Natural Language Processing (Lecture 1)

MLSS 2018

THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Mohit Bansal

(some slides adapted/borrowed from courses by Dan Klein, Richard Socher, Chris Manning, JurafskyMartin-SLP3 book, others)

About the Course (and its Goals)

- We'll start with introduction/motivation and day-to-day applications of NLP
- Then we will cover some specific recent topics useful for this summer school's theme, e.g., word embeddings and sentence embeddings, RNNs, etc.
- We will also cover some core NLP tasks/applications such as semantic parsing, Q&A, MT, Summarization, and Dialogue

Reference Books

- **SLP2**: D. Jurafsky & James H. Martin. “Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition”. Prentice Hall, Second Edition, 2009.
- **SLP3**: Some draft chapters of the third edition are available online at https://web.stanford.edu/~jurafsky/slp3/
- **FSNLP**: Chris Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press. Cambridge, MA: May 1999. http://nlp.stanford.edu/fsnlp/
- **ML Background**: Andrew Ng’s Coursera Machine Learning course https://www.coursera.org/learn/machine-learning
- **Stanford NLP + Deep Learning Class**: http://web.stanford.edu/class/cs224n/
- **My Fall2017 NLP Class:** http://www.cs.unc.edu/~mbansal/teaching/nlp-course-fall17.html

NLP Topics

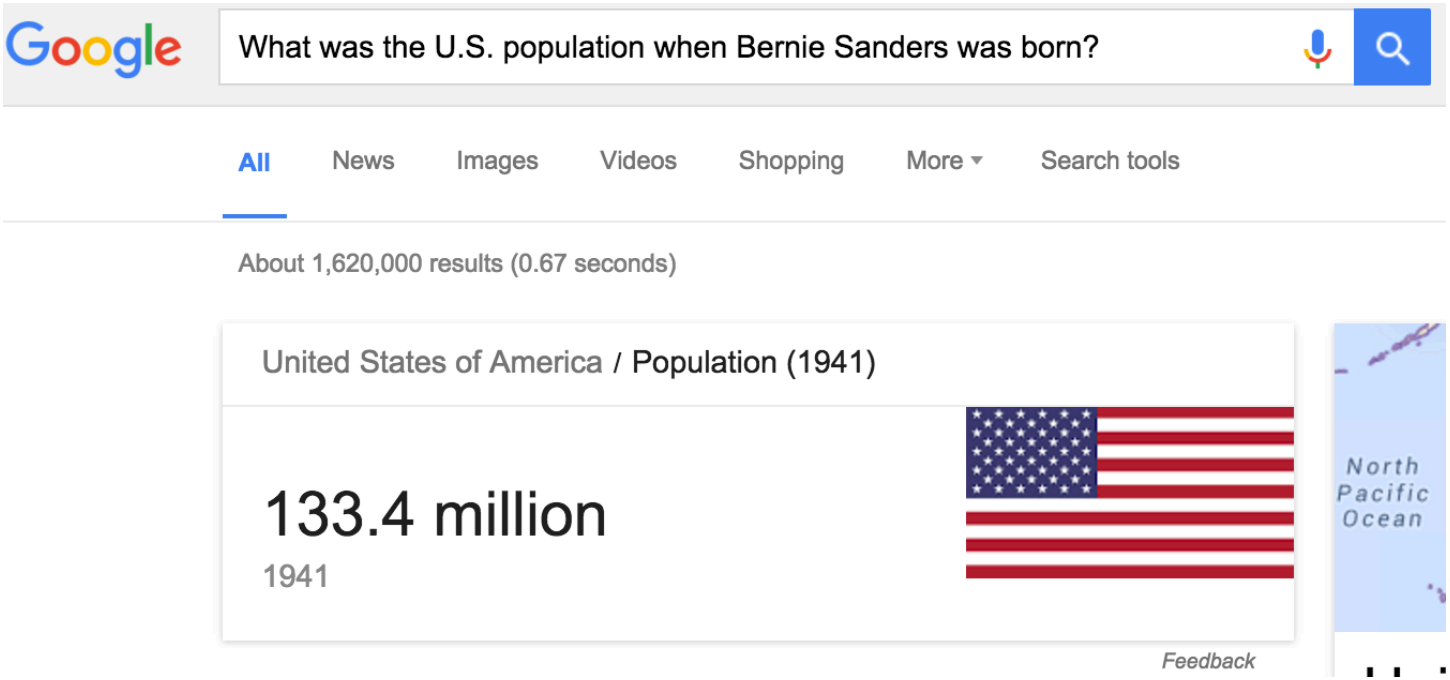
- Language Modeling
- Part-of-speech Tagging
- Syntactic Parsing: Constituent, Dependency, CCG, others
- Coreference Resolution
- Distributional Semantics: PMI, neural, CCA
- Compositional Semantics: Logical-form, Semantic Parsing, Vector-form, neural (RNNs/CNNs)
- Question Answering: Factoid-based, Passage-based
- Sentiment Analysis
- Document Summarization
- Machine Translation
- Dialogue Models
- Language and Vision: Image Captioning, Video Captioning, Visual Question Answering
- Language and Robotics: Instructions for Navigation, Manipulation, Skill Learning; Human-Robot Interaction
- Several interesting machine and deep learning models all along the way, e.g., HMMs, CRFs, factor graphs, deep+structured models, interpretable models, adversarial models, reward-based models (reinforcement learning)

What is NLP?

- Question answering

What is NLP?

- Question answering

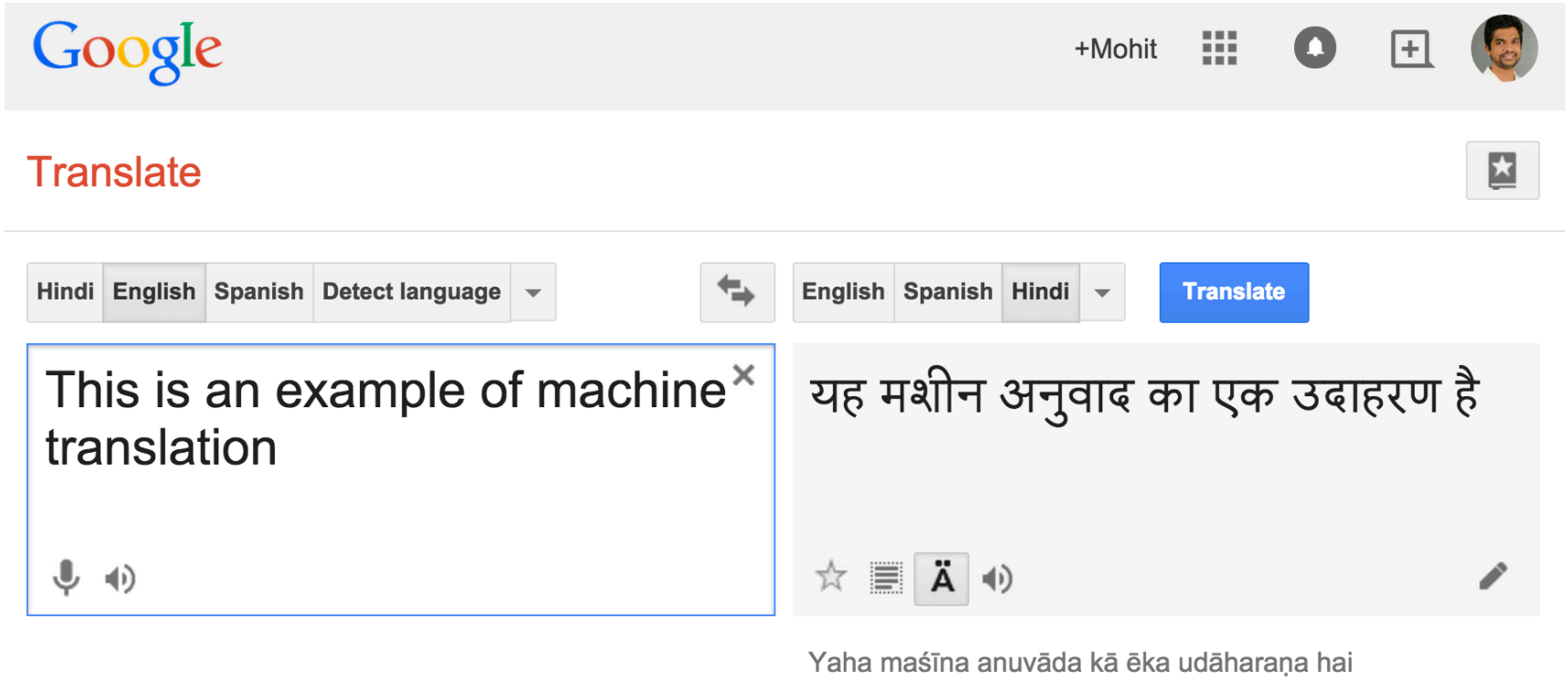


What is NLP?

- Question answering

What is NLP?

- Machine Translation



What is NLP?

- Sentiment Analysis

Sentiment Analysis with Python NLTK Text Classification

This is a demonstration of **sentiment analysis** using a `NLTK 2.0.4` powered **text classification** process. It can tell you whether it thinks the text you enter below expresses **positive sentiment**, **negative sentiment**, or if it's **neutral**. Using **hierarchical classification**, *neutrality* is determined first, and *sentiment polarity* is determined second, but only if the text is not neutral.

Analyze Sentiment

Language

english

Enter text

It always amazes me how Universal never cares to create anything remotely clever when it comes to their animations, and so once again they come up with a harmless little story that wants to be cute and funny (which it is sometimes) but is only bound to be quickly forgotten.

Enter up to 50000 characters

Analyze

Sentiment Analysis Results

The text is **neg**.

The final sentiment is determined by looking at the classification probabilities below.

Subjectivity

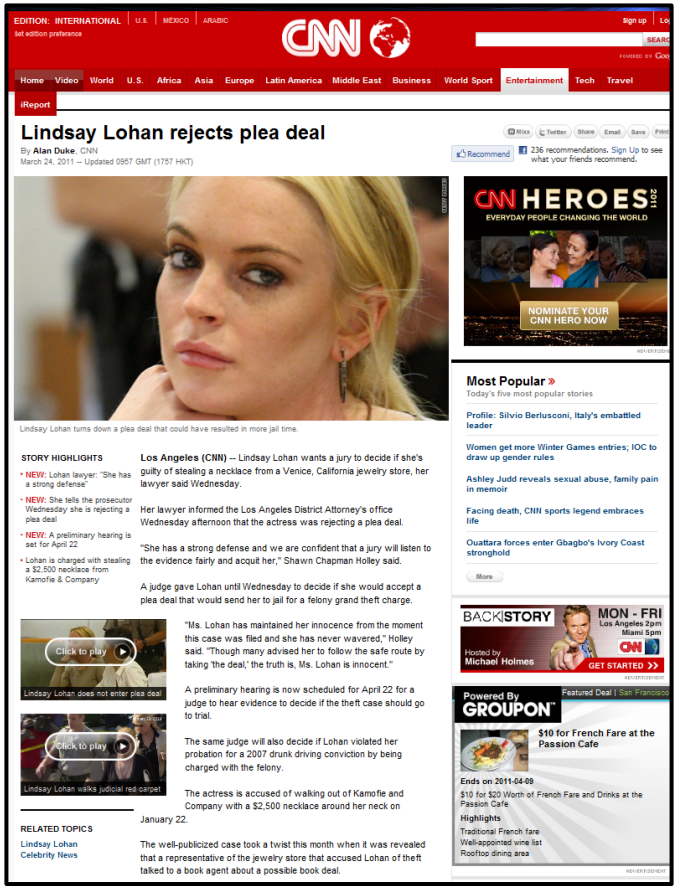
- neutral: 0.3
- **polar: 0.7**

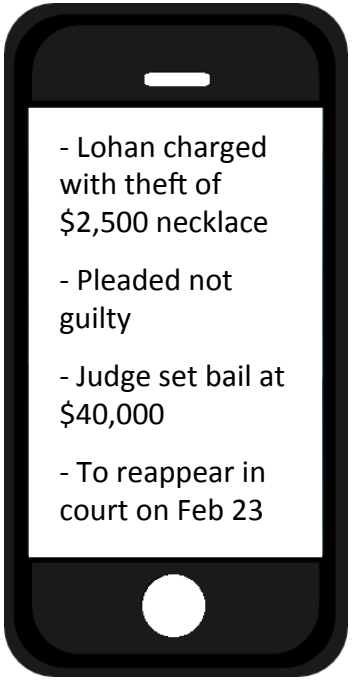
Polarity

- pos: 0.2
- **neg: 0.8**

What is NLP?

