Representations for Language: From Word Embeddings to Sentence Meanings

Stanford

Christopher Manning

Stanford University

@chrmanning ❀ @stanfordnlp

Simons Institute 2017

What's special about human language?

Most important distinctive human characteristic

The only hope for "explainable" intelligence

Communication was central to human development and dominance

Language forms come with meanings

A social system

What's special about human language?

Constructed to convey speaker/writer's meaning

Not just an environmental signal; a deliberate communication

Using an encoding which little kids learn (amazingly!) quickly

A discrete/symbolic/categorical signaling system

"rocket" = 🚀; "violin" = 🎻

Very minor exceptions for expressive signaling – "I loooove it"

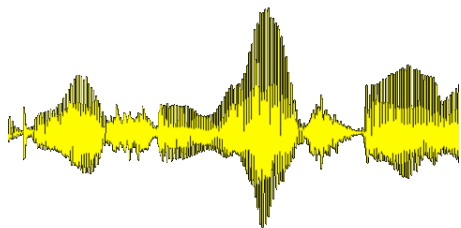Presumably because of greater signaling reliability

Symbols are not just an invention of logic / classical AI!
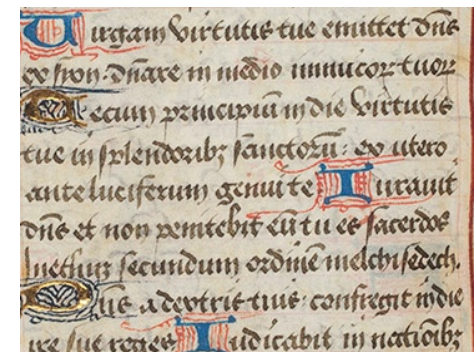
What's special about human language?

Language **symbols** are encoded as a **continuous** communication signal in several ways:

- Sound
- Gesture
- Writing (Images/Trajectories)

**Symbol is invariant** across different encodings!


CC BY 2.0 David Fulmer 2008

National Library of NZ, no known restrictions

What's special about human language?

- Traditionally, people have extended the symbolic system of language into the brain: "The language of thought"

- But a brain encoding appears to be a **continuous pattern of activation**, just like the signal used to transmit language

- Deep Learning is exploring a continuous encoding of thought

- **CogSci question:** Whether to assume symbolic representations in the brain or to directly model using continuous substrate
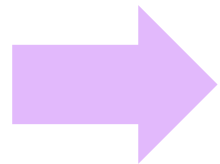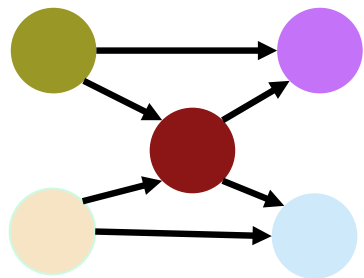
lab
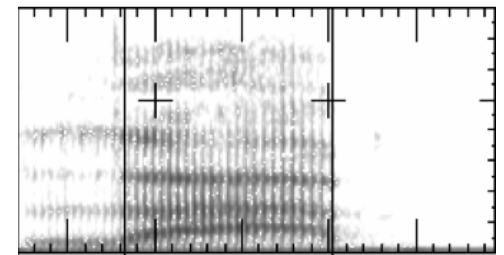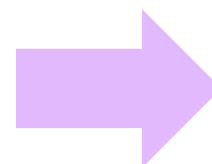
Talk outline

2. From symbolic to distributed word representations

The vast majority of (rule-based and statistical) natural language processing and information retrieval (NLP/IR) work regarded words as atomic symbols: hotel, conference

In machine learning vector space terms, this is a vector with one 1 and a lot of zeroes

$$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]$$

Deep learning people call this a "one-hot" representation

It is a **localist** representation

From symbolic to distributed word representations

Its problem, e.g., for web search:

- If user searches for [Dell notebook battery size], we would like to match documents with "Dell laptop battery capacity"

But

$$\text{size} \qquad [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]^T$$
$$\text{capacity} \quad [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0] = 0$$

Our query and document vectors are **orthogonal**

There is no natural notion of similarity in a set of one-hot vectors

Capturing similarity

There are many things you can do to capture similarity:

    Query expansion with synonym dictionaries

    Separately learning word similarities from large corpora

But a word representation that encodes similarity wins:

    Less parameters to learn (per word, not per pair)

    More sharing of statistics

    More opportunities for multi-task learning

A solution via distributional similarity-based representations

You can get a lot of value by representing a word by means of its neighbors

"You shall know a word by the company it keeps"

(J. R. Firth 1957: 11)

One of the most successful ideas of modern NLP

government debt problems turning into **banking** crises as has happened in

saying that Europe needs unified **banking** regulation to replace the hodgepodge

↖ These words will represent *banking* ↗

Basic idea of learning neural network word embeddings (**Predict!**)

We define a model that predicts between a center word w_t and context words in terms of word vectors, e.g.,
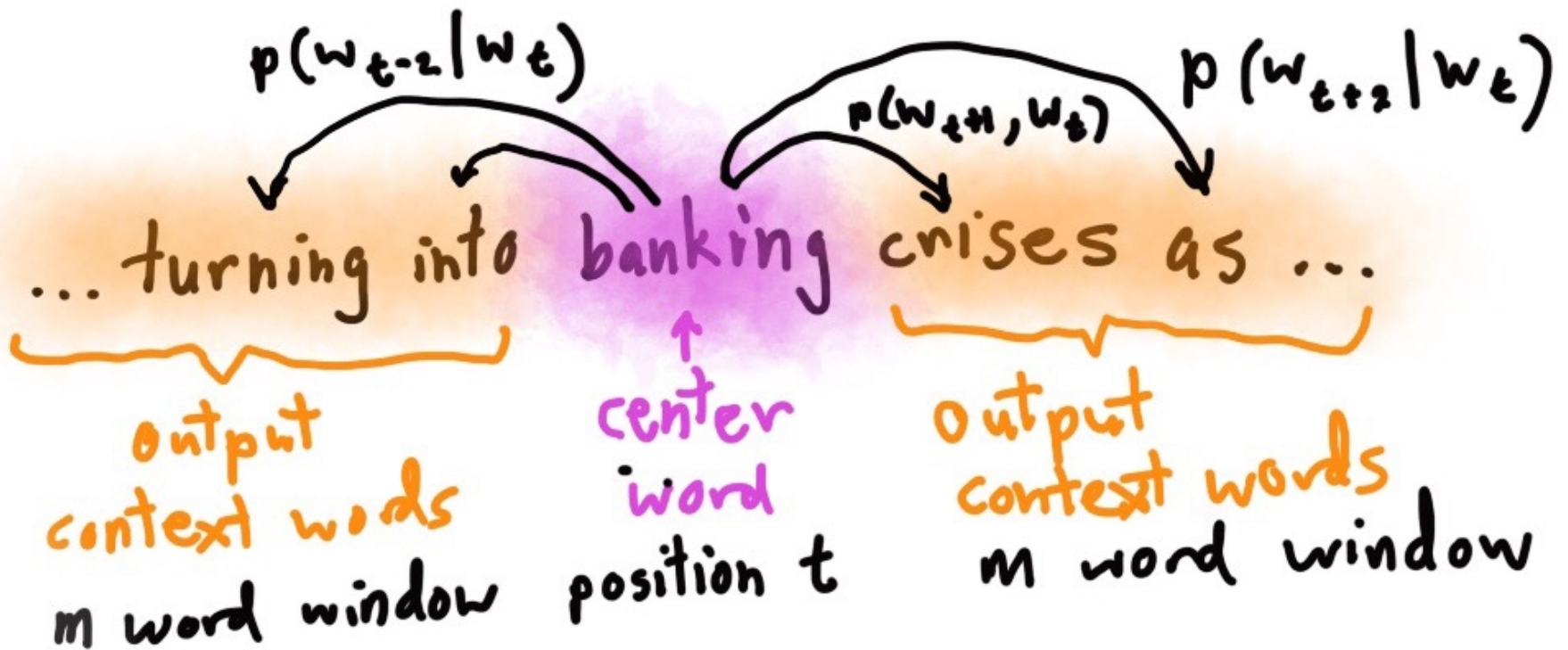
$$p(\text{context}|w_t) = \ldots$$

which has a loss function, e.g.,

$$J = 1 - p(w_{-t}|w_t)$$

We look at **many** positions t in a big language corpus

We keep adjusting the vector representations of words to minimize this loss

Word2vec skip-gram prediction

Details of Word2Vec

For $p(w_{t+j}|w_t)$ we choose:

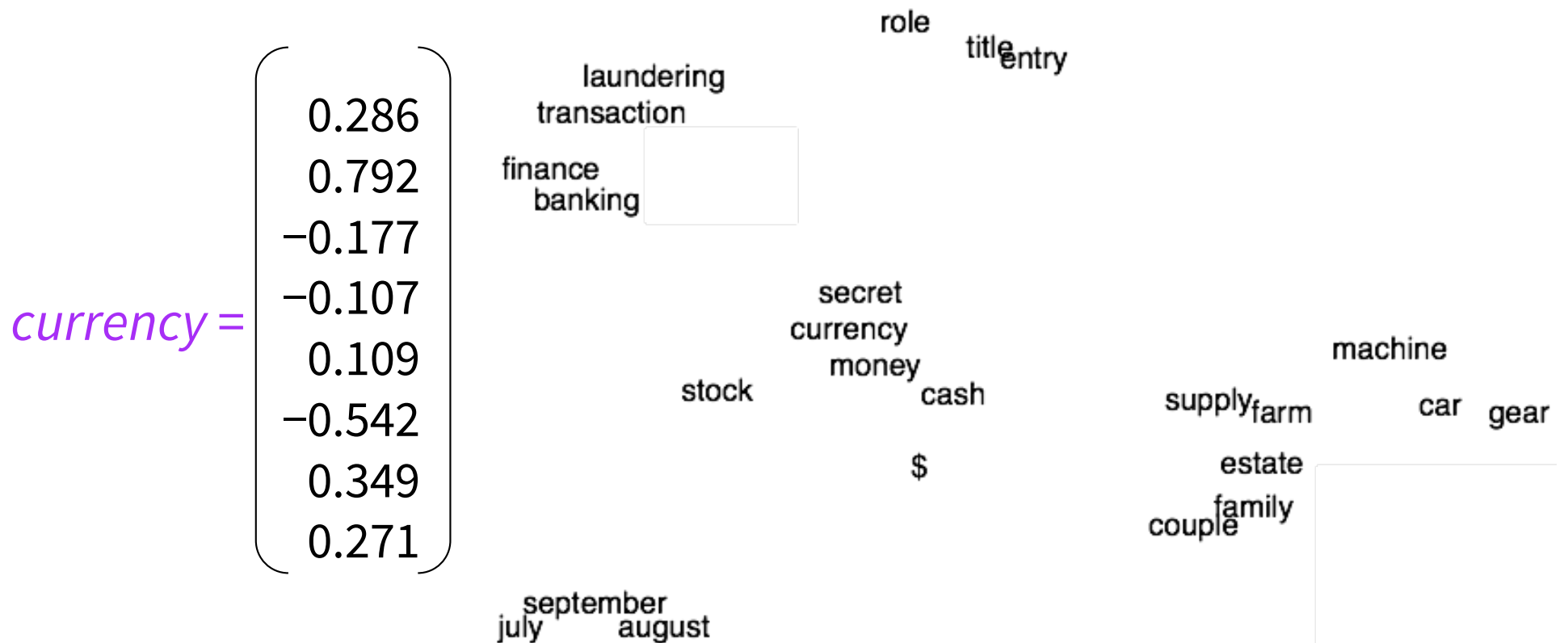$$p(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{w=1}^{V} \exp(u_w^T v_c)}$$

where o is the outside (or output) word index, c is the center word index, v_c and u_o are the "center" and "outside" vectors for word indices c and o

Softmax using word c to obtain probability of word o

Co-occurring words are driven to have similar vectors

Word meaning as a vector

The result is a dense vector for each word type, chosen so that it is good at predicting other words appearing in its context
… those other words also being represented by vectors

$$currency = \begin{bmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{bmatrix}$$

role

title entry

laundering
transaction

finance
banking

secret
currency
money

machine

stock

cash

supply farm

car gear

$

estate

couple family

september
july august

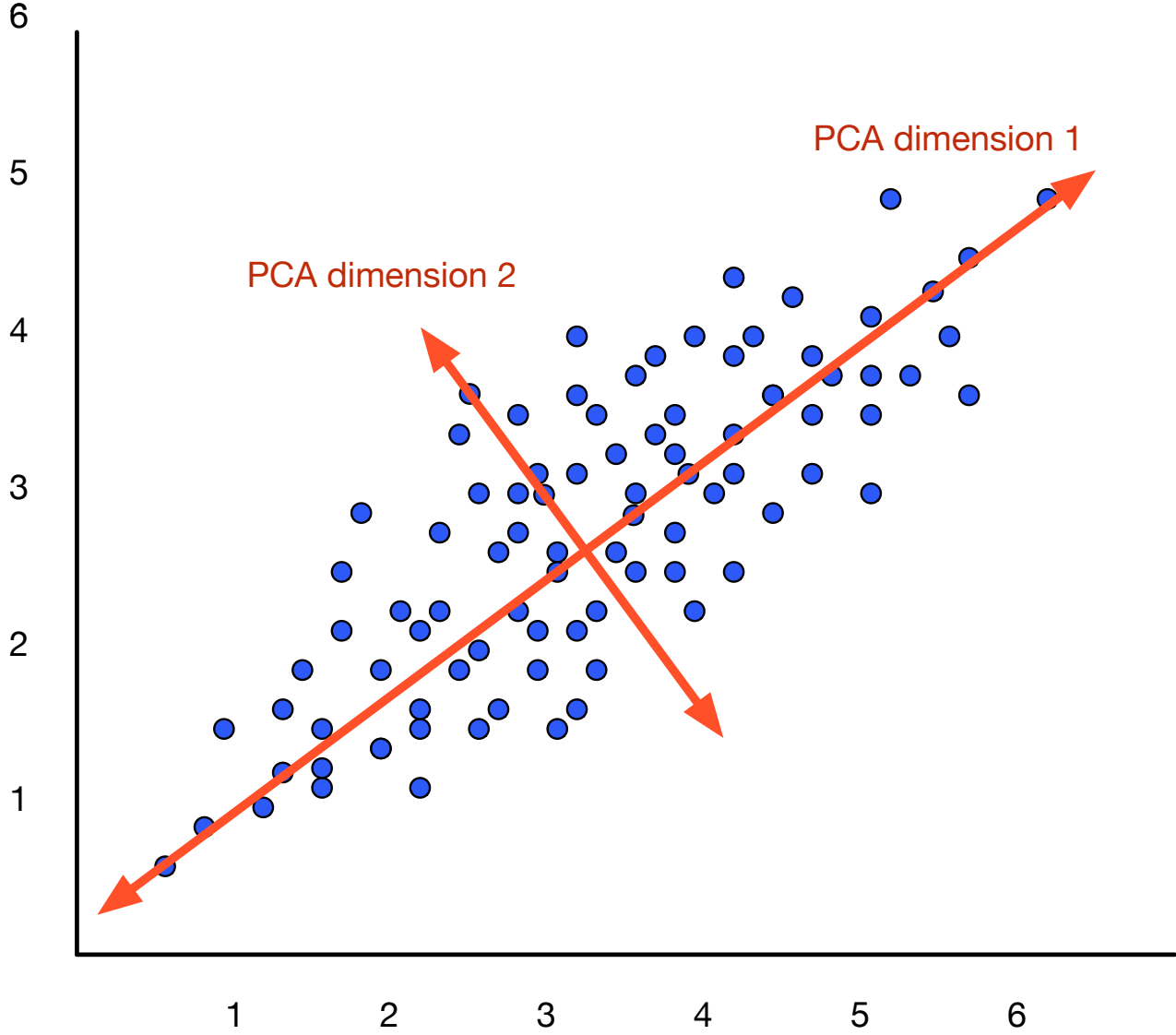# Latent Semantic Analysis (LSA) vs. "neural" models

