TTIC 31190: Natural Language Processing

Kevin Gimpel

Spring 2018

Lecture 11: Part-of-Speech Tagging and other Sequence Labeling Tasks

Assignment 2 due today

- questions?

Project Proposal

- project proposal due May 9
- details have been posted
- groups of 2-3 are ok (but think about how you will divide up the work, especially with 3)

Project Details

- ideas:
  - replicate (part of) a published paper
  - apply NLP methods to a dataset or task related to your research
  - define a new NLP task/dataset
- if you're working on a standard task, you do not need to have state of the art results
- but your project should be done carefully so that you can have confidence in your claims
- try to avoid a project that's too ambitious

Project Report

- final report due June 6
  - May 30 for graduating students
- details forthcoming on project report format

Midterm

- midterm on Wednesday, May 16th
- don’t worry about memorizing stuff
- we’ll give you most of the formulas/definitions you will need

Roadmap

- words, morphology, lexical semantics
- text classification
- language modeling
- word embeddings
- recurrent/recursive/convolutional networks in NLP
- sequence labeling, HMMs, dynamic programming
- syntax and syntactic parsing
- semantics, compositionality, semantic parsing
- machine translation and other NLP tasks

Encoders

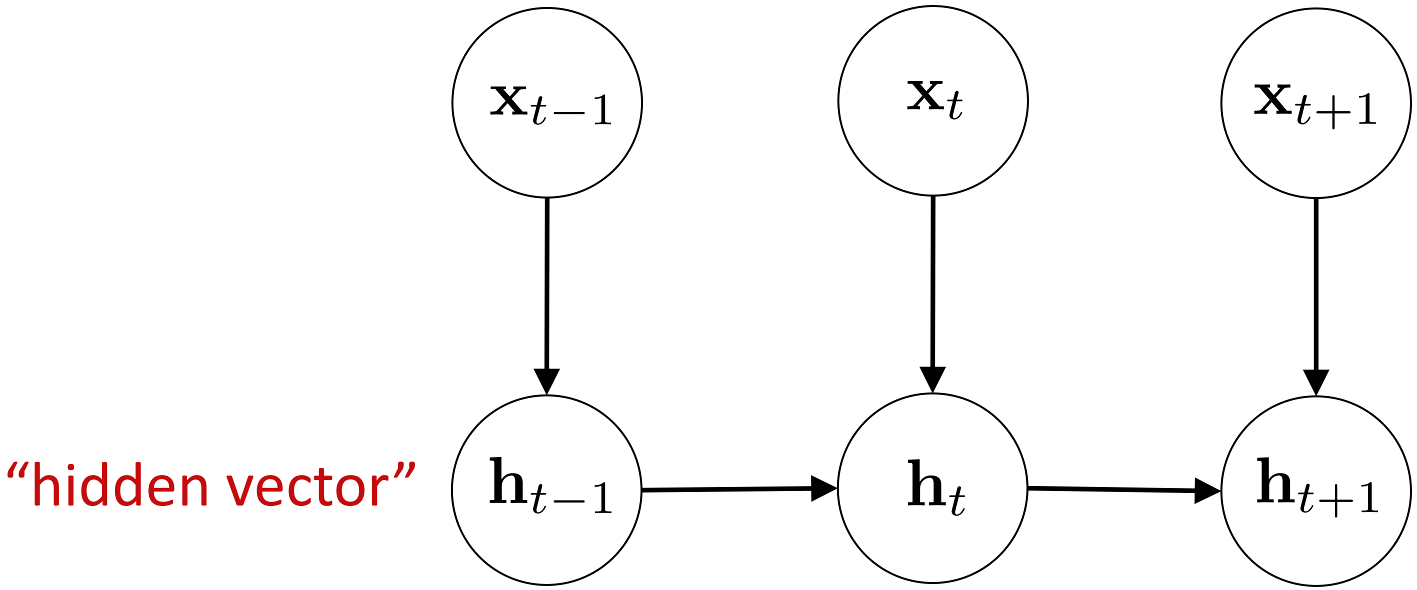
- encoder: a function to represent a word sequence as a vector
- simplest: average word embeddings:

$$\mathbf{f}_{\mathrm{avg}}(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} emb(x_i)$$

- other choices: LSTMs, GRUs, CNNs, attention-weighted sum, etc.

Recurrent Neural Networks

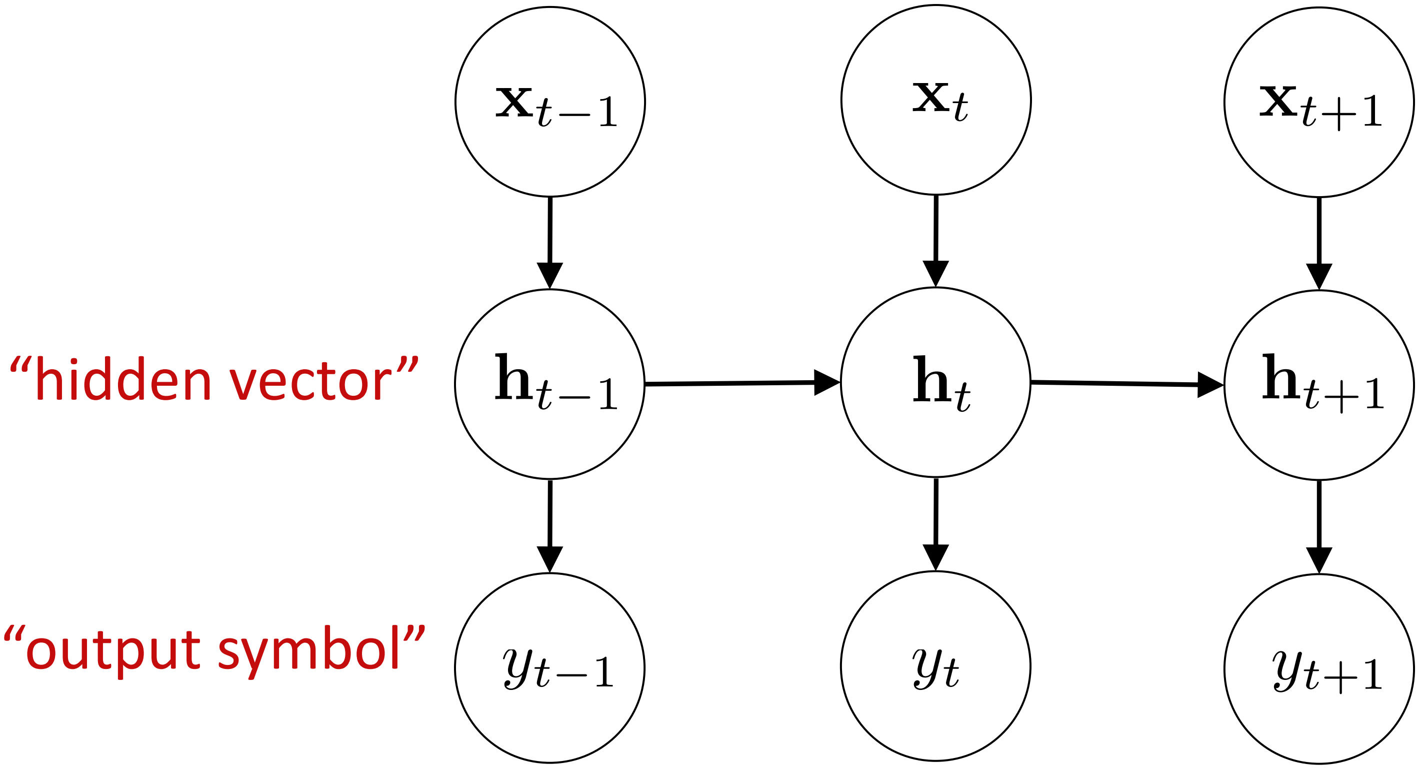
Input is a sequence:



- so far, we’ve used RNNs to encode sequences
- for tasks like sequence classification
  - also used in translation, question answering, summarization, etc.
- but RNNs are also frequently used for **generating** sequences

"Output" Recurrent Neural Networks

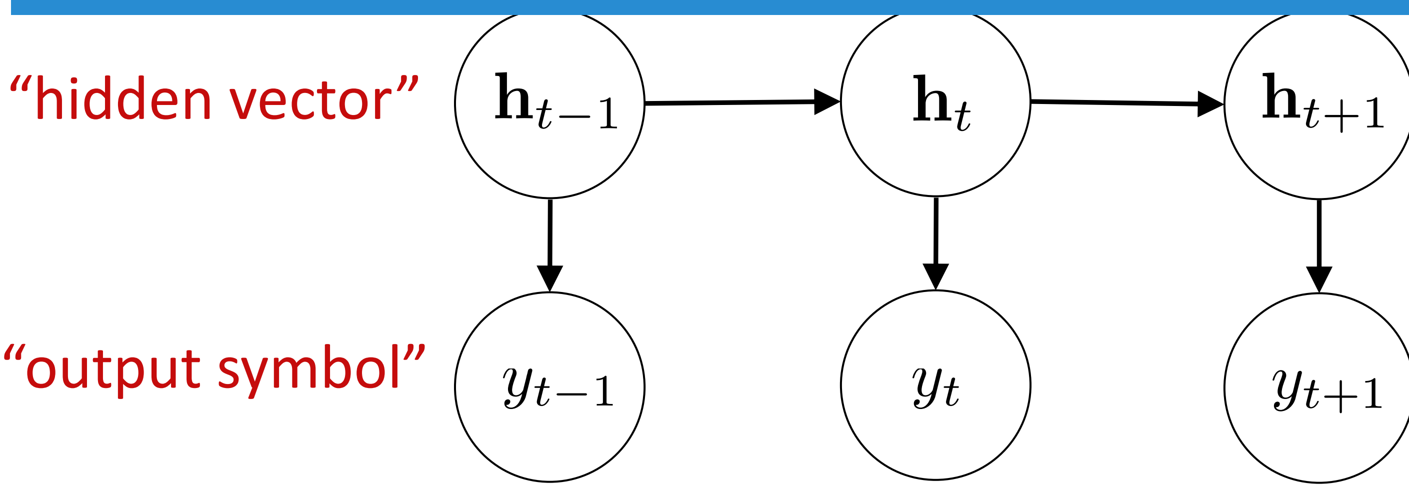
$$\mathbf{h}_t = \tanh\left(\mathbf{W}^{(x)}\mathbf{x}_t + \mathbf{W}^{(h)}\mathbf{h}_{t-1} + \mathbf{b}\right)$$



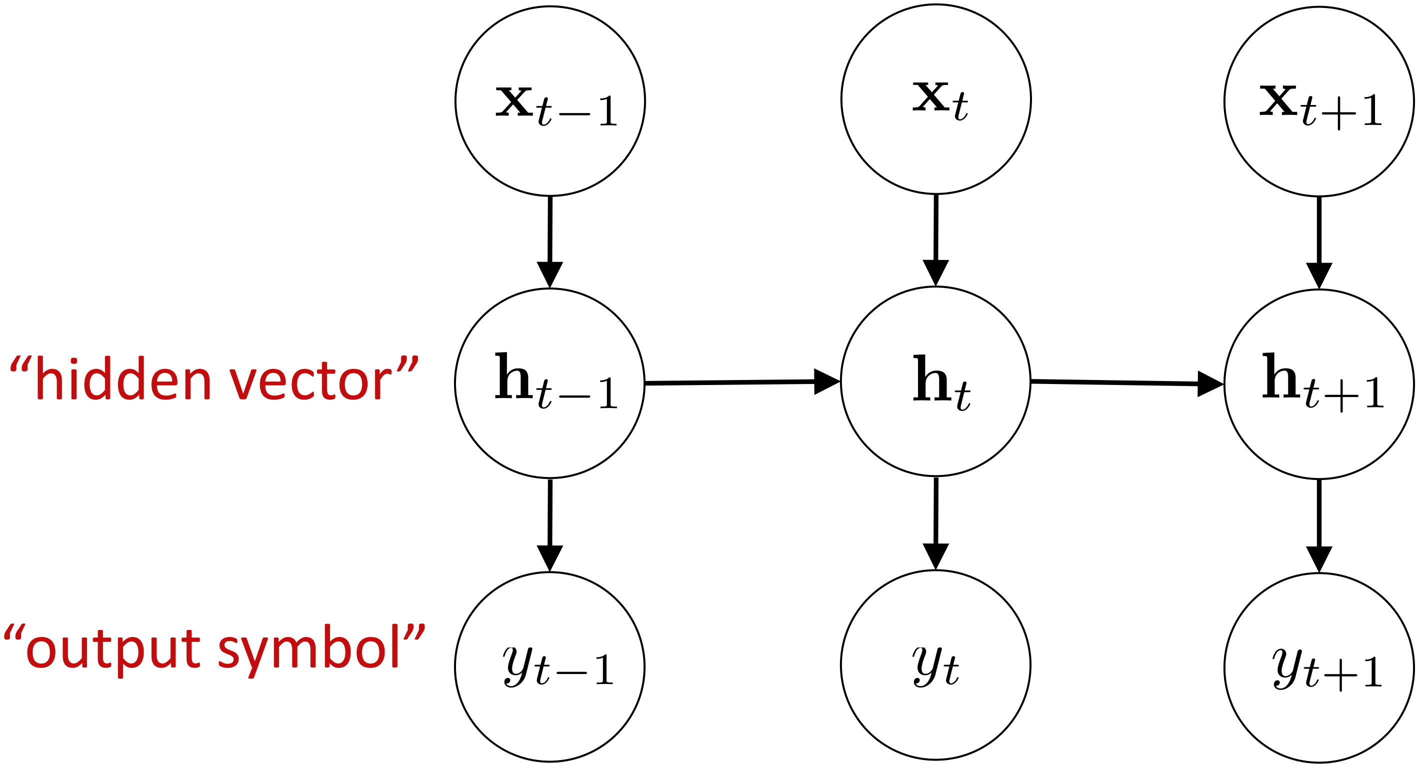
$$y_t = \operatorname*{argmax}_{y \in \mathcal{O}} \; \mathbf{h}_t^\top emb(y)$$