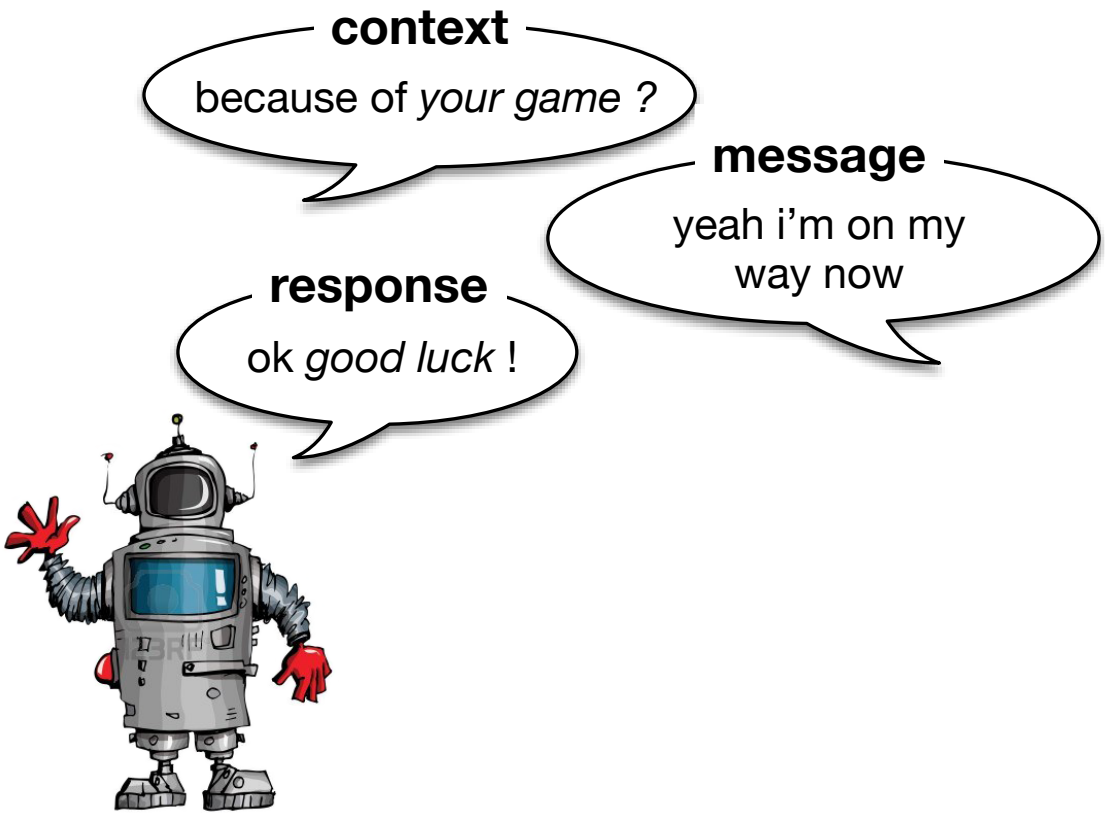
- Natural Language Generation: Summarization





What is NLP?

- Natural Language Generation: Conversation/Dialogue



[Sordoni et al., 2015]

What is NLP?

- Natural Language Generation: Image Captioning

www.cs.toronto.edu/~nitish/nips2014demo/results/8832804.html

Results

Tags

- authors
- scones
- luncheon
- breakfast
- seder

Nearest Caption in the Training Dataset

a man cuts a cake while children sit around at the table , looking on .

Generated Captions

- two people at a table with a cake .
- the two people are having a meal that is in a party .
- a man and two children in a blue table with a cake .
- a man sitting at a table with a bunch of cake on it .
- a man and woman sitting at a table with cake at a party .

back

[UToronto]

What is NLP?

- Natural Language Generation: Video Captioning



Ground truth: Two women are shopping in a store.
Two girls are shopping.
Baseline model: A man is doing a monkey in a store.
Multi-task model: A woman is shopping in a store.



Ground truth: Two men are fighting.
A group of boys are fighting.
Baseline model: A group of men are dancing.
Multi-task model: Two men are fighting.

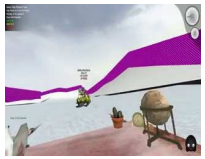


Ground truth: A man is playing a violin.
A man is playing the violin on stage.
Baseline-XE: A man is playing the drums.
CIDEr-RL: A man is playing a guitar.
CIDEnt-RL: A man is playing a violin.



Ground truth: Two men are wrestling.
Two guys are wrestling in a competition.
Baseline-XE: A group of people are playing a game.
CIDEr-RL: A man is playing a wrestling.
CIDEnt-RL: Two men are wrestling.





Ground truth: A person is playing a video game.
Someone is playing video game.
Baseline-XE: A man is riding a motorcycle.
CIDEr-RL: A man is talking about a plane.
CIDEnt-RL: A person is playing a video game.

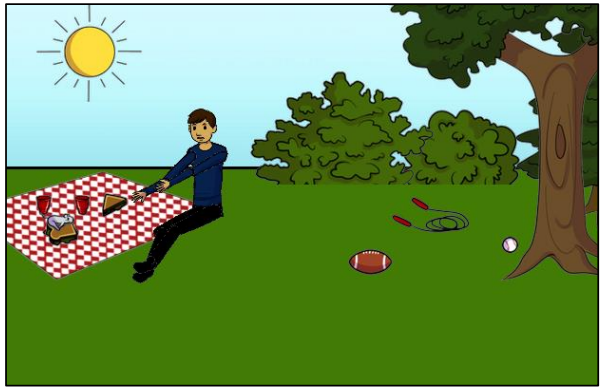
[Pasunuru and Bansal, 2017]

What is NLP?

- Natural Language Generation: Visual Question Answering

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?



Is this person expecting company?
What is just under the tree?

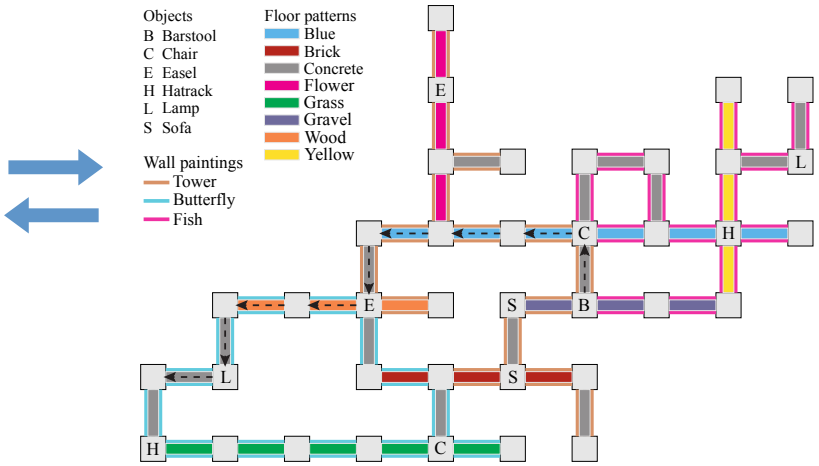
Does it appear to be rainy?
Does this person have 20/20 vision?

[Antol et al., 2015]

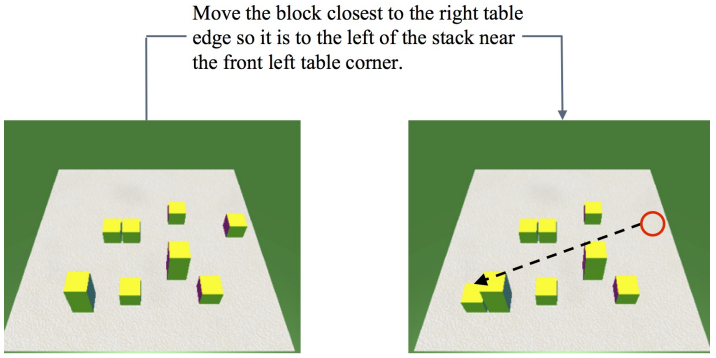
What is NLP?

- Language+Robotics: Navigation, Configuration/Assembling, etc.

Place your back against the wall of the "T" intersection. Go forward one segment to the intersection with the blue-tiled hall. This interesction [sic] contains a chair. Turn left. Go forward to the end of the hall. Turn left. Go forward one segment to the intersection with the wooden-floored hall. This intersection conatains [sic] an easel. Turn right. Go forward two segments to the end of the hall. Turn left. Go forward one segment to the intersection containing the lamp. Turn right. Go forward one segment to the empty corner.



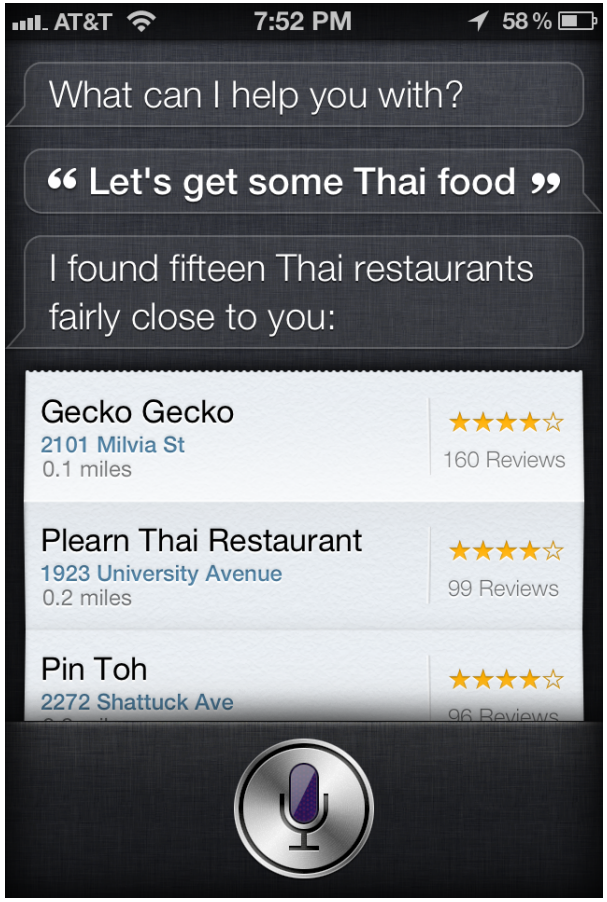
Instruction: Head upstairs and walk past the piano through an archway directly in front. Turn right when the hallway ends at pictures and table. Wait by the moose antlers hanging on the wall.



[Chen and Mooney, 2011 ;Mei et al., 2016; Daniele et al., 2017; Anderson et al., 2018; Bisk et al., 2016; Tan and Bansal, 2018]

What is NLP?

- Automatic Speech Recognition



Some Exciting NLP Challenges

Human-like Ambiguous Language

- Non-literal: Idioms, Metaphors

You: *I am under the weather today.*
Siri: *Here is the weather today… 50 F*

Human-like Ambiguous Language

- Non-literal: Idioms, Metaphors

Human-like Ambiguous Language

- Humor, Sarcasm, Politeness/Rudeness

Human-like Ambiguous Language

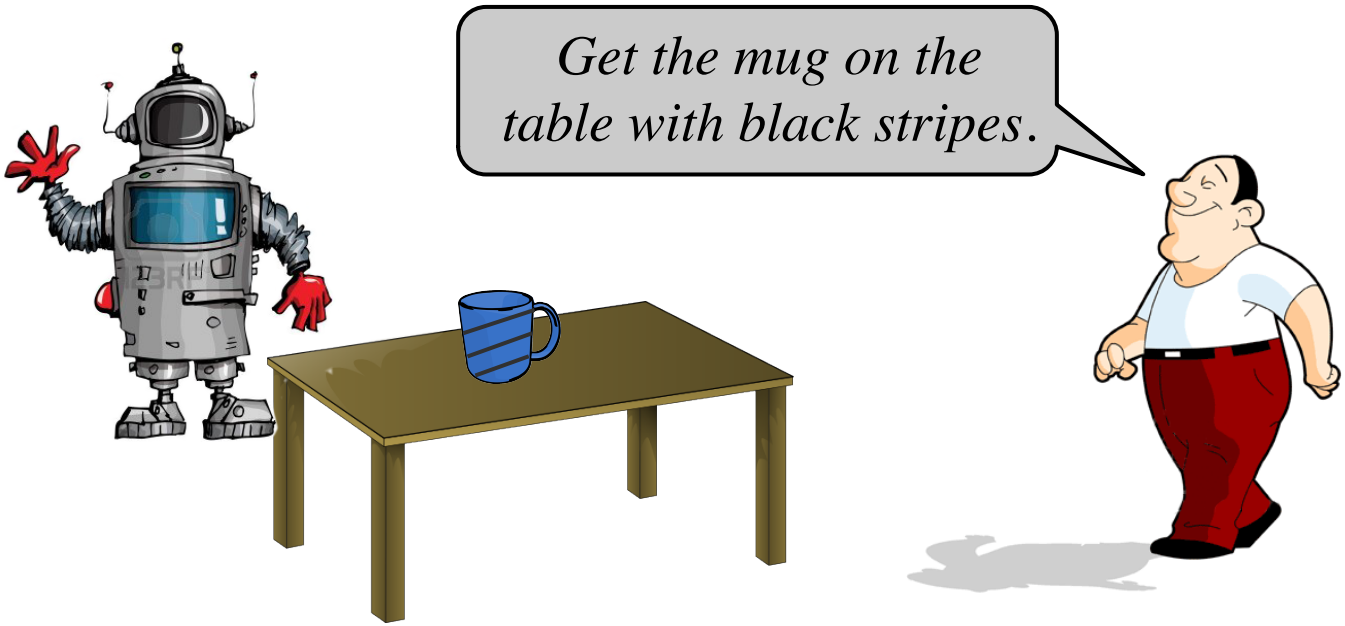
- Prepositional Attachment, Coreference Ambiguities