Comparisons to older work: LSA **Count! models**

- Factorize a (maybe weighted, often log-scaled) term-document (Deerwester et al. 1990) or word-context matrix (Schütze 1992)
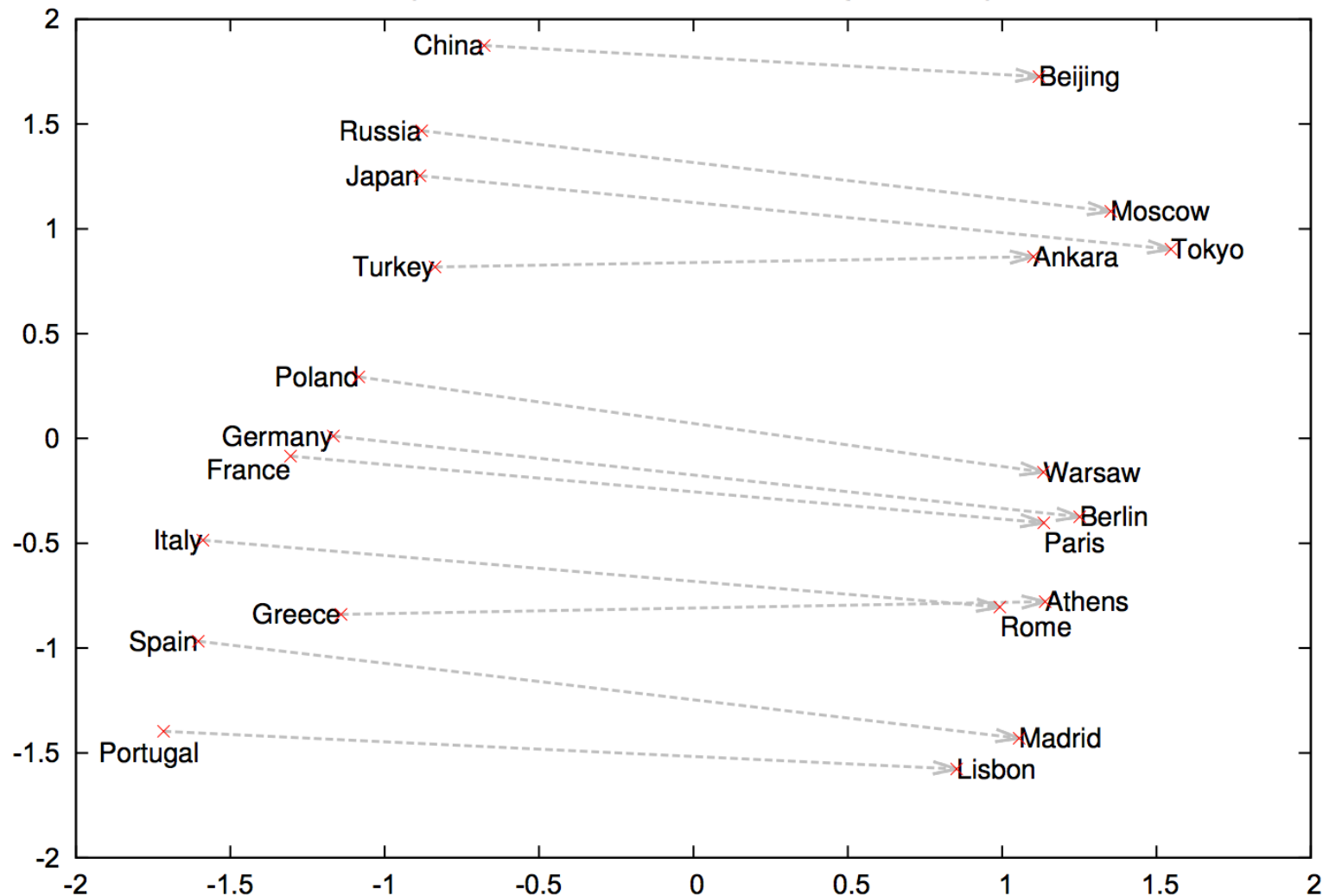
-

SVD: Intuition of Dimensionality reduction

word2vec encodes semantic components as linear vector differences

COALS model (count-modified LSA)
[Rohde, Gonnerman & Plaut, ms., 2005]
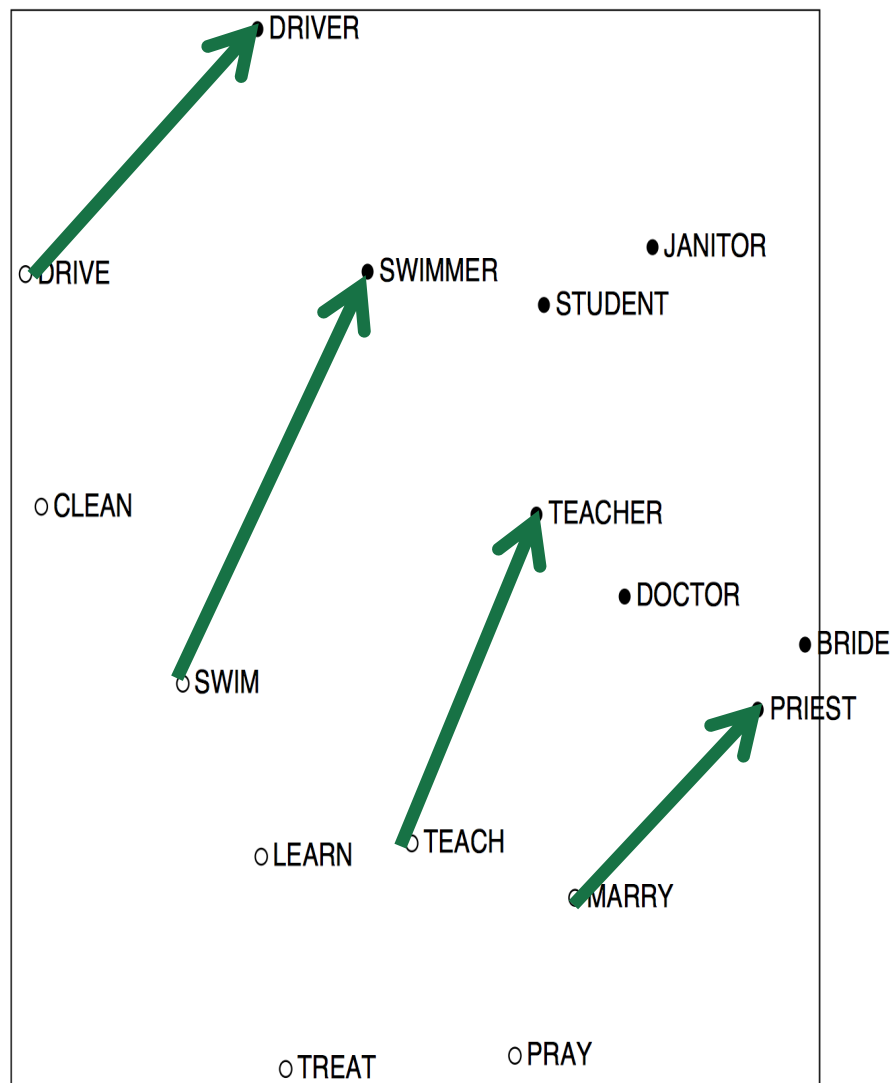
Encoding meaning in vector differences

[Pennington, Socher, and Manning, EMNLP 2014]

Crucial insight: Ratios of co-occurrence probabilities can encode meaning components

	x = solid	x = gas	x = water	x = random
$P(x\|\text{ice})$	large	small	large	small
$P(x\|\text{steam})$	small	large	large	small
$\dfrac{P(x\|\text{ice})}{P(x\|\text{steam})}$	large	small	~1	~1

Encoding meaning in vector differences

[Pennington, Socher, and Manning, EMNLP 2014]

Crucial insight: Ratios of co-occurrence probabilities can encode meaning components

	x = solid	x = gas	x = water	x = fashion
$P(x\|\text{ice})$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(x\|\text{steam})$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$\dfrac{P(x\|\text{ice})}{P(x\|\text{steam})}$	8.9	8.5×10^{-2}	1.36	0.96

Encoding meaning in vector differences

Q: How can we capture ratios of co-occurrence probabilities as meaning components in a word vector space?

A: Log-bilinear model: $\qquad w_i \cdot w_j = \log P(i|j)$

   with vector differences $\qquad w_x \cdot (w_a - w_b) = \log \dfrac{P(x|a)}{P(x|b)}$

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2 \qquad f \sim$$
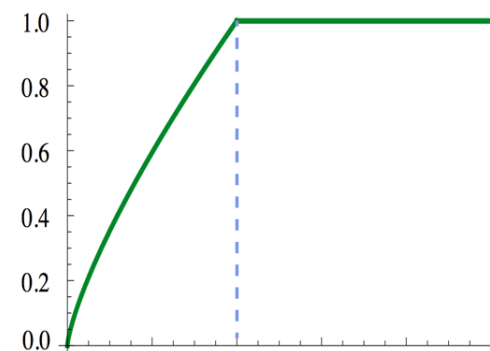
Glove Word similarities
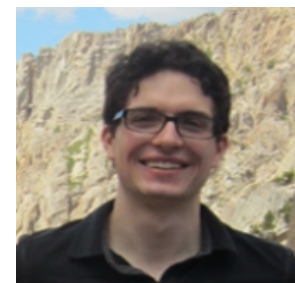
Nearest words to frog:

1. frogs
2. toad
3. litoria
4. leptodactylidae
5. rana
6. lizard
7. eleutherodactylus

http://nlp.stanford.edu/projects/glove/

litoria

leptodactylidae
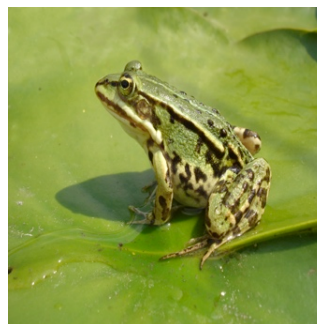
rana

eleutherodactylus

Glove Visualizations: Gender pairs

http://nlp.stanford.edu/projects/glove/
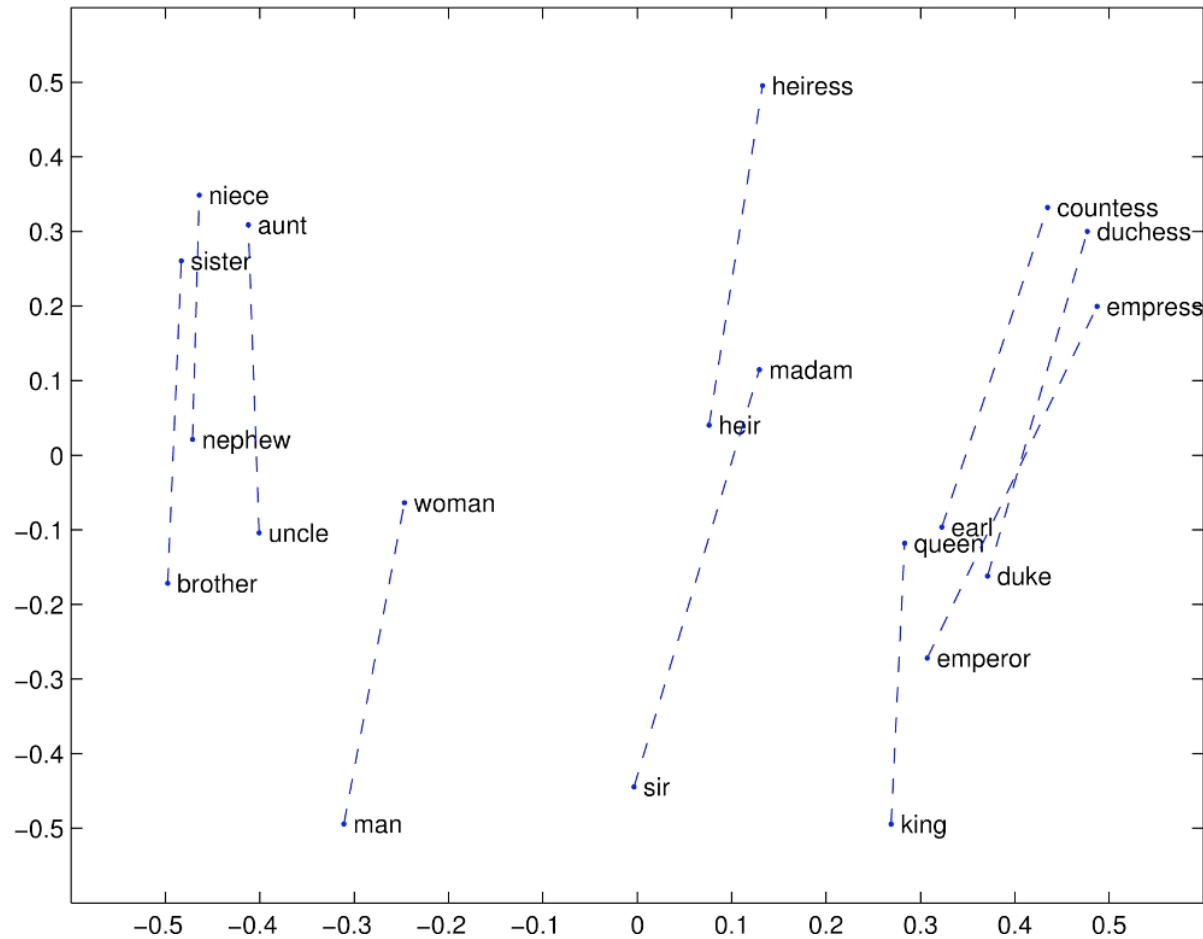
Glove Visualizations: Company - CEO
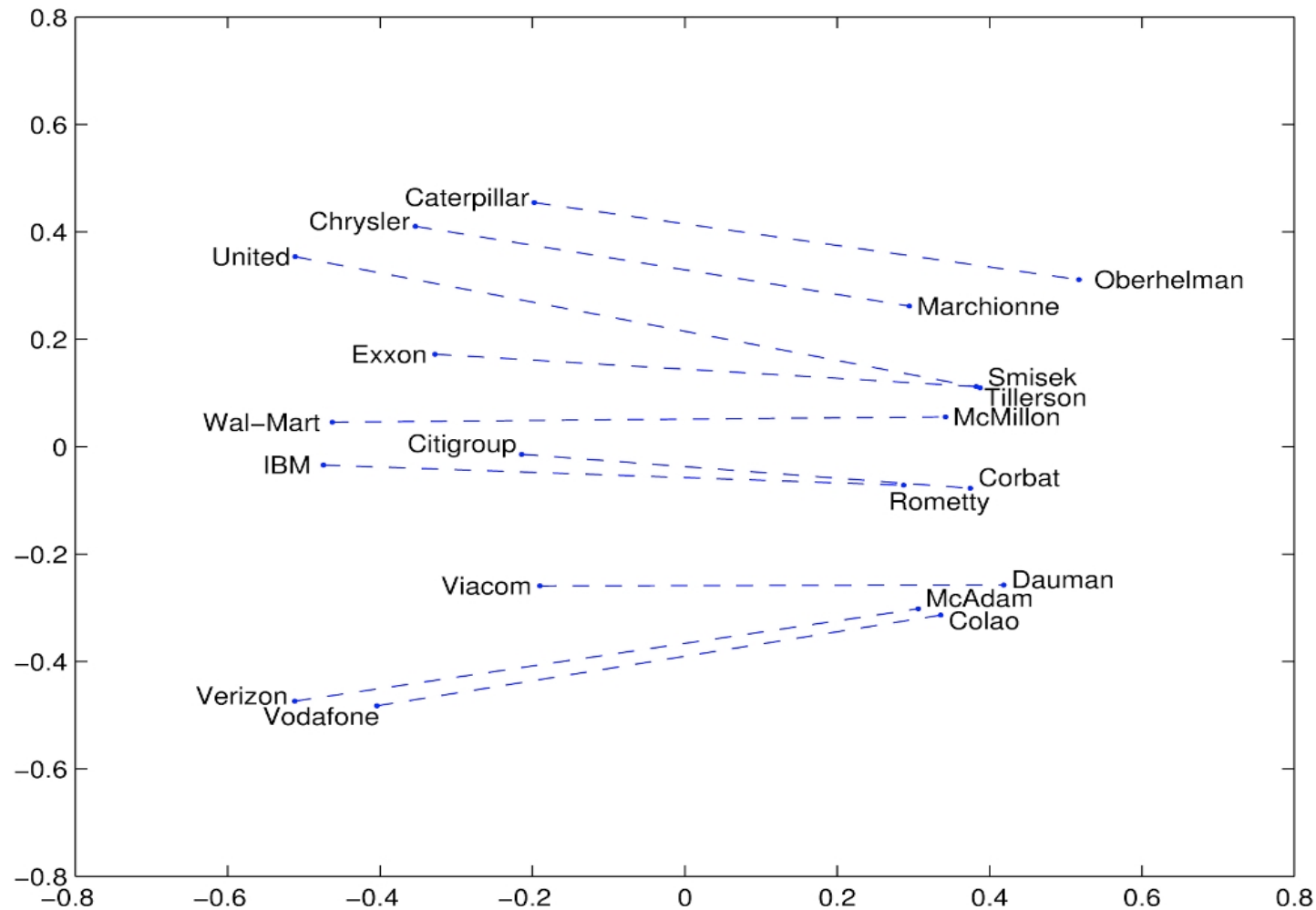
Named Entity Recognition Performance
(finding person, organization names in text)