ks

- *y* is a symbol, not a vector
- *O* is the “output” vocabulary
- we have a new parameter vector *emb(y)* for each output symbol in *O*
- *emb(y)* = **x**?
- probability distribution over output symbols?



$$y_t = \underset{y \in \mathcal{O}}{\operatorname{argmax}} \; \mathbf{h}_t^\top emb(y)$$

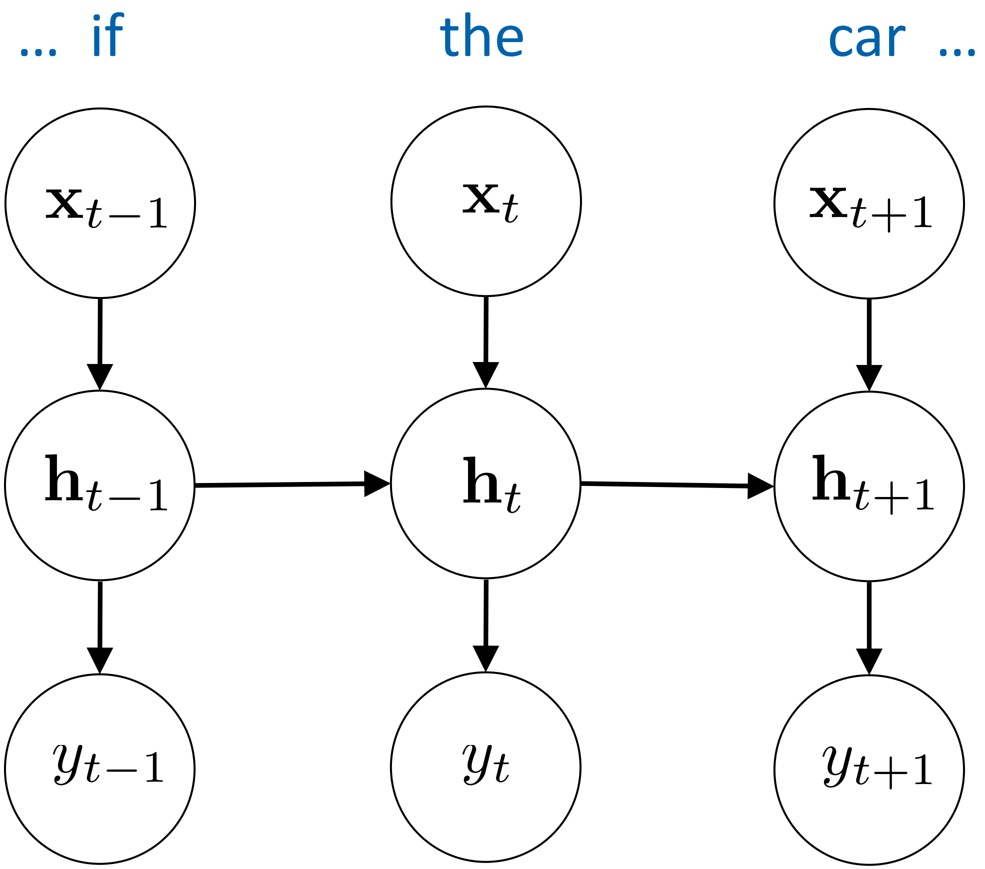