Human-like Ambiguous Language

- Prepositional Attachment, Coreference Ambiguities

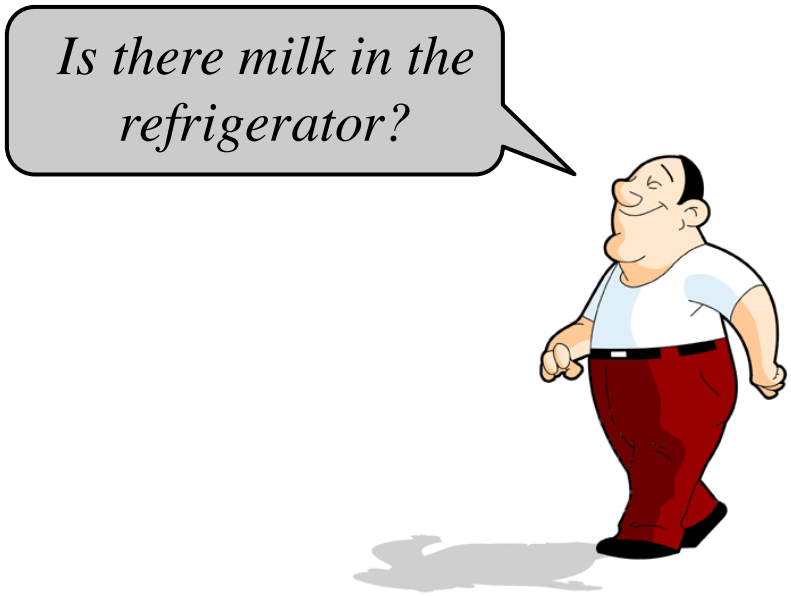
Visually Grounded Language

- Text-Image Alignment: Most of our daily communication language points to several objects in the visual world



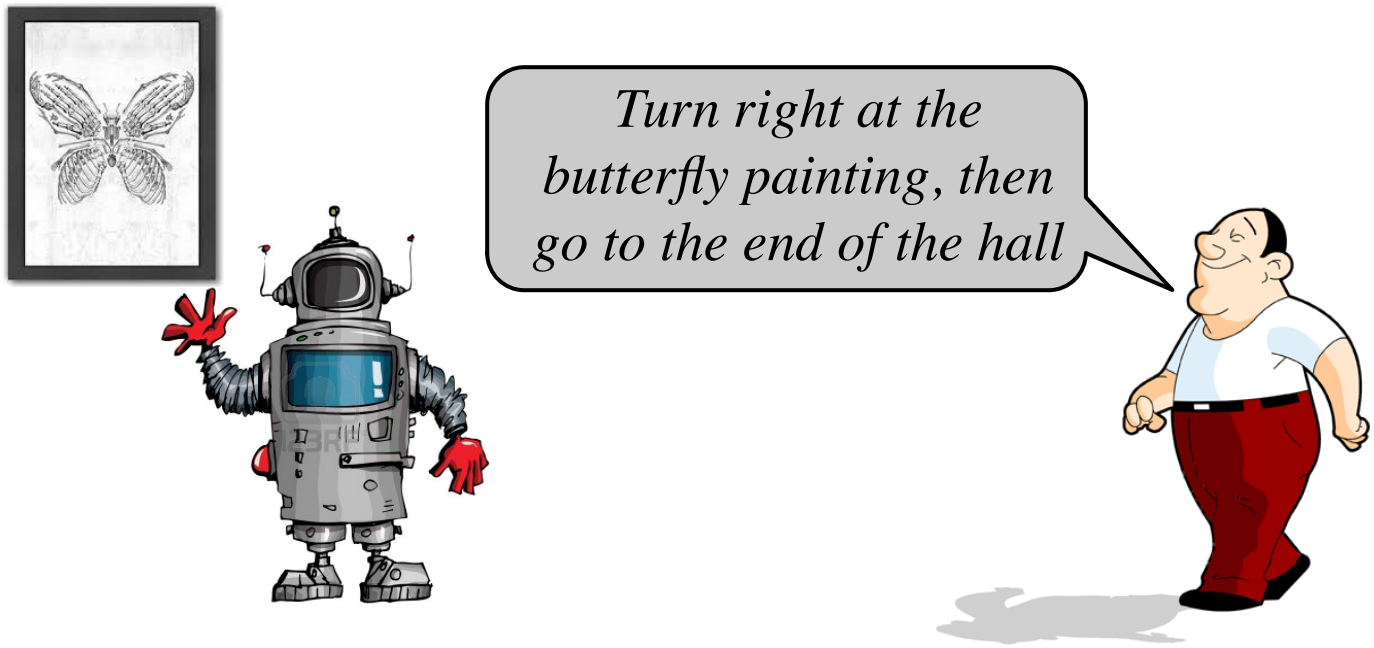
Visually Grounded Language

- Visual Question Answering: Humans asking machines about pictures/videos, e.g., for visually impaired, in remote/ dangerous scenarios, in household service settings



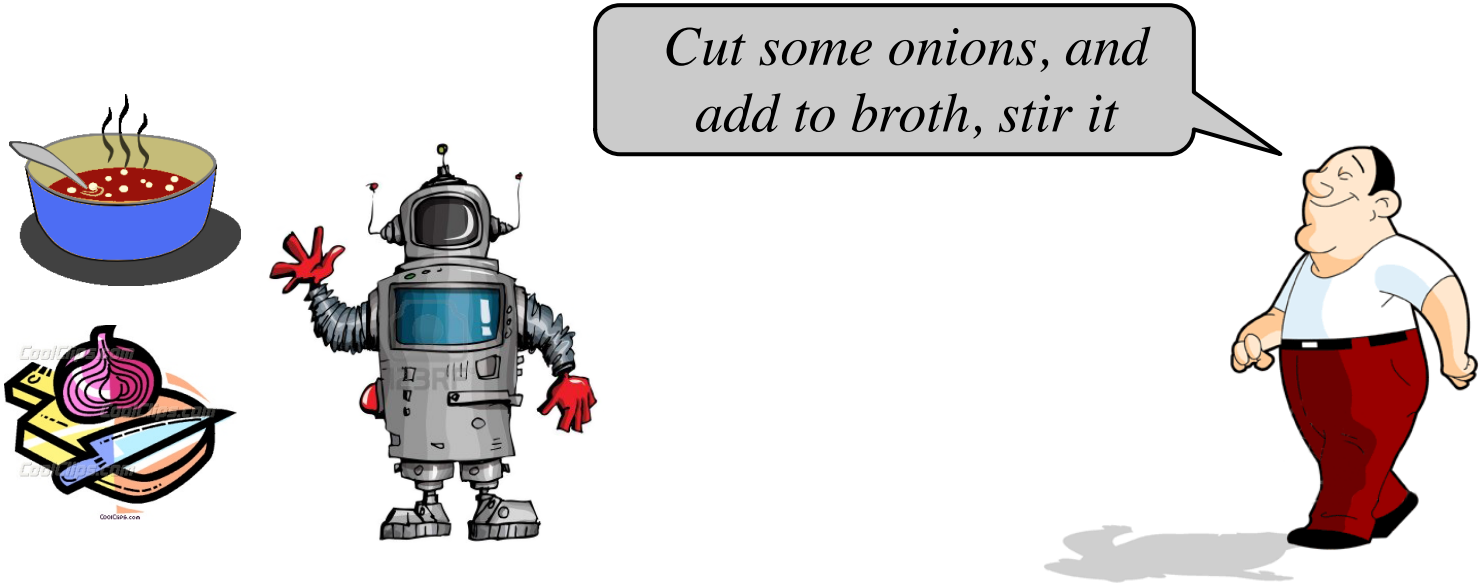
Embodied Language (Robot Instructions)

- Task-based instructions, e.g., navigation, grasping, manipulation, skill learning



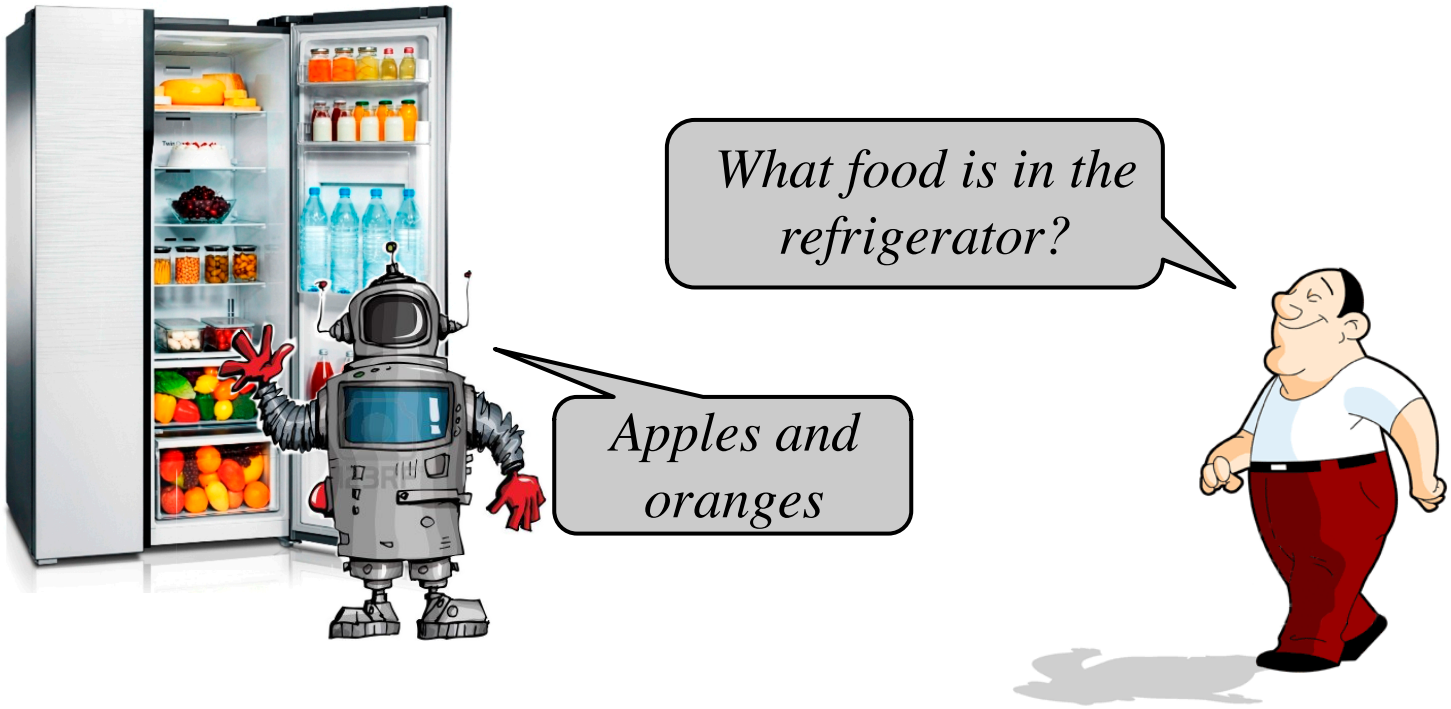
Embodied Language (Robot Instructions)

- Task-based instructions, e.g., navigation, grasping, manipulation, skill learning



