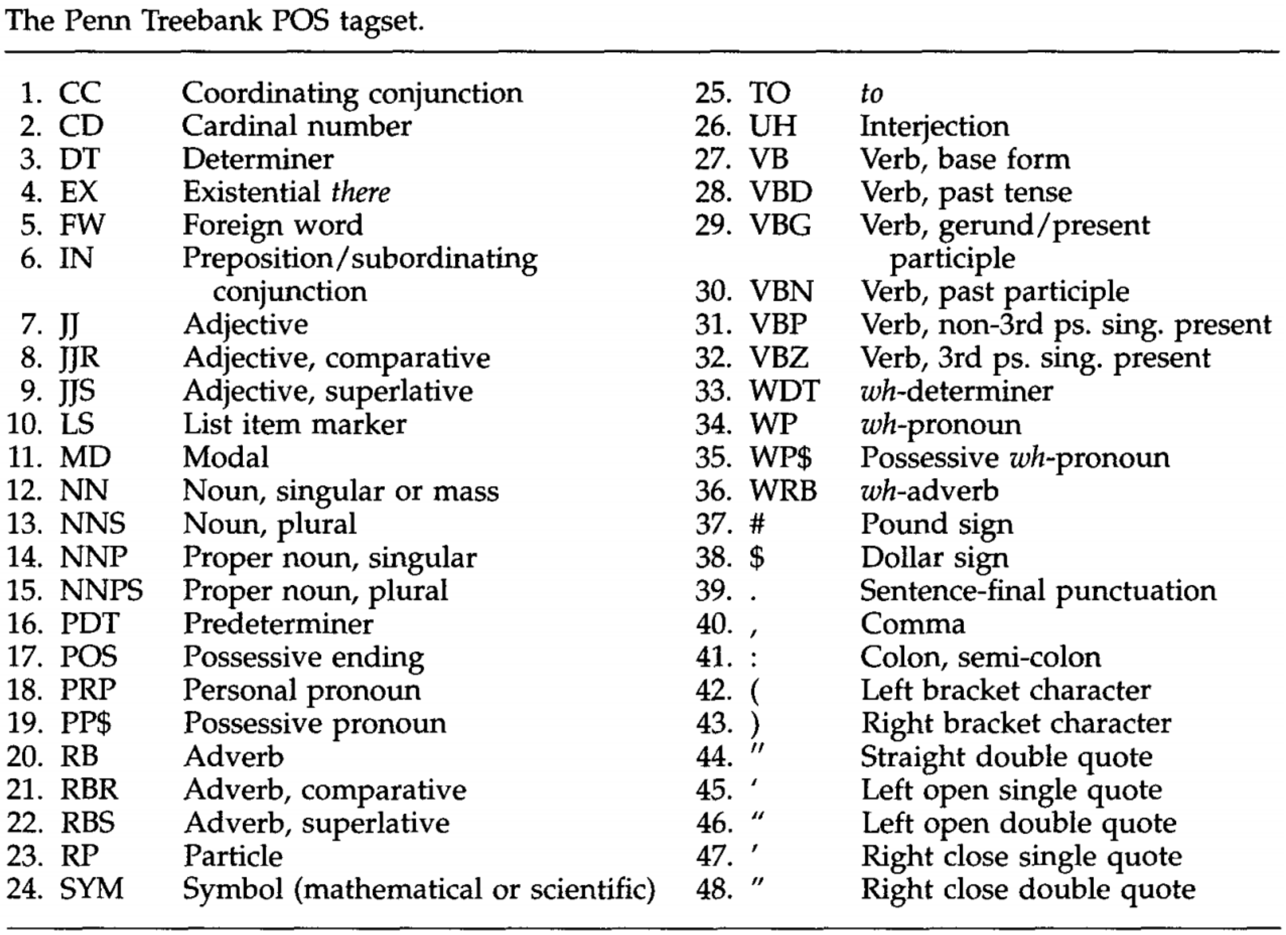
- ***Notional(or semantic)*** criteria: What sort of concept does the word refer to?

  Verbs refer to actions (observe, think, give).

1

Part-of-Speech Tagging

POS tags in Penn Treebank



The Penn Treebank POS tagset.

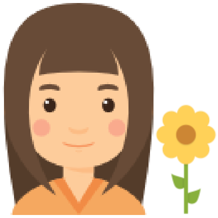
1.	CC	Coordinating conjunction	25.	TO	*to*
2.	CD	Cardinal number	26.	UH	Interjection
3.	DT	Determiner	27.	VB	Verb, base form
4.	EX	Existential *there*	28.	VBD	Verb, past tense
5.	FW	Foreign word	29.	VBG	Verb, gerund/present participle
6.	IN	Preposition/subordinating conjunction	30.	VBN	Verb, past participle
7.	JJ	Adjective	31.	VBP	Verb, non-3rd ps. sing. present
8.	JJR	Adjective, comparative	32.	VBZ	Verb, 3rd ps. sing. present
9.	JJS	Adjective, superlative	33.	WDT	*wh*-determiner
10.	LS	List item marker	34.	WP	*wh*-pronoun
11.	MD	Modal	35.	WP$	Possessive *wh*-pronoun
12.	NN	Noun, singular or mass	36.	WRB	*wh*-adverb
13.	NNS	Noun, plural	37.	#	Pound sign
14.	NNP	Proper noun, singular	38.	$	Dollar sign
15.	NNPS	Proper noun, plural	39.	.	Sentence-final punctuation
16.	PDT	Predeterminer	40.	,	Comma
17.	POS	Possessive ending	41.	:	Colon, semi-colon
18.	PRP	Personal pronoun	42.	(	Left bracket character
19.	PP$	Possessive pronoun	43.	)	Right bracket character
20.	RB	Adverb	44.	″	Straight double quote
21.	RBR	Adverb, comparative	45.	‘	Left open single quote
22.	RBS	Adverb, superlative	46.	“	Left open double quote
23.	RP	Particle	47.	’	Right close single quote
24.	SYM	Symbol (mathematical or scientific)	48.	”	Right close double quote

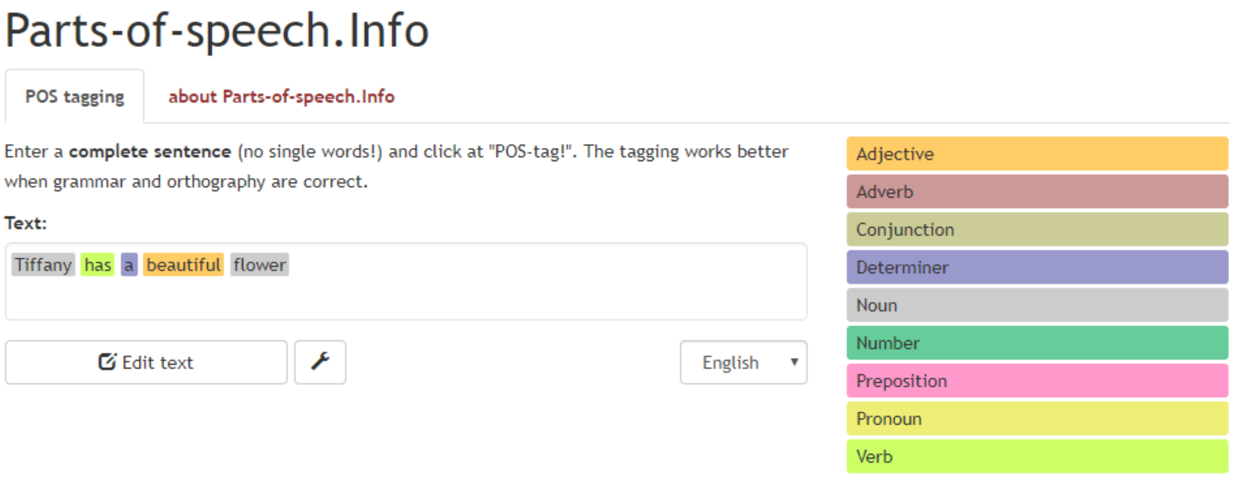
Marcus, et al. (1993)

1 Part-of-Speech Tagging

Example of POS inference

Emma	*has*	*a*	*beautiful*	*flower*
NNP	**VBZ**	**DT**	**JJ**	**NN**





Parts-of-speech.Info

POS tagging | about Parts-of-speech.Info

Enter a **complete sentence** (no single words!) and click at "POS-tag!". The tagging works better when grammar and orthography are correct.

Text:

Tiffany has a beautiful flower

Edit text

English

- Adjective
- Adverb
- Conjunction
- Determiner
- Noun
- Number
- Preposition
- Pronoun
- Verb

https://parts-of-speech.info/ , *StanfordNLP(2018)*

POS Tagging: Issue

Given an input text, tag each word correctly:

There/ was/ still/ lemonade/ in/ the/ ***bottle****/*

- (Tag sets are quite counterintuitive!)
  - In the above, the ***bottle*** is a noun not a verb
    - *but how does our tagger tell?*
  - The *still* could be an adjective or an adverb
    - *which seems more likely?*

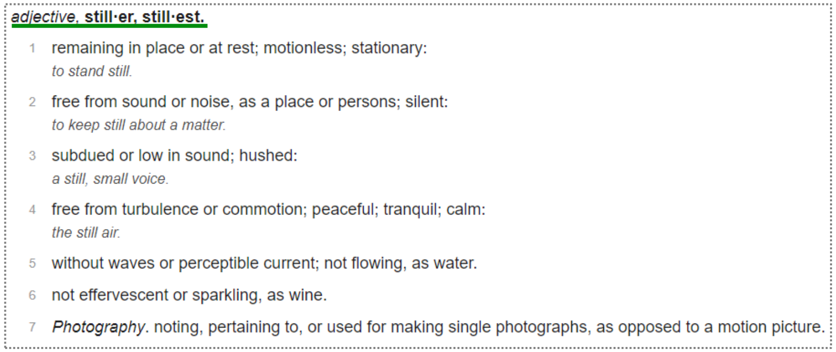
1 Part-of-Speech Tagging

POS Tagging: Issue

Given an input text, tag each word correctly:

*There/ was/ **still**/ lemonade/ in/ the/ **bottle**/*

- (Tag sets are quite counterintuitive!)
  - In the above, the ***bottle*** is a noun not a verb
    - *but how does our tagger tell?*
  - The ***still*** could be an adjective or an adverb
    - *which seems more likely?*



adjective, **still·er, still·est.**

1 remaining in place or at rest; motionless; stationary:
to stand still.

2 free from sound or noise, as a place or persons; silent:
to keep still about a matter.

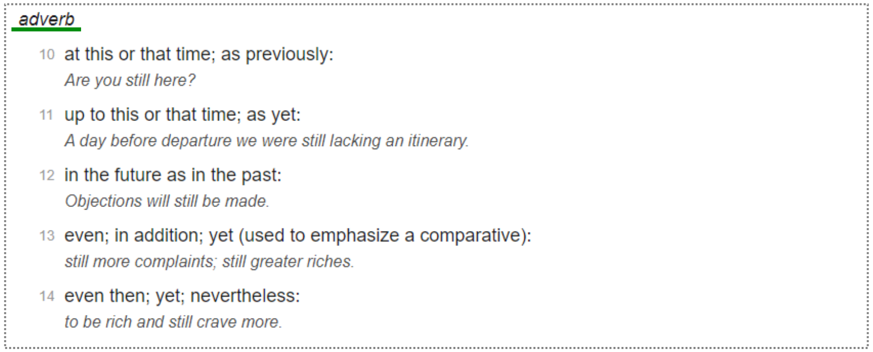
3 subdued or low in sound; hushed:
a still, small voice.

4 free from turbulence or commotion; peaceful; tranquil; calm:
the still air.

5 without waves or perceptible current; not flowing, as water.

6 not effervescent or sparkling, as wine.

7 *Photography*. noting, pertaining to, or used for making single photographs, as opposed to a motion picture.



adverb

10 at this or that time; as previously:
Are you still here?

11 up to this or that time; as yet:
A day before departure we were still lacking an itinerary.

12 in the future as in the past:
Objections will still be made.

13 even; in addition; yet (used to emphasize a comparative):
still more complaints; still greater riches.

14 even then; yet; nevertheless:
to be rich and still crave more.

The purpose of POS Tagging

Essential ingredient in natural language applications

- Useful in and of itself (more than you’d think)
  - Text-to-speech: record, lead
  - Lemmatization: saw[v] see, saw[n] saw
  - Linguistically motivated word clustering

- Useful as a pre-processing step for parsing

- Useful as features to downstream systems.

0 LECTURE PLAN

Lecture 6: Part of Speech Tagging

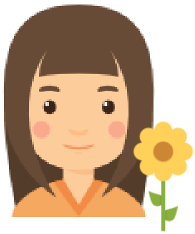
1. Part-of-Speech Tagging
2. **Baseline Approaches**
    1. Rule-based Model
    2. Look-up Table Model
    3. N-Gram Model
3. Probabilistic Approaches
    1. Hidden Markov Model
    2. Conditional Random Field
4. Deep Learning Approaches

2 Baseline Approaches

Part of Speech Tagging

N V D A N

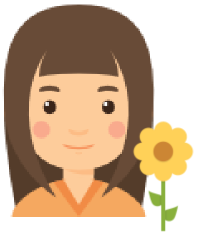
Emma has a beautiful flower

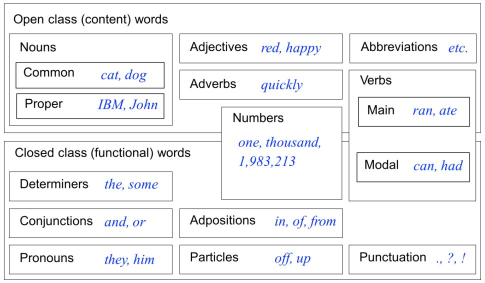


2 Baseline Approaches

Part of Speech Tagging

N	V	D	A	N
Emma	has	a	beautiful	flower





Open class (content) words

Nouns

Common *cat, dog*

Proper *IBM, John*

Adjectives *red, happy*

Adverbs *quickly*

Numbers *one, thousand, 1,983,213*

Abbreviations *etc.*

Verbs

Main *ran, ate*

Modal *can, had*

Closed class (functional) words

Determiners *the, some*

Conjunctions *and, or*

Pronouns *they, him*

Adpositions *in, of, from*

Particles *off, up*

Punctuation *., ?, !*

NOTE: The example includes the high level of classes from the PoS tagset

Rule-based POS Tagging

Basic idea:

Old POS taggers used to work in two stages, based on hand-written rules:

- the first stage identifies a set of possible POS for each word in the sentence (based on a lexicon), and
- the second uses a set of hand-crafted rules in order to select a POS from each of the lists for each word

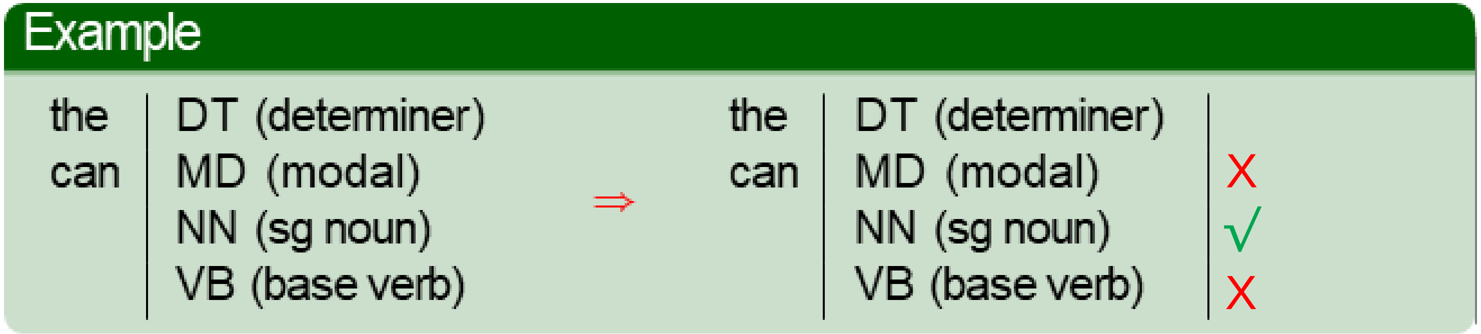
IF **Condition**,
Then **Conclusion**

2 Baseline Approaches

Rule-based POS Tagging

Basic idea:

- Assign each token all its possible tags.
- Apply rules that eliminate all tags for a token that are inconsistent with its context.



Example

the	DT (determiner)		the	DT (determiner)	
can	MD (modal)	⇒	can	MD (modal)	X
	NN (sg noun)			NN (sg noun)	√
	VB (base verb)			VB (base verb)	X

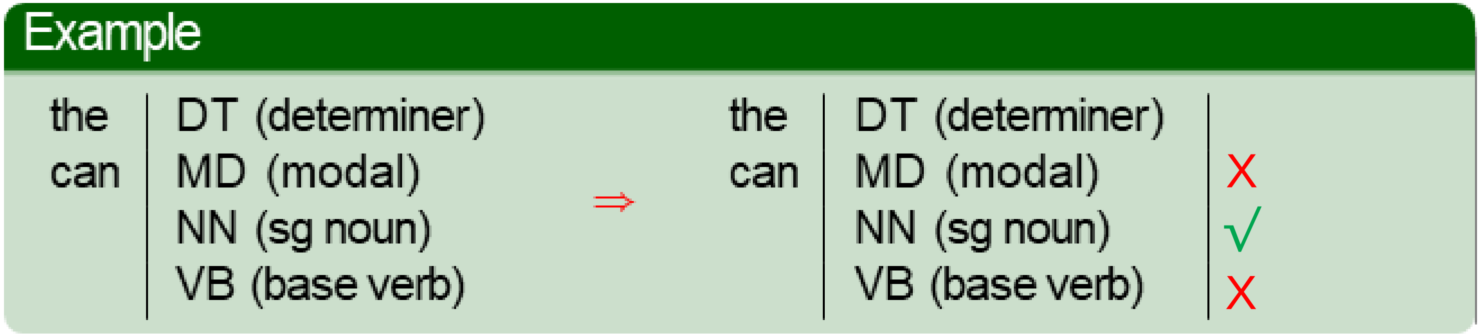
- Assign any unknown word tokens a tag that is consistent with its context (eg, the most frequent tag).

2 Baseline Approaches

Rule-based POS Tagging

- Rule-based tagging often used a large set of hand-crafted context-sensitive rules.

Example (schematic):



Example

the	DT (determiner)		the	DT (determiner)	
can	MD (modal)	⇒	can	MD (modal)	X
	NN (sg noun)			NN (sg noun)	√
	VB (base verb)			VB (base verb)	X

"Cannot eliminate all POS ambiguity."

2 Baseline Approaches

Part of Speech Tagging

Emma

John

Will

Emma likes John

2 Baseline Approaches

Part of Speech Tagging

Emma

John

Will

Emma likes John

Database

John likes Will

N V N

Will likes Emma

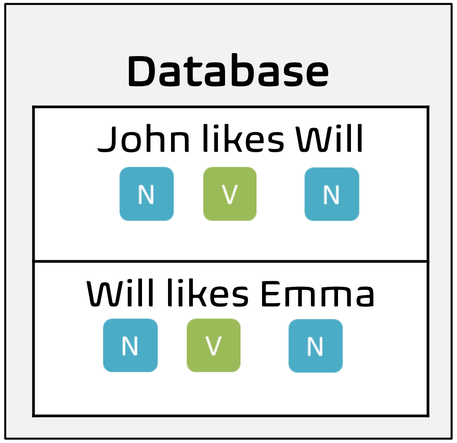
N V N

2 Baseline Approaches

Part of Speech Tagging: Lookup Table

	N	V
John	1	0
liked	0	2
Will	2	0
Emma	1	0

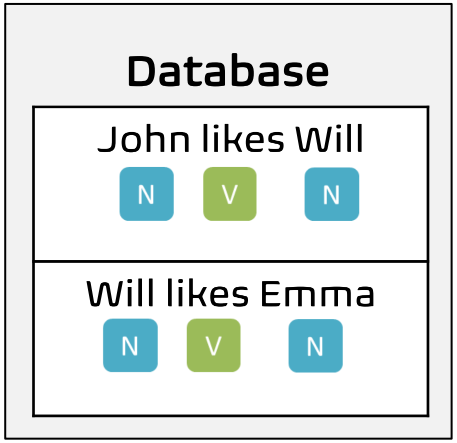
Emma likes John



Part of Speech Tagging: Lookup Table

	N	V
John	**1**	0
liked	0	**2**
Will	**2**	0
Emma	**1**	0

Emma likes John



Database

John likes Will

N V N

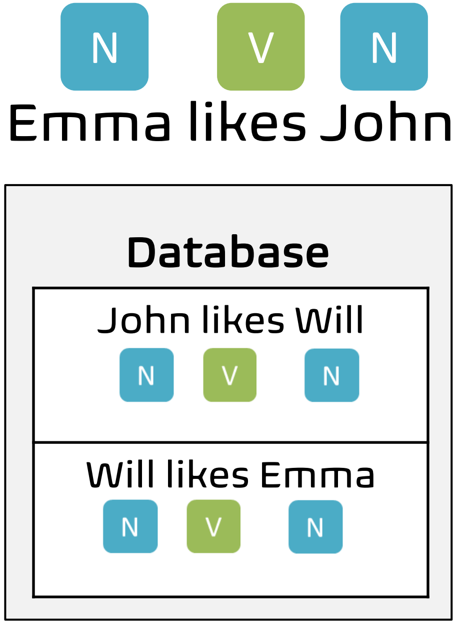
Will likes Emma

N V N

Part of Speech Tagging: Lookup Table

Pick the largest number of the corresponding row

	N	V
John	**1**	0
liked	0	**2**
Will	**2**	0
Emma	**1**	0



N V N

Emma likes John

Database

John likes Will

N V N

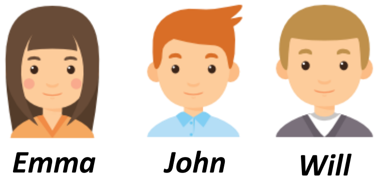
Will likes Emma

N V N

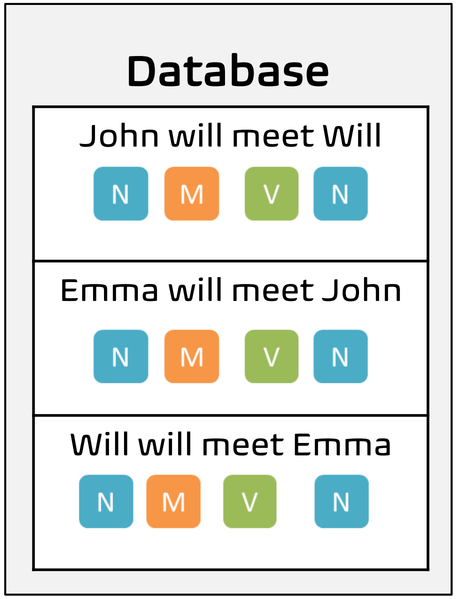
What about more complicated sentences?