Model on CoNLL	CoNLL '03 dev	CoNLL '03 test
Categorical CRF	91.0	85.4
SVD (log tf)	90.5	84.8
HPCA	92.6	**88.7**
C&W	92.2	87.4
CBOW	93.1	88.2
GloVe	**93.2**	88.3

F1 score of CRF trained on CoNLL 2003 English with 50 dim word vectors

Named Entity Recognition Performance
(finding person, organization names in text)

Model on CoNLL	CoNLL '03 dev	CoNLL '03 test	ACE 2	MUC 7
Categorical CRF	91.0	85.4	77.4	73.4
SVD (log tf)	90.5	84.8	73.6	71.5
HPCA	92.6	**88.7**	81.7	80.7
C&W	92.2	87.4	81.7	80.2
CBOW	93.1	88.2	82.2	81.1
GloVe	**93.2**	88.3	**82.9**	**82.2**

F1 score of CRF trained on CoNLL 2003 English with 50 dim word vectors

Word embeddings: Conclusion

Glove shows the connection between **Count!** work and **Predict!** work – an appropriate scaling and objective gives **Count!** models the properties and performance of **Predict!** models

Lots of other important recent work in this area:

[Levy & Goldberg, 2014]

[Arora, Li, Liang, Ma & Risteski, 2016]

[Hashimoto, Alvarez-Melis & Jaakkola, 2016]

3. The BiLSTM Hegemony

**To a first approximation,
the de facto consensus in NLP in 2017 is
that no matter what the task,
you throw a BiLSTM at it, with
attention if you need information flow**

An RNN encoder-decoder network

Encoder   Decoder

Je   suis   étudiant <EOS>

0.2	0.2	0.4	0.2	0.2	0.2	0.2	-0.1
-0.3	0.4	-0.2	0.6	0.6	0.6	0.6	0.3
-0.1	0.1	-0.3	-0.1	-0.1	-0.1	-0.1	-0.1
-0.4	-0.5	-0.4	-0.7	-0.7	-0.7	-0.7	-0.7
0.2	-0.2	-0.2	0.1	0.1	0.1	0.1	0.1

I   am   a   student   <EOS>   Je   suis   étudiant
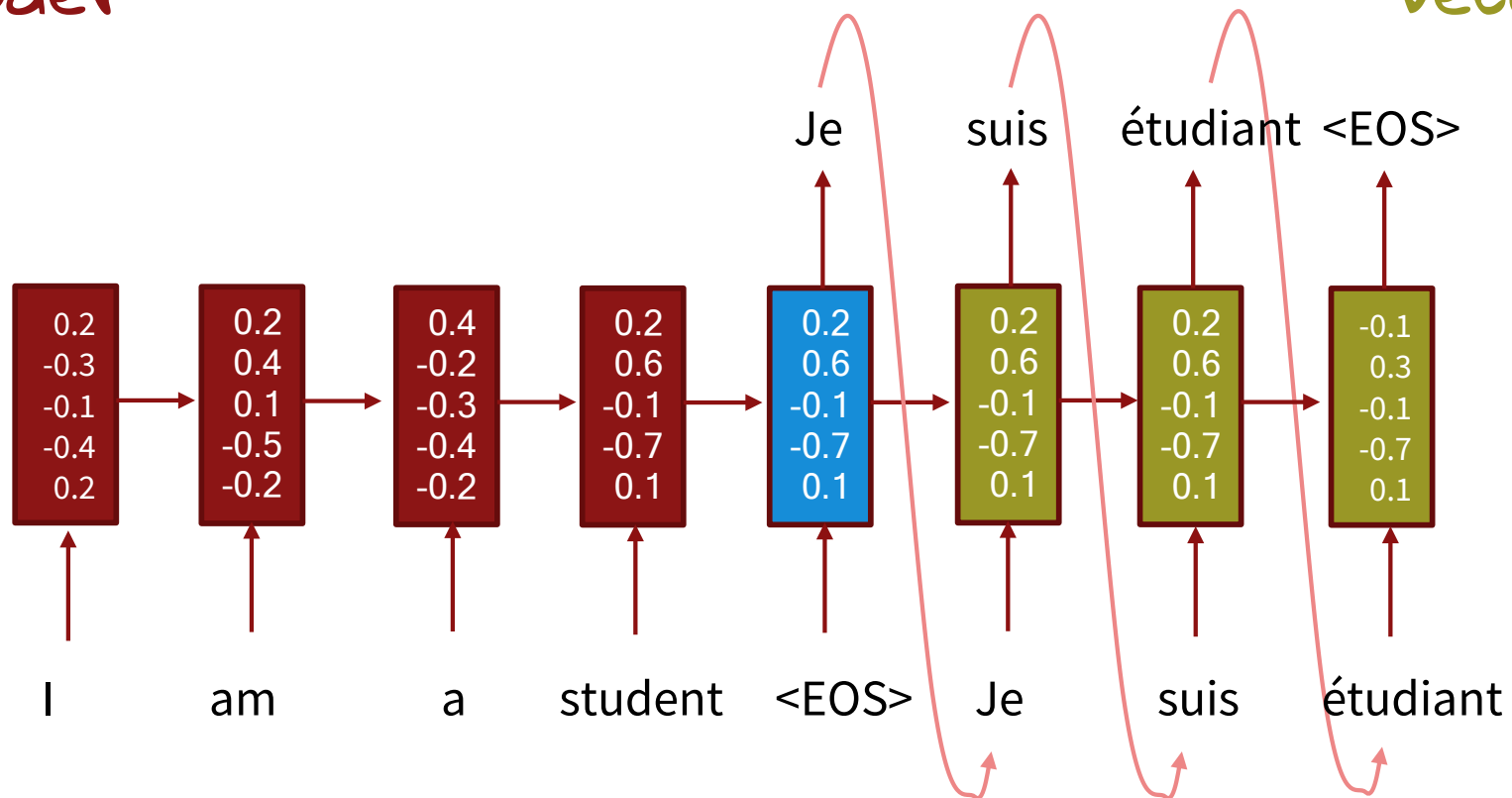
$$h_t = \tanh(W[x_t] + Uh_{t-1} + b)$$

Gated Recurrent Units ≈ "LSTMs"

Equations of the two most widely used gated recurrent units

Gated Recurrent Unit

[Cho et al., EMNLP2014; Chung, Gulcehre, Cho, Bengio, DLUFL2014]

$$h_t = u_t \odot \tilde{h}_t + (1 - u_t) \odot h_{t-1}$$

$$\tilde{h} = \tanh(W\,[x_t] + U(r_t \odot h_{t-1}) + b)$$

$$u_t = \sigma(W_u\,[x_t] + U_u h_{t-1} + b_u)$$

$$r_t = \sigma(W_r\,[x_t] + U_r h_{t-1} + b_r)$$

Basic update to memory cell (GRU h = LSTM c) is via a standard neural net layer

Long Short-Term Memory

[Hochreiter & Schmidhuber, NC1999; Gers, Thesis2001]

$$h_t = o_t \odot \tanh(c_t)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$\tilde{c}_t = \tanh(W_c\,[x_t] + U_c h_{t-1} + b_c)$$

$$o_t = \sigma(W_o\,[x_t] + U_o h_{t-1} + b_o)$$

$$i_t = \sigma(W_i\,[x_t] + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f\,[x_t] + U_f h_{t-1} + b_f)$$

Gated Recurrent Units ≈ "LSTMs"

Equations of the two most widely used gated recurrent units

Gated Recurrent Unit
[Cho et al., EMNLP2014;
 Chung, Gulcehre, Cho, Bengio, DLUFL2014]

$$h_t = u_t \odot \tilde{h}_t + (1 - u_t) \odot h_{t-1}$$

$$\tilde{h} = \tanh(W[x_t] + U(r_t \odot h_{t-1}) + b)$$

$$u_t = \sigma(W_u[x_t] + U_u h_{t-1} + b_u)$$

$$r_t = \sigma(W_r[x_t] + U_r h_{t-1} + b_r)$$

Long Short-Term Memory
[Hochreiter & Schmidhuber, NC1999;
 Gers, Thesis2001]

$$h_t = o_t \odot \tanh(c_t)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$\tilde{c}_t = \tanh(W_c[x_t] + U_c h_{t-1} + b_c)$$

$$o_t = \sigma(W_o[x_t] + U_o h_{t-1} + b_o)$$

$$i_t = \sigma(W_i[x_t] + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f[x_t] + U_f h_{t-1} + b_f)$$

Bernoulli variable "gates" control how much history is kept & input is attended to

Gated Recurrent Units ≈ "LSTMs"

Equations of the two most widely used gated recurrent units

Gated Recurrent Unit
**[Cho et al., EMNLP2014;
Chung, Gulcehre, Cho, Bengio, DLUFL2014]**

$$h_t = \boxed{u_t \odot \tilde{h}_t + (1 - u_t) \odot h_{t-1}}$$

$$\tilde{h} = \tanh(W\,[x_t] + U(r_t \odot h_{t-1}) + b)$$

$$u_t = \sigma(W_u\,[x_t] + U_u h_{t-1} + b_u)$$

$$r_t = \sigma(W_r\,[x_t] + U_r h_{t-1} + b_r)$$

Summing previous & new candidate hidden states gives direct gradient flow & more effective memory

Long Short-Term Memory
**[Hochreiter & Schmidhuber, NC1999;
Gers, Thesis2001]**

$$h_t = o_t \odot \tanh(c_t)$$

$$c_t = \boxed{f_t \odot c_{t-1} + i_t \odot \tilde{c}_t}$$

$$\tilde{c}_t = \tanh(W_c\,[x_t] + U_c h_{t-1} + b_c)$$

$$o_t = \sigma(W_o\,[x_t] + U_o h_{t-1} + b_o)$$

$$i_t = \sigma(W_i\,[x_t] + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f\,[x_t] + U_f h_{t-1} + b_f)$$

Gated Recurrent Units ≈ "LSTMs"

Equations of the two most widely used gated recurrent units

Gated Recurrent Unit
[Cho et al., EMNLP2014;
Chung, Gulcehre, Cho, Bengio, DLUFL2014]

$$h_t = u_t \odot \tilde{h}_t + (1 - u_t) \odot h_{t-1}$$

$$\tilde{h} = \tanh(W[x_t] + U(r_t \odot h_{t-1}) + b)$$

$$u_t = \sigma(W_u[x_t] + U_u h_{t-1} + b_u)$$

$$r_t = \sigma(W_r[x_t] + U_r h_{t-1} + b_r)$$

Long Short-Term Memory
[Hochreiter & Schmidhuber, NC1999;
Gers, Thesis2001]

$$h_t = o_t \odot \tanh(c_t)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$\tilde{c}_t = \tanh(W_c[x_t] + U_c h_{t-1} + b_c)$$
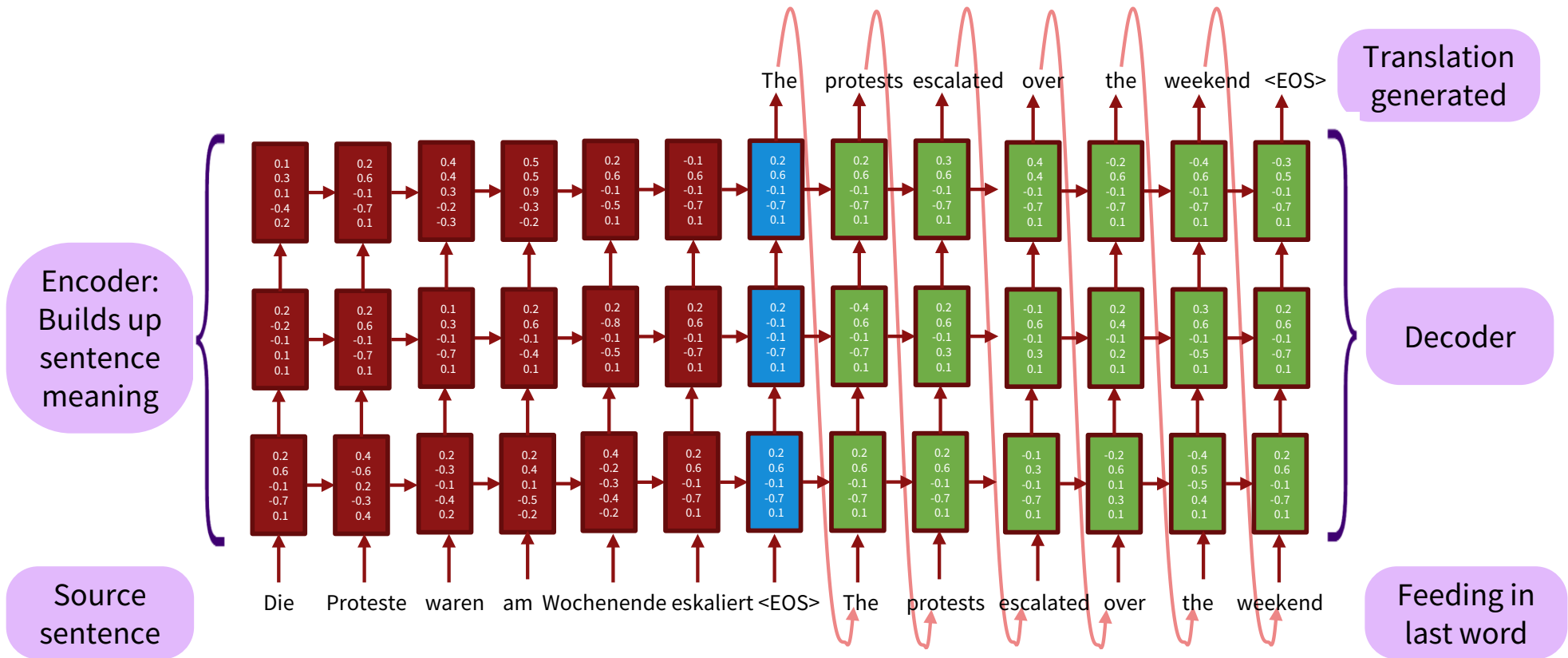
$$o_t = \sigma(W_o[x_t] + U_o h_{t-1} + b_o)$$

$$i_t = \sigma(W_i[x_t] + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f[x_t] + U_f h_{t-1} + b_f)$$

Note that recurrent state mixes control and memory. Good? (Freedom to represent.) Or bad? (Mush.)