$$y_t = \operatorname*{argmax}_{y \in \mathcal{O}} \ \mathbf{h}_t^\top emb(y)$$

$$P(Y_t) = \operatorname{softmax}\left(\mathbf{W}\mathbf{h}_t\right)$$

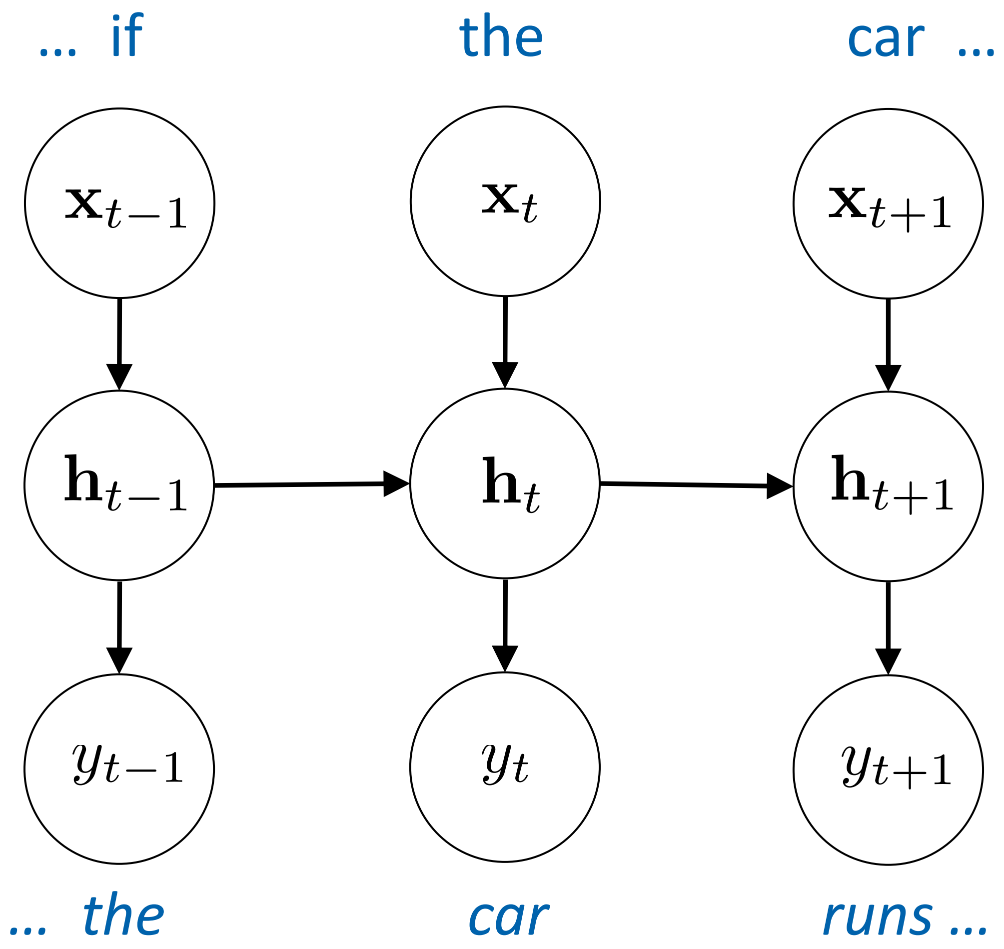
$$\mathbf{W} = \left[emb(y_1)^\top; emb(y_2)^\top; \ldots; emb(y_{|\mathcal{O}|})^\top\right]$$

Example: Language Modeling



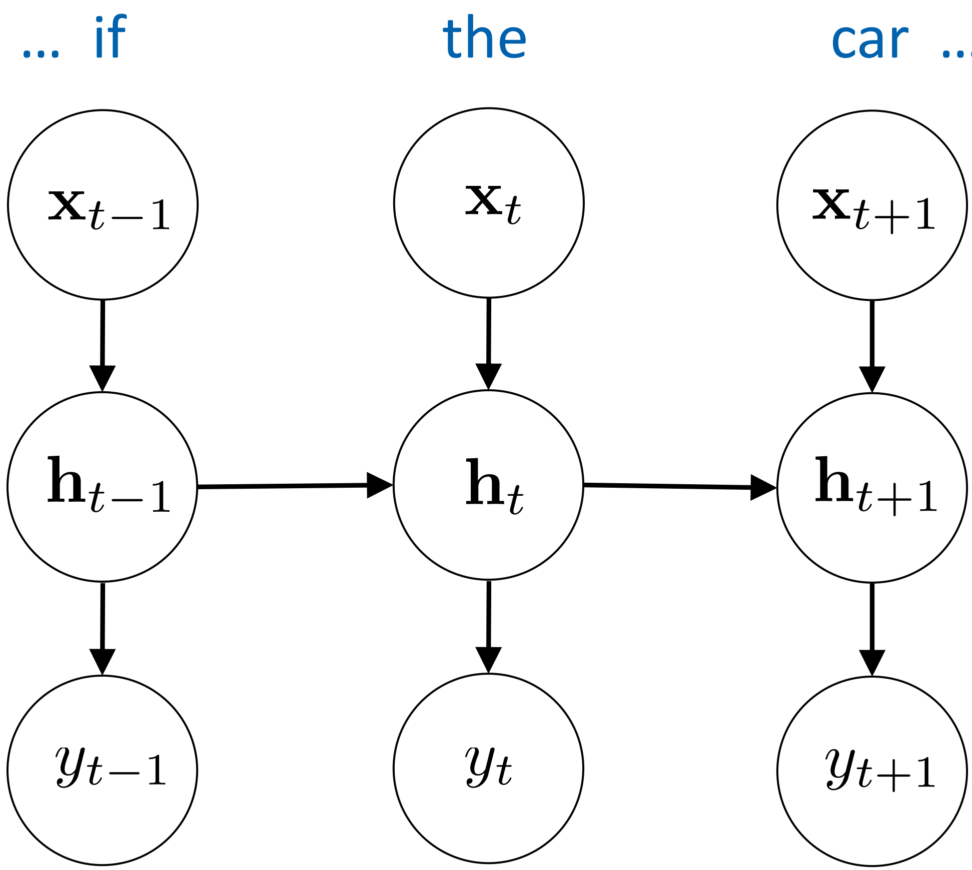
- input: a word sequence
- output?