2 Baseline Approaches

Part of Speech Tagging



Emma will meet Will

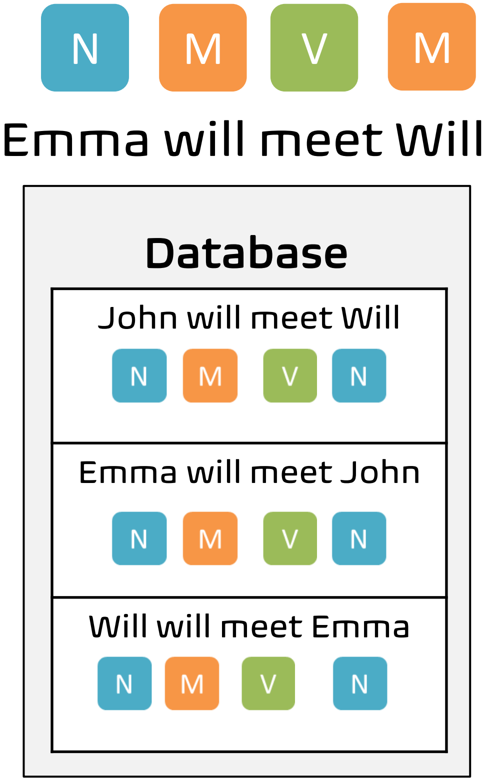


2 Baseline Approaches

Part of Speech Tagging

Pick the largest number of the corresponding row

	N	V	M
John	**2**	0	0
meet	0	**3**	0
Will	2	0	**3**
Emma	**2**	0	0



N M V M

Emma will meet Will

Database

John will meet Will

N M V N

Emma will meet John

N M V N

Will will meet Emma

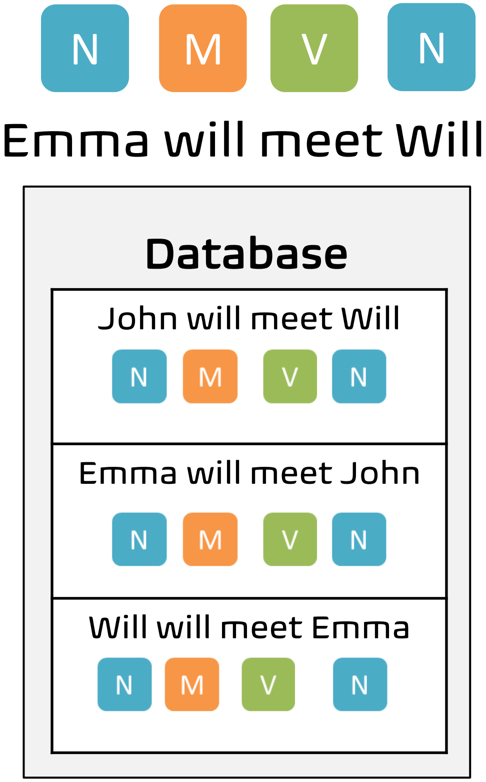
N M V N

2 Baseline Approaches

Part of Speech Tagging

Pick the largest number of the corresponding row

	N	V	M
John	**2**	0	0
meet	0	**3**	0
Will	2	0	**3**
Emma	**2**	0	0



Emma will meet Will

Database

John will meet Will

Emma will meet John

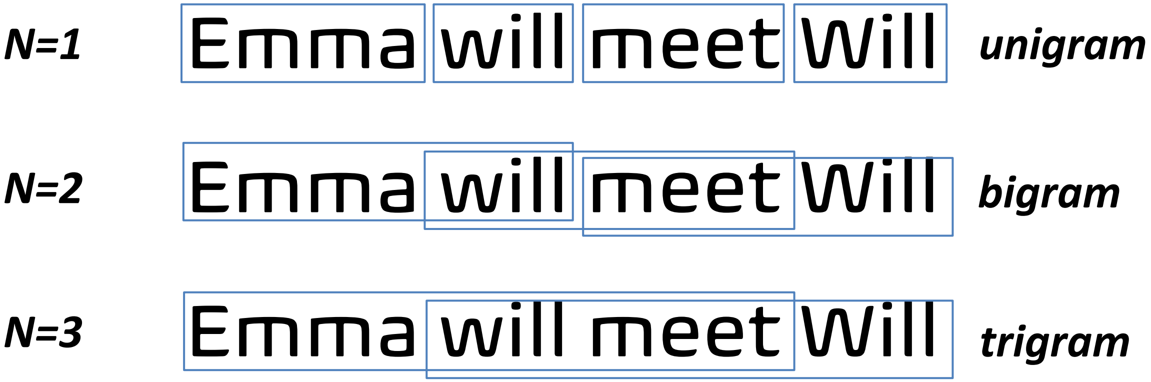
Will will meet Emma

Better Solution? What about considering the Neighbors?

2 Baseline Approaches

Part of Speech Tagging: N-gram

A contiguous sequence of N items from a given sample of text



2 Baseline Approaches

Part of Speech Tagging: N-gram

	N - M	M - V	V - N
john-will	1	0	0
will-meet	0	3	0
meet-will	0	0	1
emma-will	1	0	0
meet-john	0	0	1
will-will	1	0	0
meet-emma	0	0	1

Emma will meet Will

Database

John will meet Will
N M V N

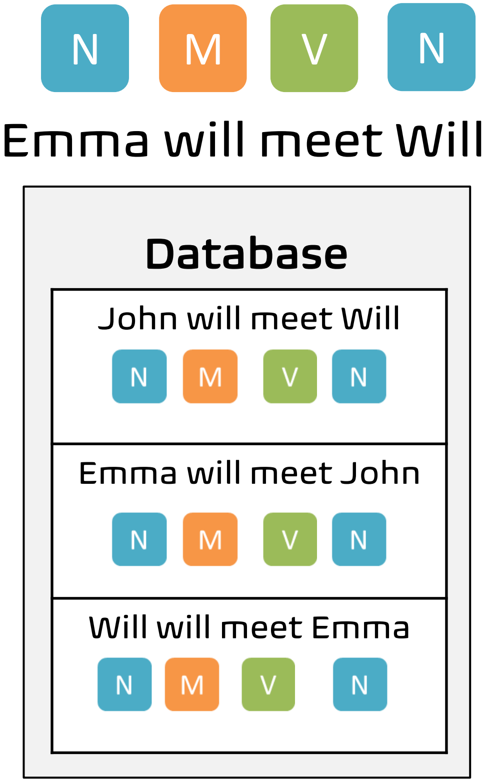
Emma will meet John
N M V N

Will will meet Emma
N M V N

2 Baseline Approaches

Part of Speech Tagging: N-gram

	N - M	M - V	V - N
john-will	1	0	0
will-meet	0	3	0
meet-will	0	0	1
emma-will	1	0	0
meet-john	0	0	1
will-will	1	0	0
meet-emma	0	0	1


N M V N

Emma will meet Will

Database

John will meet Will

N M V N

Emma will meet John

N M V N

Will will meet Emma

N M V N

2 Baseline Approaches

Part of Speech Tagging: N-gram

	N - M	M - V	V - N
john-will	1	0	0
will-meet	0	3	0
meet-will	0	0	1
emma-will	1	0	0
meet-john	0	0	1
will-will	1	0	0
meet-emma	0	0	1

N M V N

Emma will meet Will

Database

John will meet Will

N M V N

Emma will meet John

N M V N

Will will meet Emma

N M V N

What if we don't have in the database?

2 Baseline Approaches

Part of Speech Tagging: N-gram

	N - M	M - V	V - N
john-will	1	0	0
will-meet	0	3	0
meet-will	0	0	1
emma-will	1	0	0
meet-john	0	0	1
will-will	1	0	0
meet-emma	0	0	1

SORRY !!!
No Results Found

Emma will see Will

Database

John will meet Will
N M V N

Emma will meet John
N M V N

Will will meet Emma
N M V N

What if we don't have in the database?

0

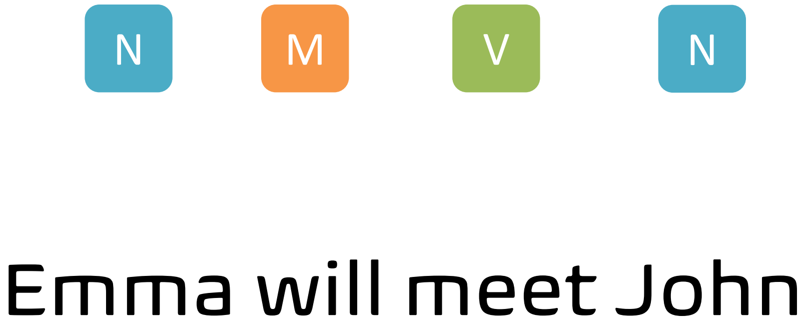
LECTURE PLAN

Lecture 6: Part of Speech Tagging

1. Part-of-Speech Tagging
2. Baseline Approaches
    1. Rule-based Model
    2. Look-up Table Model
    3. N-Gram Model
3. **Probabilistic Approaches**
    1. Hidden Markov Model
    2. Conditional Random Field
4. Deep Learning Approaches

3 Probabilistic Approaches

Hidden Markov Model: Idea

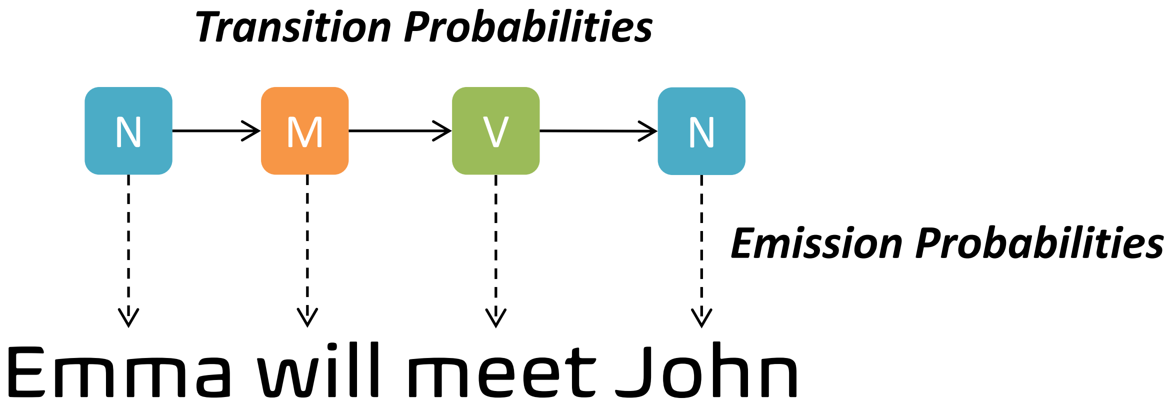


N M V N

Emma will meet John

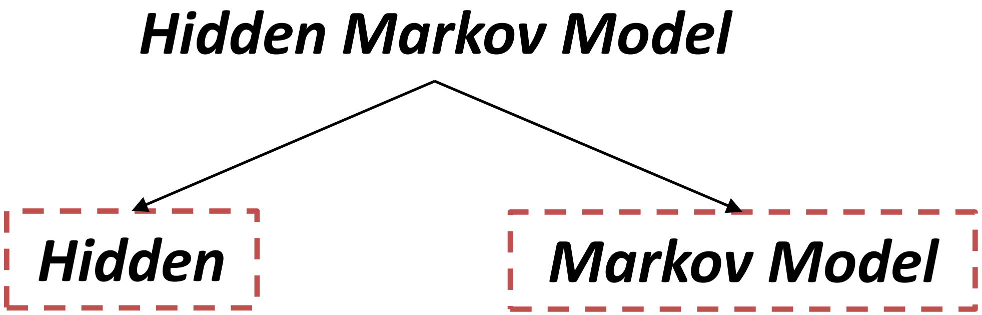
3 Probabilistic Approaches

Hidden Markov Model: Idea



3 Probabilistic Approaches

Hidden Markov Model (HMM)



Hidden Markov Model

Hidden

Markov Model

What is ‘hidden’?

What is ‘Markov Model’?

3 Probabilistic Approaches

Markov Model

Andrei Andreyevich Markov

The purpose of introducing Markov Chain
An example of statistical investigation in the text of `Eugene Onyegin' illustrating coupling of `tests' in chains.

- A stochastic model used to model randomly changing system

- Has the Markov property if the ***conditional probability distribution of future states*** of the process ***depends only upon the present state***, not on the events that occurred before it.

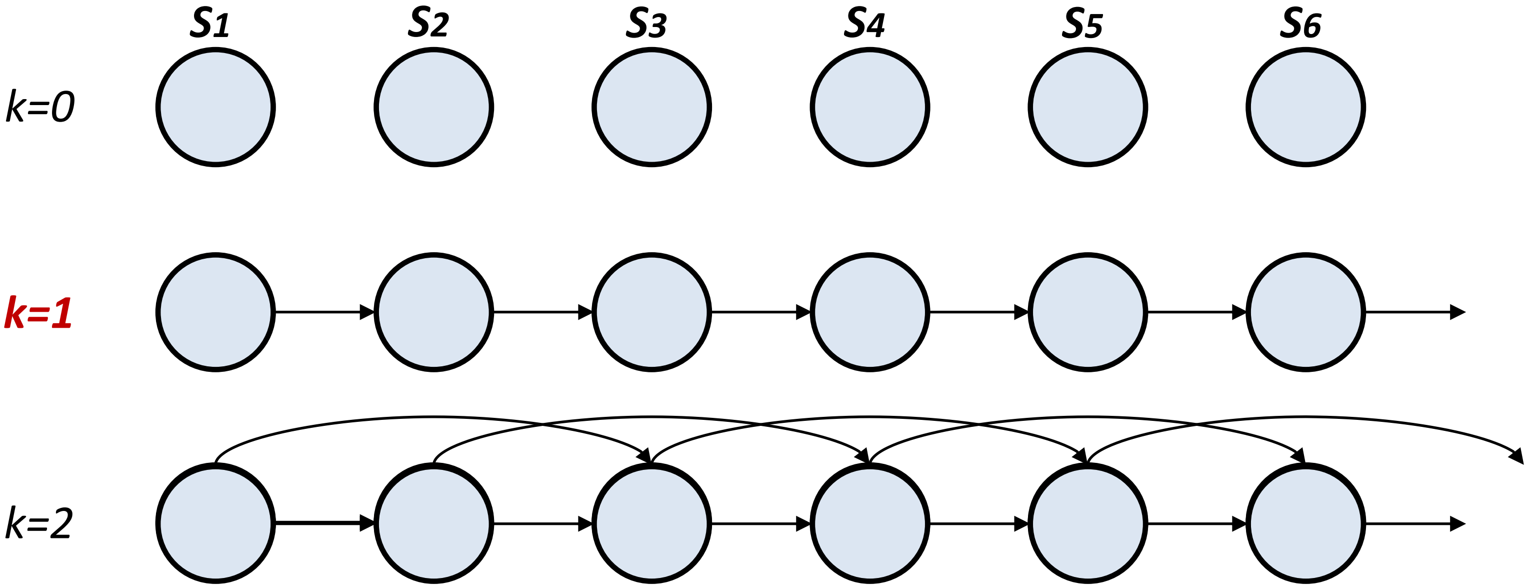
Markov Model (MM): K-Order Markov Property

- Assumption: last ***k*** states are sufficient
  - ***(k=1)*** First-order Markov Process (***Most Commonly used)***

$$P(S_t \mid S_{t-1}, \ldots, S_0) = P(S_t \mid S_{t-1})$$

  - *(k=2)* Second-order Markov Process

$$P(S_t \mid S_{t-1}, \ldots, S_0) = P(S_t \mid S_{t-1}, S_{t-2})$$



Markov Model (MM): Example

Let’s predict tomorrow weather in Sydney. Assume we have three classes

Class 1: Rainy

Class 2: Cloudy

Class 3: Sunny

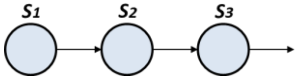
NOTE: Tomorrow weather depends only on today’s!

First order Markov Model

We found the weather change pattern based on the 1-year data.

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

$$S_{ij} = P(S_t = j | S_{t-1} = i)$$



Markov Model (MM): Example

Let’s predict tomorrow weather in Sydney. Assume we have three classes

Class 1: Rainy

Class 2: Cloudy

Class 3: Sunny

NOTE: Tomorrow weather depends only on today’s!

First order Markov Model

If it is raining today, how will be the weather tomorrow?

We found the weather change pattern based on the 1-year data.

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

$$S_{ij} = P(S_t = j | S_{t-1} = i)$$

$$\mathbf{S_{rainyrainy} = 0.4}$$

$$S_{rainycloudy} = 0.3$$

$$S_{rainysunny} = 0.3$$

Markov Model (MM): Example

Let’s predict tomorrow weather in Sydney. Assume we have three classes

Class 1: Rainy

Class 2: Cloudy

Class 3: Sunny

NOTE: Tomorrow weather depends only on today’s!

First order Markov Model

If it is raining today, how will be the weather tomorrow? ***Rainy!***

We found the weather change pattern based on the 1-year data.

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

$$S_{ij} = P(S_t = j | S_{t-1} = i)$$

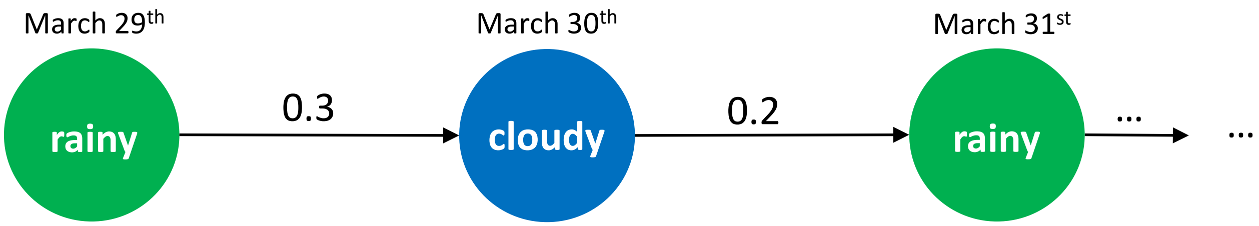
$$\mathbf{S_{rainyrainy} = 0.4}$$

$$S_{rainycloudy} = 0.3$$

$$S_{rainysunny} = 0.3$$

Markov Model (MM): Example

Transition Probabilities



We found the weather change pattern based on the 1-year data.

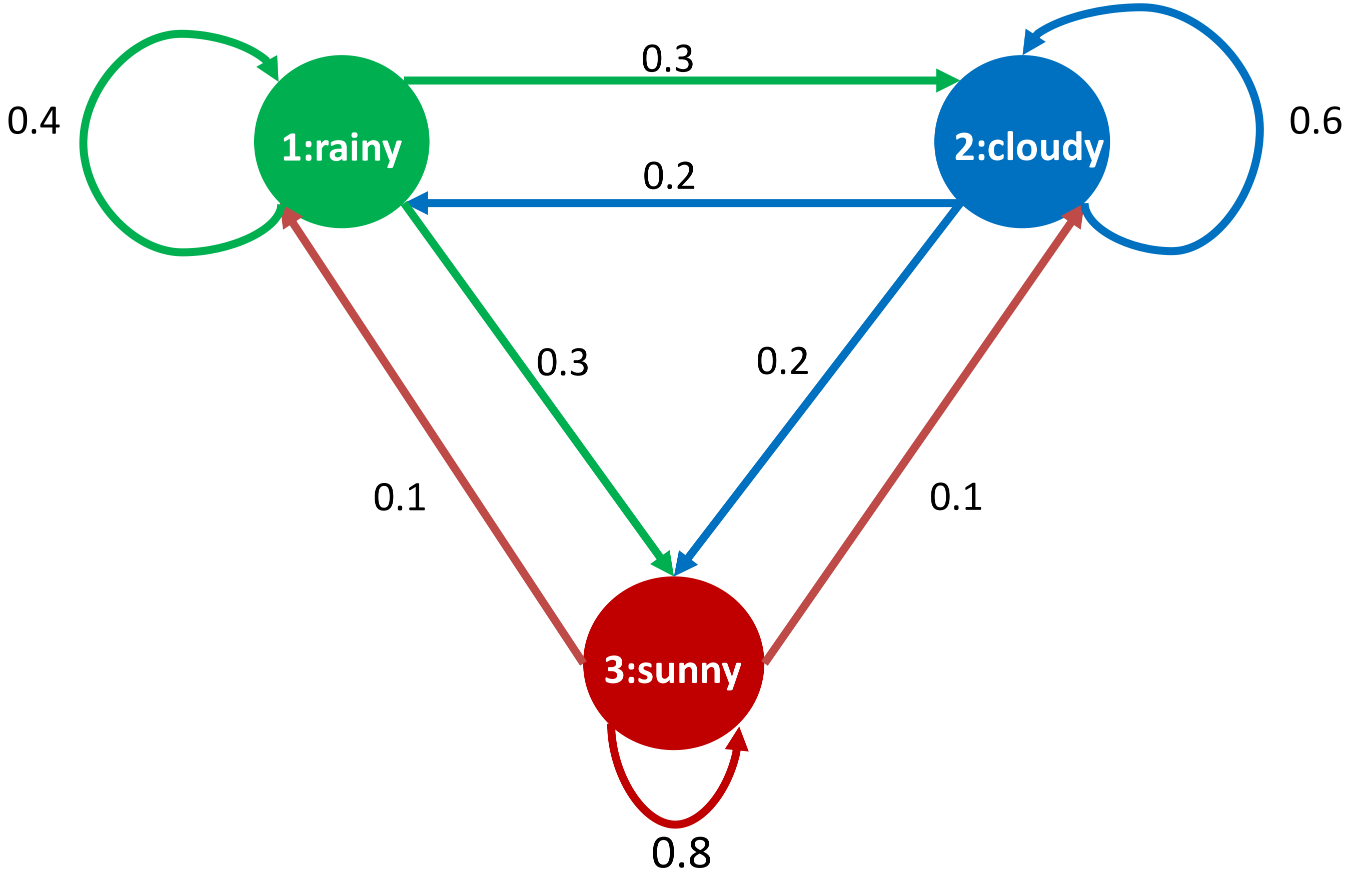
		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

Markov Model (MM): Example

Visual illustration with diagram

- *Each state corresponds to one observation*
- *Sum of outgoing edge weights is one*

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8



3 Probabilistic Approaches

Markov Model (MM): Example

State Transition Matrix

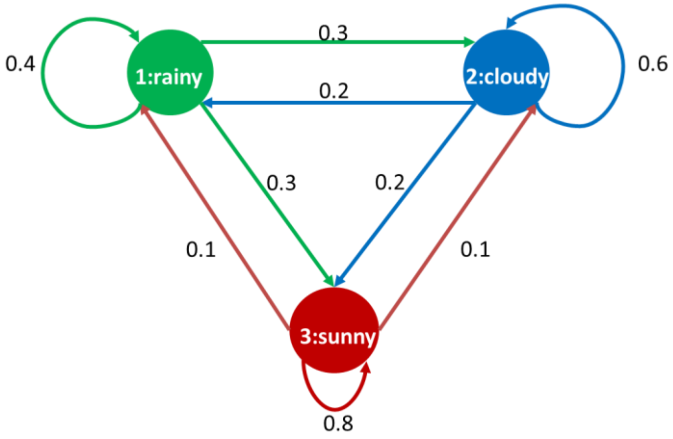
$$S_{ij} = P(S_t = j \,|\, S_{t-1} = i) \quad 1 \le i, \;\; j \ge N$$

$$S_{ij} \ge 0$$

$$S = \begin{bmatrix} S_{11} & S_{12} & \dots & S_{1N} \\ S_{21} & S_{22} & \dots & S_{2N} \\ S_{31} & S_{32} & \dots & S_{3N} \\ \vdots & \vdots & \vdots & \vdots \\ S_{N1} & S_{N2} & \dots & S_{NN} \end{bmatrix}$$

		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.4	0.3	0.3
	Cloudy	0.2	0.6	0.2
	Sunny	0.1	0.1	0.8

		Time ***t+1***		
		S_1	S_2	S_3
Time ***t***	S_1	0.4	0.3	0.3
	S_2	0.2	0.6	0.2
	S_3	0.1	0.1	0.8