An LSTM encoder-decoder network
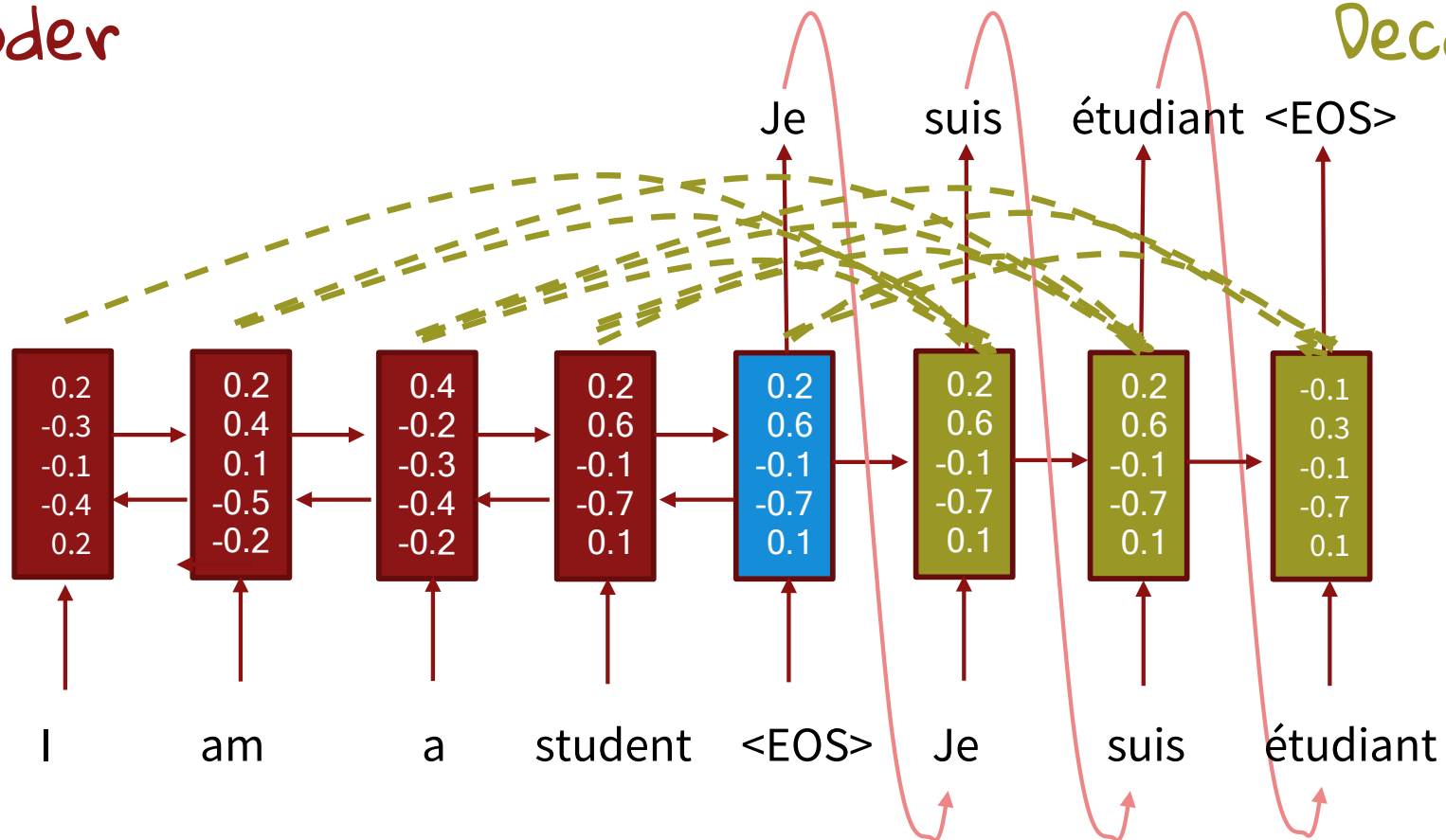## [Sutskever et al. 2014]

Translation generated

Encoder: Builds up sentence meaning

Decoder

Source sentence

Feeding in last word
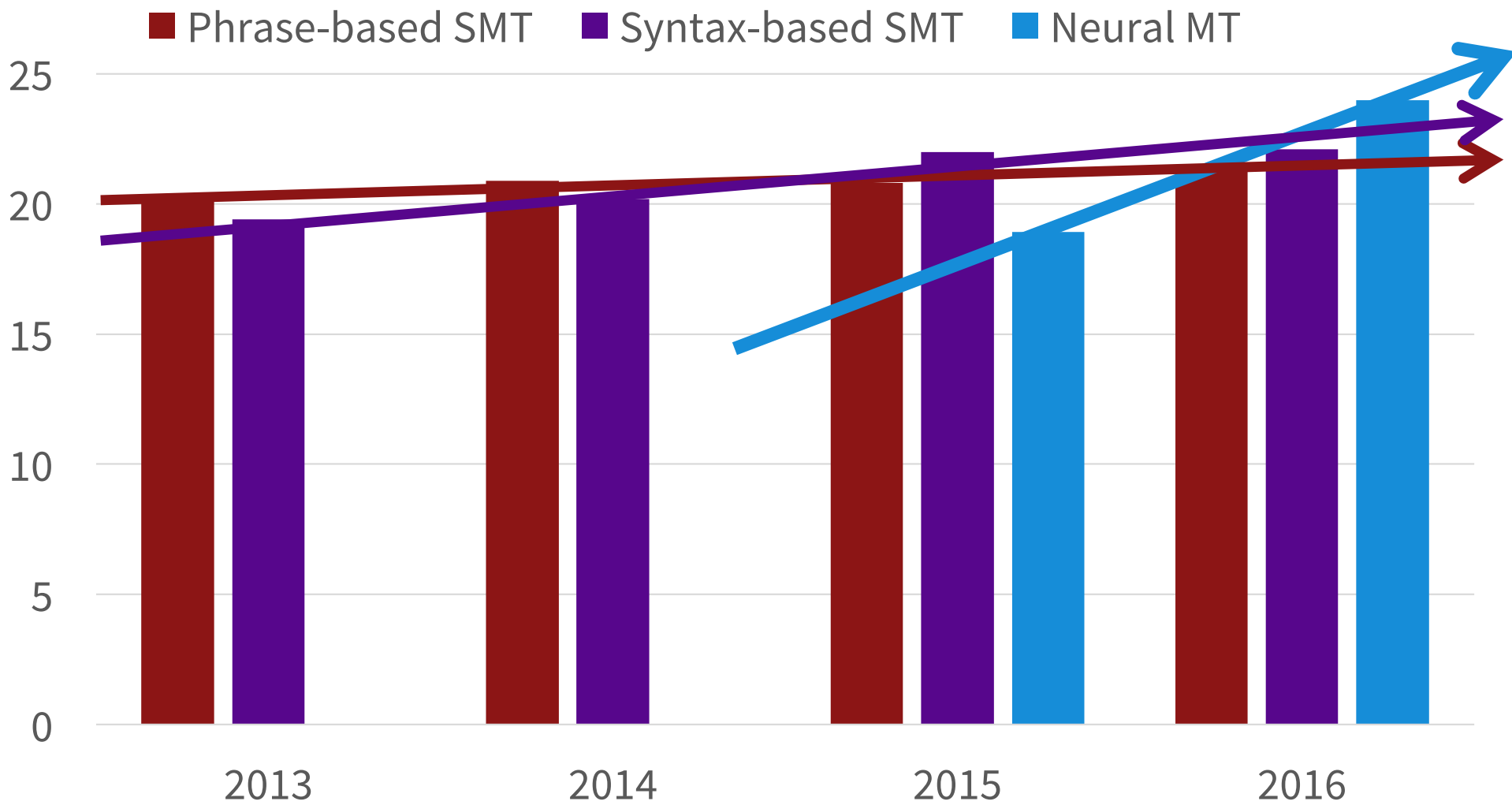
Bottleneck

A BiLSTM encoder and LSTM-with-attention decoder

Encoder

Decoder

Progress in Machine Translation

[Edinburgh En-De WMT newstest2013 Cased BLEU; NMT 2015 from U. Montréal]

■ Phrase-based SMT   ■ Syntax-based SMT   ■ Neural MT

From [Sennrich 2016, http://www.meta-net.eu/events/meta-forum-2016/slides/09_sennrich.pdf]

Four big wins of Neural MT

1. **End-to-end training**
   All parameters are simultaneously optimized to minimize a loss function on the network's output

2. **Distributed representations share strength**
   Better exploitation of word and phrase similarities

3. **Better exploitation of context**
   NMT can use a much bigger context – both source and partial target text – to translate more accurately

4. **More fluent text generation**
   Deep learning text generation is much higher quality

BiLSTMs(+Attn) not just for neural MT

Part of speech tagging

Named entity recognition
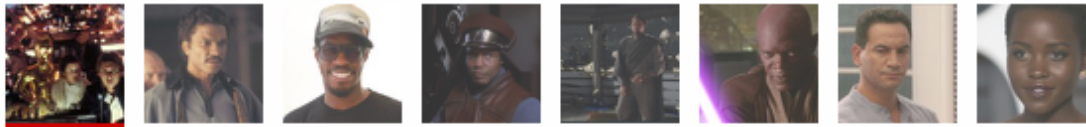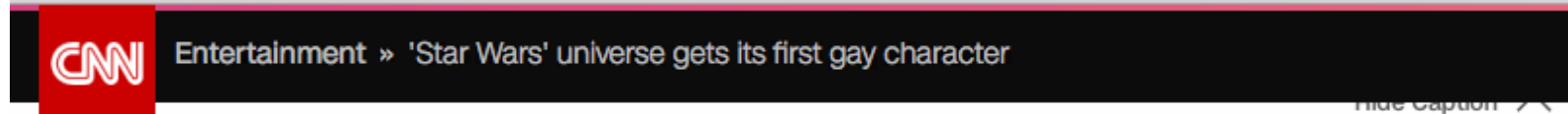
Syntactic parsing (constituency & dependency)

Reading comprehension

Question answering

Text summarization

…

Reading Comprehension on the DeepMind CNN & Daily Mail datasets [Hermann et al, 2015]

CNN | Entertainment » 'Star Wars' universe gets its first gay character

Hide Caption

Story highlights

Official "Star Wars" universe gets its first gay character, a lesbian governor

The character appears in the upcoming novel "Lords of the Sith"

Characters in ▯▯▯▯ movies have gradually become more diverse

(CNN) — If you feel a ripple in the Force today, it may be the news that the official Star Wars universe is getting its first gay character.

According to the sci-fi website Big Shiny Robot, the upcoming novel "Lords of the Sith" will feature a capable but flawed Imperial official named Moff Mors who "also happens to be a lesbian."

The character is the first gay figure in the official Star Wars universe -- the movies, television shows, comics and books approved by Star Wars franchise owner Disney -- according to Shelly Shapiro, editor of "Star Wars" books at Random House imprint Del Rey Books.
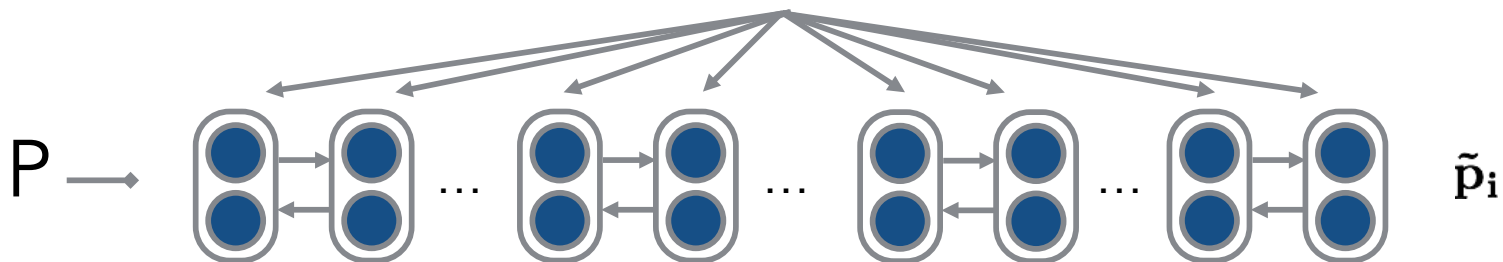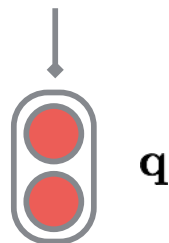
39

End-to-end Neural Network
[Chen, Bolton, & Manning, ACL 2016]

Q characters in " @placeholder " movies
have gradually become more diverse

Bidirectional RNNs

$\mathbf{q}$

P → ... ... $\tilde{\mathbf{p}}_i$

(@entity4) if you feel a ripple in the force today , it may be
the news that the official @entity6 is getting its first gay
character . according to the sci-fi website @entity9 , the
upcoming novel " @entity11 " will feature a capable but
flawed @entity13 official named @entity14 who " also
happens to be a lesbian " the character is the first gay
figure in the official @entity6 the movies , television shows,
comics and books approved by @entity6 franchise owner
@entity22 -- according to @entity24 , editor of " @entity6 "

Attention

A entity6

$$\alpha_i \quad = \quad \boxed{\operatorname{softmax}_i \left(\mathbf{q}^\top \mathbf{W}_s \tilde{\mathbf{p}}_i \right)}$$

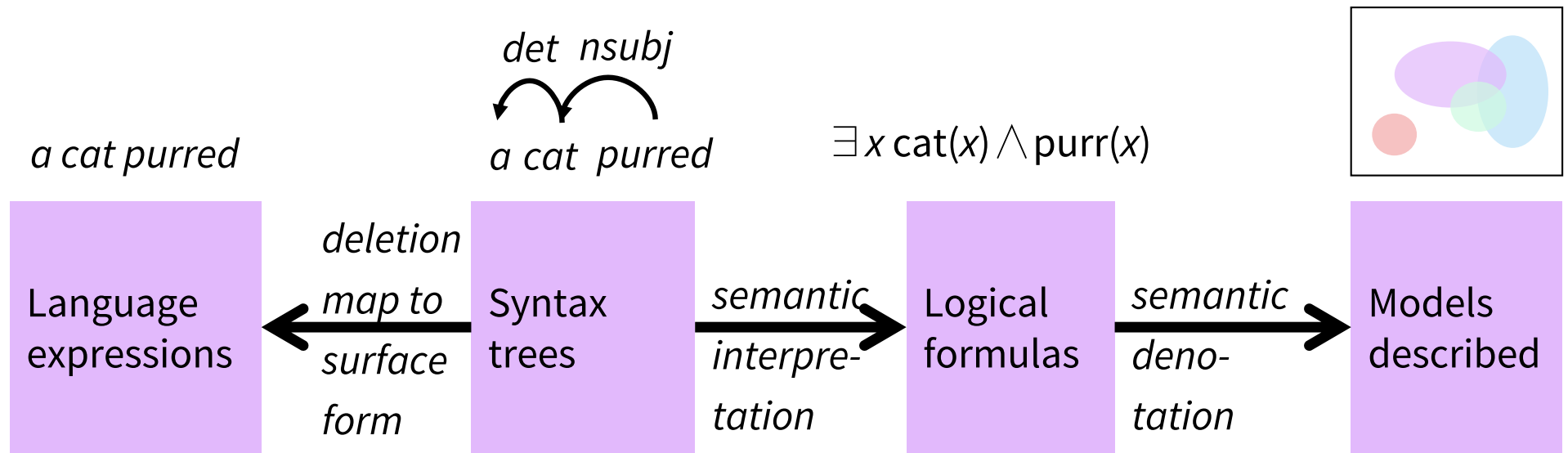$$\mathbf{o} \quad = \quad \sum_i \alpha_i \tilde{\mathbf{p}}_i$$

$$a = \arg\max_{a \in p \cap E} W_a^\top \mathbf{o}$$

40

Lots of complex models; lots of results
Nothing does much better than LSTM+Attn

		CNN		Daily Mail	
		Dev	Test	Dev	Test
(Hermann et al, 2015)	NIPS'15	61.8	63.8	69.0	68.0
(Hill et al, 2016)	ICLR'16	63.4	66.8	N/A	N/A
(Kobayashi et al, 2016)	NAACL'16	71.3	72.9	N/A	N/A
(Kadlec et al, 2016)	ACL'16	68.6	69.5	75.0	73.9
(Dhingra et al, 2016)	2016/6/5	73.0	73.8	76.7	75.7
(Sodorni et al, 2016)	2016/6/7	72.6	73.3	N/A	N/A
(Trischler et al, 2016)	2016/6/7	73.4	74.0	N/A	N/A
(Weissenborn, 2016)	2016/7/12	N/A	73.6	N/A	**77.2**
(Cui et al, 2016)	2016/7/15	73.1	**74.4**	N/A	N/A
Ours: neural net	ACL'16	**73.8**	73.6	**77.6**	76.6
Ours: neural net (ensemble)	ACL'16	**77.2**	**77.6**	**80.2**	**79.2**