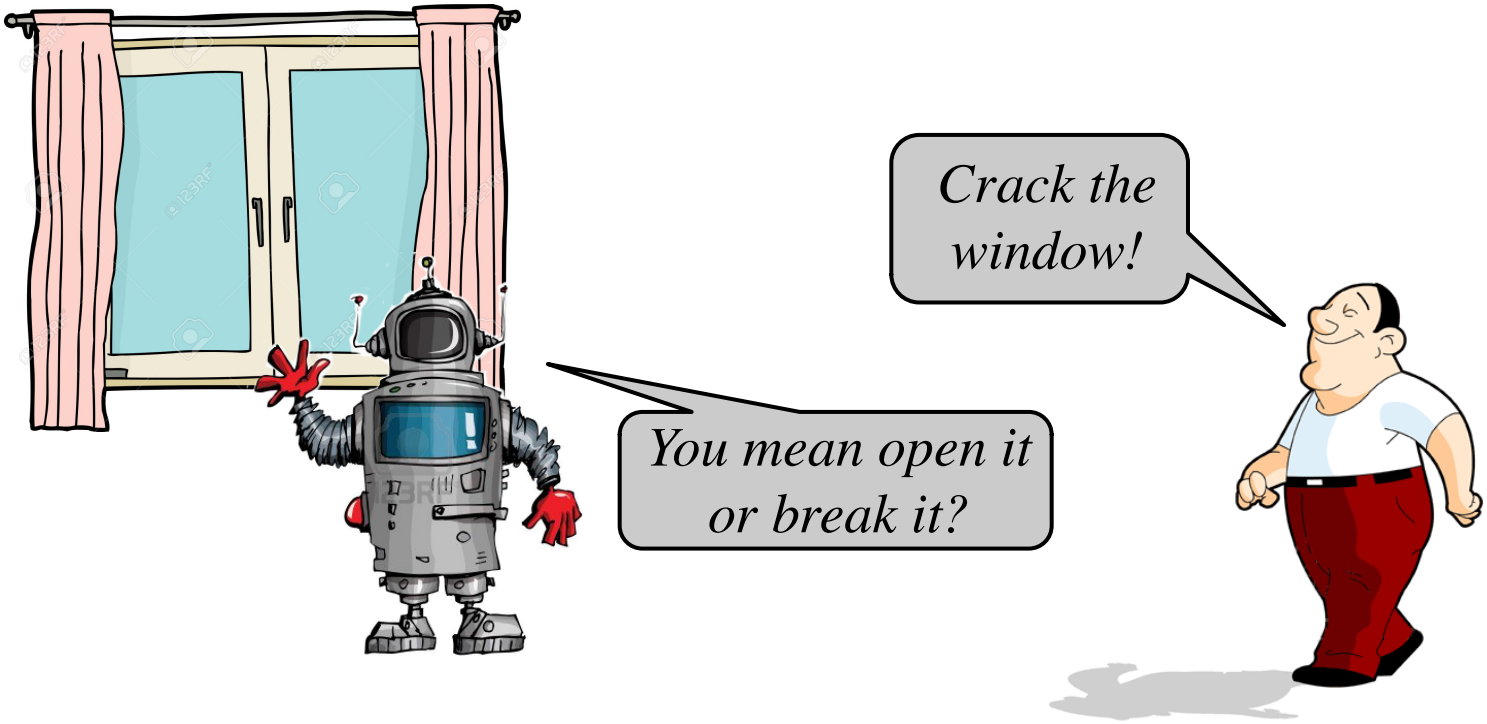
Grounded Language Generation/Dialogue

- Both for answering human questions, and to ask questions back, and for casual chit-chat



Grounded Language Generation/Dialogue

- Both for answering human questions, and to ask questions back, and for casual chit-chat



Distributional Semantics (Word Embeddings)

Distributional Semantics

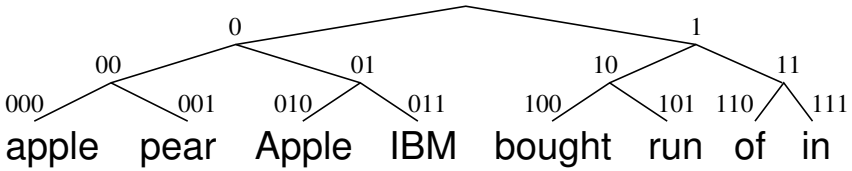
- Words occurring in similar context have similar linguistic behavior (meaning) [Harris, 1954; Firth, 1957]

food → $\begin{bmatrix} 0.6 \\ -0.2 \\ 0.9 \\ 0.3 \\ -0.4 \\ 0.5 \end{bmatrix}$

- Traditional approach: context-counting vectors
  - Count left and right context in window
  - Reweight with PMI or LLR
  - Reduce dimensionality with SVD or NNMF

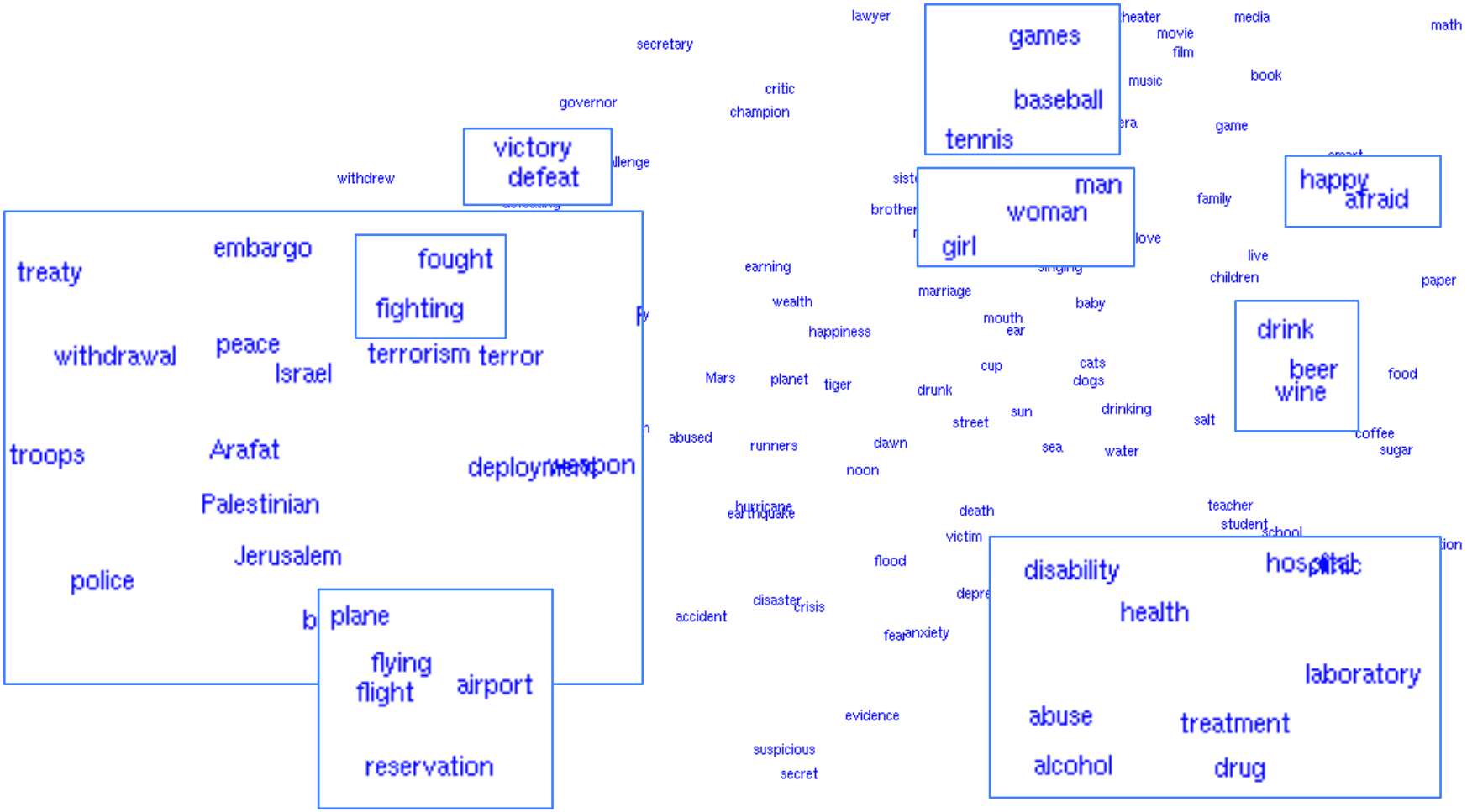
  [Pereira et al., 1993; Lund & Burgess, 1996; Lin, 1998; Lin and Pantel, 2001; Sahlgren, 2006; Pado & Lapata, 2007; Turney and Pantel, 2010; Baroni and Lenci, 2010]

- More word representations: hierarchical clustering based on bigram LM LL
  [Brown et al., 1992]



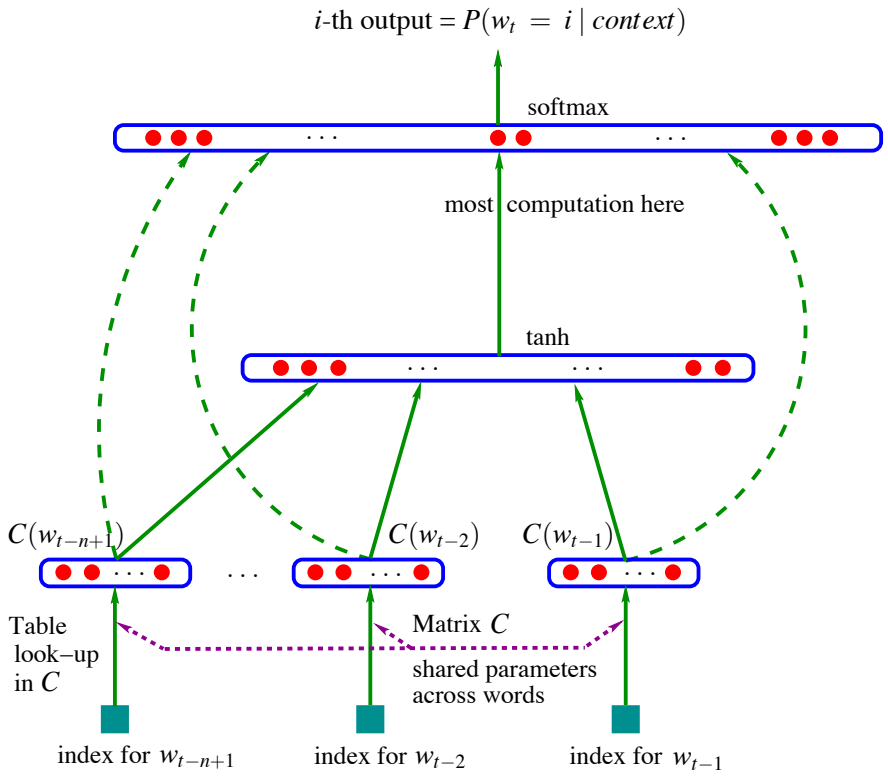
Unsupervised Embeddings

- Vector space representations learned on unlabeled linear context (i.e., left/right words): distributional semantics (Harris, 1954; Firth, 1957)



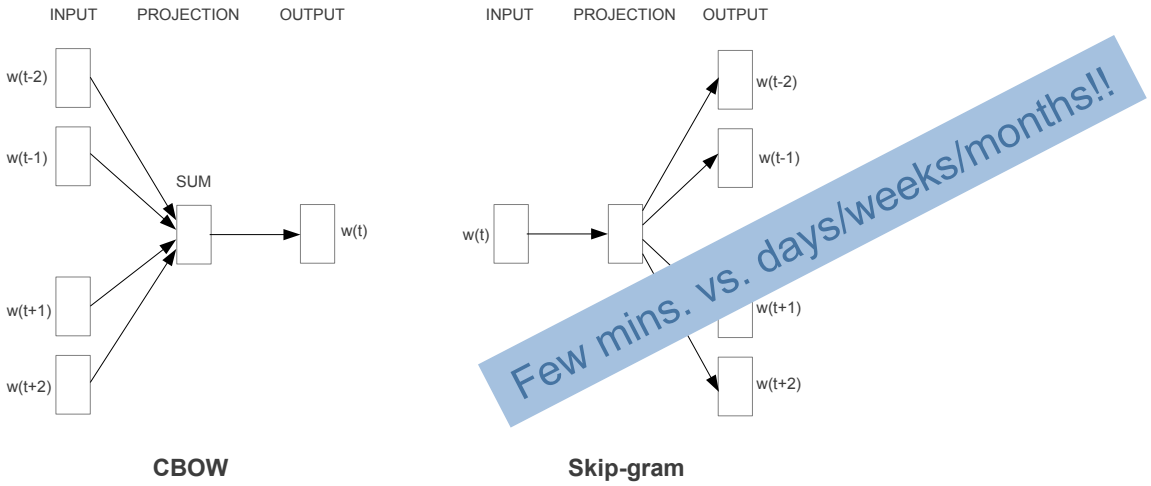
Distributional Semantics -- NNs

- Newer approach: context-predicting vectors (NNs)
  - SENNA [Collobert and Weston, 2008; Collobert et al., 2011]: Multi-layer DNN w/ ranking-loss objective; BoW and sentence-level feature layers, followed by std. NN layers. Similar to [Bengio et al., 2003].