Markov Model (MM): Example

Sequence Probability

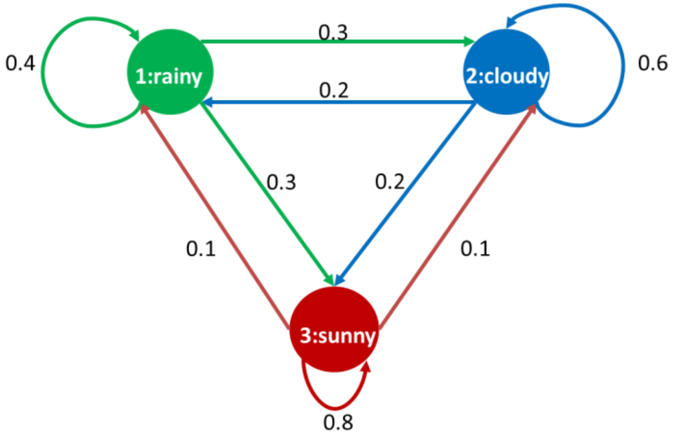
Q: *What is the probability that the weather for the next 7 days will be "sun-sun-rain-rain-sun-cloudy-sun" if it is sunny today?*

$$P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 \mid \text{model})$$

$$= P(S_3) \cdot P(S_3 \mid S_3) \cdot P(S_3 \mid S_3) \cdot P(S_1 \mid S_3) \cdot P(S_1 \mid S_1) P(S_3 \mid S_1) P(S_2 \mid S_3) P(S_3 \mid S_2)$$

$$= 1 \cdot (0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)$$

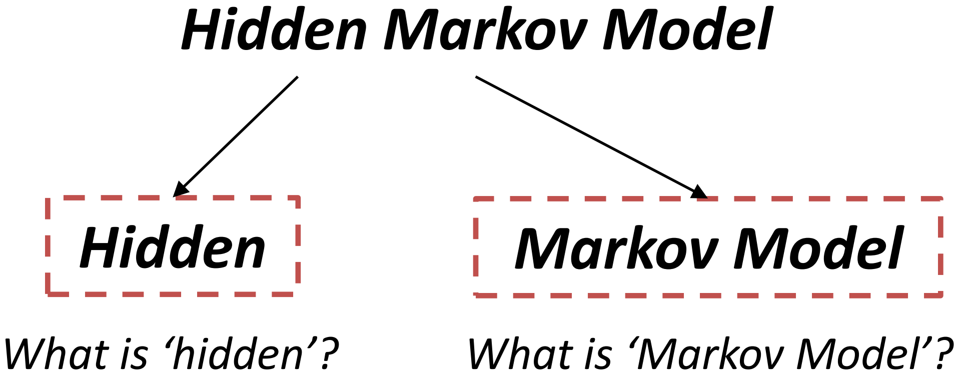
$$= 1.536 \times 10^{-4}$$



$$S_{ij} = P(S_t = j \mid \mathrm{S}_{t-1} = i)$$

3 Probabilistic Approaches

Hidden Markov Model (HMM)



Hidden Markov Model

Hidden

Markov Model

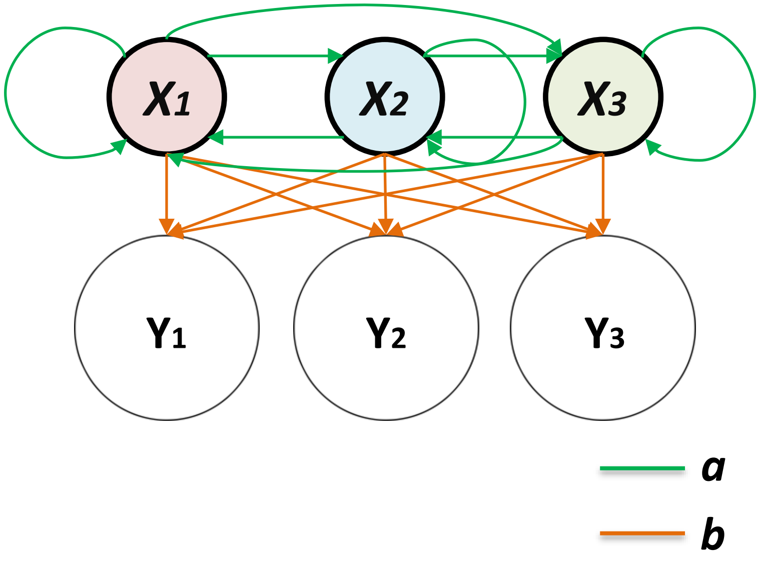
What is ‘hidden’?

What is ‘Markov Model’?

3 Probabilistic Approaches

Hidden Markov Model (HMM)

Hidden Markov Models (HMMs) are a class of probabilistic graphical model that allow us to ***predict a sequence of unknown (hidden) variables*** from a set of observed variables.



hidden

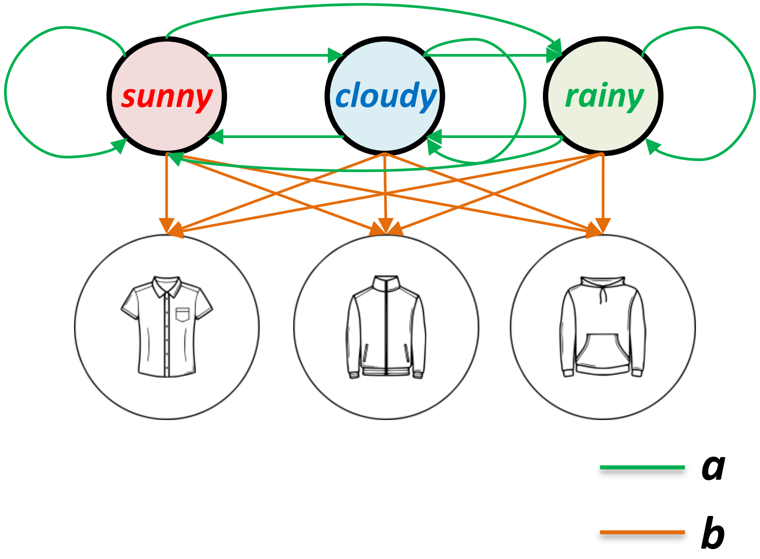
x	states
y	possible observations
a	state transition probabilities
b	output probabilities

- States are ***hidden***
- ***Observable outcome*** linked to states
- Each state has ***observation probabilities*** to determine the observable event

3 Probabilistic Approaches

Hidden Markov Model (HMM)

Hidden Markov Models (HMMs) are a class of probabilistic graphical model that allow us to ***predict a sequence of unknown (hidden) variables*** from a set of observed variables.



hidden

x	states ←
y	possible observations
a	state transition probabilities
b	output probabilities

- States are ***hidden***
- ***Observable outcome*** linked to states
- Each state has ***observation probabilities*** to determine the observable event

3 Probabilistic Approaches

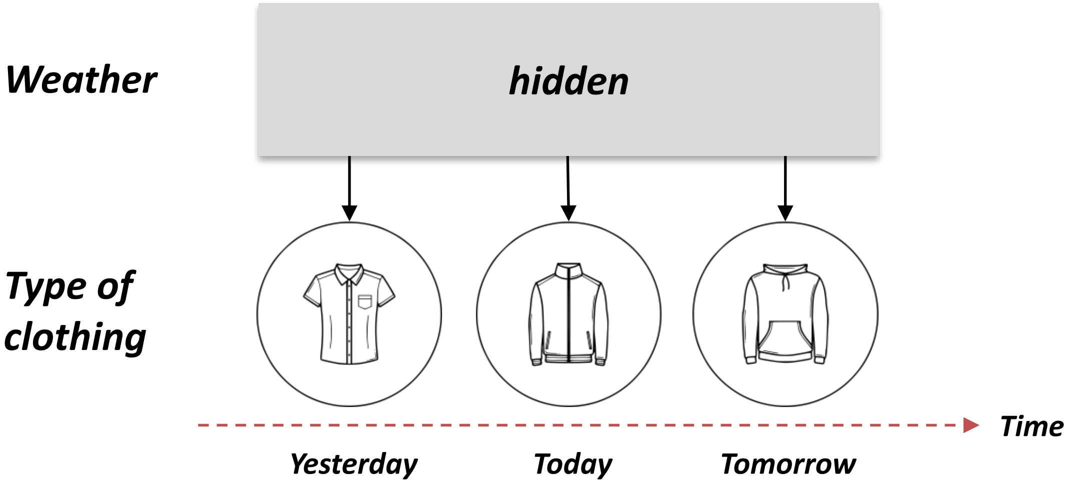
Hidden Markov Model (HMM)

Predicting the **weather** ***(state: hidden variable)*** based on the type of ***clothes that the person wears (observed event)***

- ***Weather (hidden variable): sunny, cloudy, rainy***
- ***Observed variables*** *are the* ***type of clothing the person worn***

The arrows represent:

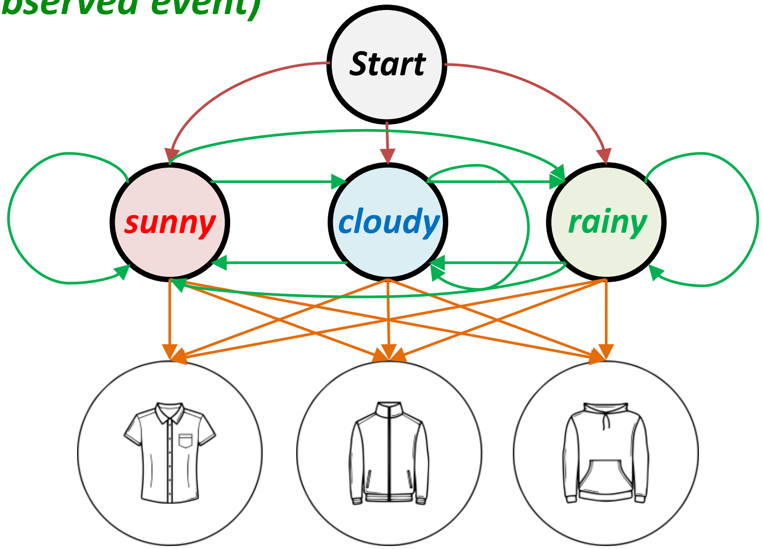
- *Transition Probabilities: from a hidden state to another hidden state*
- *Emission Probabilities: from a hidden state to an observed variable*



One or more observations allow us to make an inference about a sequence of hidden states

Hidden Markov Model (HMM)

Predicting the **weather *(state: hidden variable)*** based on the type of ***clothes that the person wears (observed event)***

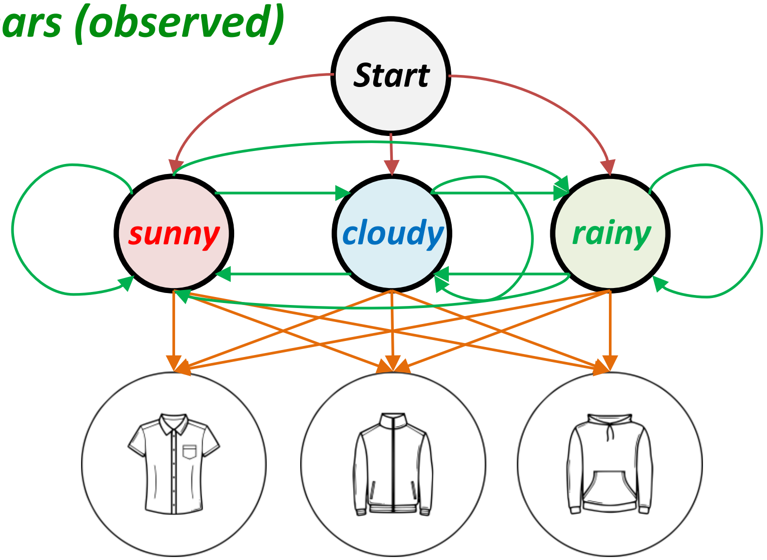


In order to compute the joint probability of a sequence of hidden states, we need to assemble three types of information:

1. ***Initial state information*** *(a.k.a.* ***prior probability****) - The initial probability of transitioning to a hidden state.*
2. ***Transition probabilities*** *— the probability of transitioning to a new state conditioned on a present state*
3. ***Emission probabilities*** *– the probability of transitioning to an observed state conditioned on a hidden state*

Hidden Markov Model (HMM)

Predicting the **weather** ***(hidden variable)*** based on the type of ***clothes that someone wears (observed)***



Priors

Rainy	0.6
Cloudy	0.3
Sunny	0.1

Transitions

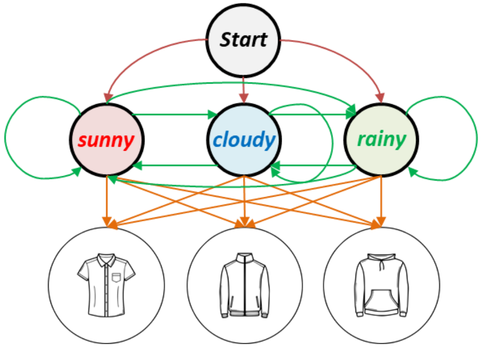
		Tomorrow		
		Rainy	Cloudy	Sunny
Today	Rainy	0.6	0.3	0.1
	Cloudy	0.4	0.3	0.2
	Sunny	0.1	0.4	0.5

Emissions

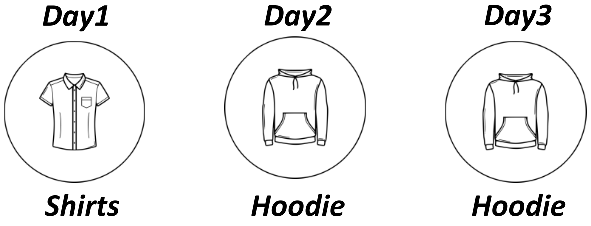
	Shirts	Jacket	Hoodies
Rainy	0.8	0.19	0.01
Cloudy	0.5	0.4	0.1
Sunny	0.01	0.2	0.79

3 Probabilistic Approaches

Hidden Markov Model (HMM)



We had the list of clothes that Caren wears for three days



Firstly, just calculate the weather condition 'cloudy-cloudy-sunny' (Random Selection)

1. Calculate the probability that Caren could wear that clothing
(with the weather condition 'cloudy – cloudy – sunny')

*P(shirts|cloudy)*P(hoodie|cloudy)*P(hoodie|sunny)*

2. Calculate the probability that weathers were 'cloudy – cloudy – sunny'

P(prior_cloudy) P(cloudy|cloudy) * P(sunny|cloudy)*

*P(shirts|cloudy)*P(hoodie|cloudy)*P(hoodie|sunny)* P(prior_cloudy)* P(cloudy|cloudy) * P(sunny|cloudy)*

This is the probability when we assume the weather (cloudy – cloudy – sunny)

3 Probabilistic Approaches

Hidden Markov Model (HMM)

The previous was the only probability when we assume the weather (cloudy – cloudy – sunny)

This is a complete set of 3^3=27 cases of weather states for three days:

{x1=s1=sunny, x2=s1=sunny, x3=s1=sunny}, {x1=s1=sunny, x2=s1=sunny, x3=s2=cloudy},
{x1=s1=sunny, x2=s1=sunny, x3=s3=rainy}, {x1=s1=sunny, x2=s2=cloudy, x3=s1=sunny},
{x1=s1=sunny, x2=s2=cloudy, x3=s2=cloudy}, {x1=s1=sunny, x2=s2=cloudy, x3=s3=rainy},
{x1=s1=sunny, x2=s3=rainy, x3=s1=sunny}, {x1=s1=sunny, x2=s3=rainy, x3=s2=cloudy},
{x1=s1=sunny, x2=s3=rainy, x3=s3=rainy}, {x1=s2=cloudy, x2=s1=sunny, x3=s1=sunny},
{x1=s2=cloudy, x2=s1=sunny, x3=s2=cloudy}, {x1=s2=cloudy, x2=s1=sunny, x3=s3=rainy},
{x1=s2=cloudy, x2=s2=cloudy, x3=s1=sunny}, {x1=s2=cloudy, x2=s2=cloudy, x3=s2=cloudy},
{x1=s2=cloudy, x2=s2=cloudy, x3=s3=rainy}, {x1=s2=cloudy, x2=s3=rainy, x3=s1=sunny},
{x1=s2=cloudy, x2=s3=rainy, x3=s2=cloudy}, {x1=s2=cloudy, x2=s3=rainy, x3=s3=rainy},
{x1=s3=rainy, x2=s1=sunny, x3=s1=sunny}, {x1=s3=rainy, x2=s1=sunny, x3=s2=cloudy},
{x1=s3=rainy, x2=s1=sunny, x3=s3=rainy}, {x1=s3=rainy, x2=s2=cloudy, x3=s1=sunny},
{x1=s3=rainy, x2=s2=cloudy, x3=s2=cloudy}, {x1=s3=rainy, x2=s2=cloudy, x3=s3=rainy},
{x1=s3=rainy, x2=s3=rainy, x3=s1=sunny}, {x1=s3=rainy, x2=s3=rainy, x3=s2=cloudy},
{x1=s3=rainy, x2=s3=rainy, x3=s3=rainy}.

Easy but slow solution: Exhaustive enumeration!

3 Probabilistic Approaches

Hidden Markov Model (HMM): Evaluation

Do we need to calculate this much all the time?

This is a complete set of 3^3=27 cases of weather states for three days:

*{x1=s1=sunny, x2=s1=sunny, x3=s1=sunny}, {x1=s1=sunny, x2=s1=sunny, x3=s2=cloudy},
{x1=s1=sunny, x2=s1=sunny, x3=s3=rainy}, {x1=s1=sunny, x2=s2=cloudy, x3=s1=sunny},
{x1=s1=sunny, x2=s2=cloudy, x3=s2=cloudy}, {x1=s1=sunny, x2=s2=cloudy, x3=s3=rainy},
{x1=s1=sunny, x2=s3=rainy, x3=s1=sunny}, {x1=s1=sunny, x2=s3=rainy, x3=s2=cloudy},
{x1=s1=sunny, x2=s3=rainy, x3=s3=rainy}, {x1=s2=cloudy, x2=s1=sunny, x3=s1=sunny},
{x1=s2=cloudy, x2=s1=sunny, x3=s2=cloudy}, {x1=s2=cloudy, x2=s1=sunny, x3=s3=rainy},
{x1=s2=cloudy, x2=s2=cloudy, x3=s1=sunny}, {x1=s2=cloudy, x2=s2=cloudy, x3=s2=cloudy},
{x1=s2=cloudy, x2=s2=cloudy, x3=s3=rainy}, {x1=s2=cloudy, x2=s3=rainy, x3=s1=sunny},
{x1=s2=cloudy, x2=s3=rainy, x3=s2=cloudy}, {x1=s2=cloudy, x2=s3=rainy, x3=s3=rainy},
{x1=s3=rainy, x2=s1=sunny, x3=s1=sunny}, {x1=s3=rainy, x2=s1=sunny, x3=s2=cloudy},
{x1=s3=rainy, x2=s1=sunny, x3=s3=rainy}, {x1=s3=rainy, x2=s2=cloudy, x3=s1=sunny},
{x1=s3=rainy, x2=s2=cloudy, x3=s2=cloudy}, {x1=s3=rainy, x2=s2=cloudy, x3=s3=rainy},
{x1=s3=rainy, x2=s3=rainy, x3=s1=sunny}, {x1=s3=rainy, x2=s3=rainy, x3=s2=cloudy},
{x1=s3=rainy, x2=s3=rainy, x3=s3=rainy}.*

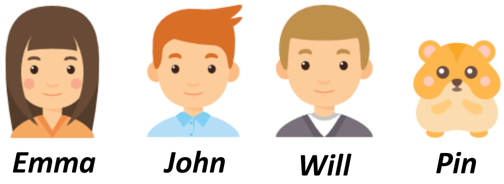
Any Efficient Solution?

Yes. Dynamic Programming technique

Now, Let's Apply this HMM
to the Part of Speech Tagging Task!