The Standard Theory of Natural Language Interpretation

a cat purred

det nsubj

a cat purred

$\exists\, x\, \mathrm{cat}(x) \wedge \mathrm{purr}(x)$

| Language expressions | *deletion map to surface form* ← → | Syntax trees | *semantic interpre-tation* → | Logical formulas | *semantic deno-tation* → | Models described |

Model of:

- most linguistic and philosophical work (till the present)
- most computational linguistic work (till 1990)
- modern "semantic parsing" (Liang, Zettlemoyer, etc.)
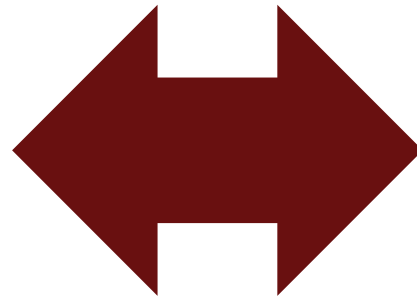
Semantic interpretation of language – Not just word vectors

How can we minimally know when larger language units are similar in meaning?
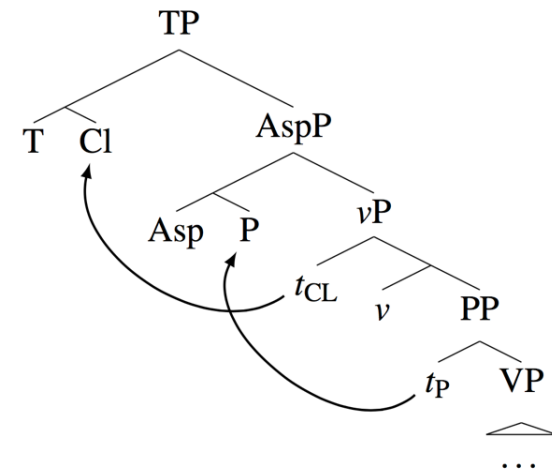
- *The **snowboarder** is leaping over a mogul*

- *A **person on a snowboard** jumps into the air*

People interpret the meaning of larger text units – entities, descriptive terms, facts, arguments, stories – by **semantic composition** of smaller elements

4. Choices for multi-word language representations

$$\begin{bmatrix} word \\ \text{PHON} & /\eth\varepsilon\textrm{\j}/ \\ \text{SYNSEM} & \begin{bmatrix} \text{LOCAL} & \begin{bmatrix} \text{CAT} & \begin{bmatrix} \text{HEAD} & \begin{bmatrix} verb \\ \text{VFORM} & finite \end{bmatrix} \\ \text{VAL} & \begin{bmatrix} \text{SUBJ} & \langle\,\rangle \\ \text{COMPS} & \langle \boxed{1} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \\ \text{ARG-ST} & \left\langle \boxed{3}\text{NP}[\textit{3pl}],\ \boxed{1}\begin{bmatrix} \text{PRED} + \\ \text{SUBJ} & \langle\boxed{3}\rangle \end{bmatrix} \right\rangle \end{bmatrix}$$

Neural bag-of-words models

- Simply average (or just sum) word vectors:

$$\left(\begin{bmatrix} 0.4 \\ 0.3 \end{bmatrix} + \begin{bmatrix} 2.1 \\ 3.3 \end{bmatrix} + \begin{bmatrix} 7.0 \\ 7.0 \end{bmatrix} + \begin{bmatrix} 4.0 \\ 4.5 \end{bmatrix} + \begin{bmatrix} 2.3 \\ 3.6 \end{bmatrix} \right)/5 = \begin{bmatrix} 3.0 \\ 3.7 \end{bmatrix}$$

     the country of   my birth

- Can improve effectiveness by putting output through 1+ fully connected layers (DANs)

- **Surprisingly effective** for many tasks ☹
  - [Iyyer, Manjunatha, Boyd-Graber and Daumé III 2015 – DANs; Wieting, Bansal, Gimpel and Livescu 2016 – Periphrastic]

Recurrent neural networks

- Simple recurrent neural nets do use word order but **cannot** capture phrases without prefix context

- Gated LSTM/GRU units in theory could up to a certain depth, but it seems **unlikely**

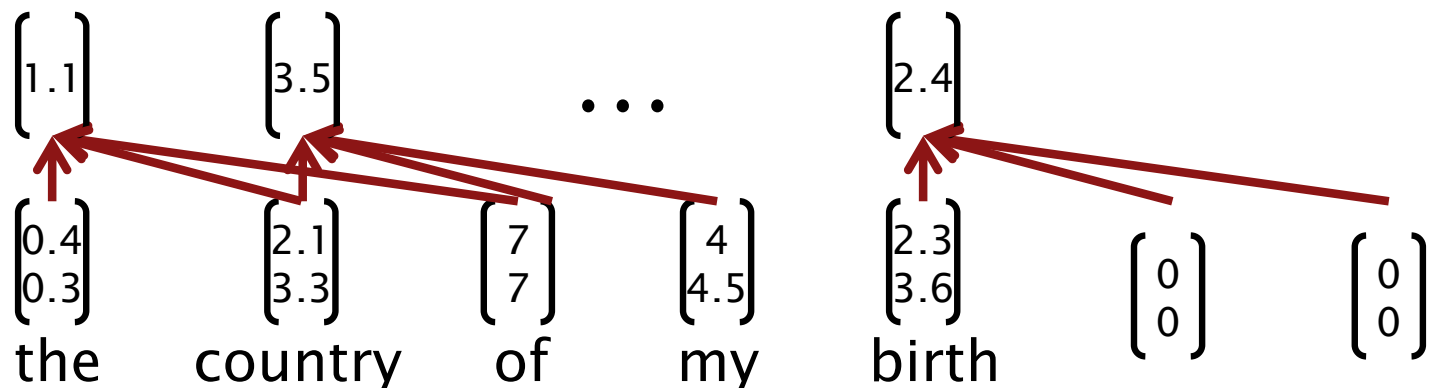- Empirically, representations capture too much of last words in final vector – focus is LM next word prediction

$$\begin{bmatrix} 1 \\ 3.5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 \\ 5 \end{bmatrix} \rightarrow \begin{bmatrix} 5.5 \\ 6.1 \end{bmatrix} \rightarrow \begin{bmatrix} 4.5 \\ 3.8 \end{bmatrix} \rightarrow \begin{bmatrix} 2.5 \\ 3.8 \end{bmatrix}$$

$$\begin{bmatrix} 0.4 \\ 0.3 \end{bmatrix} \quad \begin{bmatrix} 2.1 \\ 3.3 \end{bmatrix} \quad \begin{bmatrix} 7 \\ 7 \end{bmatrix} \quad \begin{bmatrix} 4 \\ 4.5 \end{bmatrix} \quad \begin{bmatrix} 2.3 \\ 3.6 \end{bmatrix}$$

the   country   of   my   birth
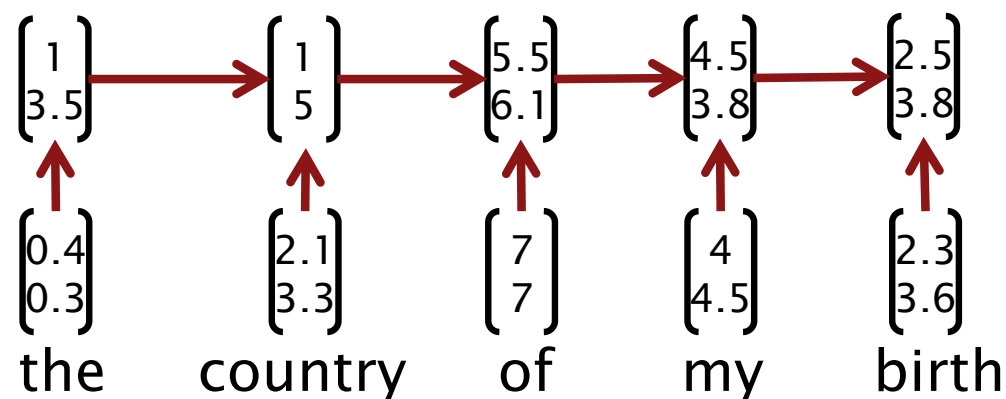
Convolutional Neural Network

- What if we compute vectors for **every** h-word phrase, often for several values of h?
  - Example: "the country of my birth" computes vectors for:
    - the country, country of, of my, my birth, the country of, country of my, of my birth, the country of my, country of my birth

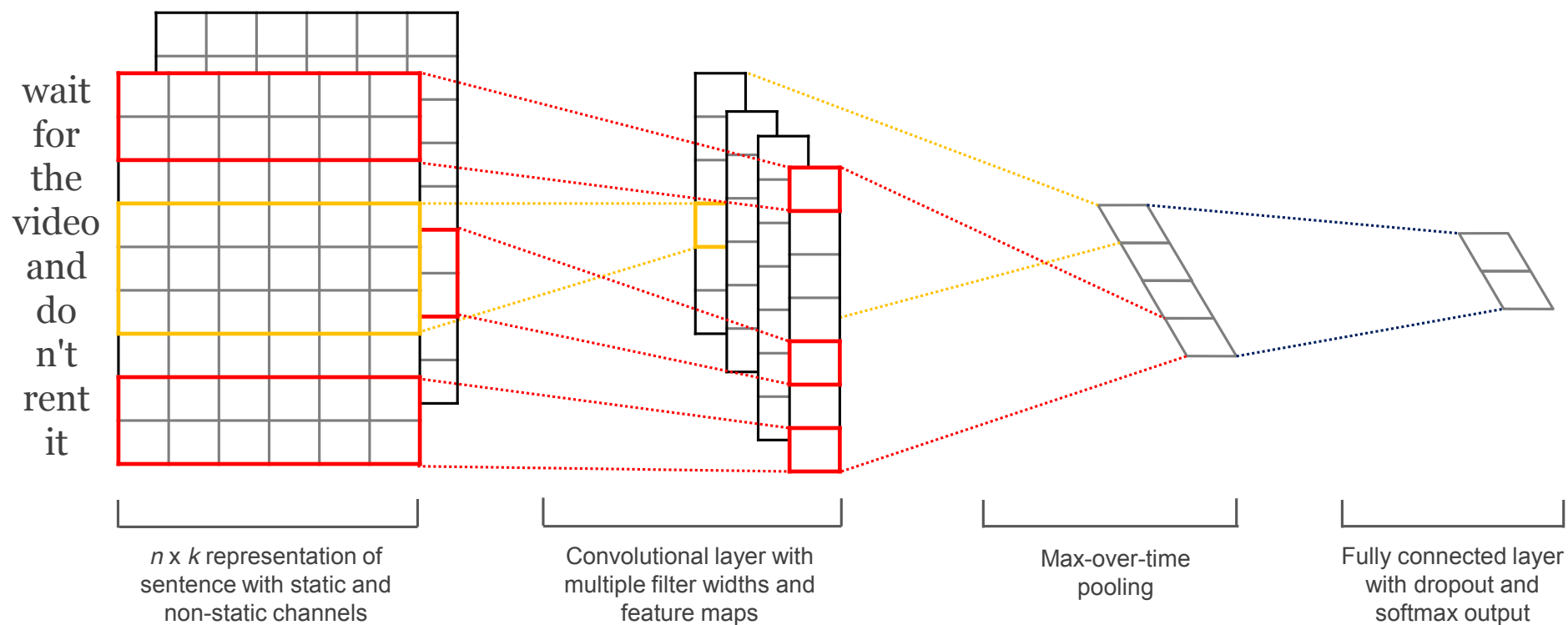- Not very linguistic, but you get everything!
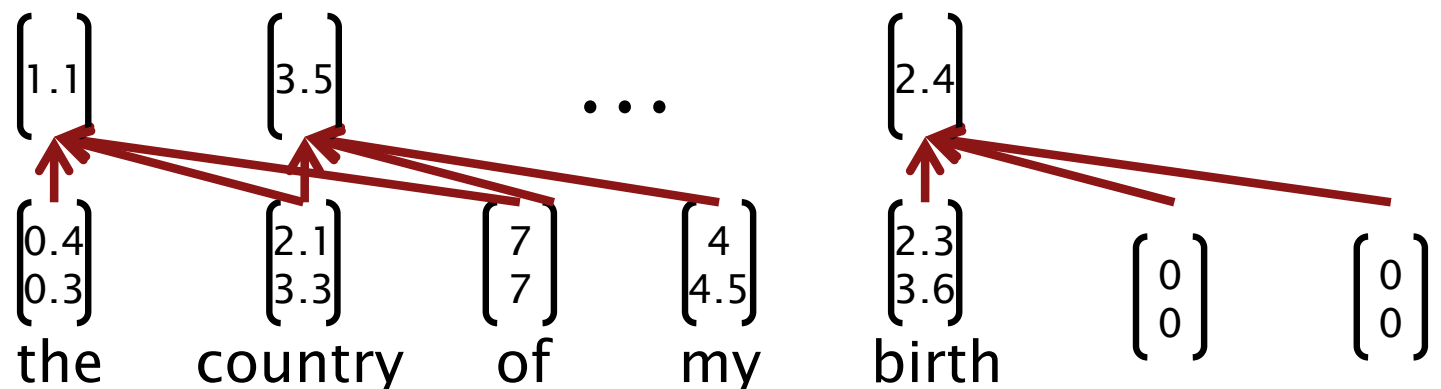
Convolutional Neural Network

- Word vectors: $\mathbf{x}_i \in \mathbb{R}^k$

- Concatenation of words in range: $\mathbf{x}_{i:i+j}$

- Convolutional filter: $\mathbf{w} \in \mathbb{R}^{hk}$

- CNN layer feature: $c_i = f(\mathbf{w}^T \mathbf{x}_{i:i+h-1} + b)$

- Get feature map: $\mathbf{c} = [c_1, c_2, \ldots, c_{n-h+1}]$

- Max pool (better than ave.): $\hat{c} = \max\{\mathbf{c}\}$

1D Convolutional neural network with max pooling and FC layer

wait for the video and do n't rent it

n x k representation of sentence with static and non-static channels

Convolutional layer with multiple filter widths and feature maps

Max-over-time pooling

Fully connected layer with dropout and softmax output

- For more features, use multiple filter weights and multiple window sizes
- Figure from [Kim 2014 "Convolutional Neural Networks for Sentence Classification"]