Distributional Semantics -- NNs

- CBOW, SKIP, word2vec [Mikolov et al., 2013]: Simple, super-fast NN w/ no hidden layer. Continuous BoW model predicts word given context, skip-gram model predicts surrounding context words given current word

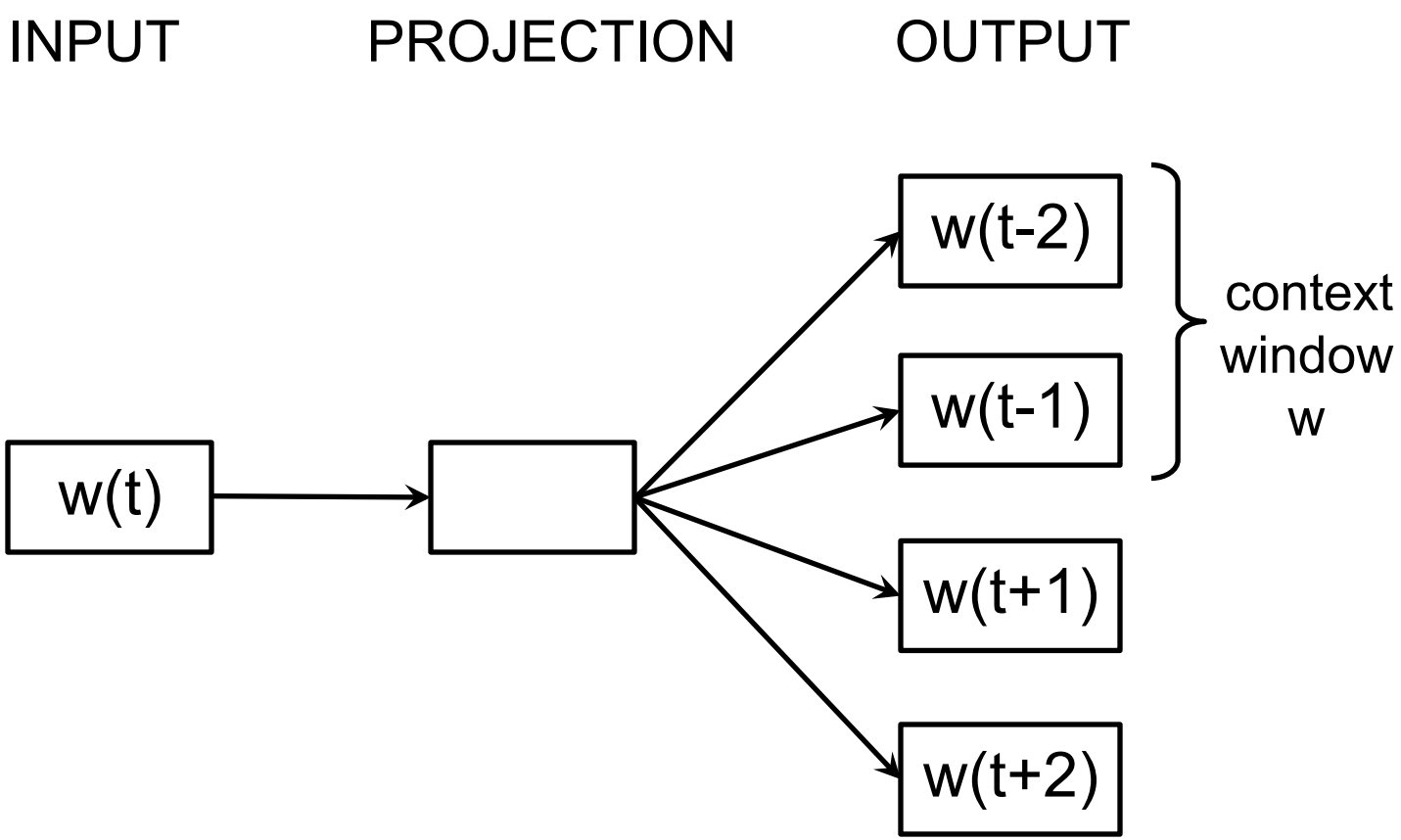


  - Other: [Mnih and Hinton, 2007; Turian et al., 2010]

- Demos: https://code.google.com/p/word2vec, http://metaoptimize.com/projects/wordreprs/, http://ml.nec-labs.com/senna/

Skipgram word2vec

[Mikolov et al., 2013]



Few mins. vs. days/weeks/months!!

Skip-gram word2vec Objective Function

[Mikolov et al., 2013]

- Objective of Skip-gram model is to max. the avg. log probability:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t)$$

- The above conditional probability is defined via the softmax function:

$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left({v'_w}^{\top} v_{w_I}\right)}$$

where v and v′ are the “input” and “output” vector representations of w, and W is the number of words in the vocabulary

Efficient Skip-gram word2vec:

[Mikolov et al., 2013]

- Negative Sampling:

$$\log \sigma({v'_{w_O}}^\top v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \left[\log \sigma(-{v'_{w_i}}^\top v_{w_I}) \right]$$

- I.e., to distinguish the target word w_o from draws from the noise distribution $P_n(w)$ using logistic regression, where there are k negative samples for each data sample.

Efficient Skip-gram word2vec:

[Mikolov et al., 2013]

- Hierarchical Softmax:

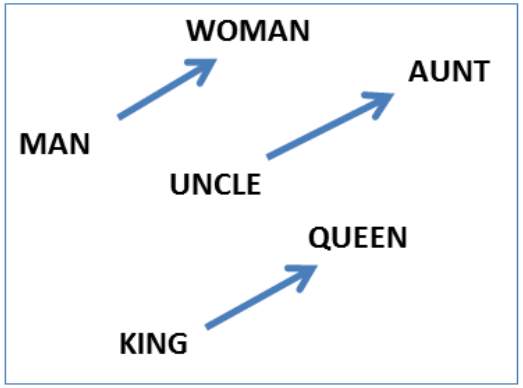
$$p(w|w_I) = \prod_{j=1}^{L(w)-1} \sigma\left([\![n(w,j+1) = \mathrm{ch}(n(w,j))]\!] \cdot {v'_{n(w,j)}}^{\top} v_{w_I}\right)$$

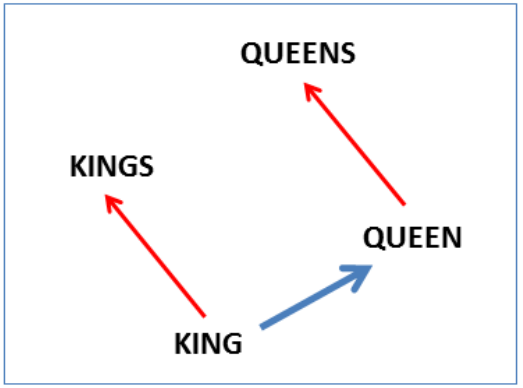
- Instead of evaluating W output nodes in the neural network to obtain the probability distribution, it is needed to evaluate only about $\log_2(W)$ nodes.

- Uses a binary tree representation of the output layer with the W words as its leaves and, for each node, explicitly represents the relative probabilities of its child nodes. These define a random walk that assigns probabilities to words.

Analogy Properties Learned

[Mikolov et al., 2013]





Analogy Properties Learned

[Mikolov et al., 2013]

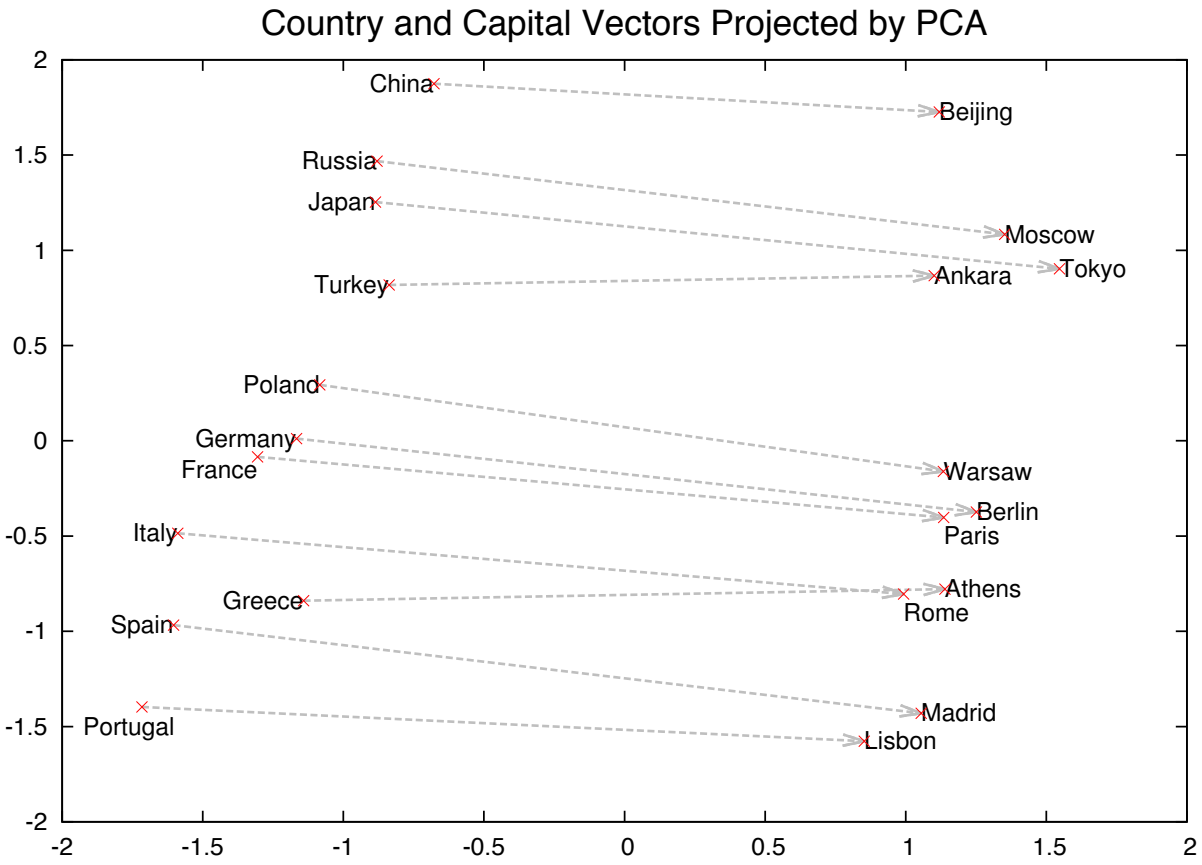


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

Analogy Properties Learned

[Mikolov et al., 2013]

Newspapers			
New York	New York Times	Baltimore	Baltimore Sun
San Jose	San Jose Mercury News	Cincinnati	Cincinnati Enquirer
NHL Teams			
Boston	Boston Bruins	Montreal	Montreal Canadiens
Phoenix	Phoenix Coyotes	Nashville	Nashville Predators
NBA Teams			
Detroit	Detroit Pistons	Toronto	Toronto Raptors
Oakland	Golden State Warriors	Memphis	Memphis Grizzlies
Airlines			
Austria	Austrian Airlines	Spain	Spainair
Belgium	Brussels Airlines	Greece	Aegean Airlines
Company executives			
Steve Ballmer	Microsoft	Larry Page	Google
Samuel J. Palmisano	IBM	Werner Vogels	Amazon

Table 2: Examples of the analogical reasoning task for phrases (the full test set has 3218 examples). The goal is to compute the fourth phrase using the first three. Our best model achieved an accuracy of 72% on this dataset.

Analogy Properties Learned

[Mikolov et al., 2013]

Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

Table 5: Vector compositionality using element-wise addition. Four closest tokens to the sum of two vectors are shown, using the best Skip-gram model.

Other Distributional Semantics Models

- Other approaches: spectral methods, e.g., CCA
  - Word-context correlation [Dhillon et al., 2011, 2012]
  - Multilingual correlation [Faruqui and Dyer, 2014; Lu et al., 2015]
- Multi-sense and contextual embeddings [Reisinger and Mooney, 2010; Neelakantan et al., 2014; Peters et al., 2018]
- Some other ideas: Train task-tailored embeddings to capture specific types of similarity/semantics, e.g.,
  - Dependency context [Bansal et al., 2014, Levy and Goldberg, 2014]
  - Predicate-argument structures [Hashimoto et al., 2014; Madhyastha et al., 2014]
  - Lexicon evidence (PPDB, WordNet, FrameNet) [Xu et al., 2014; Yu and Dredze, 2014; Faruqui et al., 2014; Wieting et al., 2015]
  - Combining advantages of global matrix factorization and local context window methods [GloVe; Pennington et al., 2014]

Multi-sense/Contextual Embeddings

- Different vectors for each sense of a word

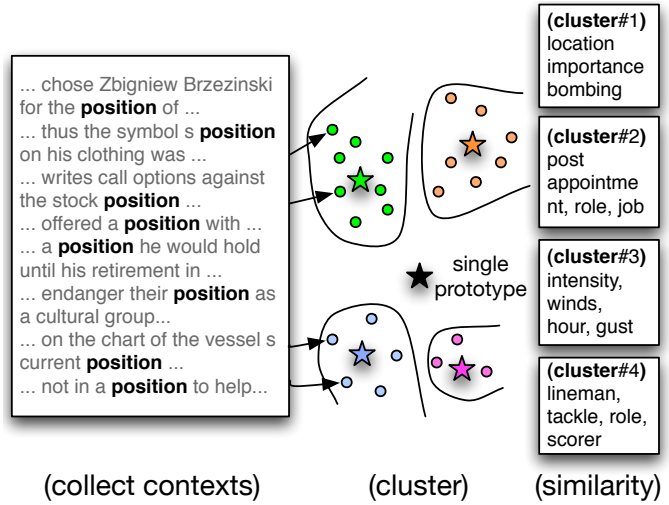


Figure 1: Overview of the multi-prototype approach to near-synonym discovery for a single target word independent of context. Occurrences are clustered and cluster centroids are used as prototype vectors. Note the "hurricane" sense of *position* (cluster 3) is not typically considered appropriate in WSD.