2 Baseline Approaches

Part of Speech Tagging: with HMM



Database

Emma, John can meet Will
N N M V N

Pin will meet Emma
N M V N

Will John pin Emma
M N V N

Emma will pat Pin
N M V N

2 Baseline Approaches

Part of Speech Tagging: with HMM

Emma

John

Will

Pin

	N	V	M
Emma	4	0	0
John	2	0	0
Will	1	0	3
Pin	2	1	0
Can	0	0	1
Meet	0	2	0
Pat	0	1	0

Database

Emma, John can meet Will

N N M V N

Pin will meet Emma

N M V N

Will John pin Emma

M N V N

Emma will pat Pin

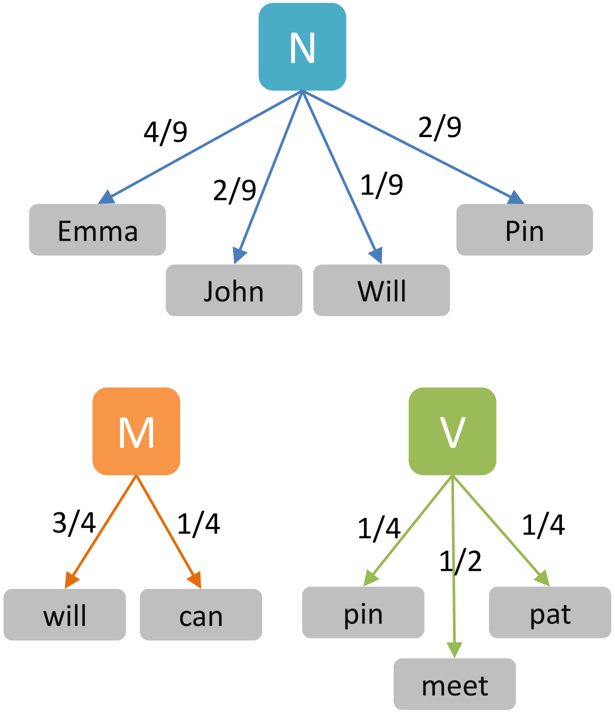
N M V N

2 Baseline Approaches

Part of Speech Tagging: with HMM

Emission Probabilities

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



2 Baseline Approaches

Part of Speech Tagging: with HMM

Transition Probabilities

	N	V	M	<E>
<S>	3	0	1	0
N	1	1	3	4
V	4	0	0	0
M	1	3	0	0

Database

Emma, John can meet Will
N N M V N

Pin will meet Emma
N M V N

Will John pin Emma
M N V N

Emma will pat Pin
N M V N

2 Baseline Approaches

Part of Speech Tagging: with HMM

Transition Probabilities

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

Database

Emma, John can meet Will
N N M V N

Pin will meet Emma
N M V N

Will John pin Emma
M N V N

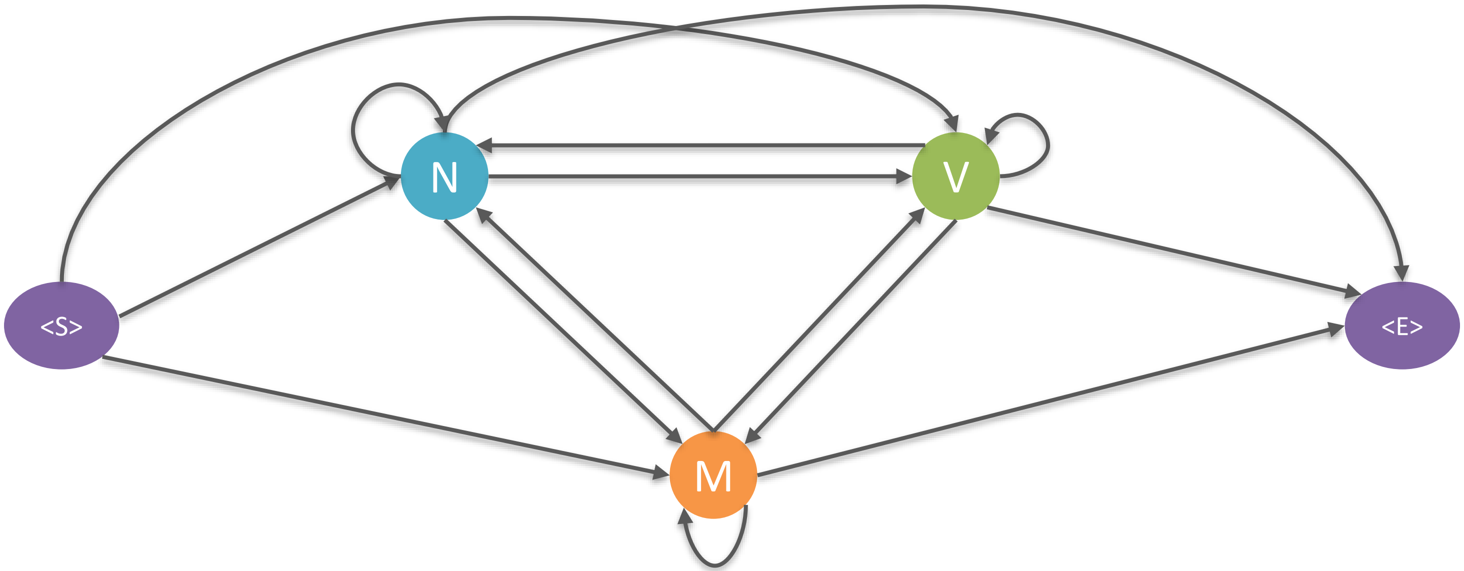
Emma will pat Pin
N M V N

2 Baseline Approaches

Part of Speech Tagging: with HMM

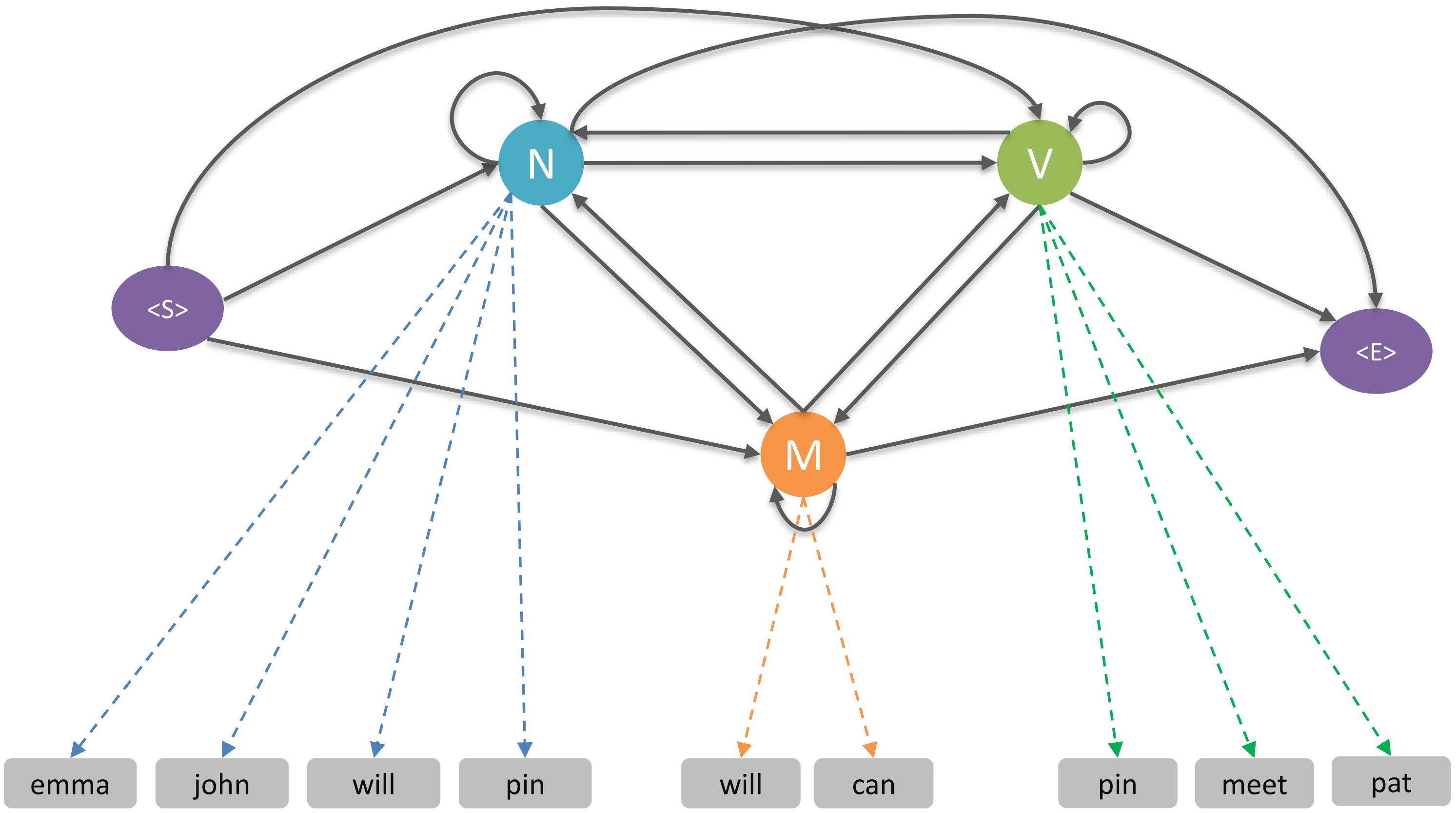
Transition Probabilities

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



Part of Speech Tagging: with HMM

Let's combine this!



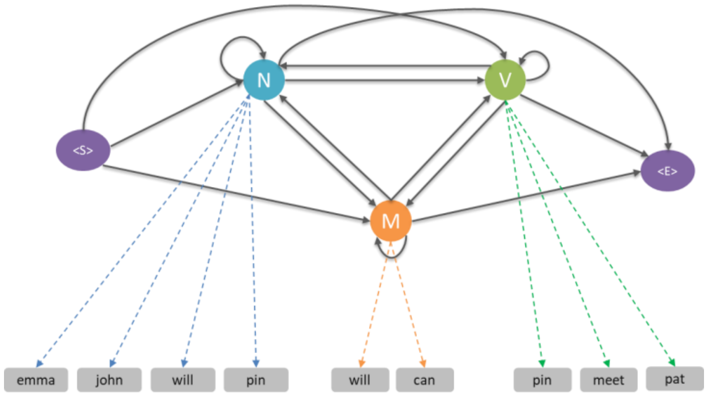
2 Baseline Approaches

Part of Speech Tagging: with HMM

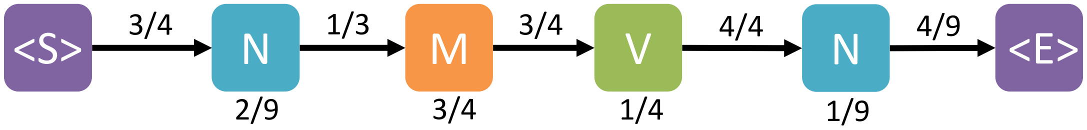
Let's combine this!

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



John will Pin Will



<S> $\xrightarrow{3/4}$ N $\xrightarrow{1/3}$ M $\xrightarrow{3/4}$ V $\xrightarrow{4/4}$ N $\xrightarrow{4/9}$ <E>

2/9 3/4 1/4 1/9

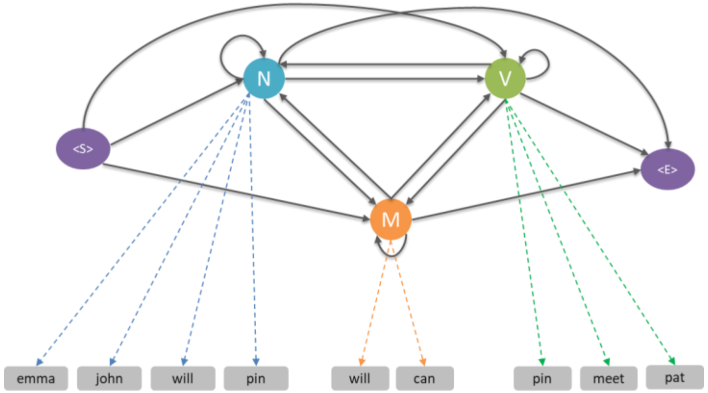
0.0003858

Part of Speech Tagging: with HMM

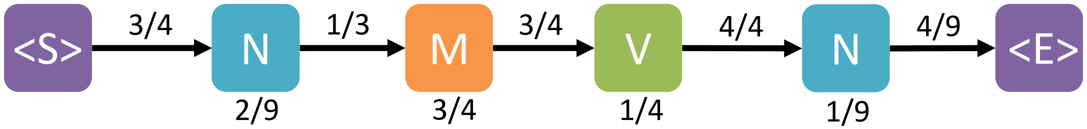
Let's combine this!

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



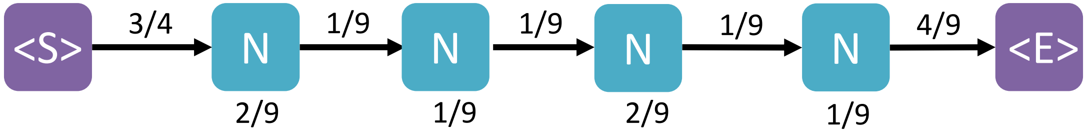
John will Pin Will


<S> 3/4 N 1/3 M 3/4 V 4/4 N 4/9 <E>

2/9 3/4 1/4 1/9

0.0003858

John will Pin Will


<S> 3/4 N 1/9 N 1/9 N 1/9 N 4/9 <E>

2/9 1/9 2/9 1/9

0.000000278

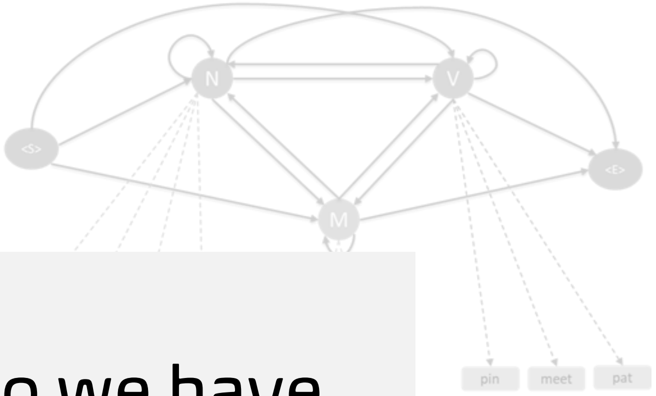
2 Baseline Approaches

Part of Speech Tagging: with HMM

Let's combine this!

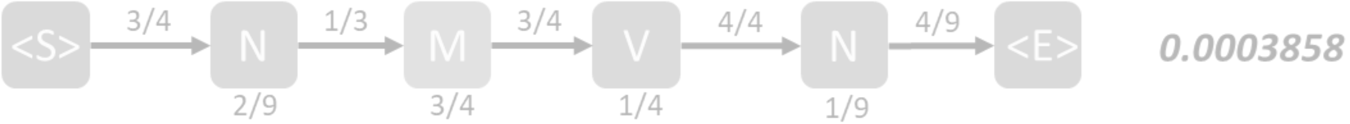
	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can			
Meet			
Pat			

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0

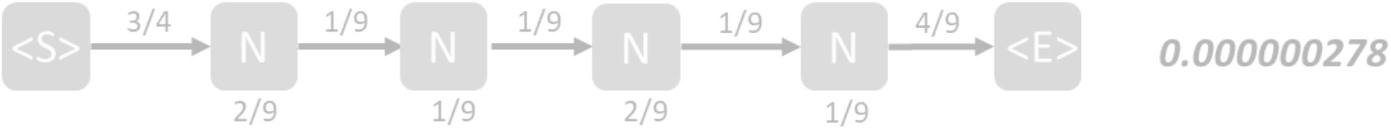


Question!

How many possibilities do we have for the sentence 'John will Pin Will'?



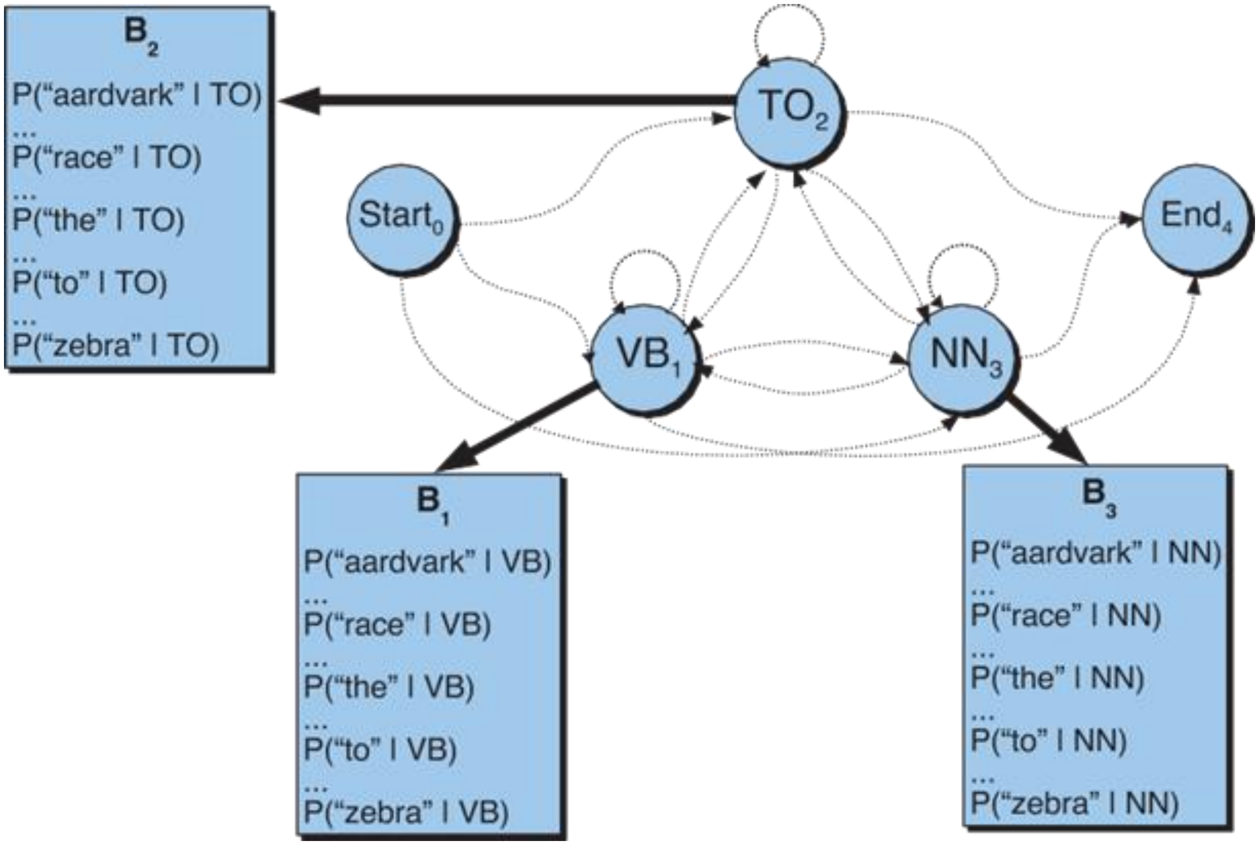
John will Pin Will



Probabilistic Approaches

Hidden Markov Model (HMM) with POS Tagging

Emissions



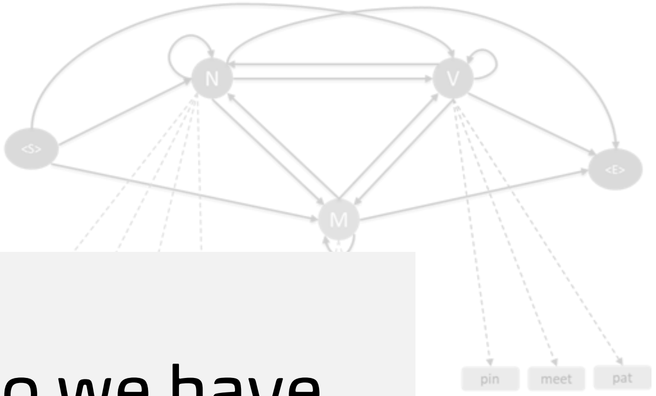
2 Baseline Approaches

Part of Speech Tagging: with HMM

Let's combine this!

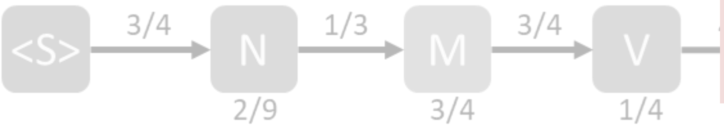
	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can			
Meet			
Pat			

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0

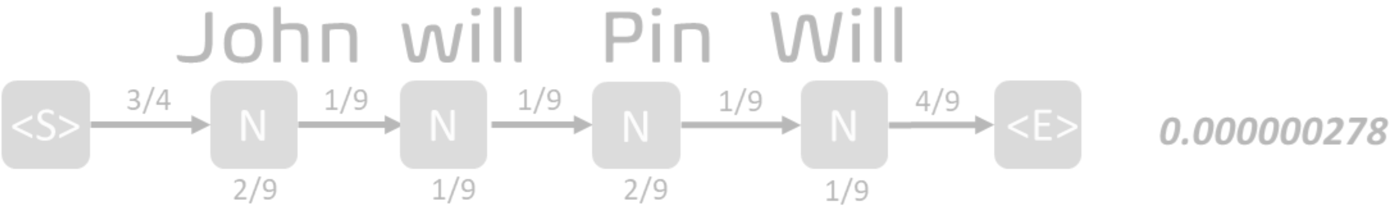


Question!

How many possibilities do we have for the sentence 'John will Pin Will'?

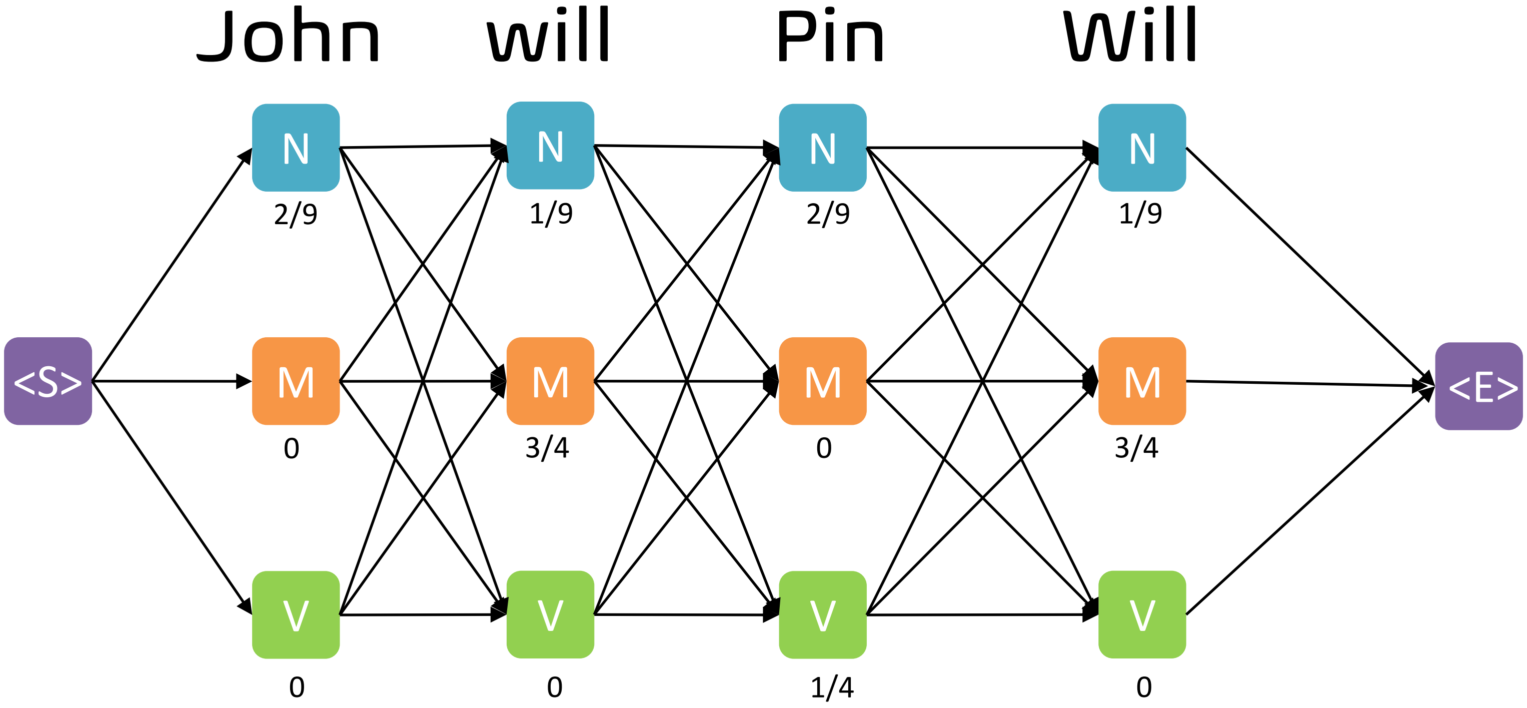


$3^3 * 3 = 81$



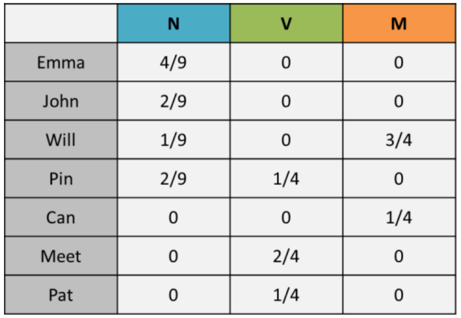
POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