Data-dependent composition

Fig. 22
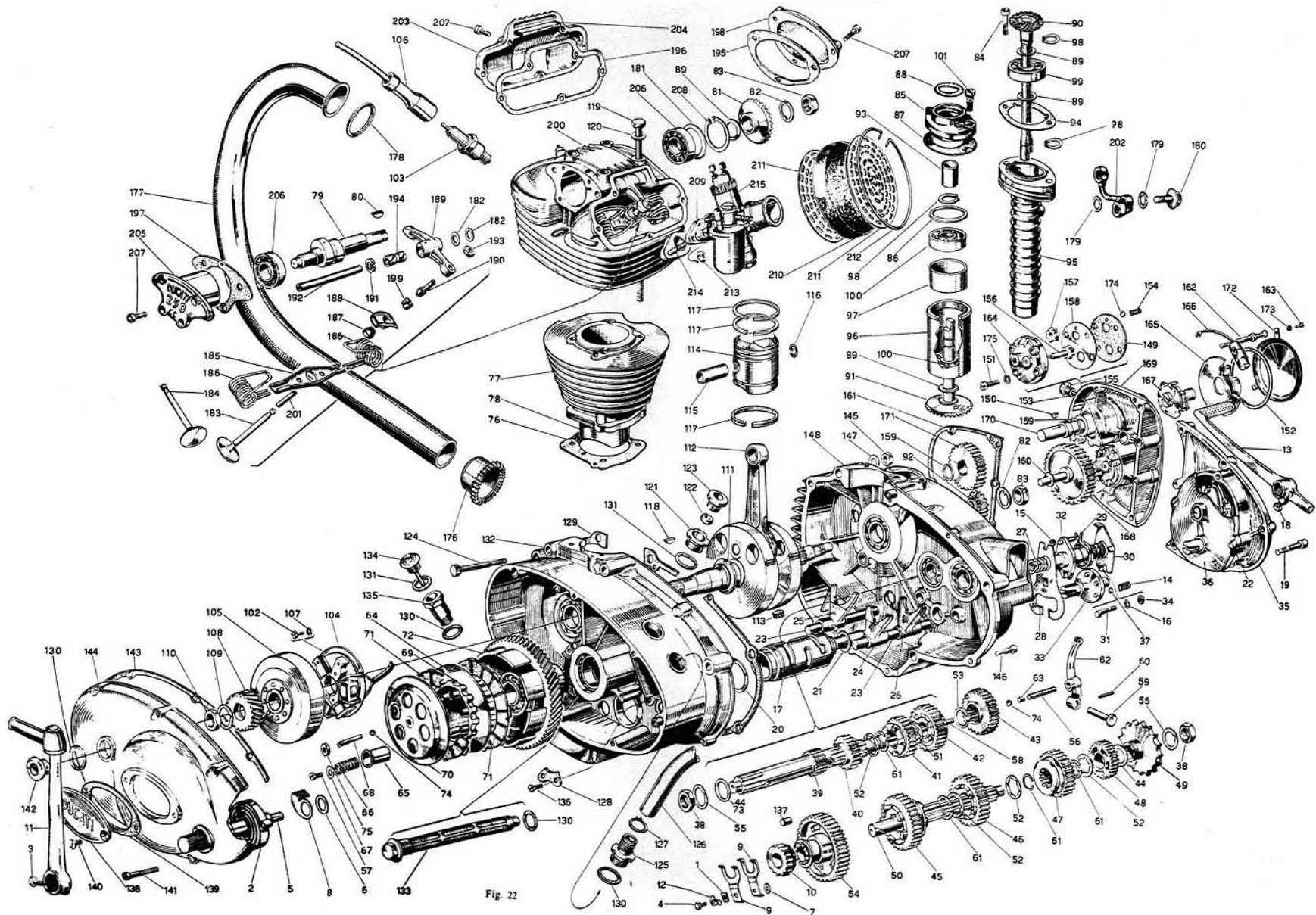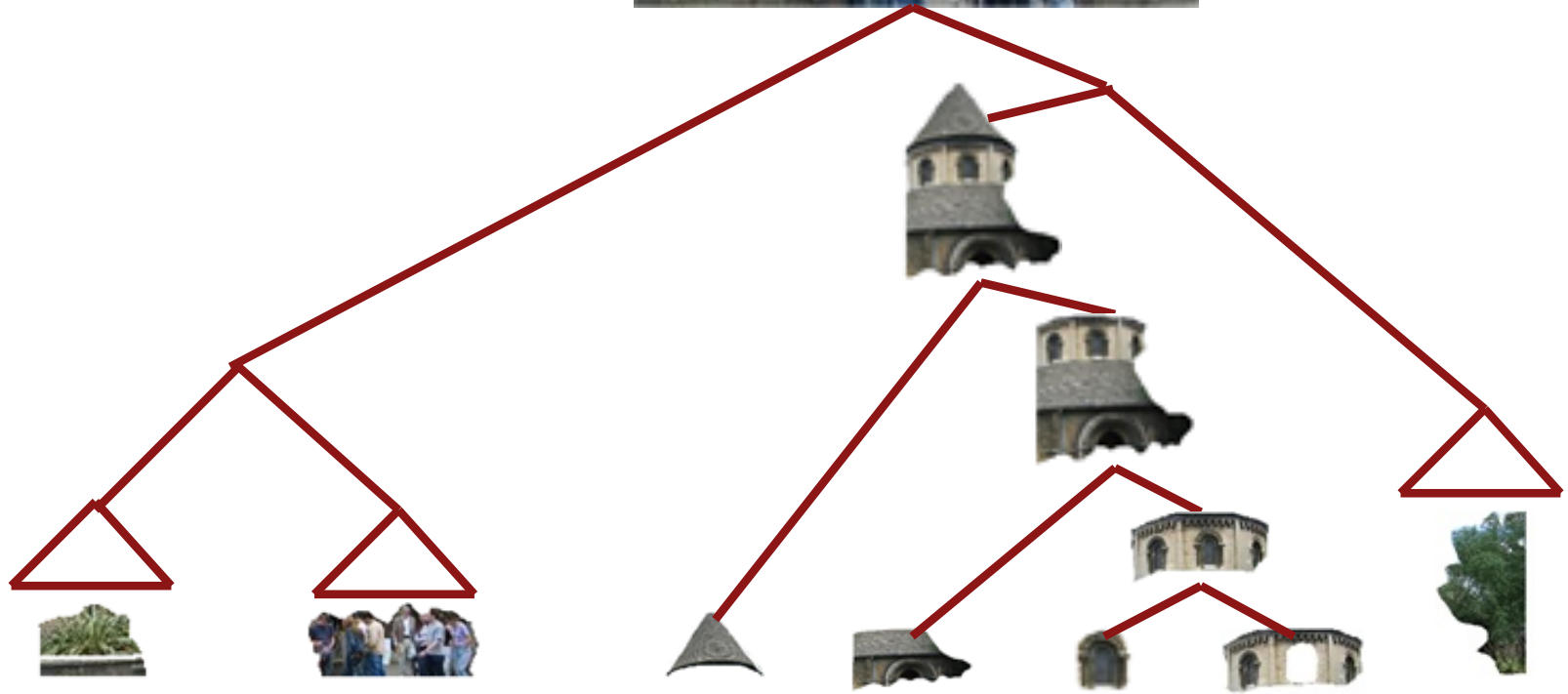
Language Understanding - & Artificial Intelligence - requires being able to understand bigger things from knowing about smaller parts

The Faculty of Language: What Is It, Who Has It, and How Did It Evolve?

Marc D. Hauser,[1]* Noam Chomsky,[2] W. Tecumseh Fitch[1]

We argue that an understanding of the faculty of language requires substantial interdisciplinary cooperation. We suggest how current developments in linguistics can be profitably wedded to work in evolutionary biology, anthropology, psychology, and neuroscience. We submit that a distinction should be made between the faculty of language in the broad sense (FLB) and in the narrow sense (FLN). FLB includes a sensory-motor system, a conceptual-intentional system, and the computational mechanisms for recursion, providing the capacity to generate an infinite range of expressions from a finite set of elements. We hypothesize that FLN only includes recursion and is the only uniquely human component of the faculty of language. We further argue that FLN may have evolved for reasons other than language, hence comparative studies might look for evidence of such computations outside of the domain of communication (for example, number, navigation, and social relations).

If a martian graced our planet, it would be struck by one remarkable similarity among Earth's living creatures and a key difference. Concerning similarity, it would note that all
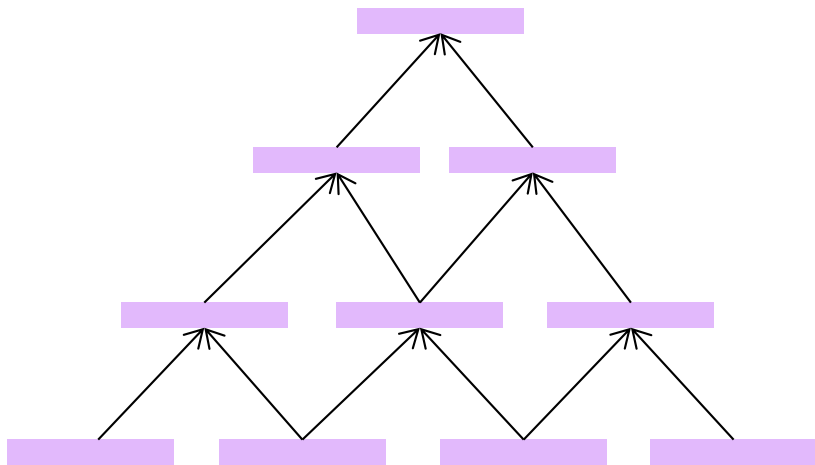
53

Language structure is recursive

- Recursion is natural for describing language
  - [The man from [the company that you spoke with about [the project] yesterday]]
  - noun phrase containing a noun phrase with a noun phrase
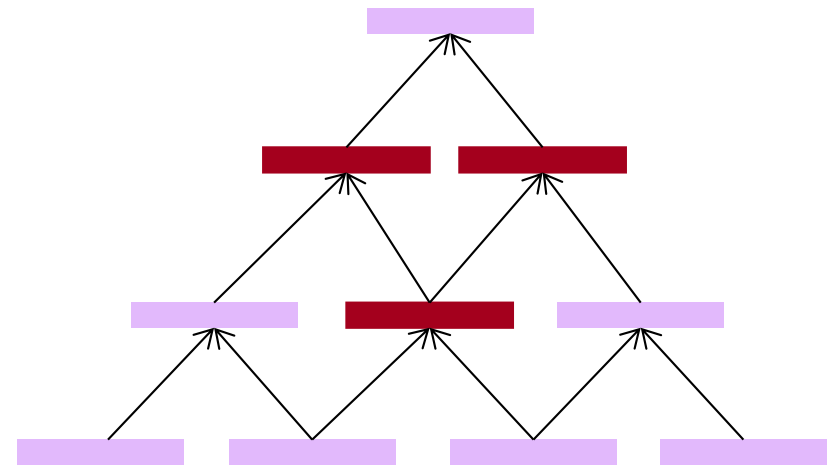- Phrases correspond to semantic units of language
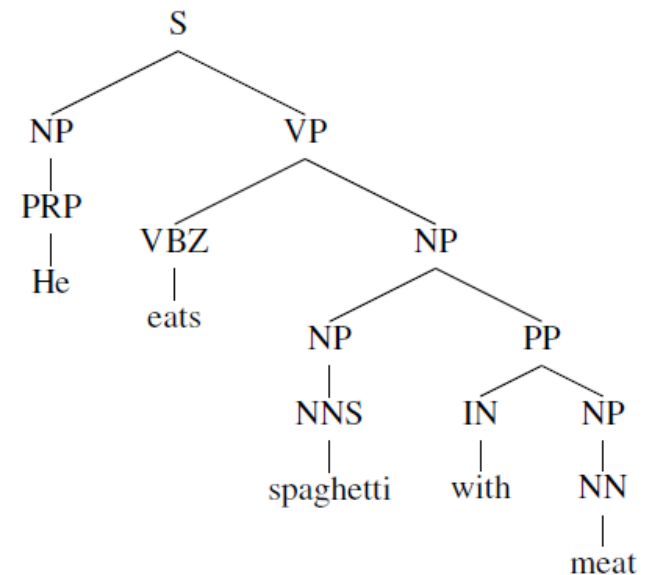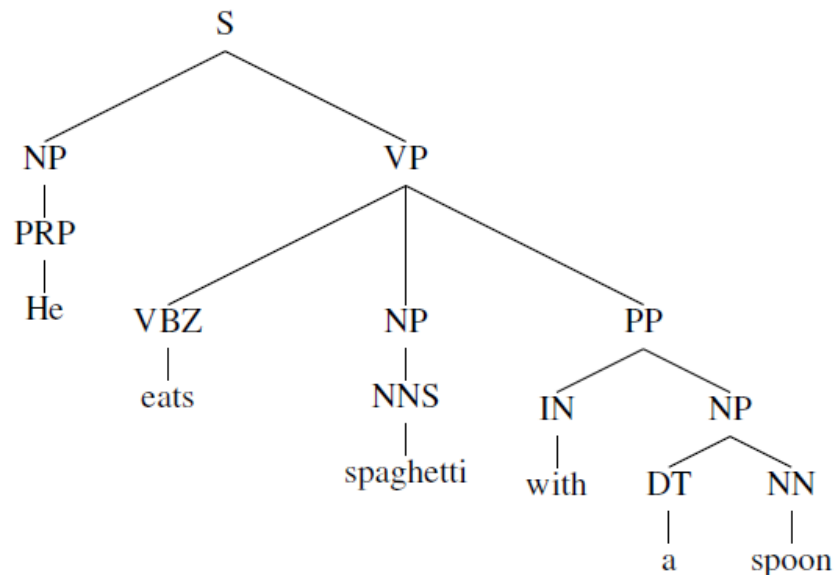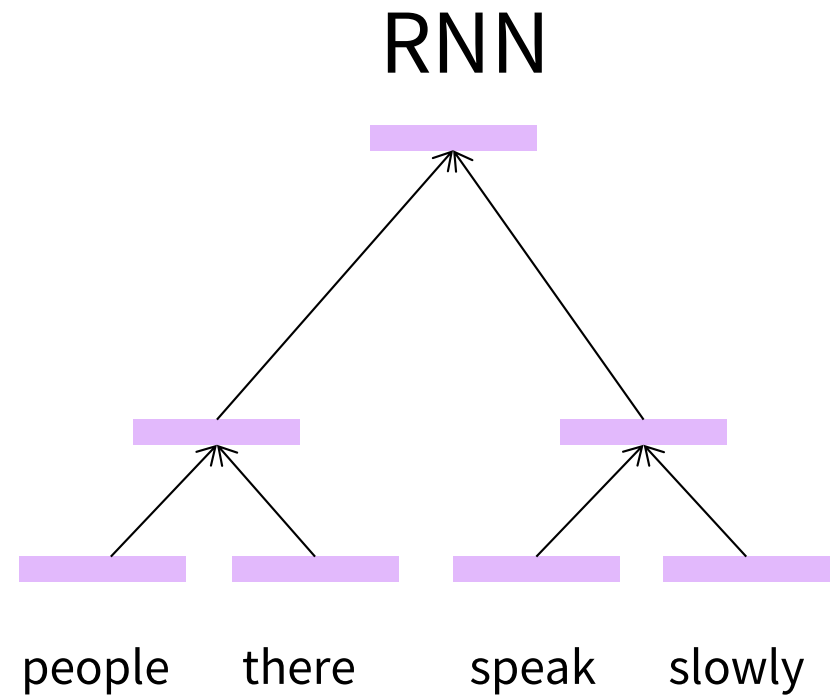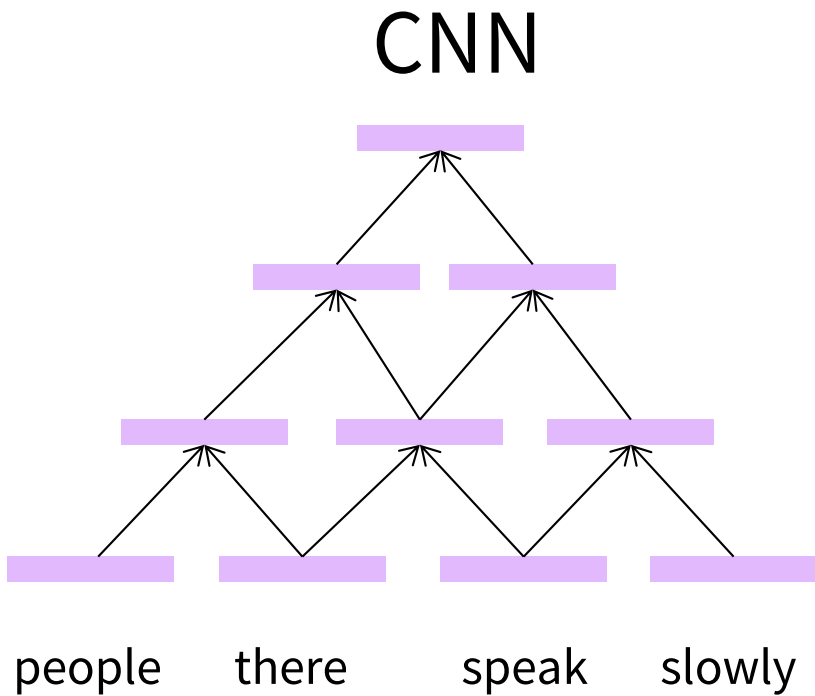
Relationship between RNNs and CNNs

CNN

RNN

Relationship between RNNs and CNNs

5. Using tree-structured models: Sentiment detection

Is the tone of a piece of text positive, negative, or neutral?