[Reisinger and Mooney, 2010]

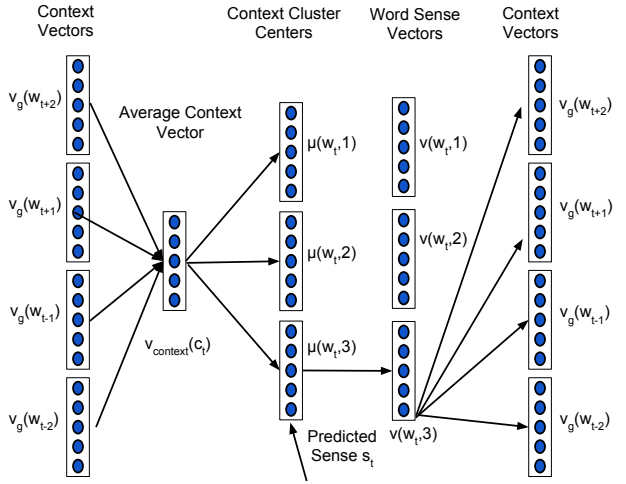


Figure 2: Architecture of Multi-Sense Skip-gram (MSSG) model with window size $R_t = 2$ and $K = 3$. Context c_t of word w_t consists of $w_{t-1}, w_{t-2}, w_{t+1}, w_{t+2}$. The sense is predicted by finding the cluster center of the context that is closest to the average of the context vectors.

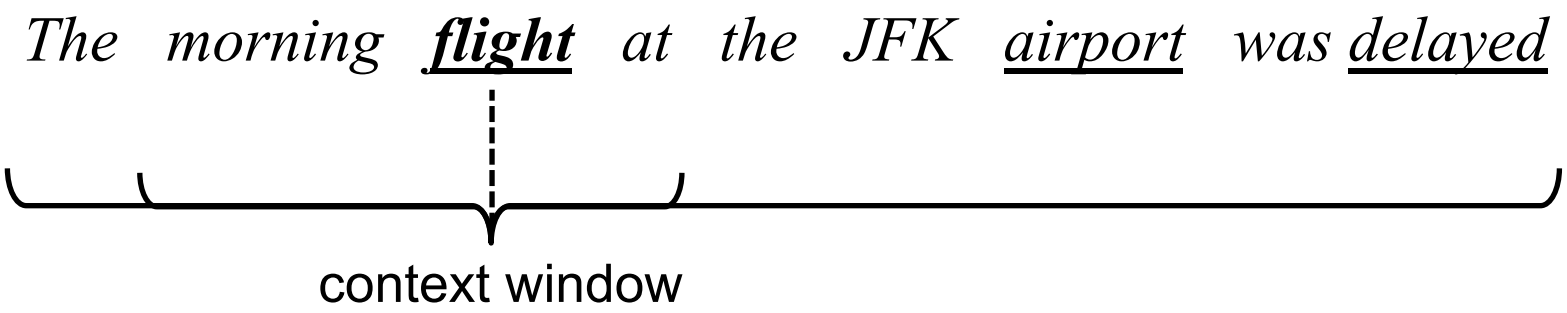
[Neelakantan et al., 2014]

- Recent: ELMo (learned functions of the internal states of a deep bidirectional language model, which is pretrained on a large text corpus) [Peters et al., 2018]

Syntactically Tailored Embeddings

[Bansal et al., 2014]

- Context window size (SKIP)
  - Smaller window → syntactic/functional similarity
  - Larger window → topical similarity



- Similar effect in distributional representations (Lin and Wu, 2009)

Multilingual Embeddings via CCA

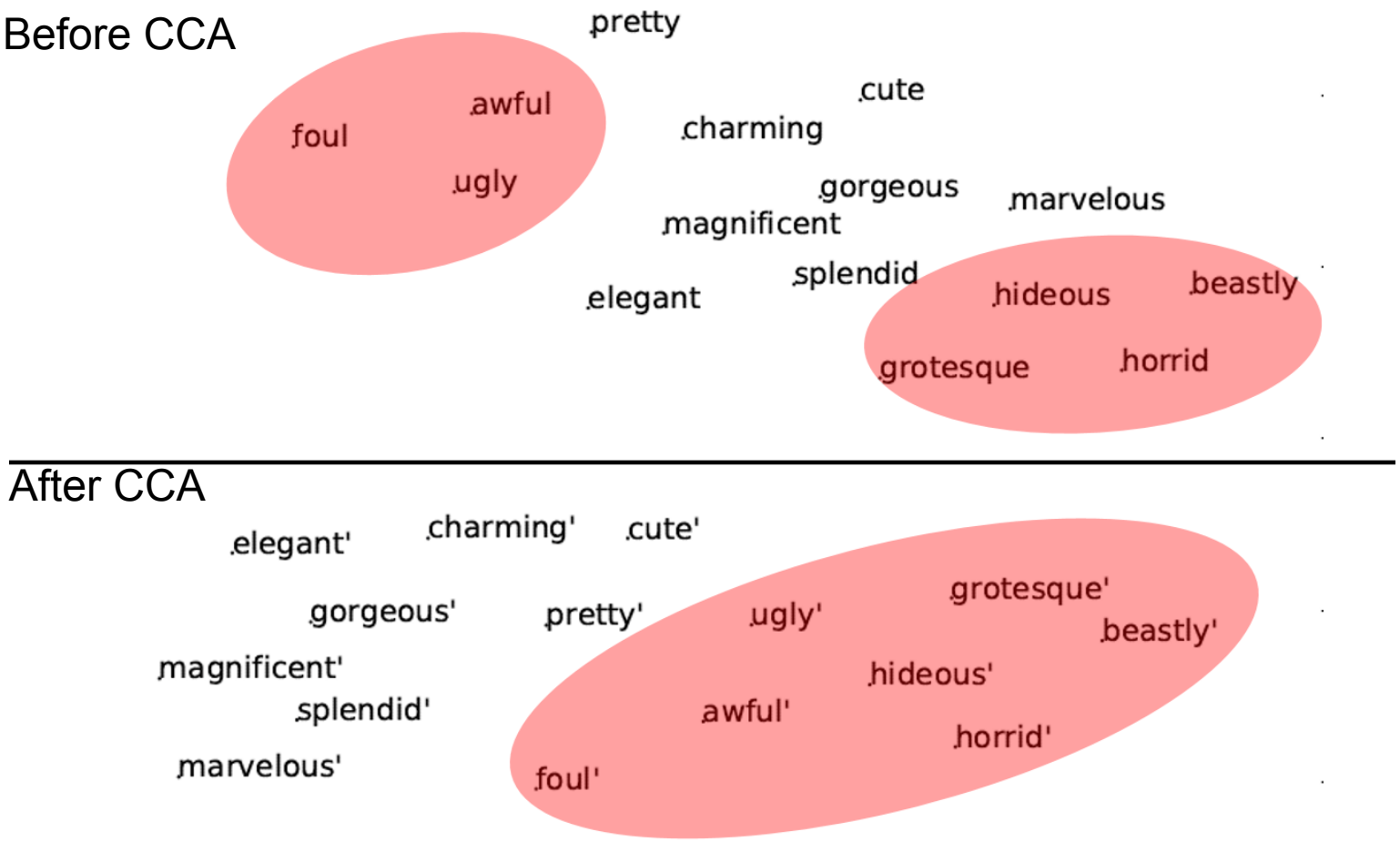
- Translational context (say, English ←→ German) can help learn stronger embeddings, e.g., separate antonyms vs. synonyms

- CCA on translation pairs to map them to shared space

$$\max_{\mathbf{u}\in\mathbb{R}^{D_x},\mathbf{v}\in\mathbb{R}^{D_y}} \frac{\mathbb{E}\left[(\mathbf{u}^\top\mathbf{x})(\mathbf{v}^\top\mathbf{y})\right]}{\sqrt{\mathbb{E}\left[(\mathbf{u}^\top\mathbf{x})^2\right]}\sqrt{\mathbb{E}\left[(\mathbf{v}^\top\mathbf{y})^2\right]}} = \frac{\mathbf{u}^\top\boldsymbol{\Sigma}_{xy}\mathbf{v}}{\sqrt{\mathbf{u}^\top\boldsymbol{\Sigma}_{xx}\mathbf{u}}\sqrt{\mathbf{v}^\top\boldsymbol{\Sigma}_{yy}\mathbf{v}}}$$

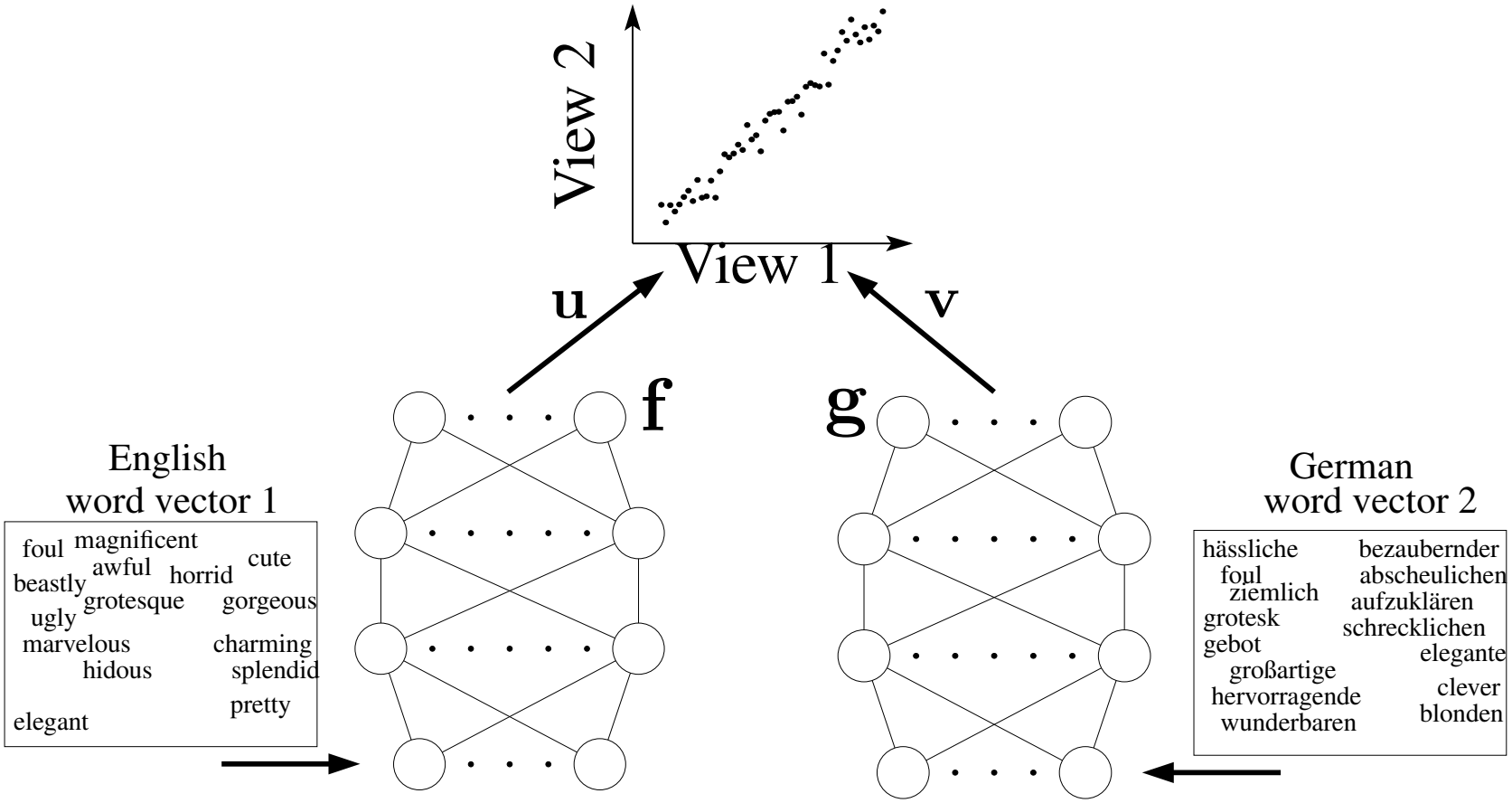
[Faruqui and Dyer, 2014]

Multi-view Embeddings via CCA



[Faruqui and Dyer, 2014]

Deep-CCA



[Lu, Wang, Bansal, Gimpel, Livescu, 2015]

Retrofitting Word Embeddings to Lexicons

- We want the inferred word vector to be close to the observed value $\hat{q}$ and close to its neighbors q_j, $\forall j$ such that $(i, j) \in E$, where E is the set of relations in a dictionary/lexicon (e.g., WordNet, PPDB, etc.)