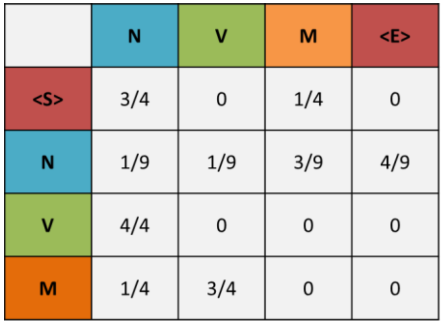


3 Probabilistic Approaches

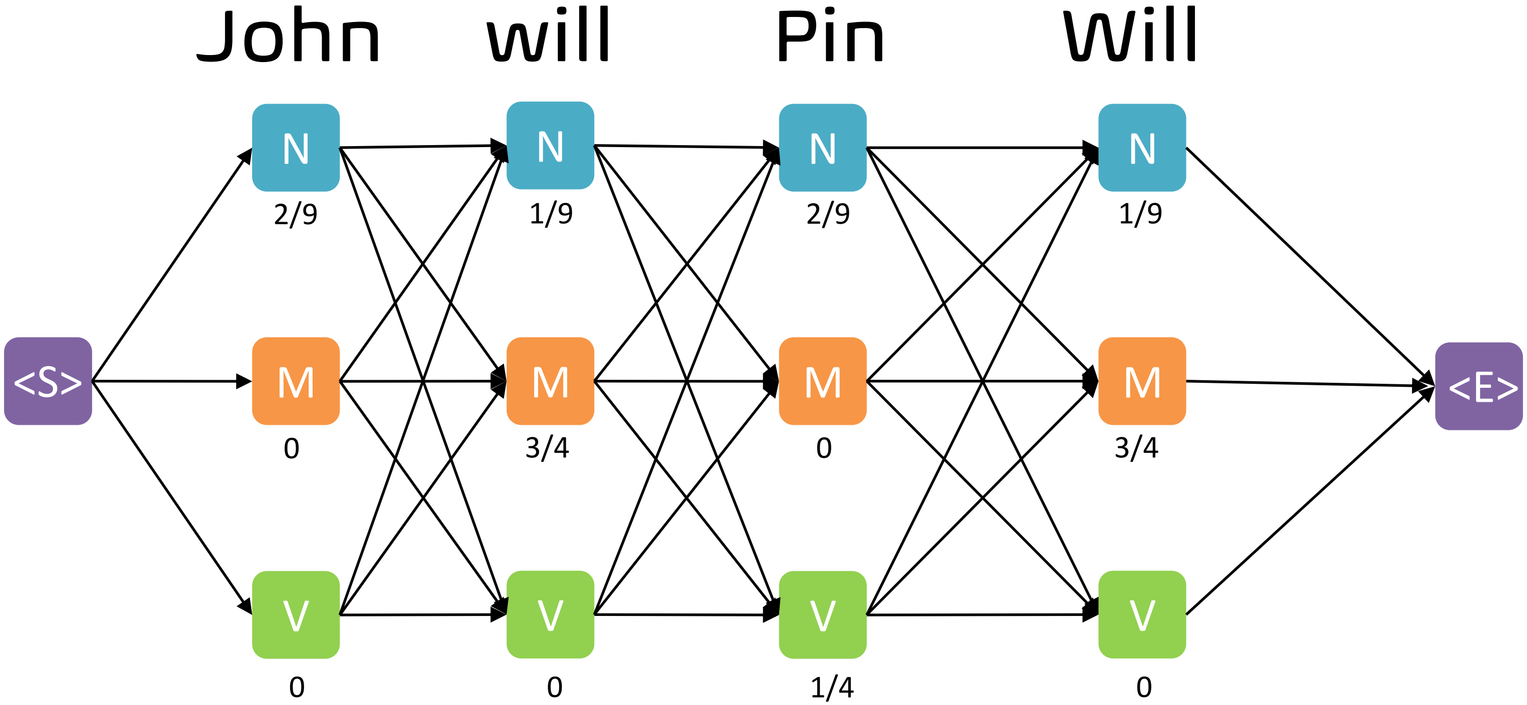
POS Tagging: with HMM



	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



Beam Search

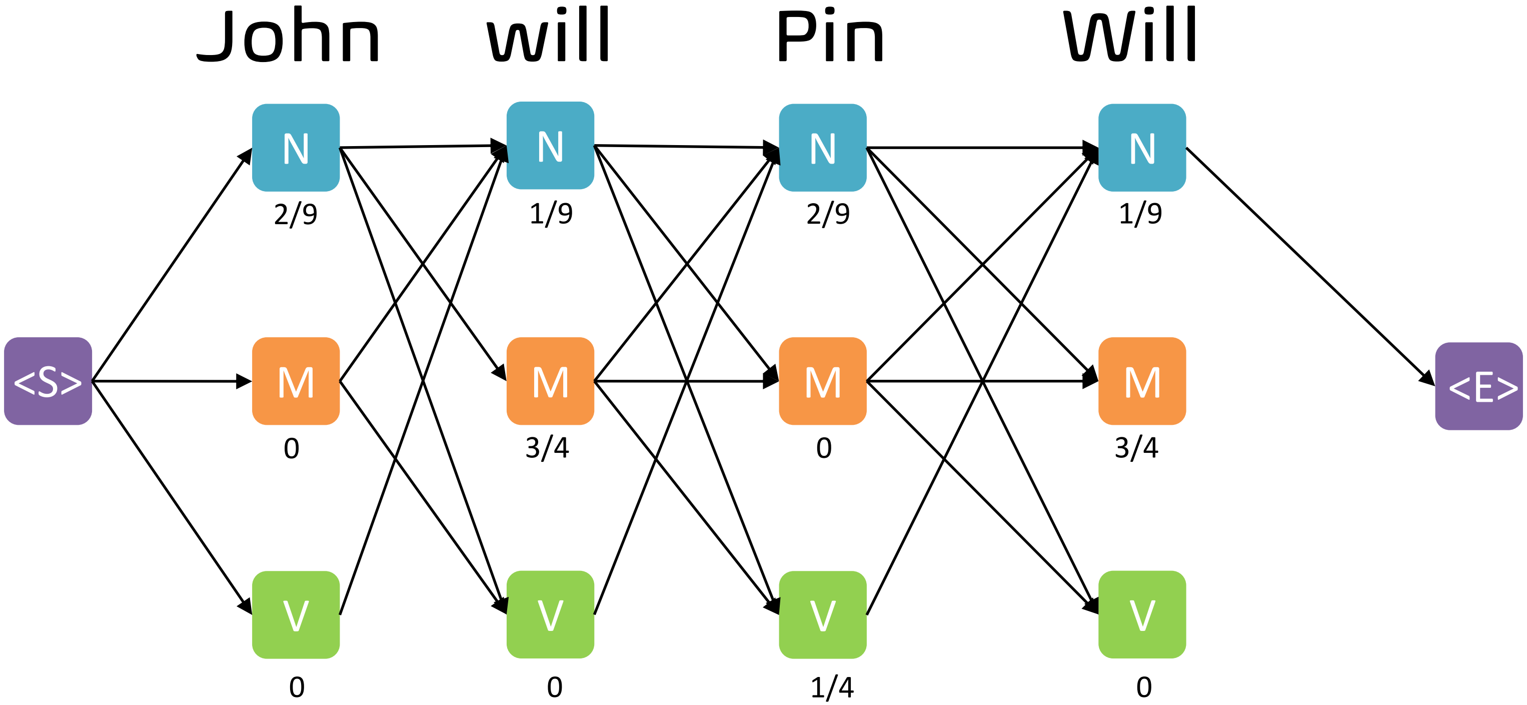
Get rid of unlikely candidates

3 Probabilistic Approaches

POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



Beam Search

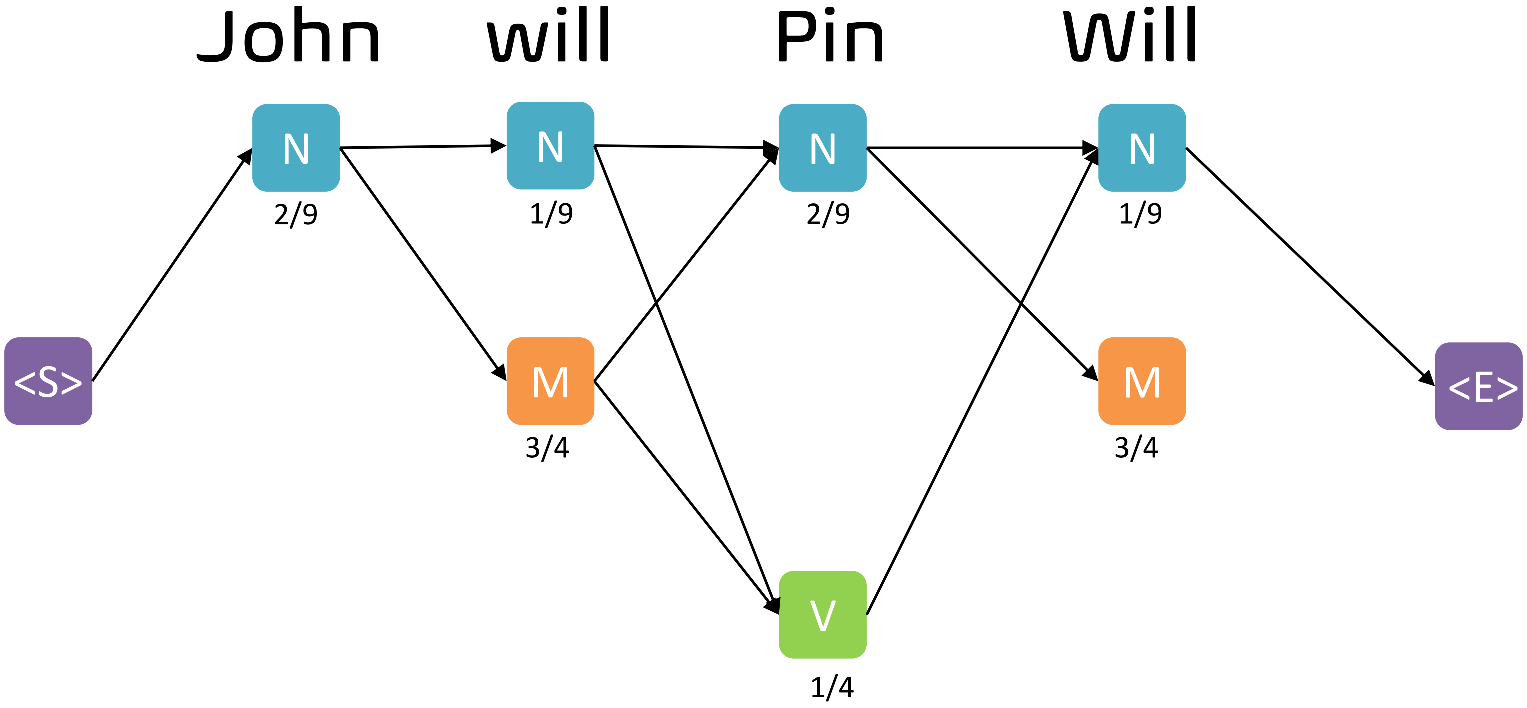
Get rid of unlikely candidates

3 Probabilistic Approaches

POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

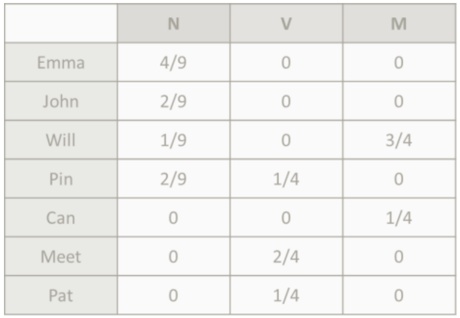


Beam Search

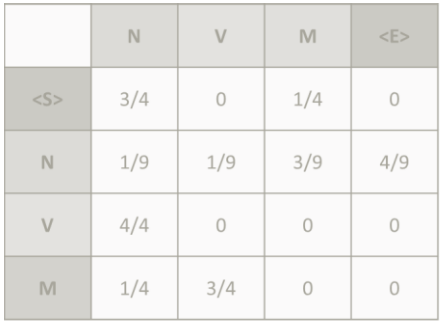
Get rid of unlikely candidates

3 Probabilistic Approaches

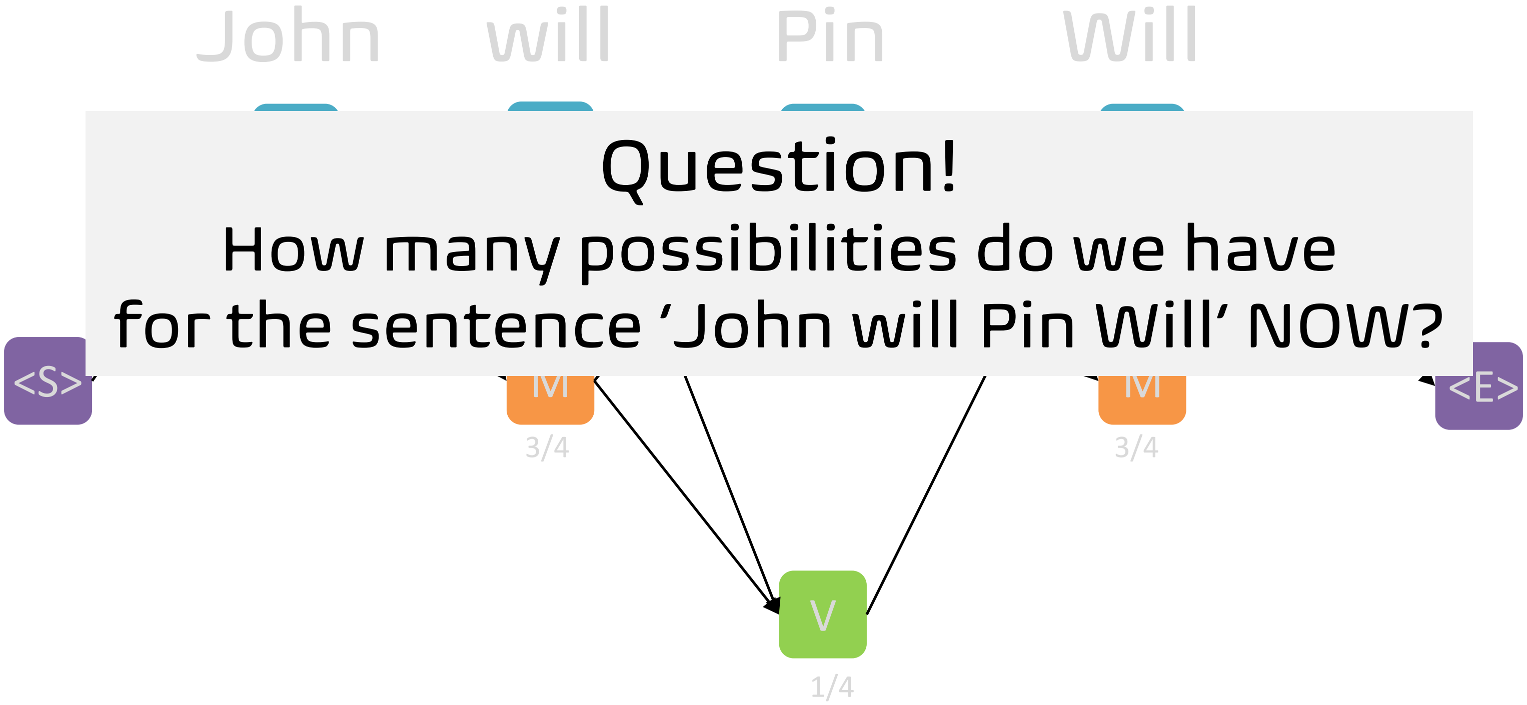
POS Tagging: with HMM



	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



Question!

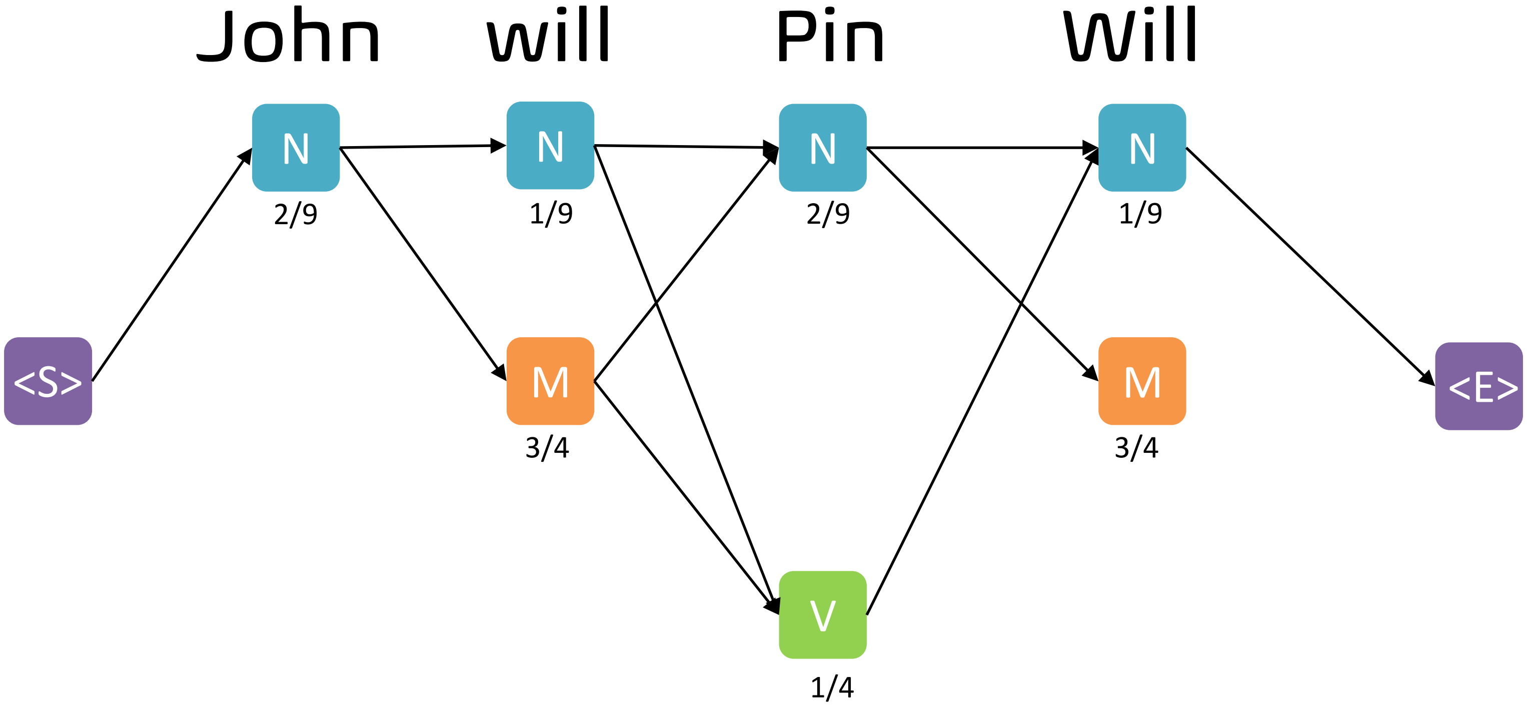
How many possibilities do we have for the sentence ‘John will Pin Will’ NOW?

3 Probabilistic Approaches

POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



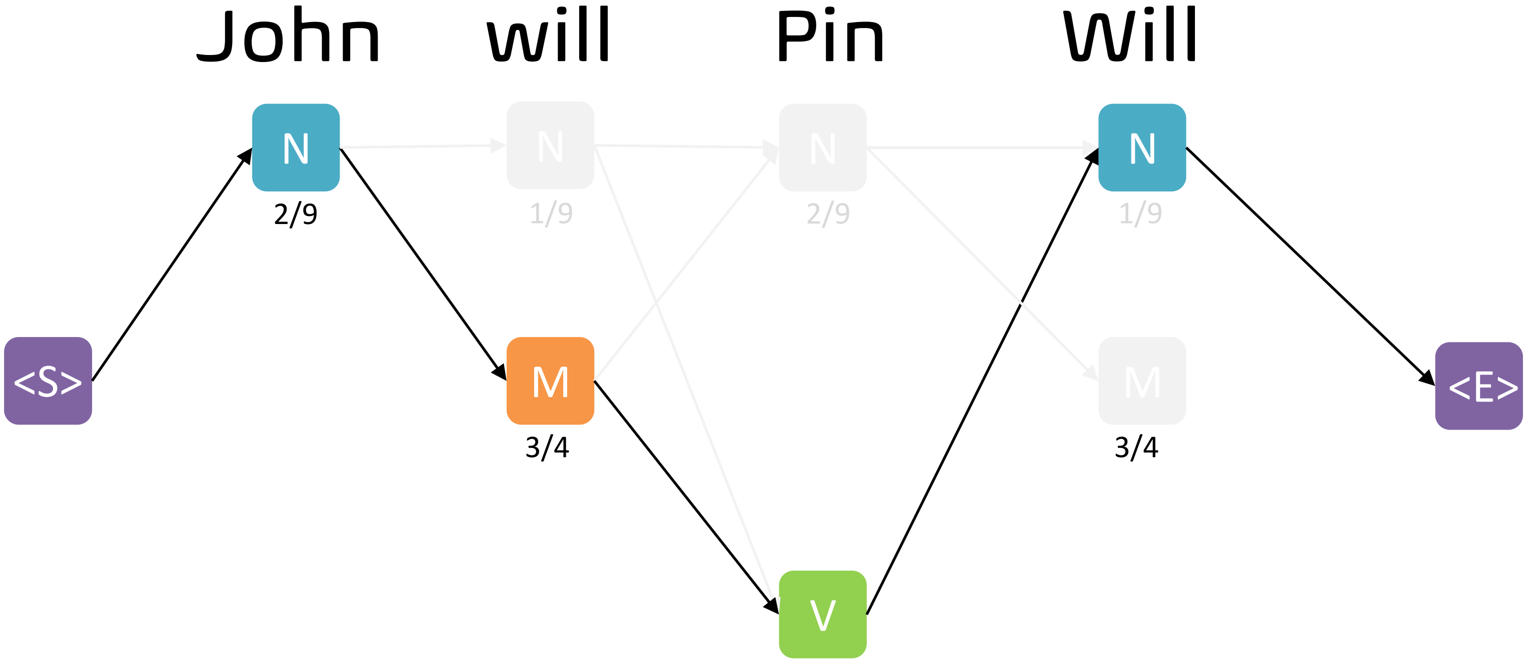
Which Path is the most likely?