Example: Language Modeling



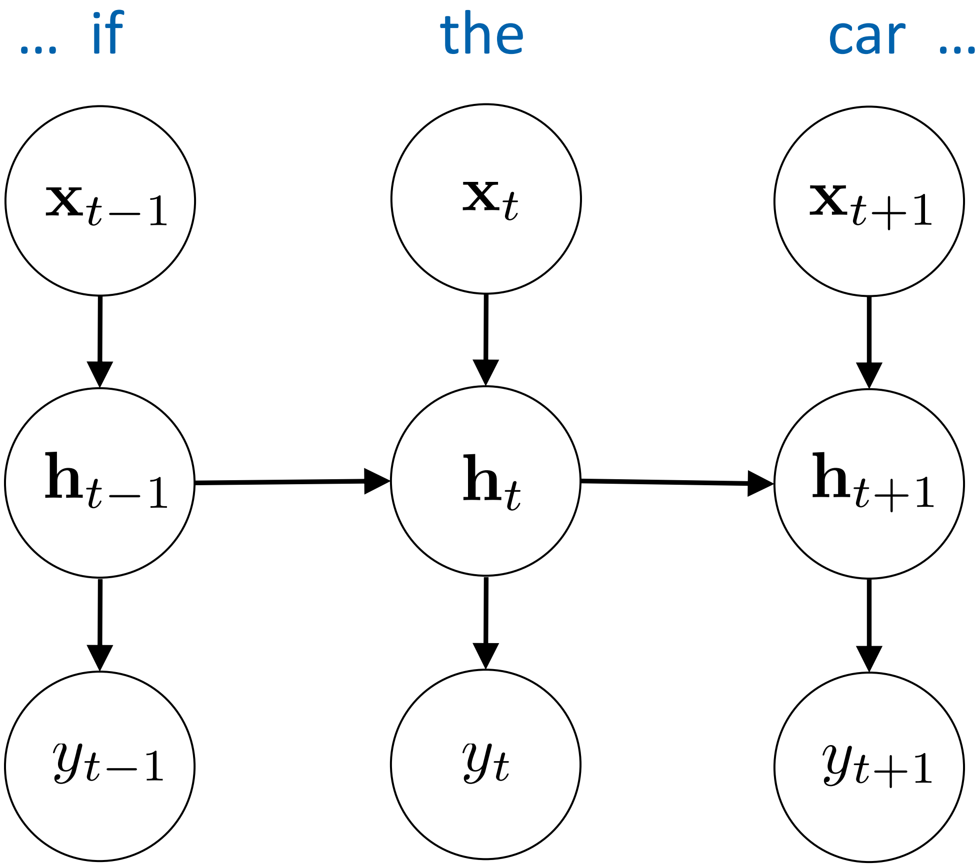
- target output at each position: next word in the sequence!

Language Modeling: Training



$$-\log P(Y_{t-1} = ?)$$

Language Modeling: Training



$$- \log P(Y_{t-1} = \text{“the”}) - \log P(Y_t = \text{“car”}) \ \ldots$$

- while we showed this for simple RNNs, it’s easy to instead use LSTMs, GRUs, etc.
- LSTMs/GRUs still produce a hidden vector at each position in the sequence, just like RNNs
- LSTM = most common choice for language modeling

Linguistic phenomena: summary so far…

- words have structure (stems and affixes)
- words have multiple meanings (senses) → word sense ambiguity
  - senses of a word can be homonymous or polysemous
  - senses have relationships:
    - synonymy, hyponymy (“is a”), meronymy (“part of”, “member of”)
- variability/flexibility of linguistic expression
  - many ways to express the same meaning (as you saw in Assignment 2)
  - word embeddings tell us when two words are similar
- today: **part-of-speech**

Part-of-Speech Tagging

Some questioned if Tim Cook ’s first product

would be a breakaway hit for Apple .

Part-of-Speech Tagging

determiner	verb (past)	prep.	proper noun	proper noun	poss.	adj.	noun
Some	questioned	if	Tim	Cook	’s	first	product

modal	verb	det.	adjective	noun	prep.	proper noun	punc.
would	be	a	breakaway	hit	for	Apple	.

Part-of-Speech (POS)

- functional category of a word:
  – noun, verb, adjective, etc.
  – how is the word functioning in its context?
- dependent on context like word sense, but different from sense:
  – sense represents word meaning, POS represents word function
  – sense uses a distinct category of senses per word, POS uses same set of categories for all words

Penn Treebank tag set

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two*	TO	“to”	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential ‘there’	*there*	VB	verb base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb gerund	*eating*
JJ	adjective	*yellow*	VBN	verb past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, sing.	*IBM*	$	dollar sign	*$*
NNPS	proper noun, plural	*Carolinas*	#	pound sign	*#*
PDT	predeterminer	*all, both*	“	left quote	‘ or “
POS	possessive ending	*’s*	”	right quote	’ or ”
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one’s*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

POS Ambiguity in Penn Treebank

- word that can be both noun and verb?
  - more often noun than verb:
    - increase: 248 NN vs. 127 VB (and 4 VBP)
    - place: 134 NN vs. 14 VB (and 4 VBP)
  - more often verb than noun:
    - makes: 182 VBZ vs. 5 NNS
    - transfer: 22 VB vs. 16 NN

POS Ambiguity in Penn Treebank

- word that can be both a singular noun and a plural noun?
  - "savings", e.g.:

DT	NN	VBD	VBN	RB
The	savings	was	given	incorrectly …

DT	JJ	NN	NN
a	Belgian	savings	bank

POS Ambiguity in Penn Treebank

- word that can be both a common noun and a proper noun?
  - "Earth": 16 NNP vs. 5 NN
  - annotation inconsistencies: nothing in the context indicates which tag is used
  - these kinds of inconsistencies are common in annotated datasets, so it's usually not possible to get perfect accuracy

POS Ambiguity in Penn Treebank

- word that can be both a common noun and a proper noun?
  - "Chapter": 21 NNP vs. 41 NN
  - annotation inconsistencies:

VB	VBG	IN	NNP	NNP	NN	NN
consider	filing	for	Chapter	11	bankruptcy	protection

NNP	VBD	IN	NN	CD	NN	NN
Continental	filed	for	Chapter	11	bankruptcy	protection

How many tags can a word have?

words in Penn Treebank with the most unique tags:

```
7 down
6 that
6 set
6 put
6 open
6 hurt
6 cut
6 bet
6 back
5 vs.
5 the
5 spread
5 split
5 say
```

How many tags can a word have?

tag counts for `down`:

```
353 down  RB
214 down  RP
142 down  IN
 10 down  JJ
  1 down  VBP
  1 down  RBR
  1 down  NN
```

How many tags can a word have?

tag counts for `down`:

```
353 down  RB  adverb
214 down  RP  particle
142 down  IN  preposition
 10 down  JJ  adjective
  1 down  VBP verb (past tense)
  1 down  RBR comparative adverb
  1 down  NN  singular noun
```

`RP` tag: particle

- test for verb particle:
- can you insert a modifier between the verb and its particle without it sounding weird?
  - `take the trash out immediately`
  - `*take the trash immediately out`

  - `take the trash outside immediately`
  - `take the trash immediately outside`
- `out` is a particle here, while `outside` is not

What about `vs.`?

tag counts for `vs.`:

```
15 vs.  FW
 9 vs.  IN
 6 vs.  CC
 2 vs.  NN
 1 vs.  JJ
```

Universal Tag Set

- many use smaller sets of coarser tags
- e.g., "universal tag set" containing 12 tags:
  – noun, verb, adjective, adverb, pronoun, determiner/article, adposition (preposition or postposition), numeral, conjunction, particle, punctuation, other

sentence:	The	oboist	Heinz	Holliger	has	taken	a	hard	line	about	the	problems	.
original:	DT	NN	NNP	NNP	VBZ	VBN	DT	JJ	NN	IN	DT	NNS	.
universal:	DET	NOUN	NOUN	NOUN	VERB	VERB	DET	ADJ	NOUN	ADP	DET	NOUN	.