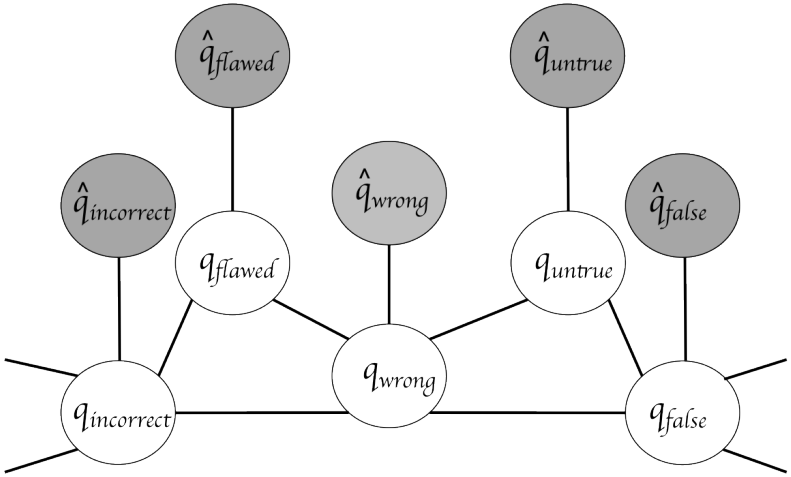


Figure 1: Word graph with edges between related words showing the observed (grey) and the inferred (white) word vector representations.

$$\Psi(Q) = \sum_{i=1}^{n} \left[\alpha_i \|q_i - \hat{q}_i\|^2 + \sum_{(i,j) \in E} \beta_{ij} \|q_i - q_j\|^2 \right]$$

[Faruqui et al., 2015]

Bias in Word Embeddings

Extreme *she*	Extreme *he*
1. homemaker	1. maestro
2. nurse	2. skipper
3. receptionist	3. protege
4. librarian	4. philosopher
5. socialite	5. captain
6. hairdresser	6. architect
7. nanny	7. financier
8. bookkeeper	8. warrior
9. stylist	9. broadcaster
10. housekeeper	10. magician

Gender stereotype *she-he* analogies

sewing-carpentry	registered nurse-physician	housewife-shopkeeper
nurse-surgeon	interior designer-architect	softball-baseball
blond-burly	feminism-conservatism	cosmetics-pharmaceuticals
giggle-chuckle	vocalist-guitarist	petite-lanky
sassy-snappy	diva-superstar	charming-affable
volleyball-football	cupcakes-pizzas	lovely-brilliant

Gender appropriate *she-he* analogies

queen-king	sister-brother	mother-father
waitress-waiter	ovarian cancer-prostate cancer	convent-monastery

Figure 1: **Left** The most extreme occupations as projected on to the *she*−*he* gender direction on w2vNEWS. Occupations such as *businesswoman*, where gender is suggested by the orthography, were excluded. **Right** Automatically generated analogies for the pair *she-he* using the procedure described in text. Each automatically generated analogy is evaluated by 10 crowd-workers to whether or not it reflects gender stereotype.

- Debiasing word embeddings via identifying pairs (sets) of words to correct/neutralize, identify bias direction (subspace), and then debias via neutralize+equalize or soften algorithms.

[Bolukbasi et al., 2016]

Task-Trained Embeddings

[Chen and Manning, 2014; CS224n]

- Can also directly train word embeddings on the task, via back-prop from the task supervision (XE errors), e.g., dependency parsing:

Softmax probabilities

Output layer y

$y = \text{softmax}(Uh + b_2)$

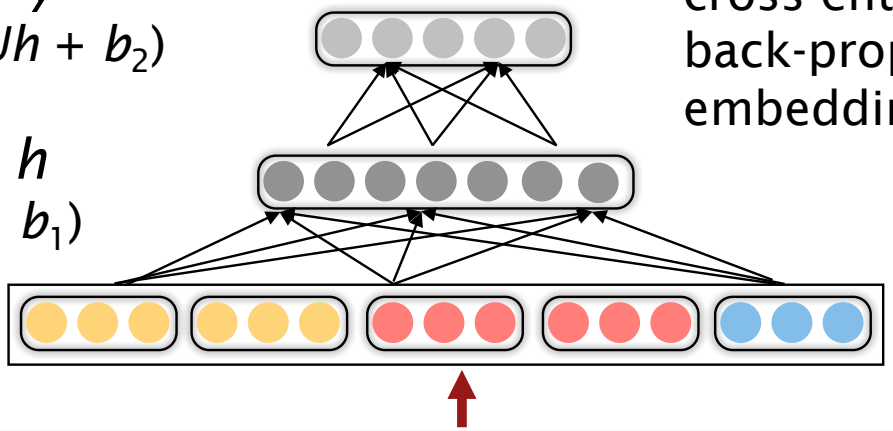
Hidden layer h

$h = \text{ReLU}(Wx + b_1)$

Input layer x

lookup + concat

cross-entropy error will be back-propagated to the embeddings.



Compositional Semantics

Compositional Semantics with NNs

- Composing, combining word vectors to representations for longer units: phrases, sentences, paragraphs, …
- Initial approaches: point-wise sum, multiplication
  [Mitchell and Lapata, 2010; Blacoe and Lapata, 2012]
- Vector-matrix compositionality [Baroni and Zamparelli, 2010; Zanzotto et al., 2010; Grefenstette and Sadrzadeh, 2011; Socher et al., 2011; Yessenalina and Cardie, 2011]
- Linguistic information added via say parses in RvNNs
  [Socher et al., 2011b, 2012, 2013a, 2013b, 2014; Hermann and Blunsom, 2013]
- Sequential RNNs (with GRU/LSTM gates)

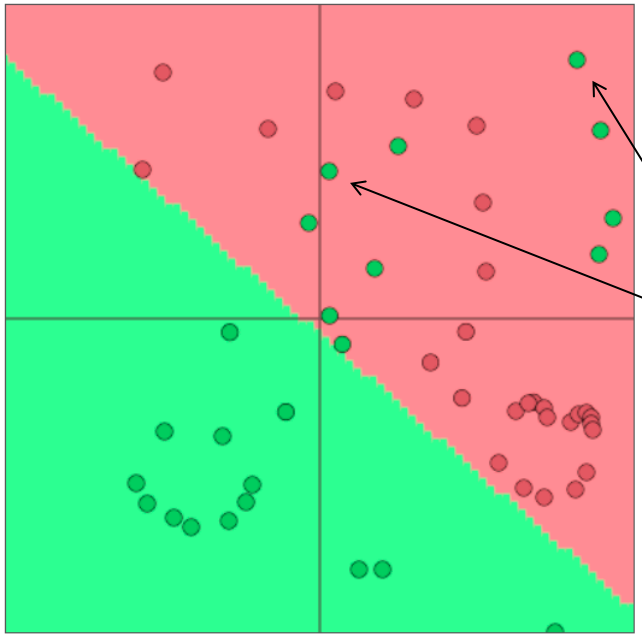
  (Simple vector averaging w/ updating sometimes competitive)

Compositional Semantics with NNs

- Feed-forward NNs with back-propagation

Softmax (= logistic regression) is not very powerful

- Softmax only linear decision boundaries



→ Lame when problem is complex

Wouldn't it be cool to get these correct?

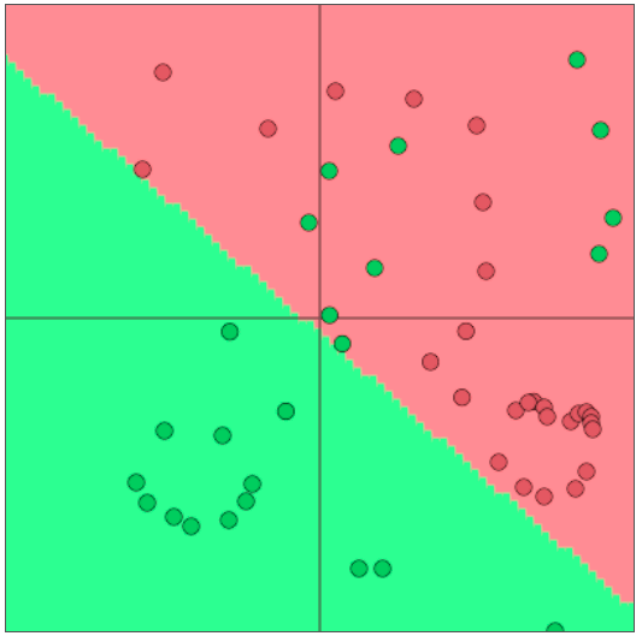
NN and backprop slides from CS224d – Richard Socher

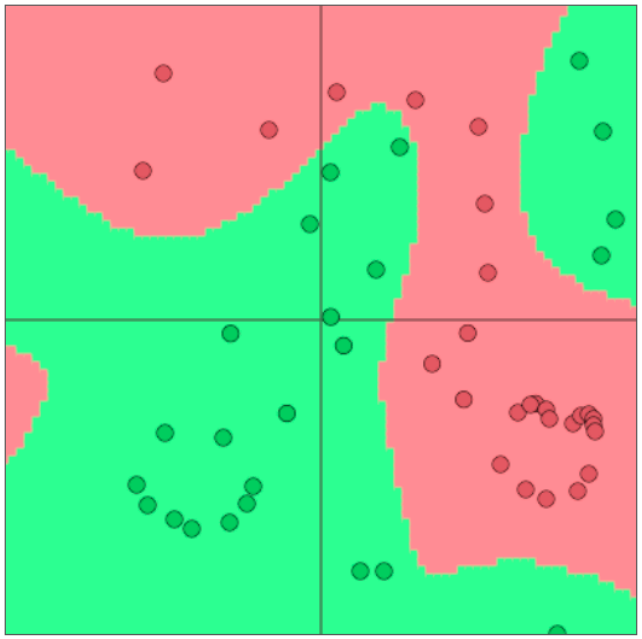
Compositional Semantics with NNs

- Feed-forward NNs with back-propagation

Neural Nets for the Win!

- Neural networks can learn much more complex functions and nonlinear decision boundaries!