3 Probabilistic Approaches

POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



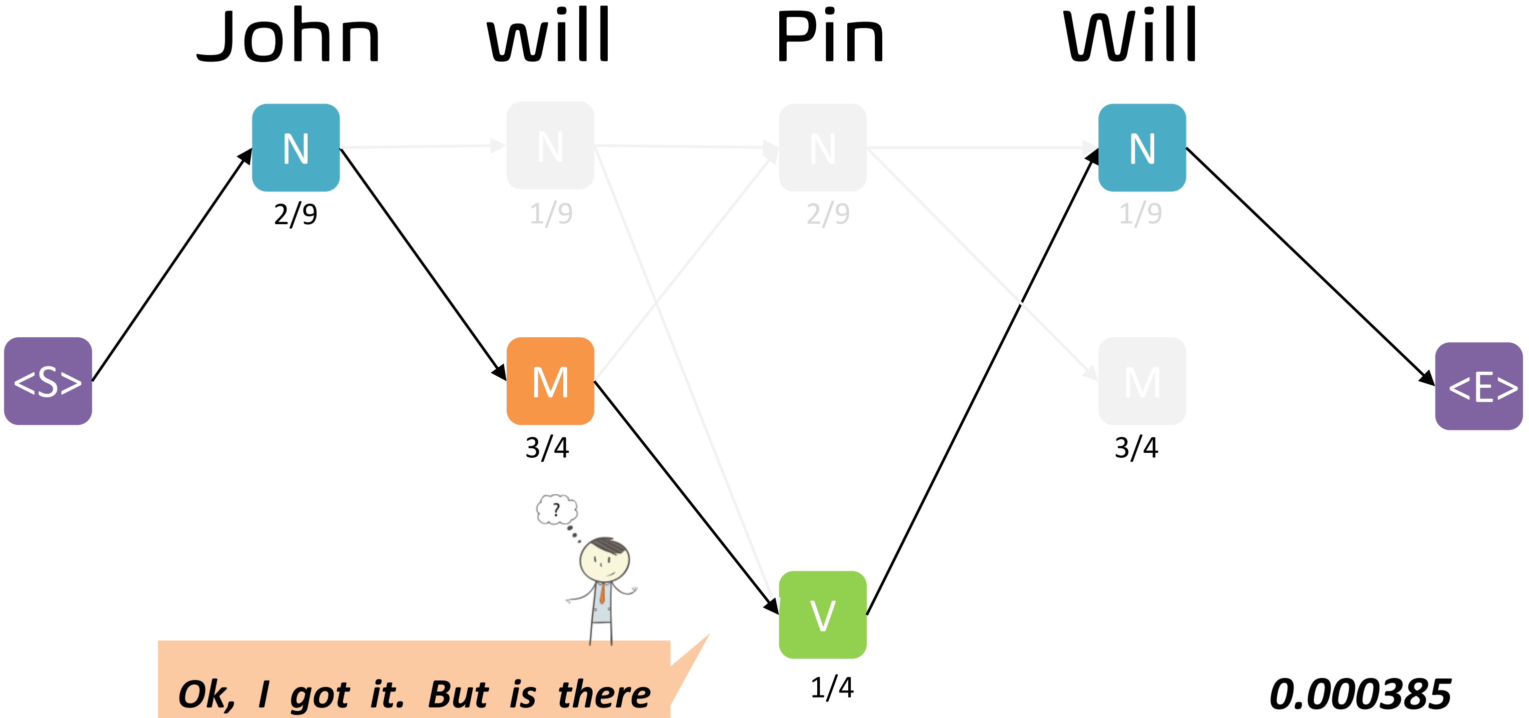
0.000385

3 Probabilistic Approaches

POS Tagging: with HMM

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



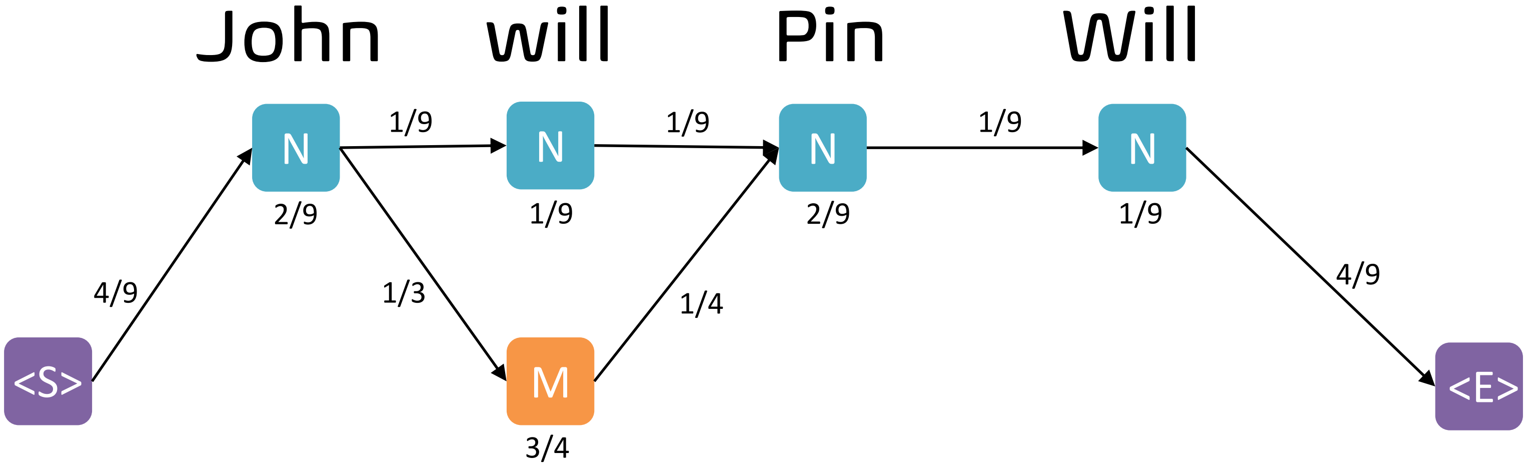
Ok, I got it. But is there any better algorithm?

0.000385

3 Probabilistic Approaches

Viterbi Algorithm!

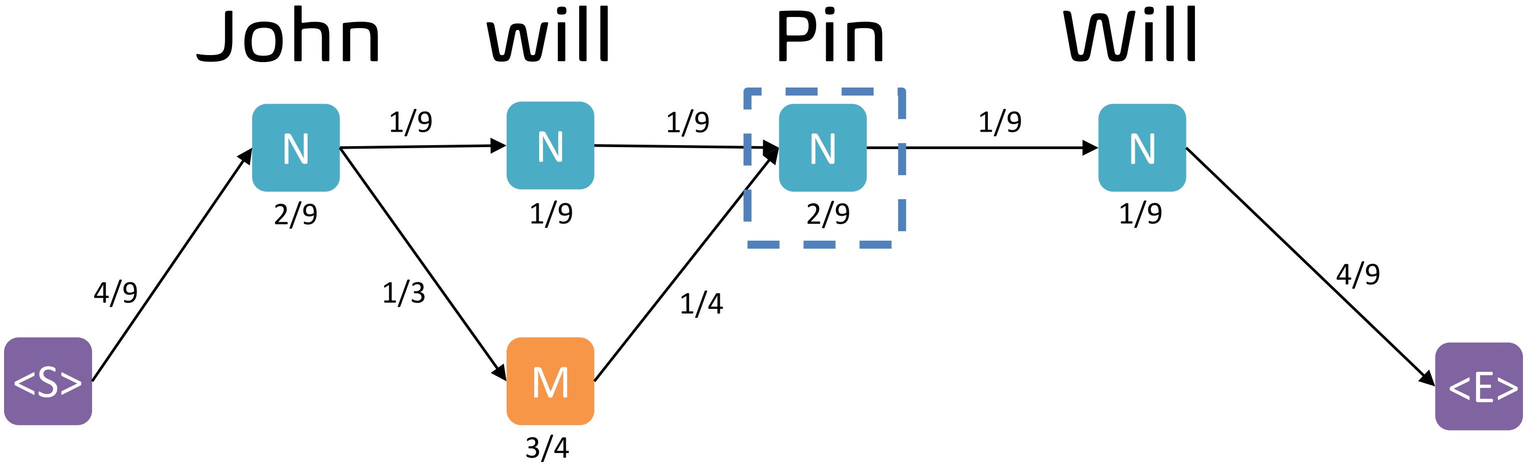
Assume we have only these options now

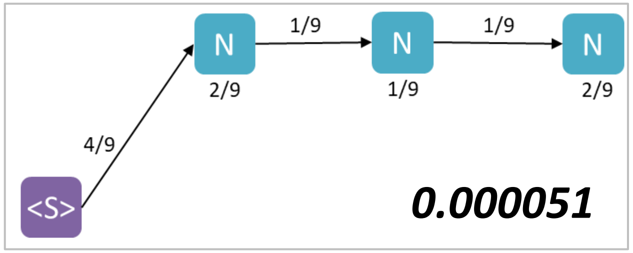


3 Probabilistic Approaches

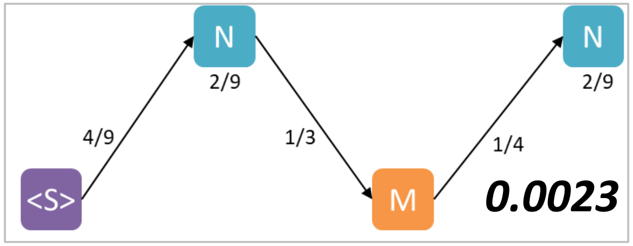
Viterbi Algorithm!

Assume we have only these options now





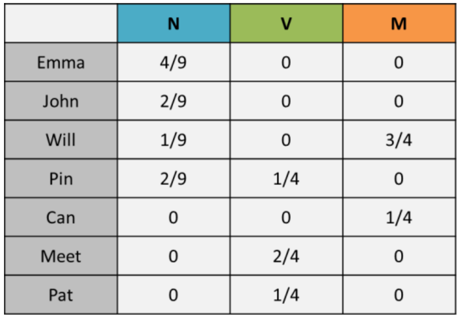
0.000051



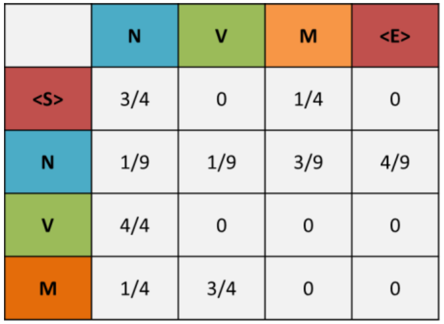
0.0023

3 Probabilistic Approaches

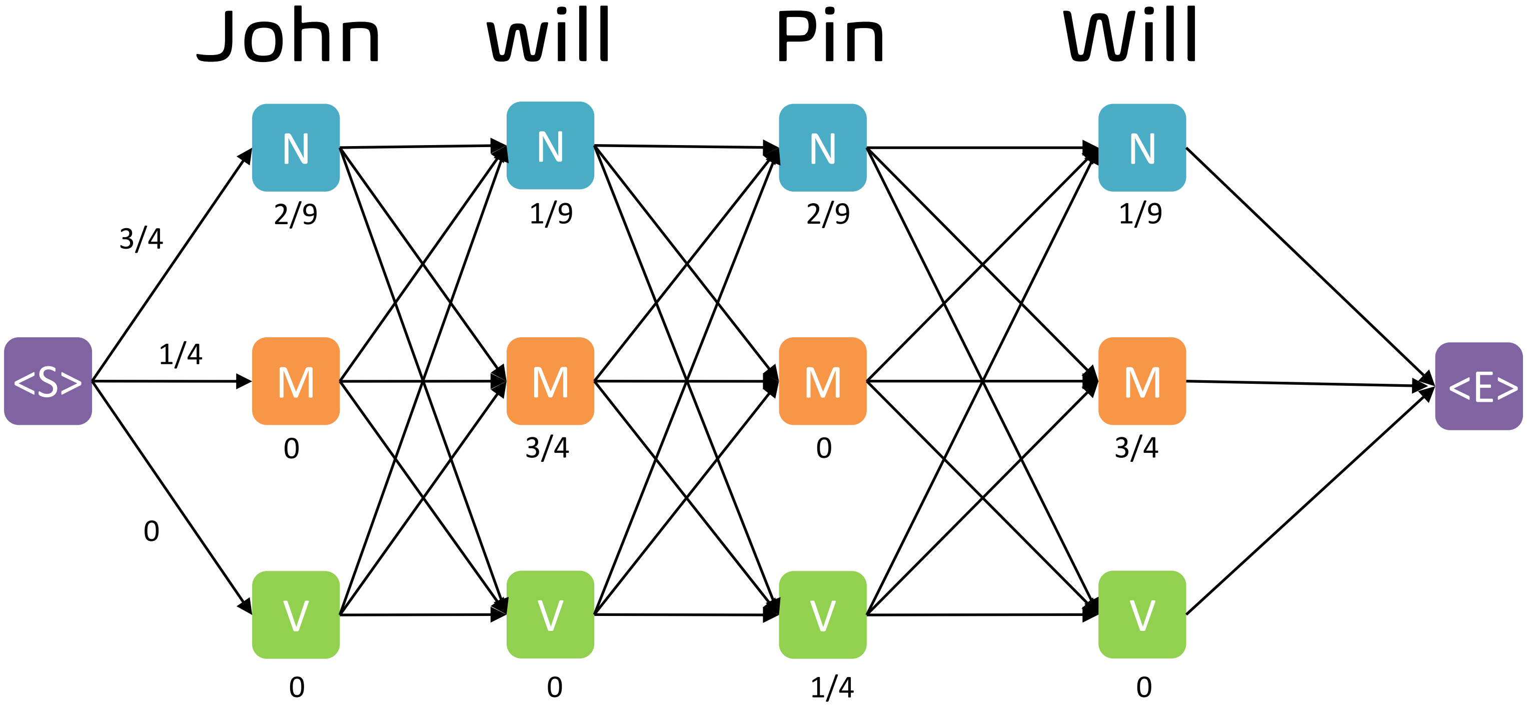
Viterbi Algorithm



	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



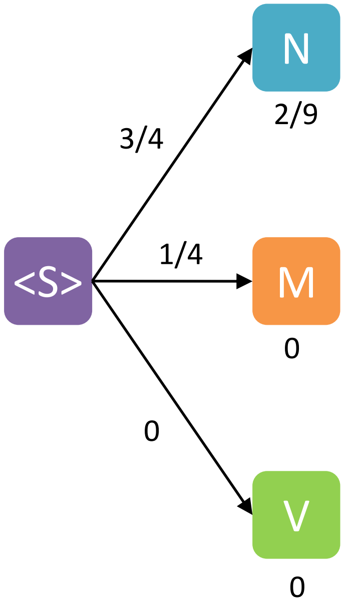
3 Probabilistic Approaches

Viterbi Algorithm

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

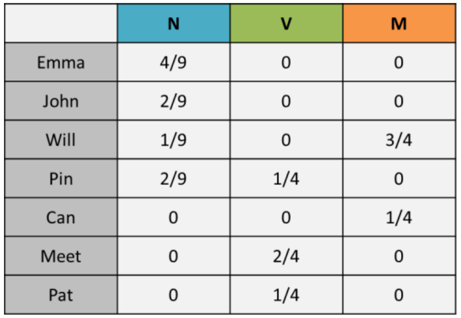
	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

John will Pin Will

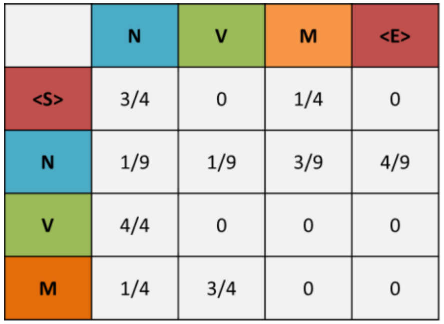


3 Probabilistic Approaches

Viterbi Algorithm

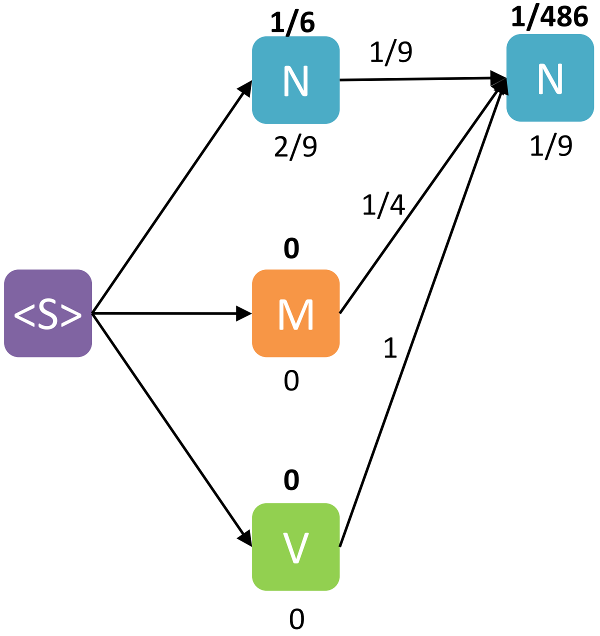


	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

John will Pin Will

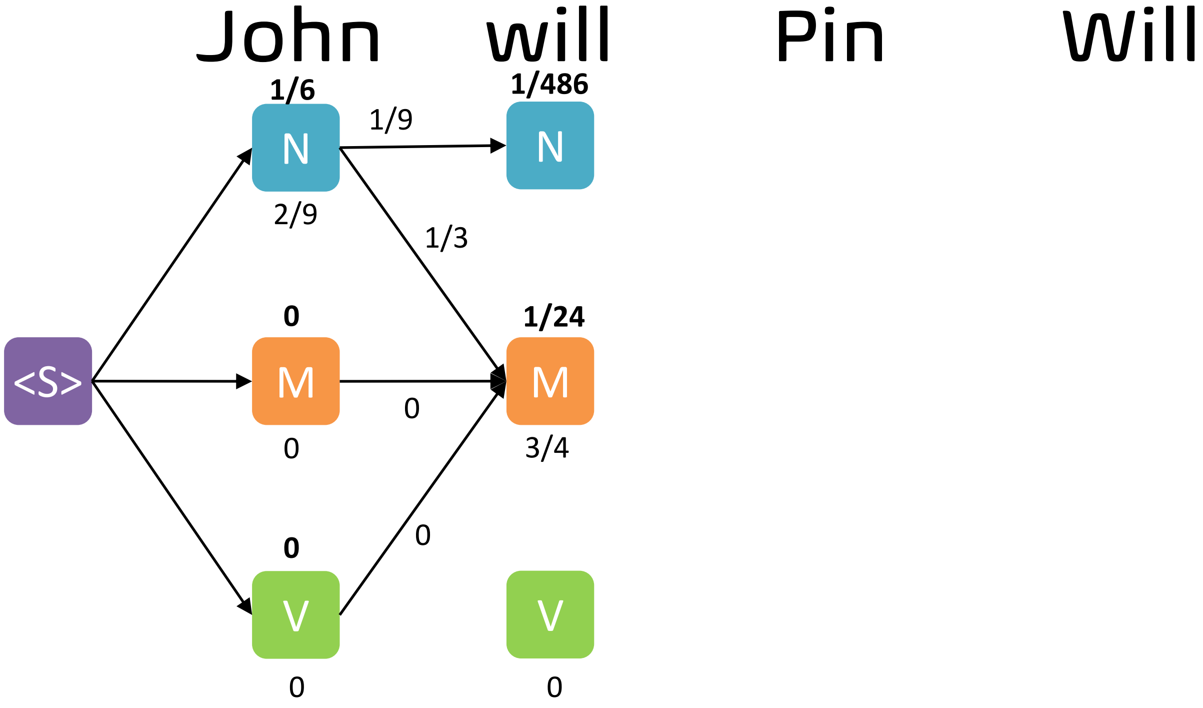


3 Probabilistic Approaches

Viterbi Algorithm

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

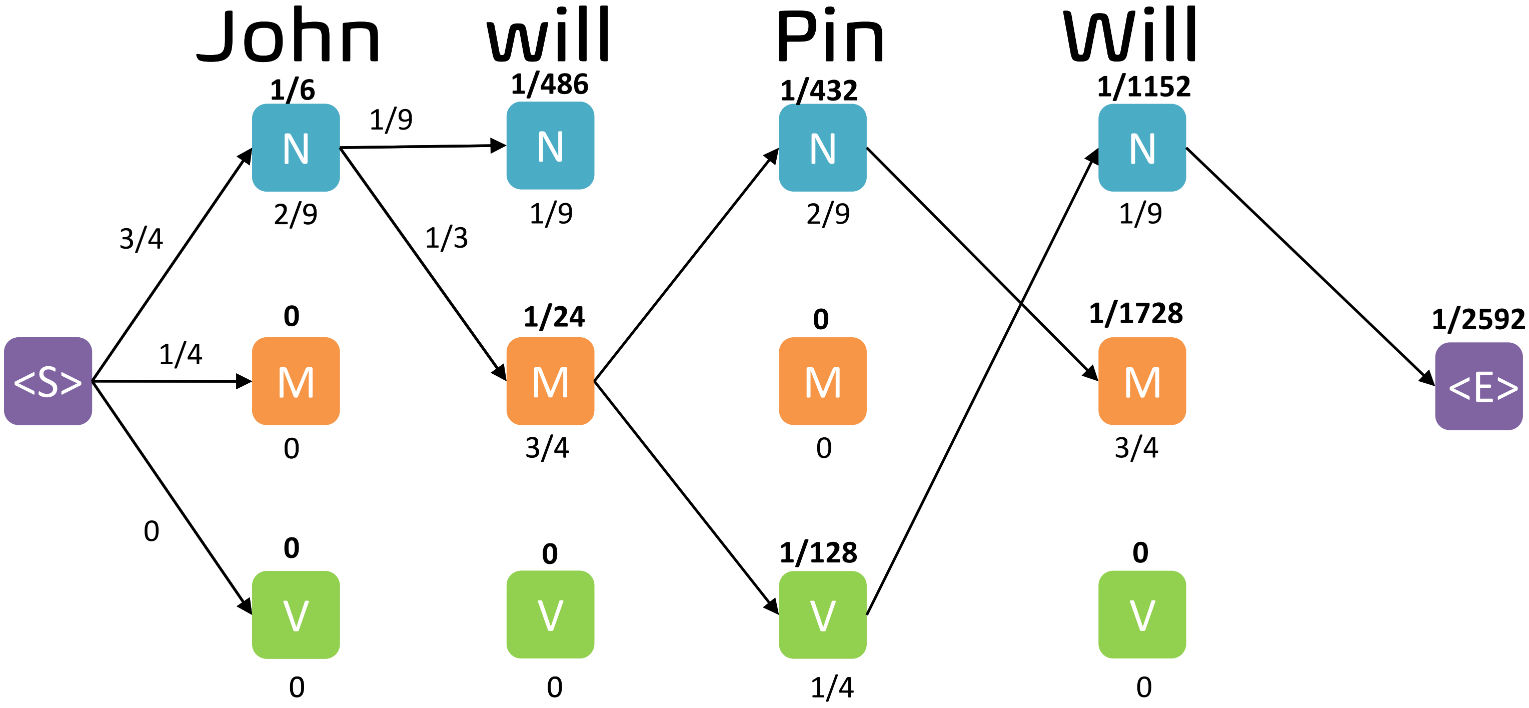


3 Probabilistic Approaches

Viterbi Algorithm

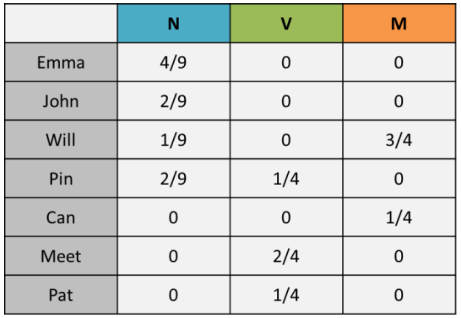
	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