- Sentiment is that sentiment is "easy"
- Detection accuracy for longer documents ~90%

… … loved … … … … … … great … … … … … … …
impressed … … … … … … … marvelous … … … …

- BUT

With this cast, and this subject matter, the movie should have been funnier and more entertaining.
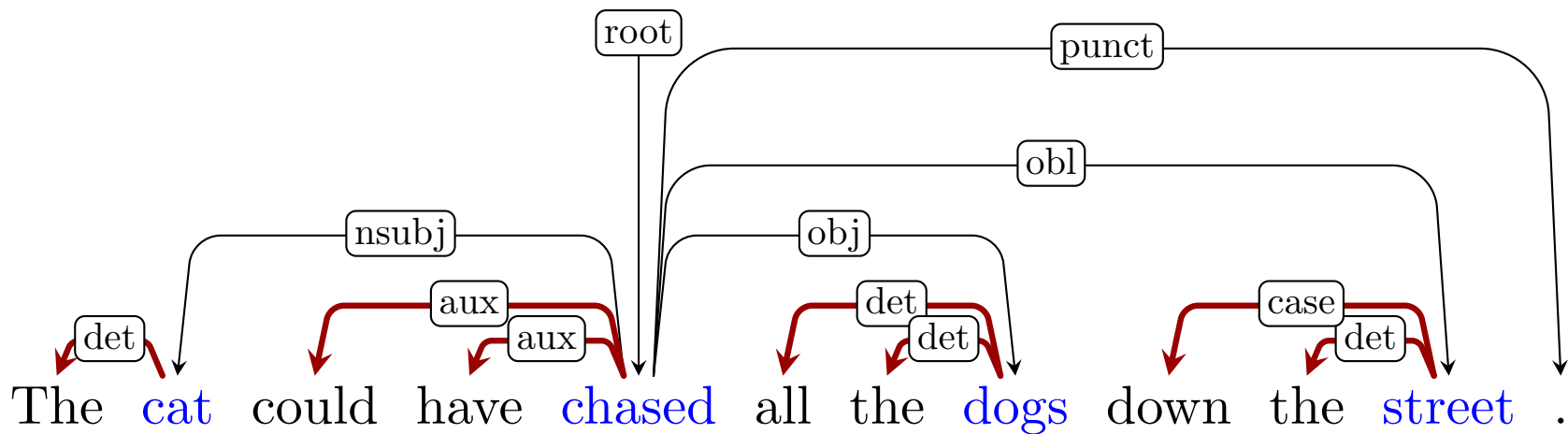
Stanford Sentiment Treebank

- 215,154 phrases labeled in 11,855 sentences
- Can train and test compositions

http://nlp.stanford.edu:8080/sentiment/
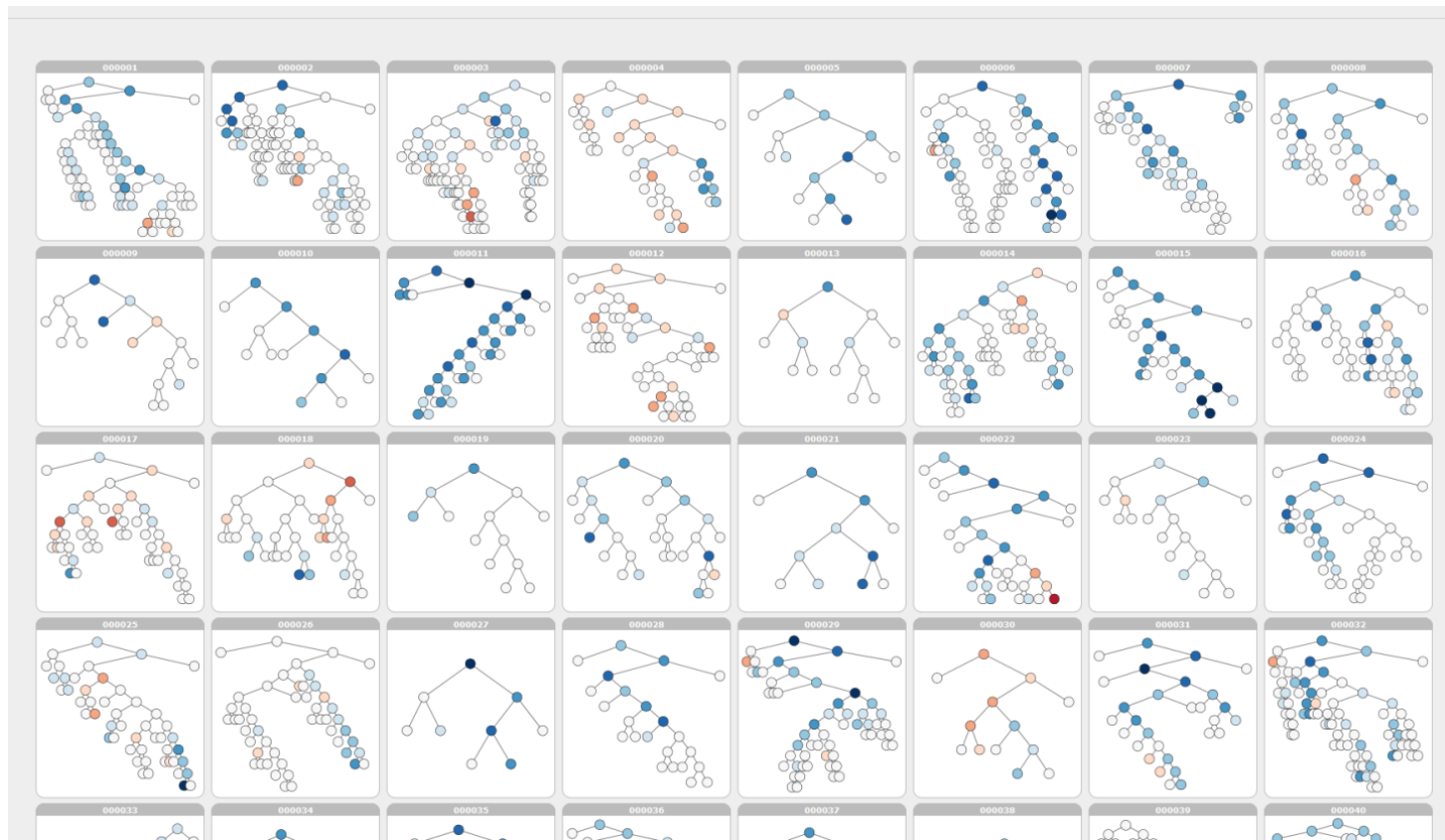
Universal Dependencies Syntax
http://universaldependencies.org/
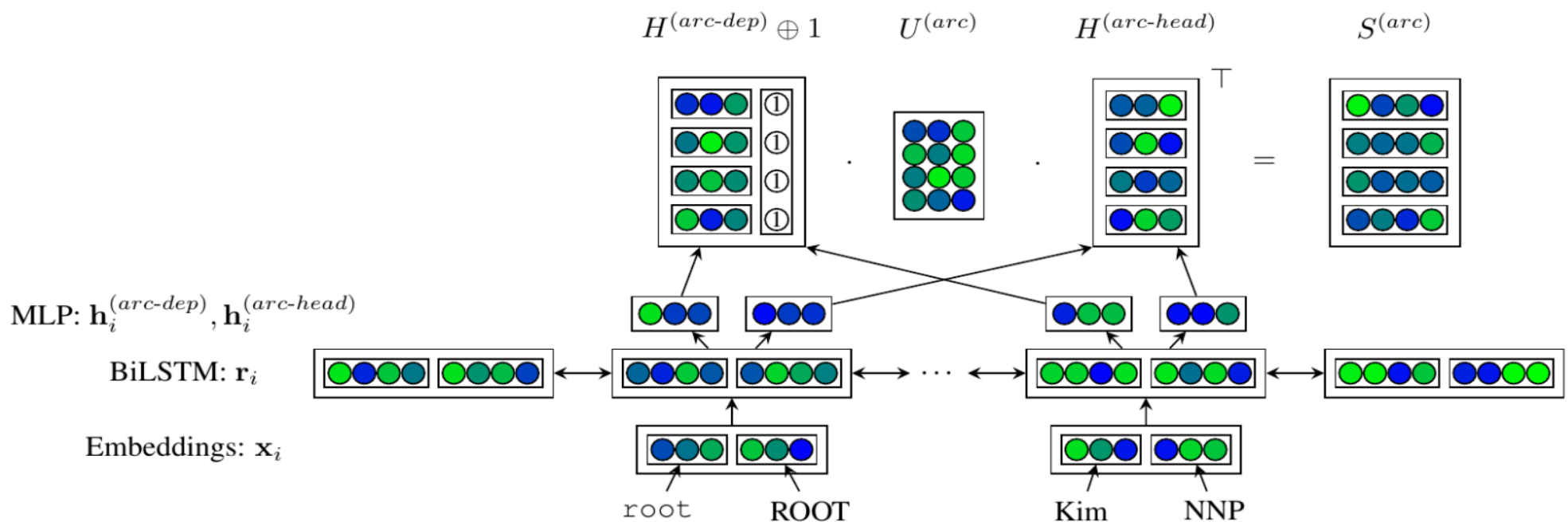

- Content words are related by dependency relations
- Function words attach to content word they modify
- Punctuation attaches to head of phrase or clause

Dozat & Manning (ICLR 2017)

- Each word predicts what it is a dependent of as a kind of head-dependent attention relation
- We then find the best tree (MST algorithm)

$H^{(arc-dep)} \oplus 1$ $U^{(arc)}$ $H^{(arc-head)}$ $S^{(arc)}$

MLP: $\mathbf{h}_i^{(arc-dep)}$, $\mathbf{h}_i^{(arc-head)}$

BiLSTM: $\mathbf{r}_i$

Embeddings: $\mathbf{x}_i$

root   ROOT   Kim   NNP

PTB-SD 3.3.0 and CTB 5.1 Results

Type	Model	PTB-SD		CTB	
		UAS	LAS	UAS	LAS
Transition	Chen & Manning (2014)	92.0	89.7	83.9	82.4
	Andor et al. (2016)	94.61	92.79	--	--
	Kuncoro et al. (2016)	**95.8**	**94.6**	--	--
Graph	K & G (2016)	93.9	91.9	87.6	86.1
	Cheng et al. (2016)	94.10	91.49	88.1	86.1
	Hashimoto et al. (2016)	94.67	92.90	--	--
	Ours	95.74	94.08	**89.30**	**88.23**

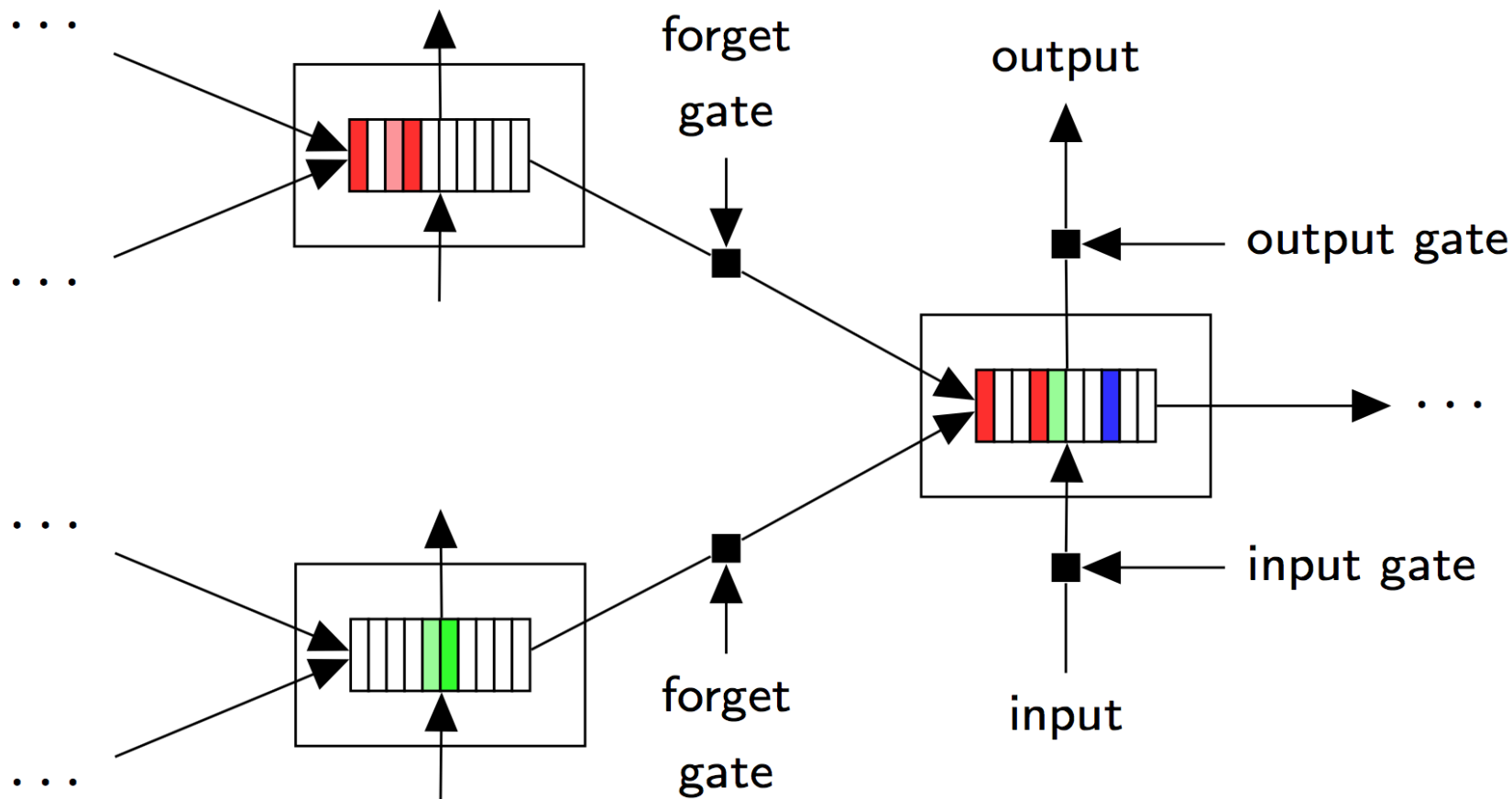# Tree-Structured Long Short-Term Memory Networks [Tai et al., ACL 2015]
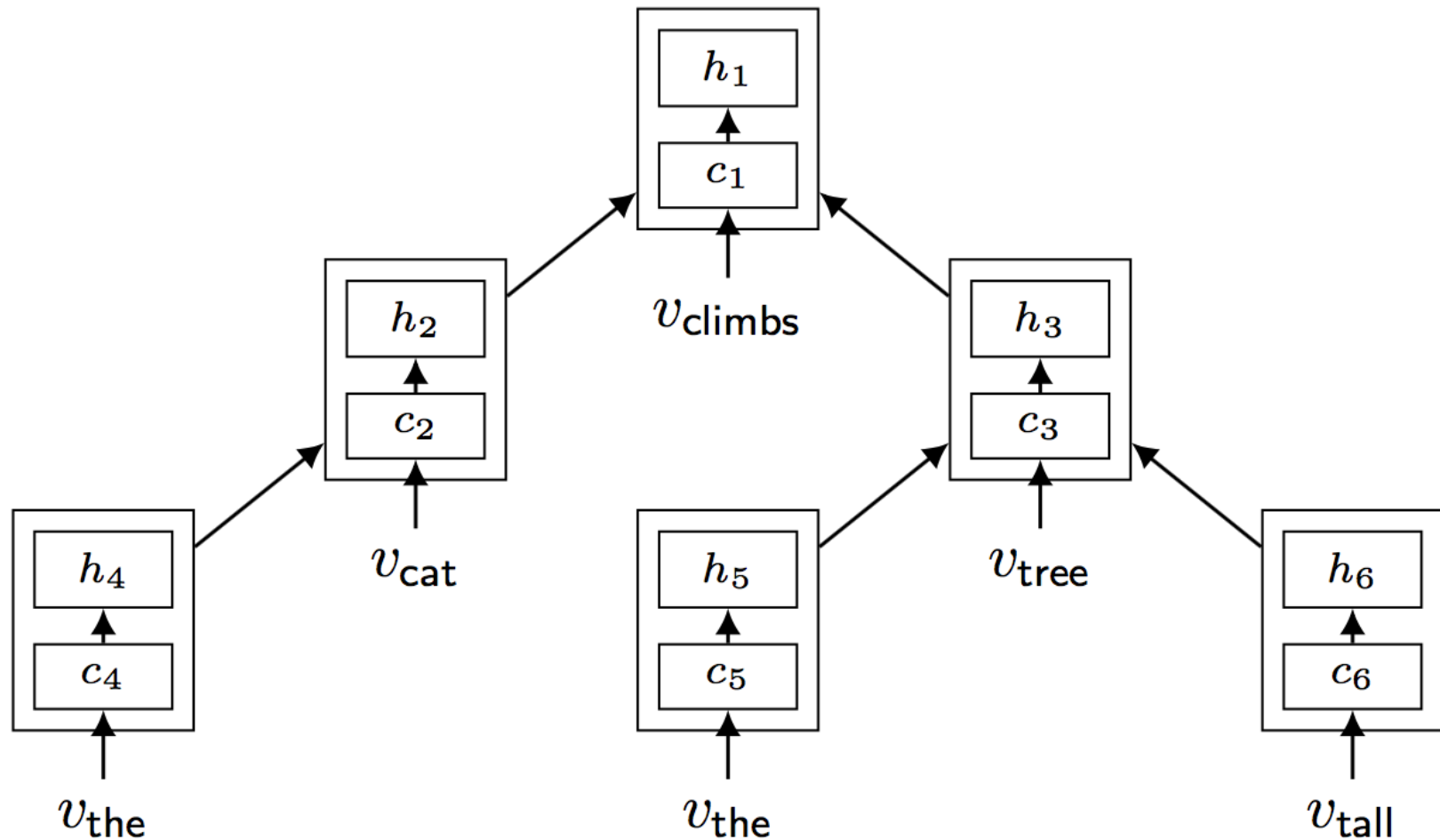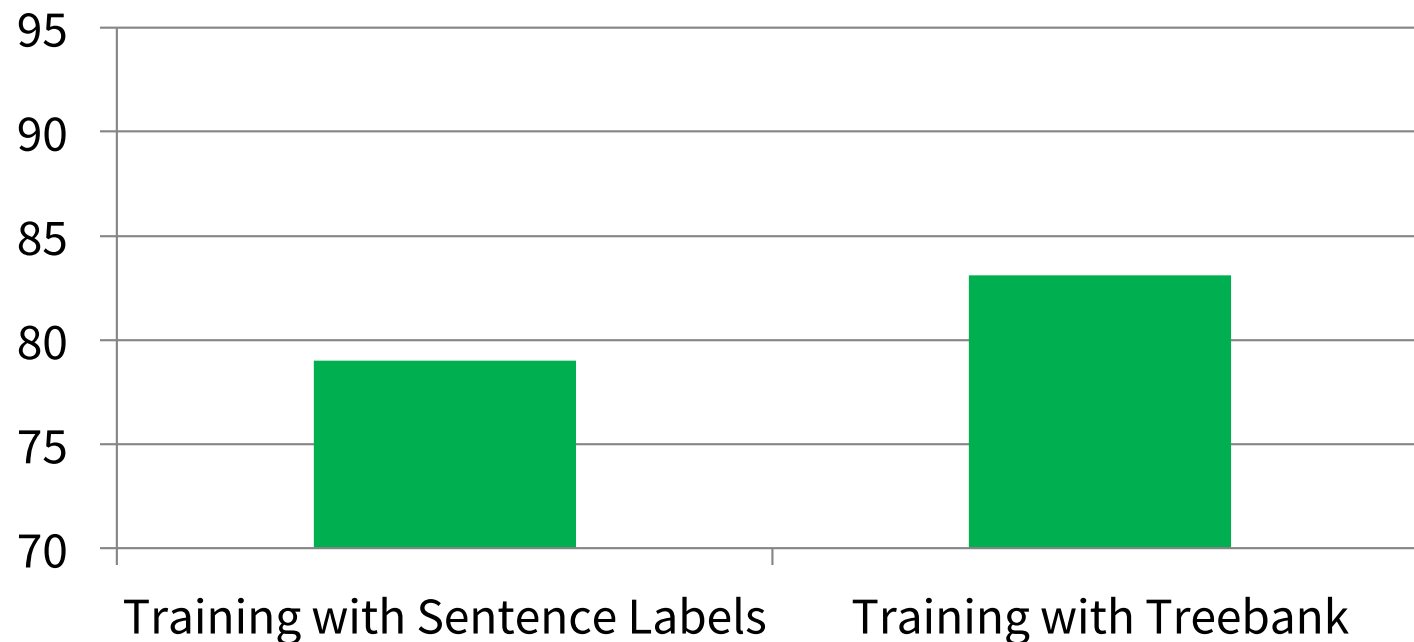
Tree-structured LSTM