Figure 1: Example English sentence with its language specific and corresponding universal POS tags.

Petrov, Das, McDonald (2011)

Twitter Part-of-Speech Tagging

other verb article noun pronoun

intj pronoun prep adj prep verb

ikr smh he asked fir yo last name so he can

add u on fb lololol =D #lolz

verb prep intj emoticon hashtag

pronoun proper noun

adj = adjective
prep = preposition
intj = interjection

- we removed some fine-grained POS tags, then added Twitter-specific tags:

  hashtag

  @-mention

  URL / email address

  emoticon

  Twitter discourse marker

  other (multi-word abbreviations, symbols, garbage)

- in Penn Treebank (1M words), word with most tags had 7 tags

- in Twitter POS-annotated data (40k words), word with most tags has how many tags?

How many tags can a word have?

words in Twitter with the most unique tags:

```
7 over
5 up
5 out
5 one
5 off
5 a
5 @
4 to
4 there
4 that
4 right
4 outside
4 no
4 n
```

How many tags can a word have?

words in Twitter with the most unique tags:

```
7 over
5 up
5 out
5 one
5 off
5 a
5
4
```

```
4 there
4 that
4 right
4 outside
4 no
4 n
```

Twitter shows a wider variety of uses for common words

word sense vs. part-of-speech

	word sense	part-of-speech
semantic or syntactic?	semantic: indicates meaning of word in its context	syntactic: indicates function of word in its context
number of categories		
inter-annotator agreement		
independent or joint classification of nearby words?		

word sense vs. part-of-speech

	word sense	part-of-speech
semantic or syntactic?	semantic: indicates meaning of word in its context	syntactic: indicates function of word in its context
number of categories	$\|V\|$ words, ~5 senses each → 5$\|V\|$ categories!	typical POS tag sets have 12 to 45 tags
inter-annotator agreement	low; some sense distinctions are highly subjective	high; relatively few POS tags and function is relatively shallow / surface-level
independent or joint classification of nearby words?	independent: can classify a single word based on context words; structured prediction is rarely used	joint: strong relationship between tags of nearby words; structured prediction often used

How might POS tags be useful?

- text classification
- machine translation
- question answering
- speech synthesis (pronounce “contract”)
- …

Models for POS Tagging

- today we'll discuss simple models that do not use structured prediction
- these are often called "local" models
- they predict a tag for each word in a sequence, (and can use the entire word sequence to make each prediction)
- but they do not use information about previous *predictions* to make later predictions
- by contrast, **structured prediction**:
  - predict structures
  - or: make multiple predictions jointly

Feed-Forward Neural Networks for Twitter POS Tagging

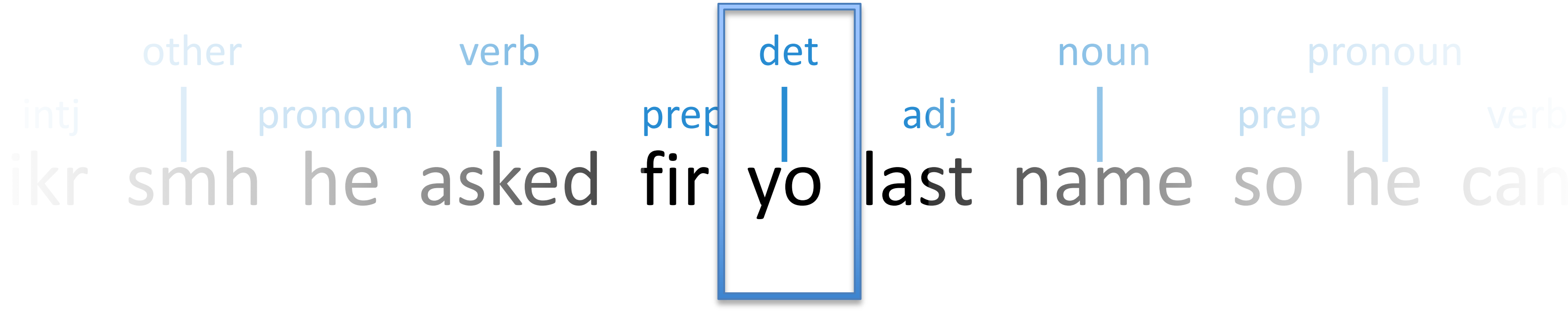
ikr	smh	he	asked	fir	yo	last	name	so	he	can
intj	other	pronoun	verb	prep	det	adj	noun	prep	pronoun	verb

add	u	on	fb	lololol
verb	pronoun	prep	proper noun	intj

adj = adjective
prep = preposition
intj = interjection

- in Assignment 3, you’ll build a neural network classifier to predict a word’s POS tag based on its context

Feed-Forward Neural Networks for Twitter POS Tagging



- e.g., predict tag of *yo* given context
- what should the input **x** be to the neural network?
  - it has to be independent of the label
  - it has to be a **fixed-length** vector

Feed-Forward Neural Networks for Twitter POS Tagging

intj	other	pronoun	verb	prep	det	adj	noun	prep	pronoun	verb
ikr	smh	he	asked	fir	**yo**	last	name	so	he	can

- e.g., predict tag of *yo* given context
- what should the input **x** be?

$$\mathbf{x} = \underbrace{[0.4\ \ 0.1\ \ \ldots\ \ 0.9]}_{\text{word vector for } yo}{}^{\top}$$

Feed-Forward Neural Networks for Twitter POS Tagging

intj other pronoun verb prep det adj noun prep pronoun verb

ikr smh he asked fir yo last name so he can

- e.g., predict tag of *yo* given context
- what should the input **x** be?