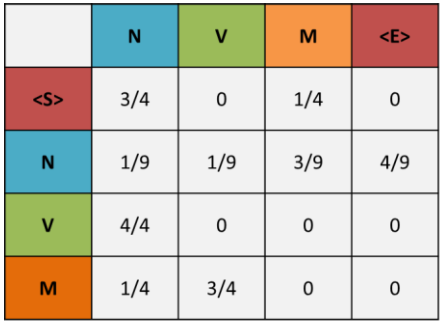


3 Probabilistic Approaches

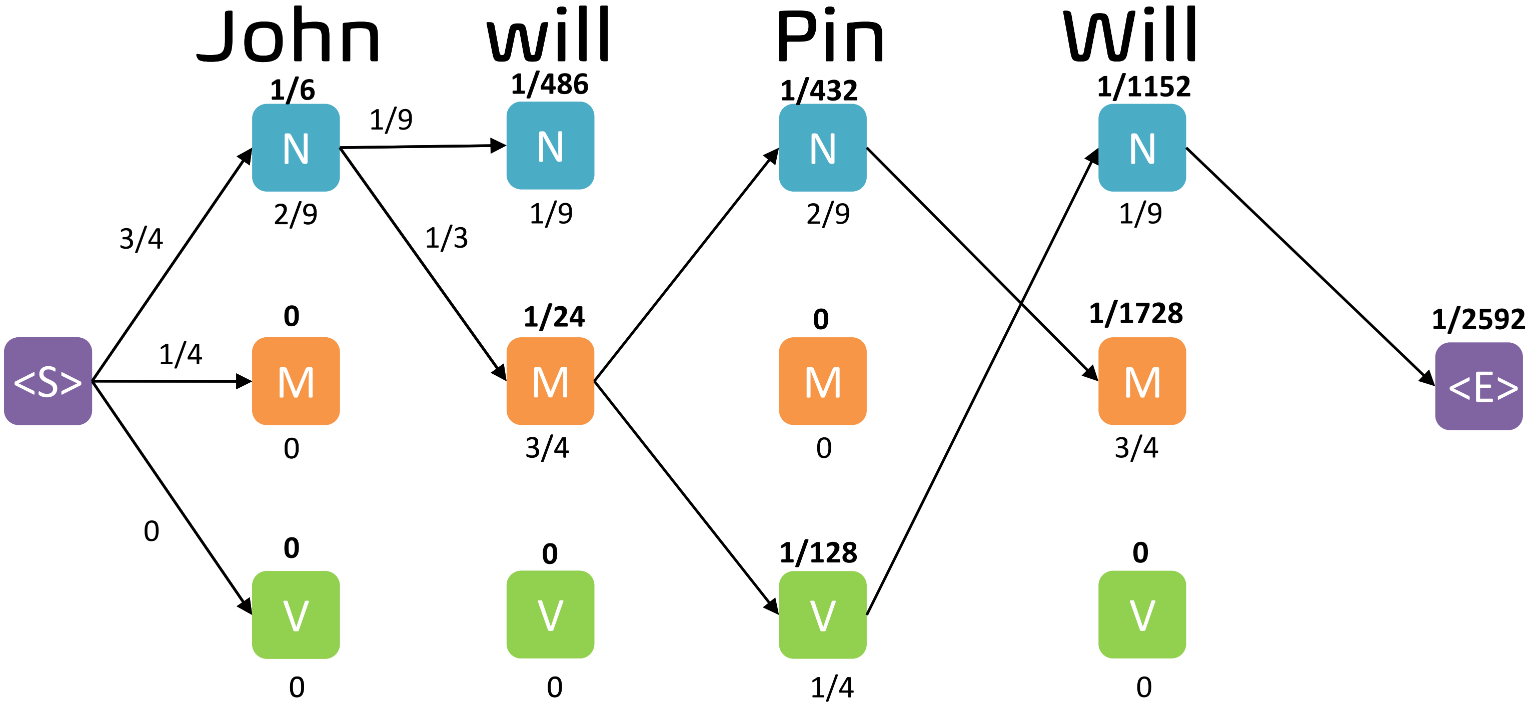
Viterbi Algorithm



	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0



	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0

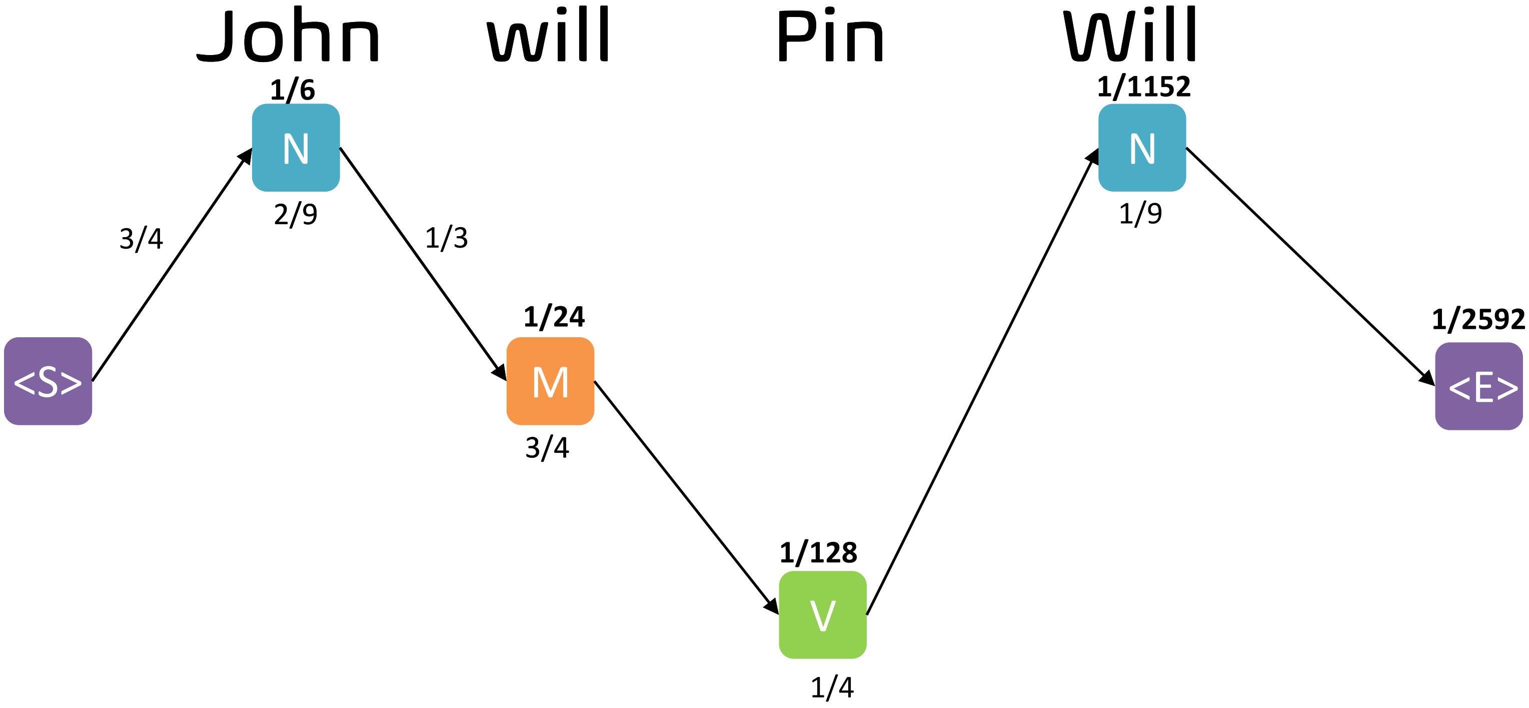


3 Probabilistic Approaches

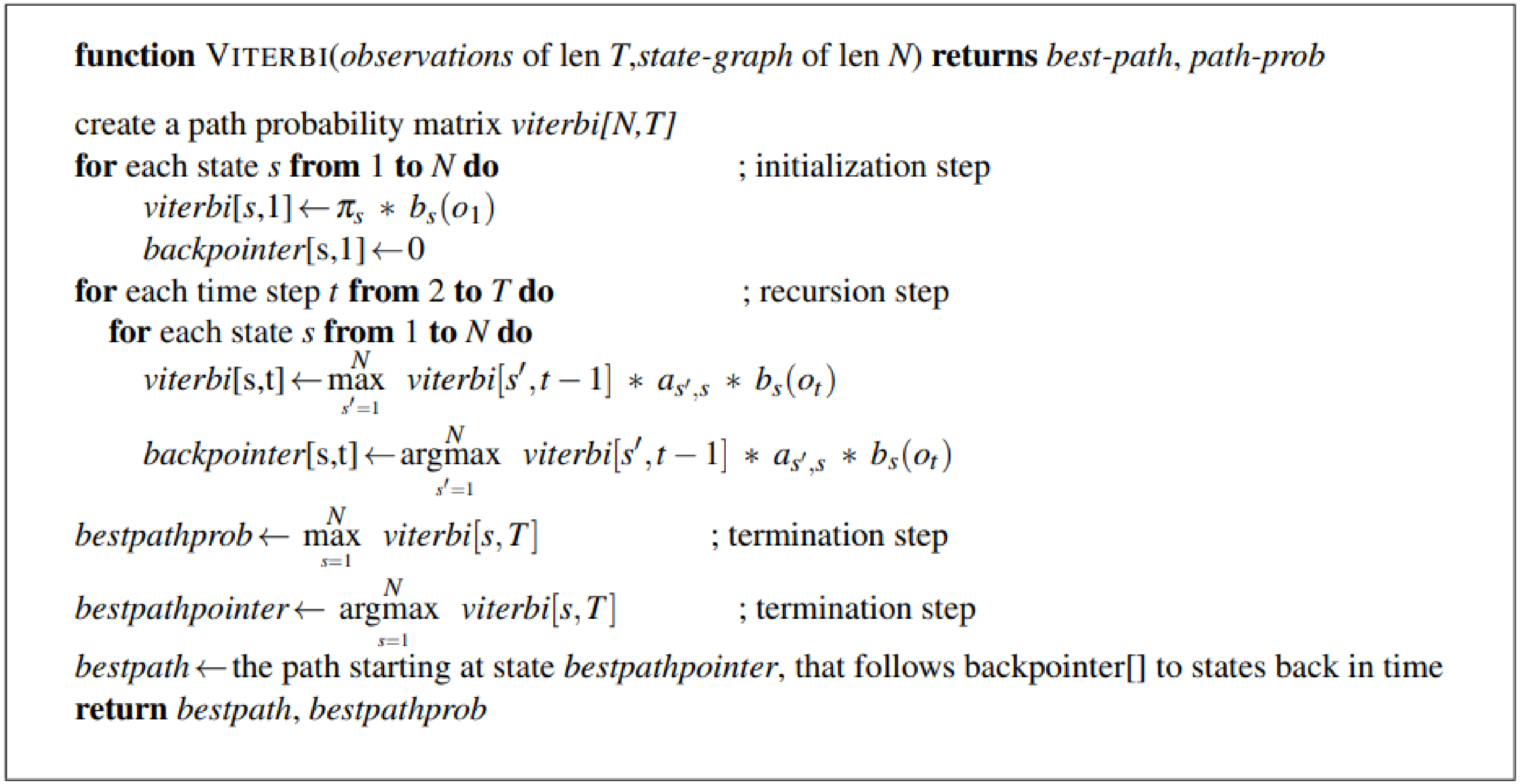
Viterbi Algorithm

	N	V	M
Emma	4/9	0	0
John	2/9	0	0
Will	1/9	0	3/4
Pin	2/9	1/4	0
Can	0	0	1/4
Meet	0	2/4	0
Pat	0	1/4	0

	N	V	M	<E>
<S>	3/4	0	1/4	0
N	1/9	1/9	3/9	4/9
V	4/4	0	0	0
M	1/4	3/4	0	0



Viterbi Algorithm



```
function VITERBI(observations of len T,state-graph of len N) returns best-path, path-prob

create a path probability matrix viterbi[N,T]
for each state s from 1 to N do                          ; initialization step
      $viterbi[s,1] \leftarrow \pi_s * b_s(o_1)$
      $backpointer[s,1] \leftarrow 0$
for each time step t from 2 to T do                      ; recursion step
   for each state s from 1 to N do
      $viterbi[s,t] \leftarrow \max_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$
      $backpointer[s,t] \leftarrow \operatorname{argmax}_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$
$bestpathprob \leftarrow \max_{s=1}^{N} viterbi[s,T]$                  ; termination step
$bestpathpointer \leftarrow \operatorname{argmax}_{s=1}^{N} viterbi[s,T]$          ; termination step
$bestpath \leftarrow$ the path starting at state bestpathpointer, that follows backpointer[] to states back in time
return bestpath, bestpathprob
```

https://web.stanford.edu/~jurafsky/slp3/8.pdf

Out-of-Vocab

HMM Tagger Issue: #1. Unknown (OOV) Words

How to handle if there are any unknown words

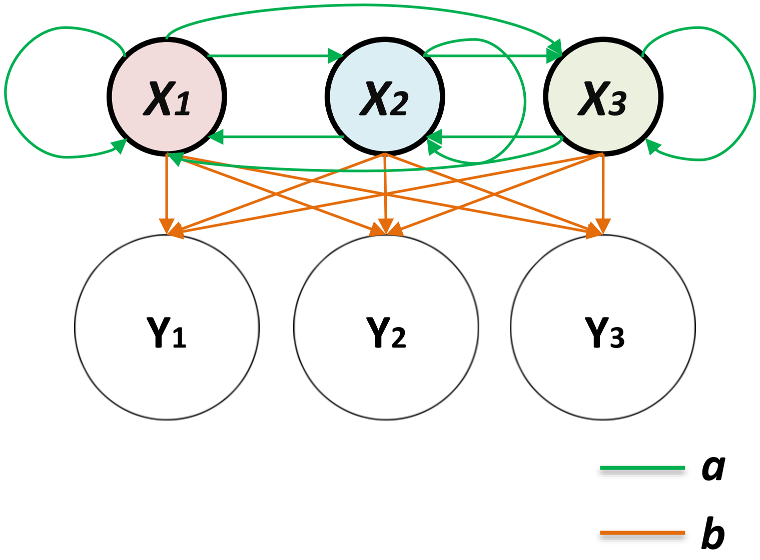
Solution 1: Use N-grams to predict the correct Tag

Solution 2: Use morphology (prefixes, suffixes) or hyphenation

3 Probabilistic Approaches

HMM Tagger Issue: #2. Independency Problem

HMM is only dependent on every state and its corresponding observed object. The sequence labeling, in addition to having a relationship with individual words, also relates to such aspects as the observed sequence length, word context and others.

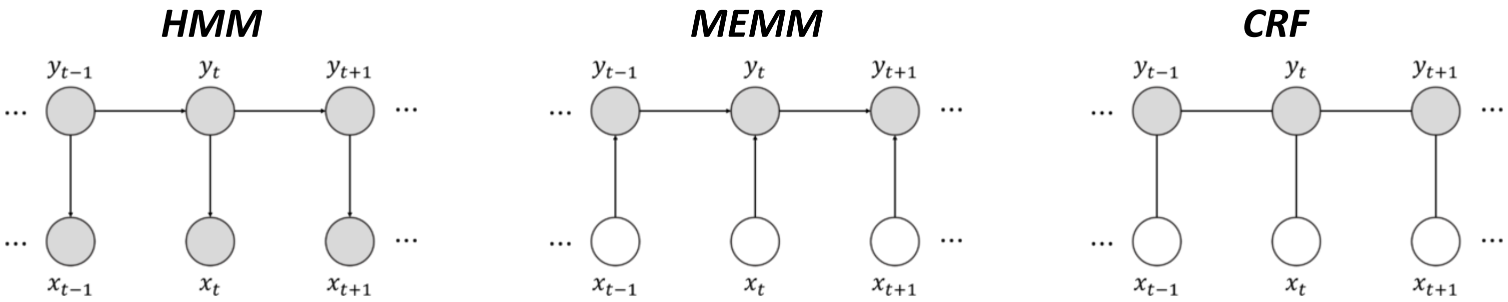


3 Probabilistic Approaches

Advanced HMM (MEMM or CRF)

The CRF model has addressed the labeling bias issue and eliminated two unreasonable hypotheses in HMM.

MEMM adopts local variance normalization while CRF adopts global variance normalization.



Conditional Random Field: Advantages

- Compared with HMM: Since CRF does not have as strict independence assumptions as HMM does, it can accommodate any context information.

- Compared with MEMM: Since CRF computes the conditional probability of global optimal output nodes, it overcomes the drawbacks of label bias in MEMM.

However,

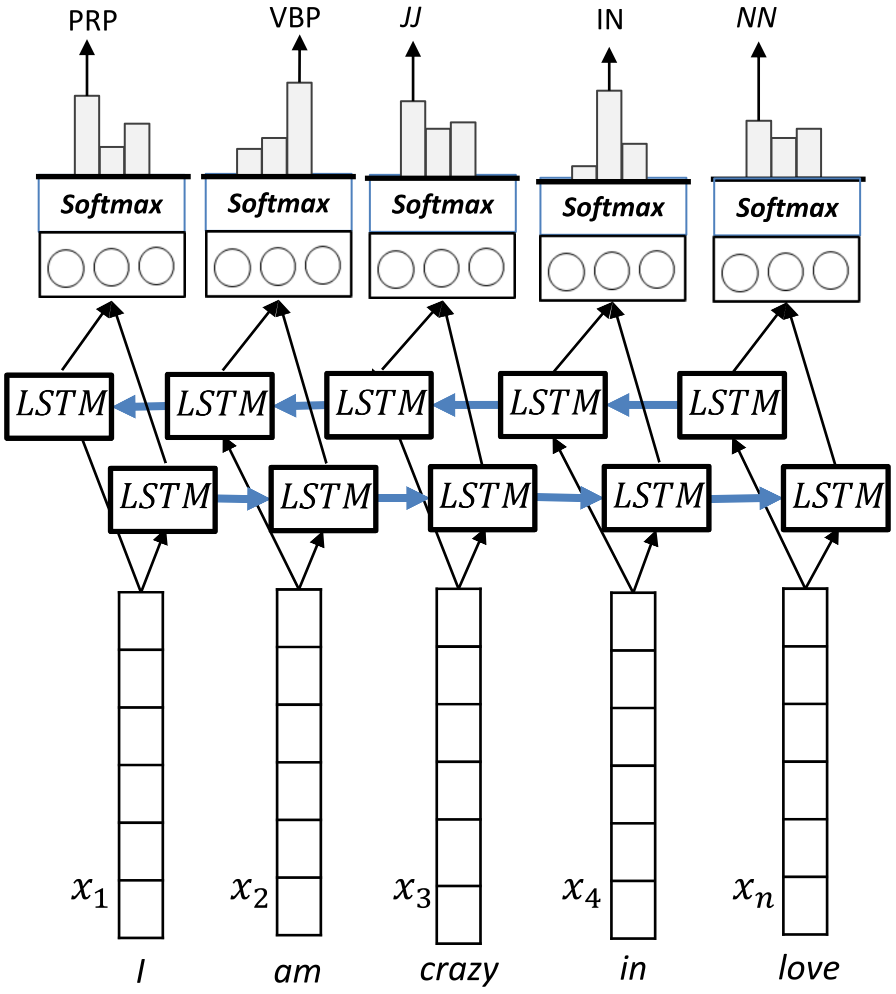
CRF is highly ***computationally complex at the training stage*** of the algorithm.
It makes it ***very difficult to re-train the model*** when newer data becomes available.

0 LECTURE PLAN

Lecture 6: Part of Speech Tagging

1. Part-of-Speech Tagging
2. Baseline Approaches
    1. Rule-based Model
    2. Look-up Table Model
    3. N-Gram Model
3. Probabilistic Approaches
    1. Hidden Markov Model
    2. Conditional Random Field
4. **Deep Learning Approaches**

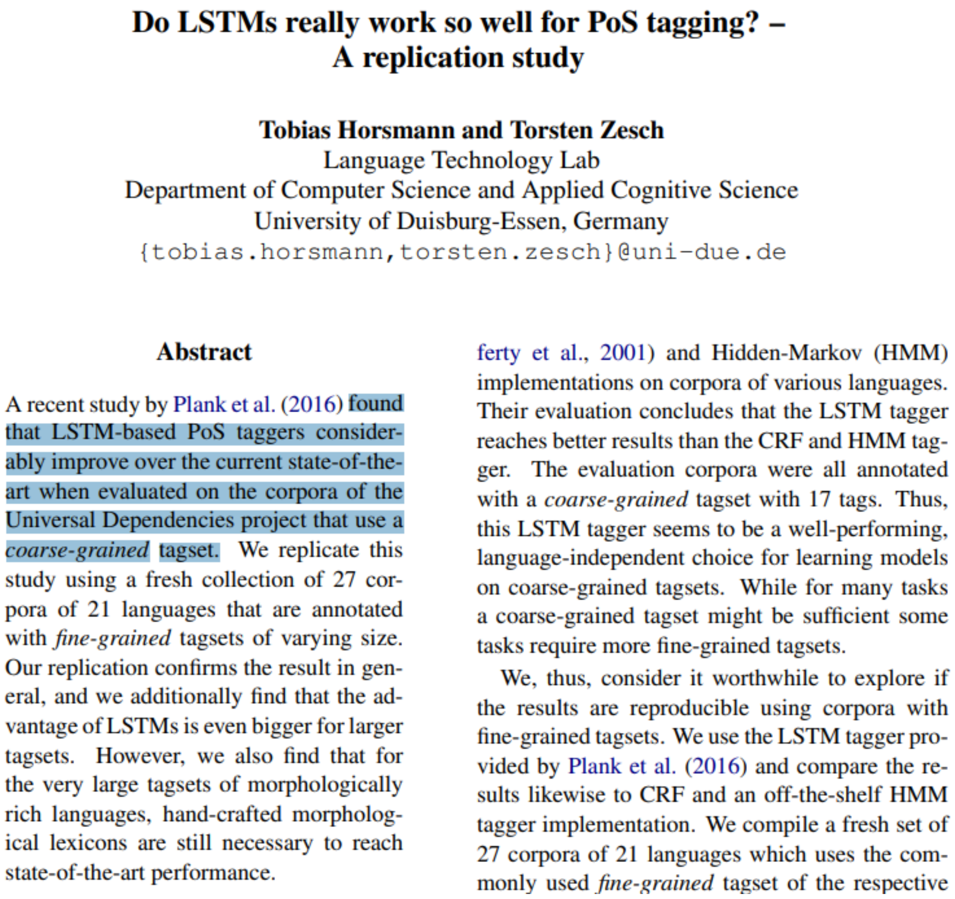
RNN/LSTM/GRU in Part of Speech Tagging



4 Deep Learning Approaches

Do LSTMs really work so well for PoS tagging?

(Horsmann and Zesch, 2017)



Do LSTMs really work so well for PoS tagging? – A replication study

Tobias Horsmann and Torsten Zesch
Language Technology Lab
Department of Computer Science and Applied Cognitive Science
University of Duisburg-Essen, Germany
{tobias.horsmann,torsten.zesch}@uni-due.de

Abstract

A recent study by Plank et al. (2016) found that LSTM-based PoS taggers considerably improve over the current state-of-the-art when evaluated on the corpora of the Universal Dependencies project that use a *coarse-grained* tagset. We replicate this study using a fresh collection of 27 corpora of 21 languages that are annotated with *fine-grained* tagsets of varying size. Our replication confirms the result in general, and we additionally find that the advantage of LSTMs is even bigger for larger tagsets. However, we also find that for the very large tagsets of morphologically rich languages, hand-crafted morphological lexicons are still necessary to reach state-of-the-art performance.

ferty et al., 2001) and Hidden-Markov (HMM) implementations on corpora of various languages. Their evaluation concludes that the LSTM tagger reaches better results than the CRF and HMM tagger. The evaluation corpora were all annotated with a *coarse-grained* tagset with 17 tags. Thus, this LSTM tagger seems to be a well-performing, language-independent choice for learning models on coarse-grained tagsets. While for many tasks a coarse-grained tagset might be sufficient some tasks require more fine-grained tagsets.