Generalizes sequential LSTM to trees with any branching factor
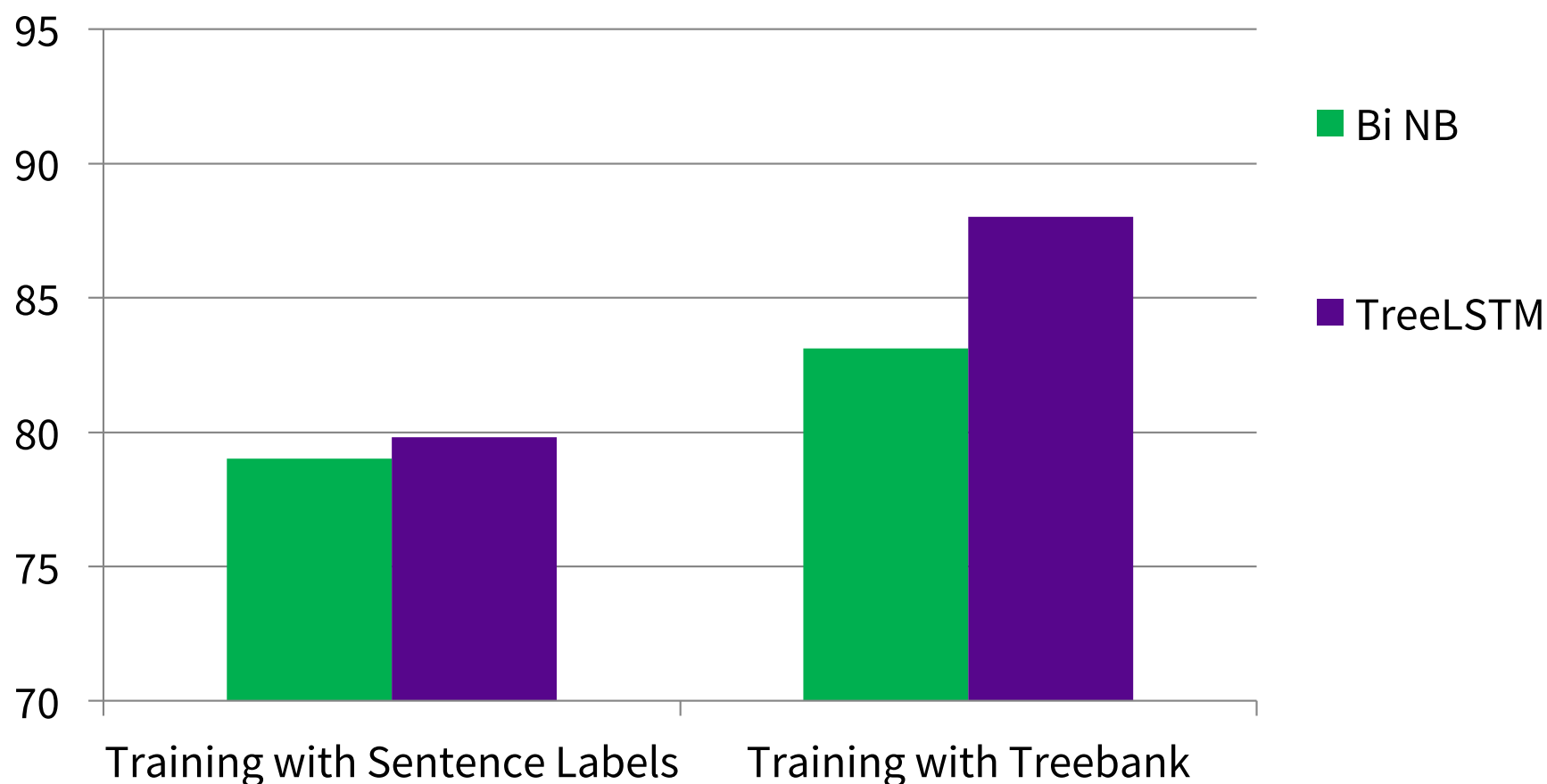
Better Dataset Helped Even Simple Models

Positive/negative sentence classification Uni+Bigram Naïve Bayes

- But hard negation cases are still mostly incorrect
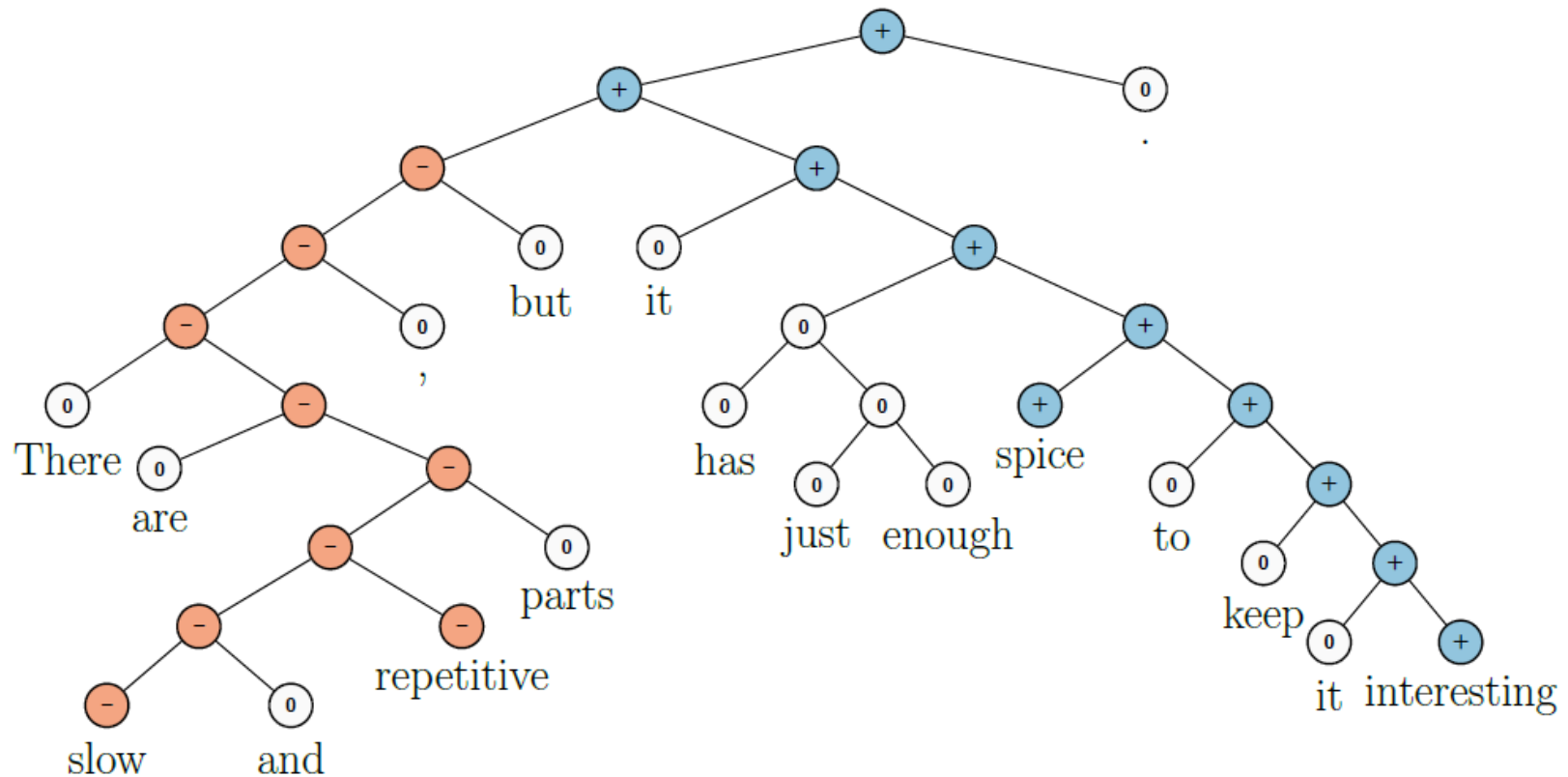- We also need a more powerful model!

Positive/Negative Results on Treebank

Classifying Sentences: Accuracy improves to 88%

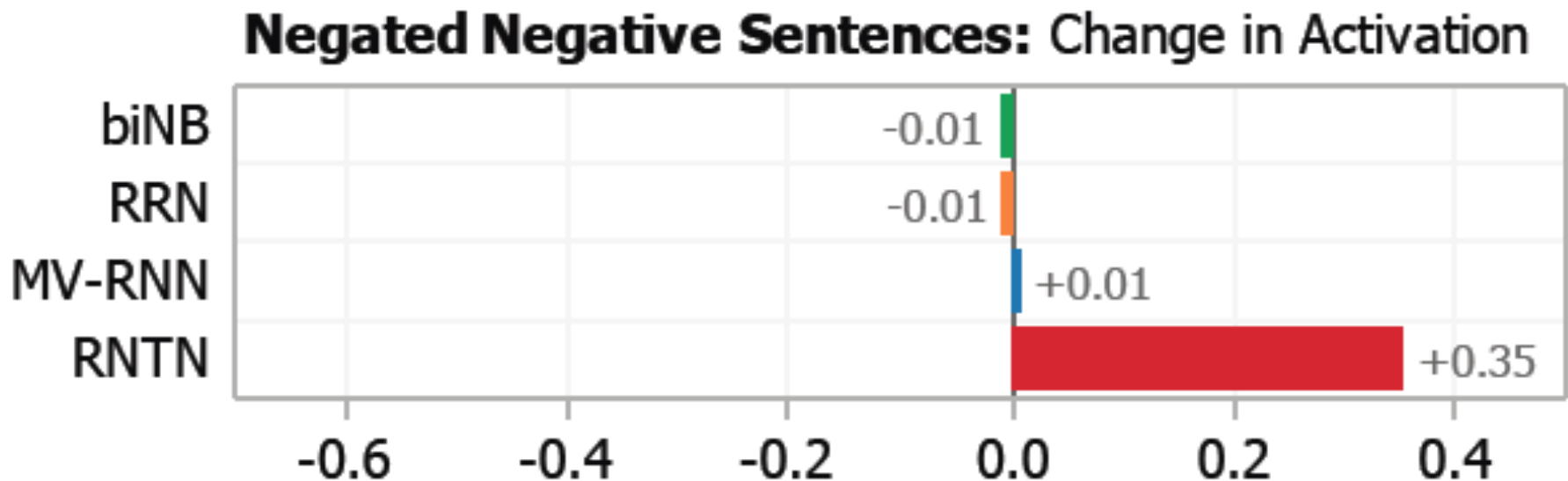# Experimental Results on Treebank

- TreeRNN can capture constructions like *X but Y*
- Biword Naïve Bayes is only 58% on these

Results on Negating Negatives

E.g., sentiment of "not uninteresting"
Goal: Positive activation should increase

Negated Negative Sentences: Change in Activation

biNB	-0.01
RRN	-0.01
MV-RNN	+0.01
RNTN	+0.35

-0.6   -0.4   -0.2   0.0   0.2   0.4

Envoi

- Deep learning – distributed representations, end-to-end training, and richer modeling of state – has brought great gains to NLP

- At the moment, it seems like we can't win, or we can only barely win, by having more structure than a vector space mush

- However, I deeply believe that we do need more structure and modularity for language, memory, knowledge, and planning; it'll just take some time