$$\mathbf{x} = [\underbrace{-0.2 \;\; 0.5 \;\; \ldots \;\; 0.8}_{\text{word vector for } \textit{fir}} \;\; \underbrace{0.4 \;\; 0.1 \;\; \ldots \;\; 0.9}_{\text{word vector for } \textit{yo}}]^\top$$

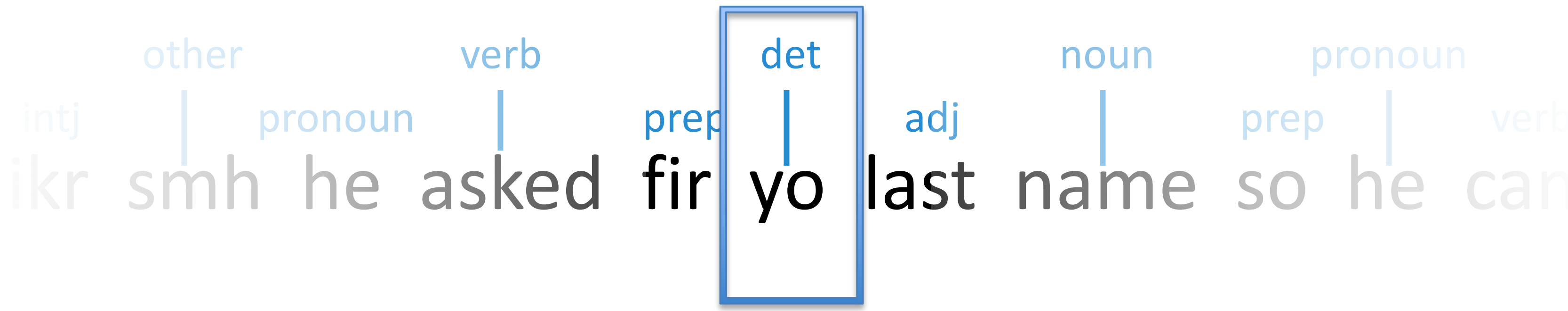
Feed-Forward Neural Networks for Twitter POS Tagging

intj	other	pronoun	verb	prep	det	adj	noun	prep	pronoun	verb
ikr	smh	he	asked	fir	**yo**	last	name	so	he	can

- when using word vectors as part of input, we can also treat them as more parameters to be learned!
- this is called "updating" or "fine-tuning" the vectors (since they are initialized using something like `word2vec`)

$$\mathbf{x} = [\underbrace{-0.2\ \ 0.5\ \ \ldots\ \ 0.8}_{\text{word vector for } \textit{fir}}\ \ \underbrace{0.4\ \ 0.1\ \ \ldots\ \ 0.9}_{\text{word vector for } \textit{yo}}]^\top$$

Feed-Forward Neural Networks for Twitter POS Tagging



- let’s use the center word + two words to the right:

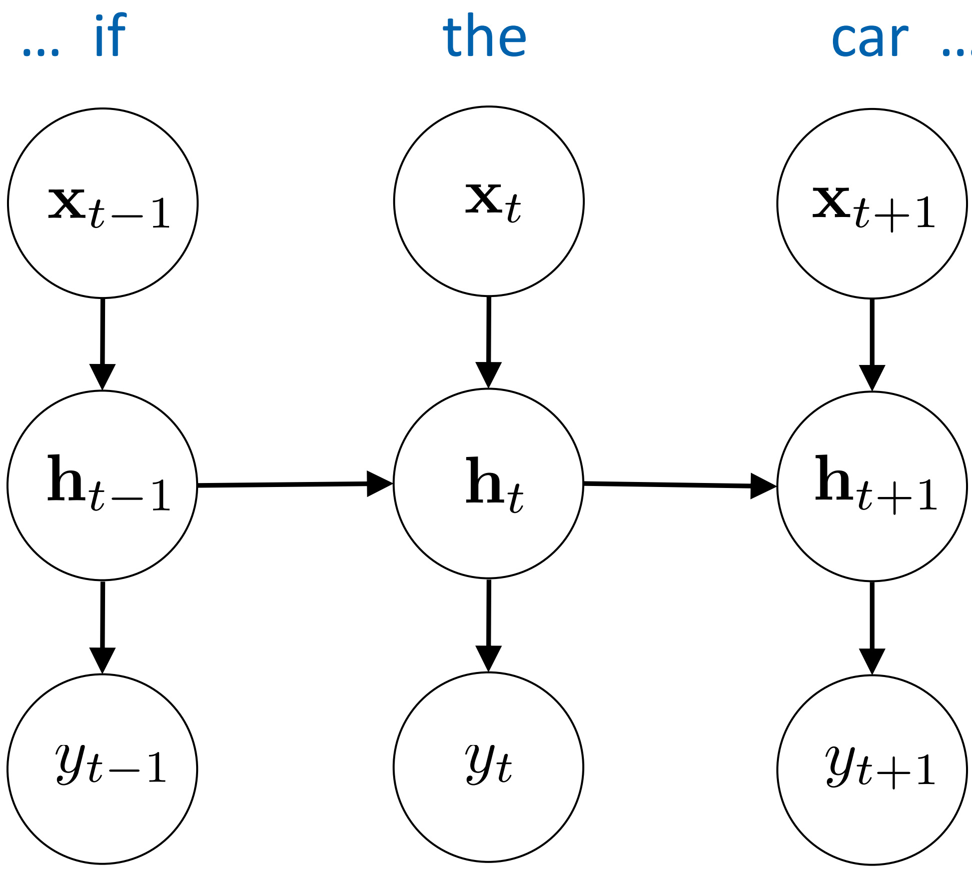
$$\mathbf{x} = [\underbrace{0.4 \;\ldots\; 0.9}_{\text{vector for } \textit{yo}} \; \underbrace{0.2 \;\ldots\; 0.7}_{\text{vector for } \textit{last}} \; \underbrace{0.3 \;\ldots\; 0.6}_{\text{vector for } \textit{name}}]^\top$$

- if *name* is to the right of *yo*, then *yo* is probably a form of *your*
- but our **x** above uses separate dimensions for each position!
  - i.e., *name* is two words to the right
  - what if *name* is one word to the right?