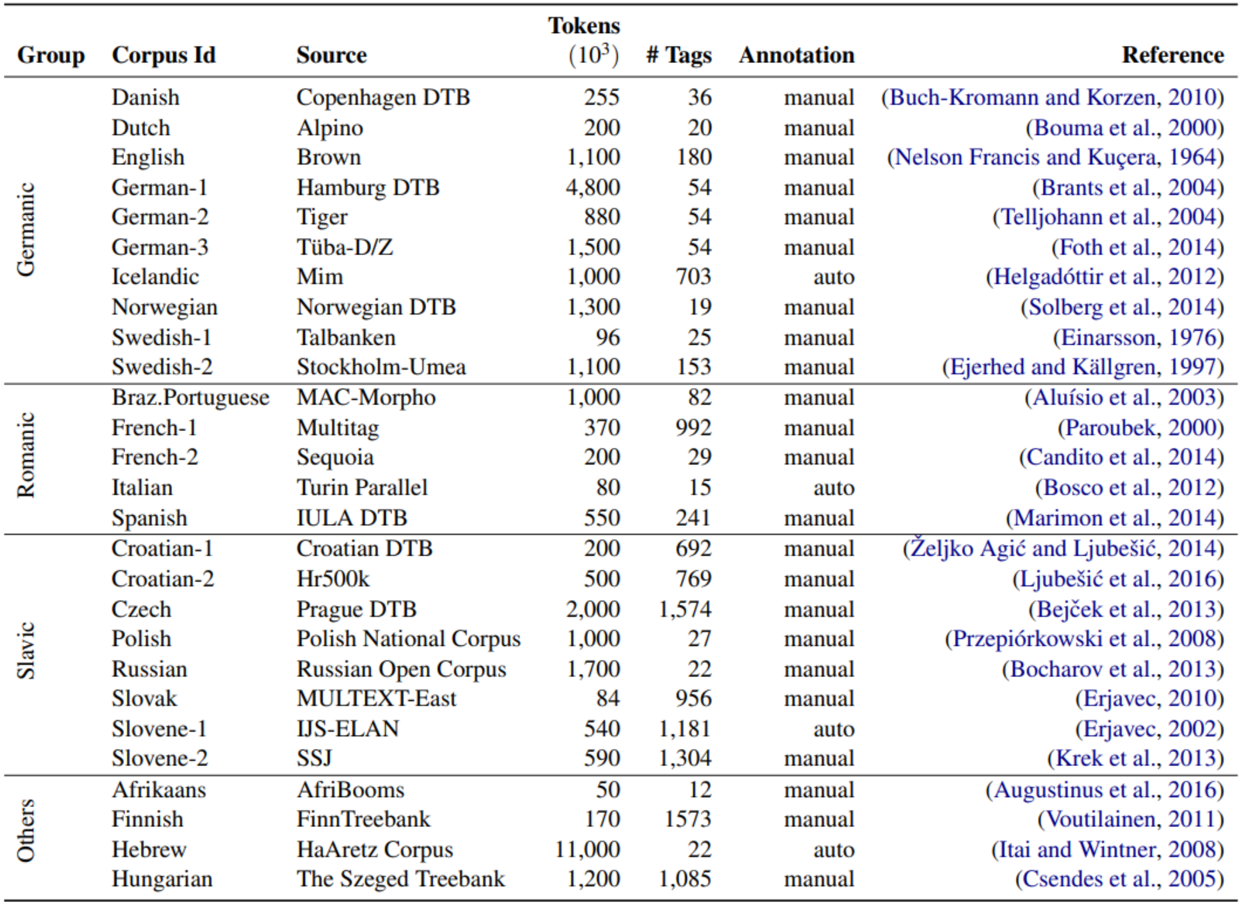
We, thus, consider it worthwhile to explore if the results are reproducible using corpora with fine-grained tagsets. We use the LSTM tagger provided by Plank et al. (2016) and compare the results likewise to CRF and an off-the-shelf HMM tagger implementation. We compile a fresh set of 27 corpora of 21 languages which uses the commonly used *fine-grained* tagset of the respective

4

Deep Learning Approaches

Do LSTMs really work so well for PoS tagging?

Corpora used in the experiments

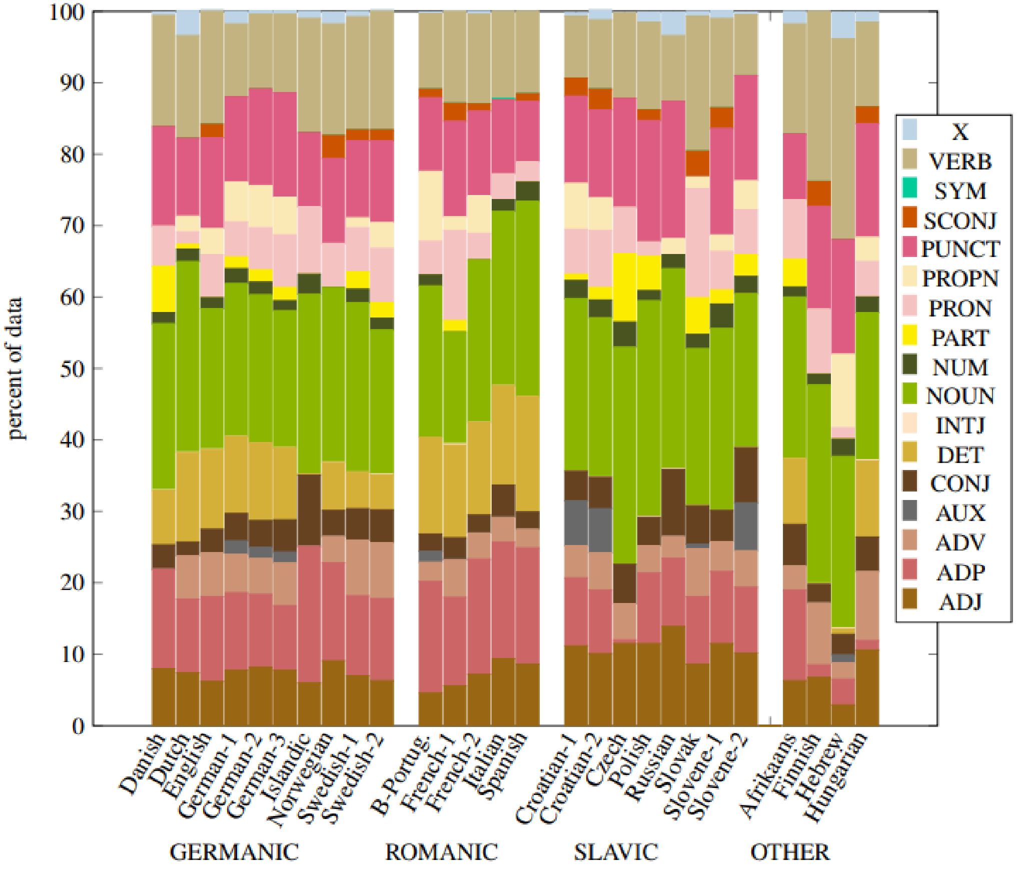


Group	Corpus Id	Source	Tokens (10^3)	# Tags	Annotation	Reference
Germanic	Danish	Copenhagen DTB	255	36	manual	(Buch-Kromann and Korzen, 2010)
	Dutch	Alpino	200	20	manual	(Bouma et al., 2000)
	English	Brown	1,100	180	manual	(Nelson Francis and Kuçera, 1964)
	German-1	Hamburg DTB	4,800	54	manual	(Brants et al., 2004)
	German-2	Tiger	880	54	manual	(Telljohann et al., 2004)
	German-3	Tüba-D/Z	1,500	54	manual	(Foth et al., 2014)
	Icelandic	Mim	1,000	703	auto	(Helgadóttir et al., 2012)
	Norwegian	Norwegian DTB	1,300	19	manual	(Solberg et al., 2014)
	Swedish-1	Talbanken	96	25	manual	(Einarsson, 1976)
	Swedish-2	Stockholm-Umea	1,100	153	manual	(Ejerhed and Källgren, 1997)
Romanic	Braz.Portuguese	MAC-Morpho	1,000	82	manual	(Aluísio et al., 2003)
	French-1	Multitag	370	992	manual	(Paroubek, 2000)
	French-2	Sequoia	200	29	manual	(Candito et al., 2014)
	Italian	Turin Parallel	80	15	auto	(Bosco et al., 2012)
	Spanish	IULA DTB	550	241	manual	(Marimon et al., 2014)
Slavic	Croatian-1	Croatian DTB	200	692	manual	(Željko Agić and Ljubešić, 2014)
	Croatian-2	Hr500k	500	769	manual	(Ljubešić et al., 2016)
	Czech	Prague DTB	2,000	1,574	manual	(Bejček et al., 2013)
	Polish	Polish National Corpus	1,000	27	manual	(Przepiórkowski et al., 2008)
	Russian	Russian Open Corpus	1,700	22	manual	(Bocharov et al., 2013)
	Slovak	MULTEXT-East	84	956	manual	(Erjavec, 2010)
	Slovene-1	IJS-ELAN	540	1,181	auto	(Erjavec, 2002)
	Slovene-2	SSJ	590	1,304	manual	(Krek et al., 2013)
Others	Afrikaans	AfriBooms	50	12	manual	(Augustinus et al., 2016)
	Finnish	FinnTreebank	170	1573	manual	(Voutilainen, 2011)
	Hebrew	HaAretz Corpus	11,000	22	auto	(Itai and Wintner, 2008)
	Hungarian	The Szeged Treebank	1,200	1,085	manual	(Csendes et al., 2005)

4 Deep Learning Approaches

Do LSTMs really work so well for PoS tagging?

Coarse-grained PoS tag distribution of corpora by language group

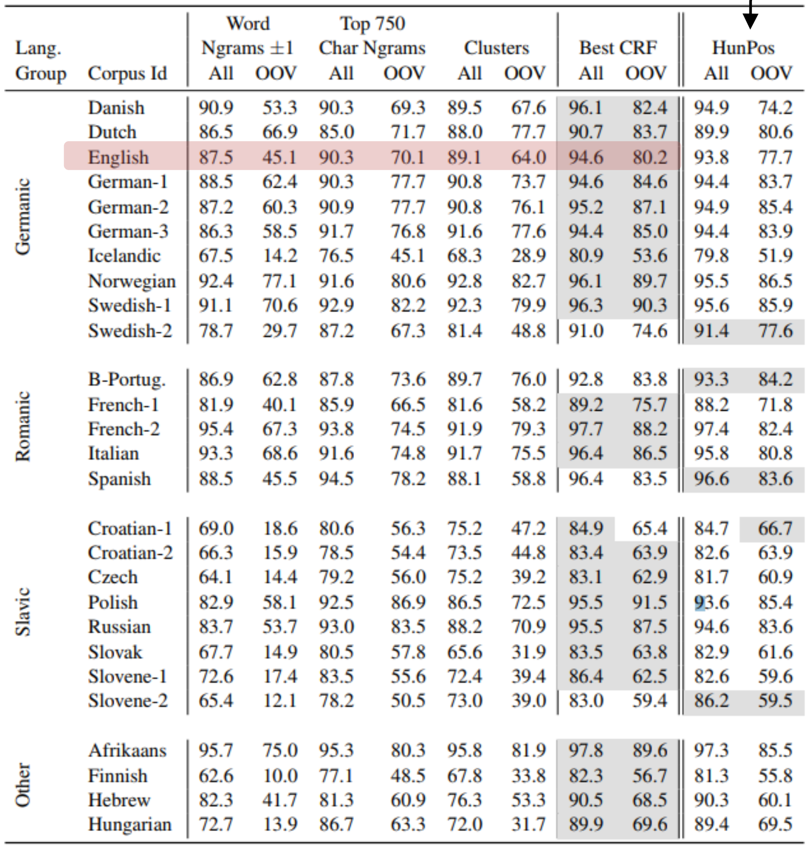


4 Deep Learning Approaches

Do LSTMs really work so well for PoS tagging?

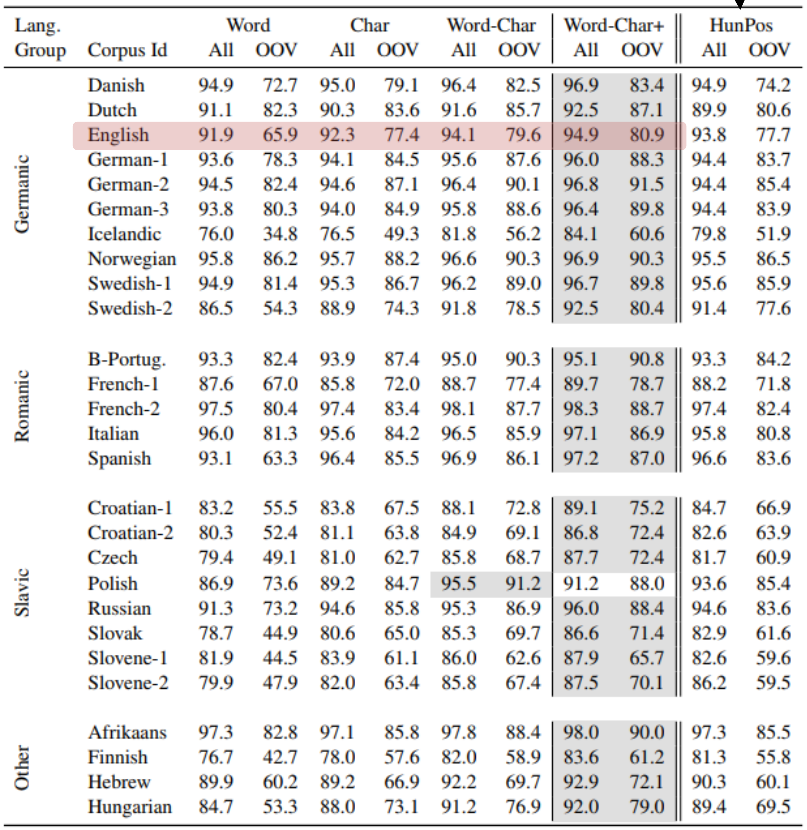
(Horsmann and Zesch, 2017)

HMM POS Tagger



Lang. Group	Corpus Id	Word Ngrams ±1 All	Word Ngrams ±1 OOV	Top 750 Char Ngrams All	Top 750 Char Ngrams OOV	Clusters All	Clusters OOV	Best CRF All	Best CRF OOV	HunPos All	HunPos OOV
Germanic	Danish	90.9	53.3	90.3	69.3	89.5	67.6	96.1	82.4	94.9	74.2
	Dutch	86.5	66.9	85.0	71.7	88.0	77.7	90.7	83.7	89.9	80.6
	English	87.5	45.1	90.3	70.1	89.1	64.0	94.6	80.2	93.8	77.7
	German-1	88.5	62.4	90.3	77.7	90.8	73.7	94.6	84.6	94.4	83.7
	German-2	87.2	60.3	90.9	77.7	90.8	76.1	95.2	87.1	94.9	85.4
	German-3	86.3	58.5	91.7	76.8	91.6	77.6	94.4	85.0	94.4	83.9
	Icelandic	67.5	14.2	76.5	45.1	68.3	28.9	80.9	53.6	79.8	51.9
	Norwegian	92.4	77.1	91.6	80.6	92.8	82.7	96.1	89.7	95.5	86.5
	Swedish-1	91.1	70.6	92.9	82.2	92.3	79.9	96.3	90.3	95.6	85.9
	Swedish-2	78.7	29.7	87.2	67.3	81.4	48.8	91.0	74.6	91.4	77.6
Romanic	B-Portug.	86.9	62.8	87.8	73.6	89.7	76.0	92.8	83.8	93.3	84.2
	French-1	81.9	40.1	85.9	66.5	81.6	58.2	89.2	75.7	88.2	71.8
	French-2	95.4	67.3	93.8	74.5	91.9	79.3	97.7	88.2	97.4	82.4
	Italian	93.3	68.6	91.6	74.8	91.7	75.5	96.4	86.5	95.8	80.8
	Spanish	88.5	45.5	94.5	78.2	88.1	58.8	96.4	83.5	96.6	83.6
Slavic	Croatian-1	69.0	18.6	80.6	56.3	75.2	47.2	84.9	65.4	84.7	66.7
	Croatian-2	66.3	15.9	78.5	54.4	73.5	44.8	83.4	63.9	82.6	63.9
	Czech	64.1	14.4	79.2	56.0	75.2	39.2	83.1	62.9	81.7	60.9
	Polish	82.9	58.1	92.5	86.9	86.5	72.5	95.5	91.5	93.6	85.4
	Russian	83.7	53.7	93.0	83.5	88.2	70.9	95.5	87.5	94.6	83.6
	Slovak	67.7	14.9	80.5	57.8	65.6	31.9	83.5	63.8	82.9	61.6
	Slovene-1	72.6	17.4	83.5	55.6	72.4	39.4	86.4	62.5	82.6	59.6
	Slovene-2	65.4	12.1	78.2	50.5	73.0	39.0	83.0	59.4	86.2	59.5
Other	Afrikaans	95.7	75.0	95.3	80.3	95.8	81.9	97.8	89.6	97.3	85.5
	Finnish	62.6	10.0	77.1	48.5	67.8	33.8	82.3	56.7	81.3	55.8
	Hebrew	82.3	41.7	81.3	60.9	76.3	53.3	90.5	68.5	90.3	60.1
	Hungarian	72.7	13.9	86.7	63.3	72.0	31.7	89.9	69.6	89.4	69.5

Table 2: Accuracy of CRF taggers (10fold CV)

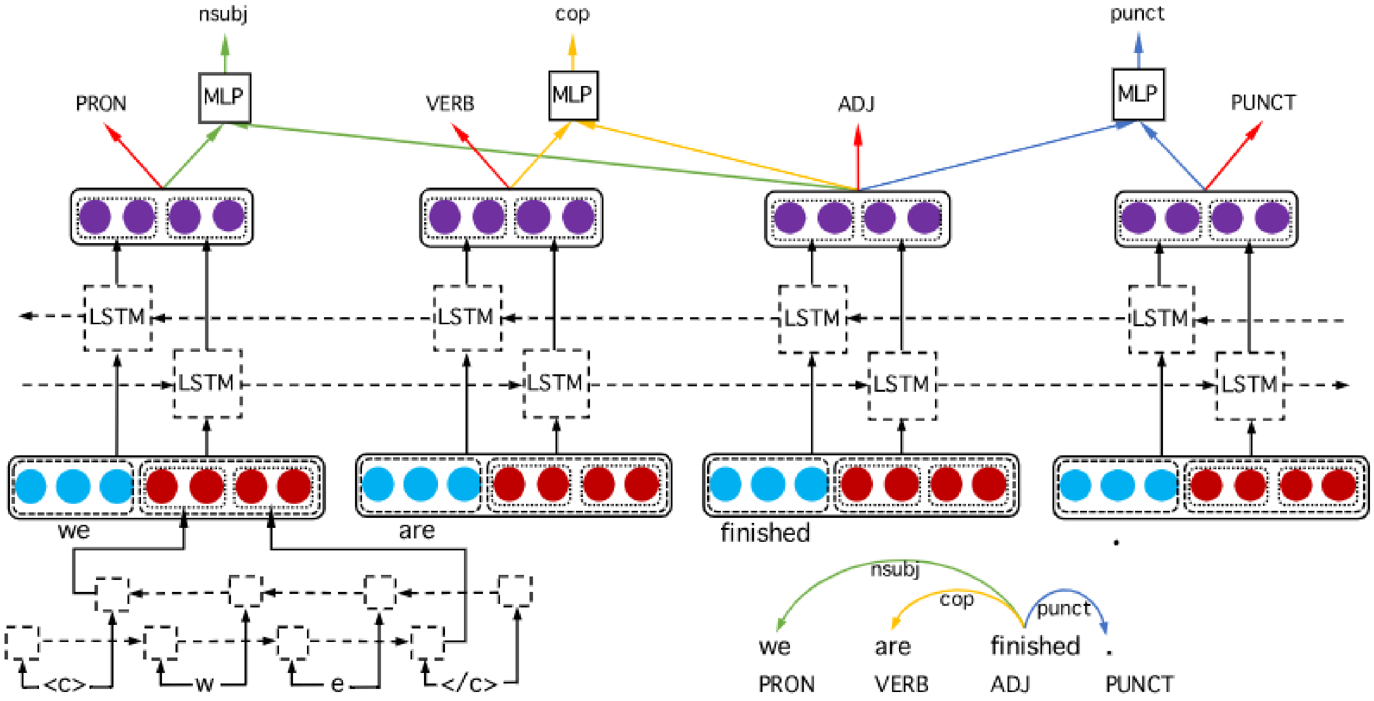


Lang. Group	Corpus Id	Word All	Word OOV	Char All	Char OOV	Word-Char All	Word-Char OOV	Word-Char+ All	Word-Char+ OOV	HunPos All	HunPos OOV
Germanic	Danish	94.9	72.7	95.0	79.1	96.4	82.5	96.9	83.4	94.9	74.2
	Dutch	91.1	82.3	90.3	83.6	91.6	85.7	92.5	87.1	89.9	80.6
	English	91.9	65.9	92.3	77.4	94.1	79.6	94.9	80.9	93.8	77.7
	German-1	93.6	78.3	94.1	84.5	95.6	87.6	96.0	88.3	94.4	83.7
	German-2	94.5	82.4	94.6	87.1	96.4	90.1	96.8	91.5	94.4	85.4
	German-3	93.8	80.3	94.0	84.9	95.8	88.6	96.4	89.8	94.4	83.9
	Icelandic	76.0	34.8	76.5	49.3	81.8	56.2	84.1	60.6	79.8	51.9
	Norwegian	95.8	86.2	95.7	88.2	96.6	90.3	96.9	90.3	95.5	86.5
	Swedish-1	94.9	81.4	95.3	86.7	96.2	89.0	96.7	89.8	95.6	85.9
	Swedish-2	86.5	54.3	88.9	74.3	91.8	78.5	92.5	80.4	91.4	77.6
Romanic	B-Portug.	93.3	82.4	93.9	87.4	95.0	90.3	95.1	90.8	93.3	84.2
	French-1	87.6	67.0	85.8	72.0	88.7	77.4	89.7	78.7	88.2	71.8
	French-2	97.5	80.4	97.4	83.4	98.1	87.7	98.3	88.7	97.4	82.4
	Italian	96.0	81.3	95.6	84.2	96.5	85.9	97.1	86.9	95.8	80.8
	Spanish	93.1	63.3	96.4	85.5	96.9	86.1	97.2	87.0	96.6	83.6
Slavic	Croatian-1	83.2	55.5	83.8	67.5	88.1	72.8	89.1	75.2	84.7	66.9
	Croatian-2	80.3	52.4	81.1	63.8	84.9	69.1	86.8	72.4	82.6	63.9
	Czech	79.4	49.1	81.0	62.7	85.8	68.7	87.7	72.4	81.7	60.9
	Polish	86.9	73.6	89.2	84.7	95.5	91.2	91.2	88.0	93.6	85.4
	Russian	91.3	73.2	94.6	85.8	95.3	86.9	96.0	88.4	94.6	83.6
	Slovak	78.7	44.9	80.6	65.0	85.3	69.7	86.6	71.4	82.9	61.6
	Slovene-1	81.9	44.5	83.9	61.1	86.0	62.6	87.9	65.7	82.6	59.6
	Slovene-2	79.9	47.9	82.0	63.4	85.8	67.4	87.5	70.1	86.2	59.5
Other	Afrikaans	97.3	82.8	97.1	85.8	97.8	88.4	98.0	90.0	97.3	85.5
	Finnish	76.7	42.7	78.0	57.6	82.0	58.9	83.6	61.2	81.3	55.8
	Hebrew	89.9	60.2	89.2	66.9	92.2	69.7	92.9	72.1	90.3	60.1
	Hungarian	84.7	53.3	88.0	73.1	91.2	76.9	92.0	79.0	89.4	69.5

Table 3: Accuracy of LSTM taggers (10fold CV)

LSTM-based POS Tagging

llustration of LSTM-based joint POS tagging and graph-based dependency parsing.



/

Reference

Summary

- M. Marcus, B. Santorini and M.A. Marcinkiewicz (1993). Building a large annotated corpus of English: The Penn Treebank. In Computational Linguistics, volume 19, number 2, pp. 313–330.
- Nguyen, D. Q., Dras, M., & Johnson, M. (2017). A novel neural network model for joint pos tagging and graph-based dependency parsing. arXiv preprint arXiv:1705.05952.
- Ling, W., Luís, T., Marujo, L., Astudillo, R. F., Amir, S., Dyer, C., ... & Trancoso, I. (2015). Finding function in form: Compositional character models for open vocabulary word representation. arXiv preprint arXiv:1508.02096.
- Horsmann, T., & Zesch, T. (2017). Do LSTMs really work so well for PoS tagging?–A replication study. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 727-736).
- Speech and Language Processing. Daniel Jurafsky & James H. Martin. Copyright c 2019. All rights reserved. Draft of October 2, 2019.