Feed-Forward Networks for POS Tagging

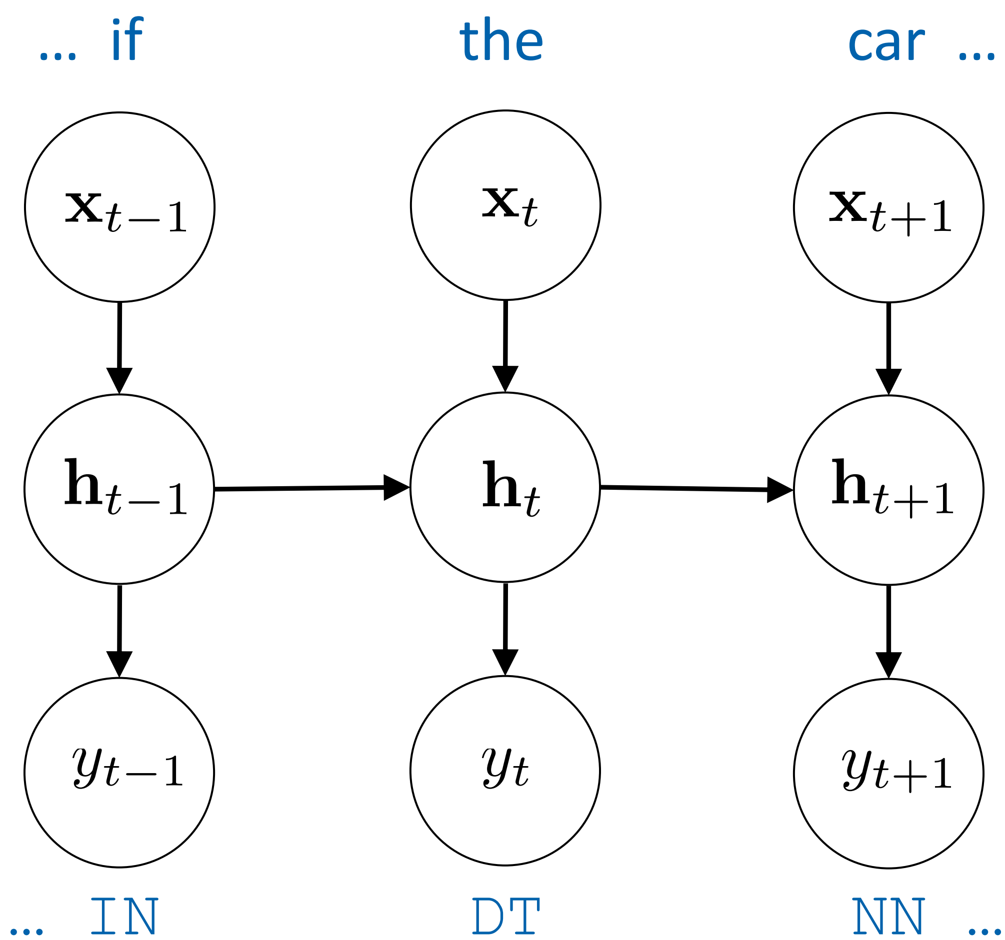
- feed-forward networks are OK for tagging
- they tend to work best with very small contexts (e.g., 1 word to left & right)
- with larger windows, probably not enough data to learn a good model
- also, distant words not very informative for POS tagging
- can also use convolutional networks defined on a window centered on the target word

RNNs for Part-of-Speech Tagging



- input: a word sequence

RNNs for Part-of-Speech Tagging



- target output at each position: POS tag for corresponding word

RNN Taggers

- RNN POS taggers are simple and effective
- most common is to use some sort of bidirectional RNN, like a BiLSTM or BiGRU

RNN Taggers

- note: RNN taggers are not structured predictors
- yes, a structure is being predicted, but predictions for neighboring words are independent!
- BiRNN taggers do compute input representations that depend on the sentence context
- but they do not make any predictions jointly; each prediction is independent of all others

Sequence Labeling

- roughly: for each item in an input sequence, predict a label
- many sequence labeling tasks in NLP and other areas
  - computational biology, speech processing, video processing, etc.
- related class of tasks: segmentation, possibly with labeling of segments

Formulating segmentation tasks as sequence labeling via B-I-O labeling:

Named Entity Recognition

O	O	O	B-PERSON	I-PERSON	O	O	O
Some	questioned	if	Tim	Cook	’s	first	product

O	O	O	O	O	O	B-ORGANIZATION	O
would	be	a	breakaway	hit	for	Apple	.

B = “begin”
I = “inside”
O = “outside”

- there are many downloadable part-of-speech taggers and named entity recognizers:
  - Stanford POS tagger, NER labeler
  - TurboTagger, TurboEntityRecognizer
  - Illinois Entity Tagger
  - CMU Twitter POS tagger
  - Alan Ritter's Twitter POS/NER labeler

Stanford CoreNLP

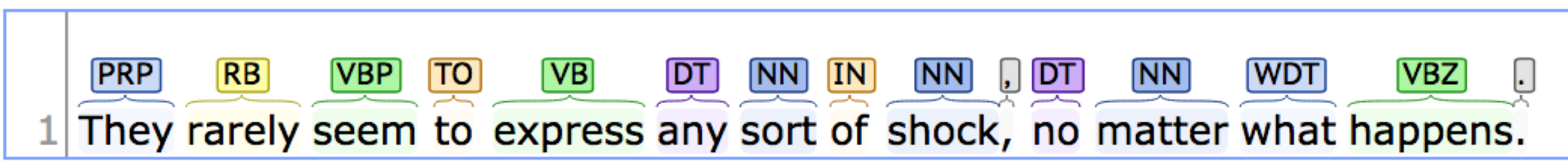
Output format: Visualise

Please enter your text here:

They rarely seem to express any sort of shock, no matter what happens.

Submit Clear

Part-of-Speech:



Stanford Named Entity Tagger

Classifier: english.all.3class.distsim.crf.ser.gz

Output Format: highlighted

Preserve Spacing: no

Please enter your text here:

Some questioned if Tim Cook's first product would be a breakaway hit for Apple.

Submit Clear

Some questioned if Tim Cook's first product would be a breakaway hit for Apple.

Potential tags:

ORGANIZATION

LOCATION

PERSON