Compositional Semantics with NNs

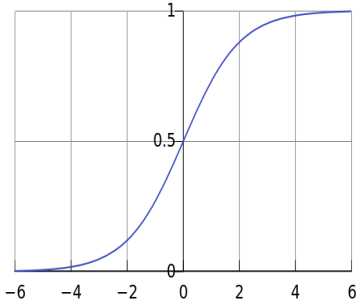
- Feed-forward NNs with back-propagation

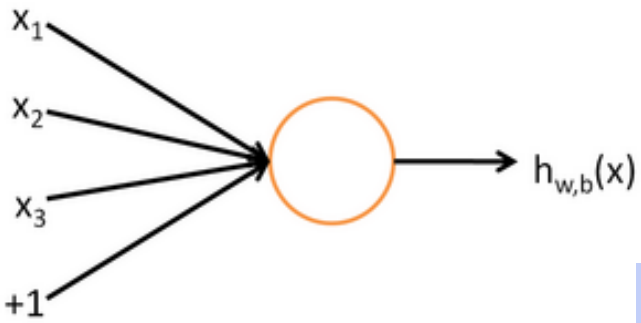
A neuron is essentially a binary logistic regression unit

$$h_{w,b}(x) = f(w^{\mathsf{T}}x + b)$$

$$f(z) = \frac{1}{1+e^{-z}}$$

b: We can have an "always on" feature, which gives a class prior, or separate it out, as a bias term





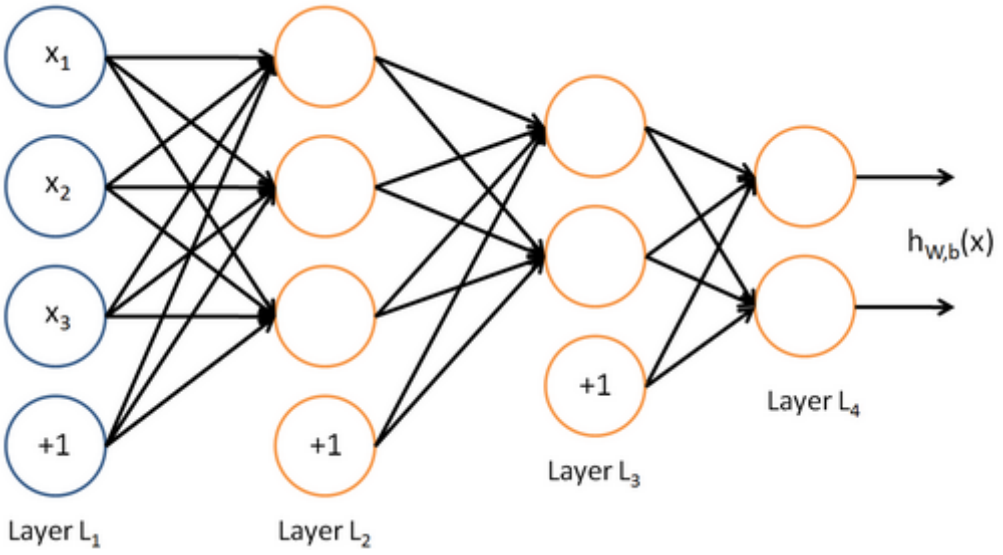
w, *b* are the parameters of this neuron i.e., this logistic regression model

Compositional Semantics with NNs

- Feed-forward NNs with back-propagation

A neural network
= running several logistic regressions at the same time

Before we know it, we have a multilayer neural network….



Compositional Semantics with NNs

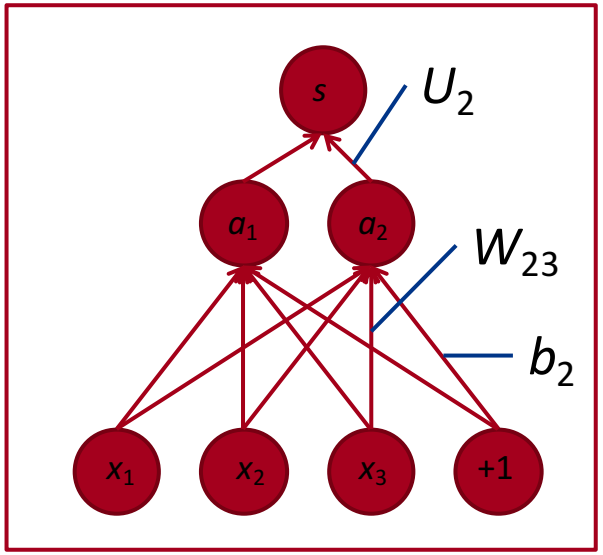
- Feed-forward NNs with back-propagation

Training with Backpropagation

- Let's consider the derivative of a single weight W_{ij}

$$\frac{\partial s}{\partial W} = \frac{\partial}{\partial W} U^T a = \frac{\partial}{\partial W} U^T f(z) = \frac{\partial}{\partial W} U^T f(Wx + b)$$

- This only appears inside a_i
- For example: W_{23} is only used to compute a_2



Compositional Semantics with NNs

- Feed-forward NNs with back-propagation

Training with Backpropagation

$$\frac{\partial s}{\partial W} = \frac{\partial}{\partial W} U^T a = \frac{\partial}{\partial W} U^T f(z) = \frac{\partial}{\partial W} U^T f(Wx + b)$$

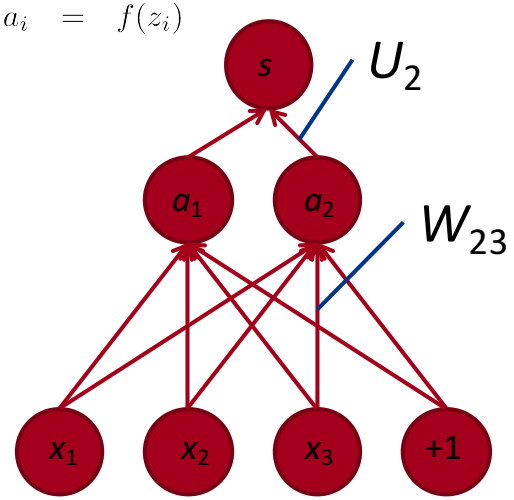
Derivative of weight W_{ij}:

$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial u} \frac{\partial u}{\partial x}$$

$$\frac{\partial}{\partial W_{ij}} U^T a \quad \rightarrow \quad \frac{\partial}{\partial W_{ij}} U_i a_i$$

$$\begin{aligned} U_i \frac{\partial}{\partial W_{ij}} a_i &= U_i \frac{\partial a_i}{\partial z_i} \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i \frac{\partial f(z_i)}{\partial z_i} \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i f'(z_i) \frac{\partial z_i}{\partial W_{ij}} \\ &= U_i f'(z_i) \frac{\partial W_i . x + b_i}{\partial W_{ij}} \end{aligned}$$

$$\begin{aligned} z_i &= W_i . x + b_i = \sum_{j=1}^{3} W_{ij} x_j + b_i \\ a_i &= f(z_i) \end{aligned}$$



Compositional Semantics with NNs

- Feed-forward NNs with back-propagation

Training with Backpropagation

Derivative of single weight W_{ij} :

$$z_i = W_i.x + b_i = \sum_{j=1}^{3} W_{ij}x_j + b_i$$

$$a_i = f(z_i)$$

$$U_i \frac{\partial}{\partial W_{ij}} a_i = U_i f'(z_i) \frac{\partial W_i.x + b_i}{\partial W_{ij}}$$

$$= U_i f'(z_i) \frac{\partial}{\partial W_{ij}} \sum_k W_{ik}x_k$$

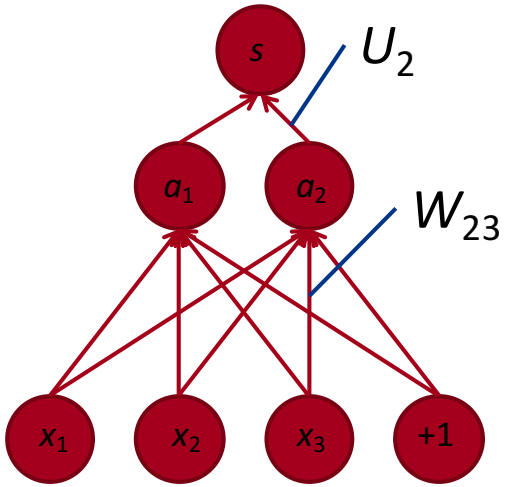
$$= \underbrace{U_i f'(z_i)}\, x_j$$

$$= \delta_i \; x_j$$

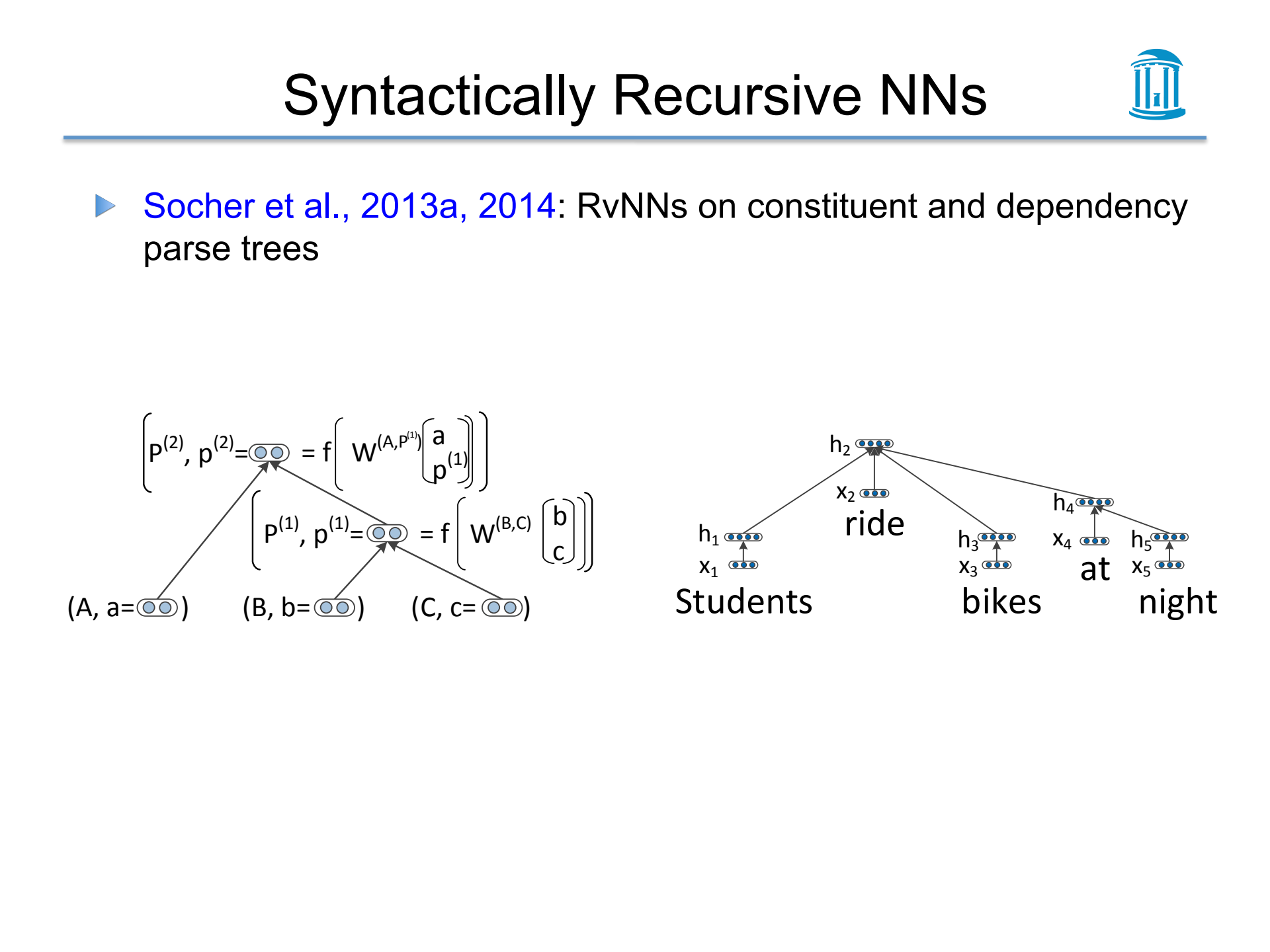
Local error signal

Local input signal

where $f'(z) = f(z)(1 - f(z))$ for logistic f



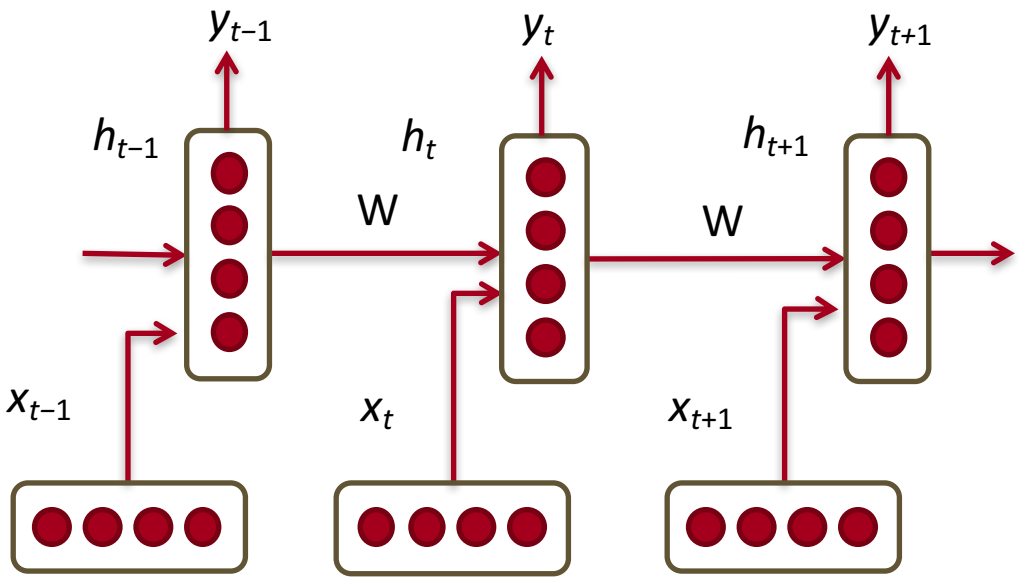
Syntactically Recursive NNs



- Socher et al., 2013a, 2014: RvNNs on constituent and dependency parse trees

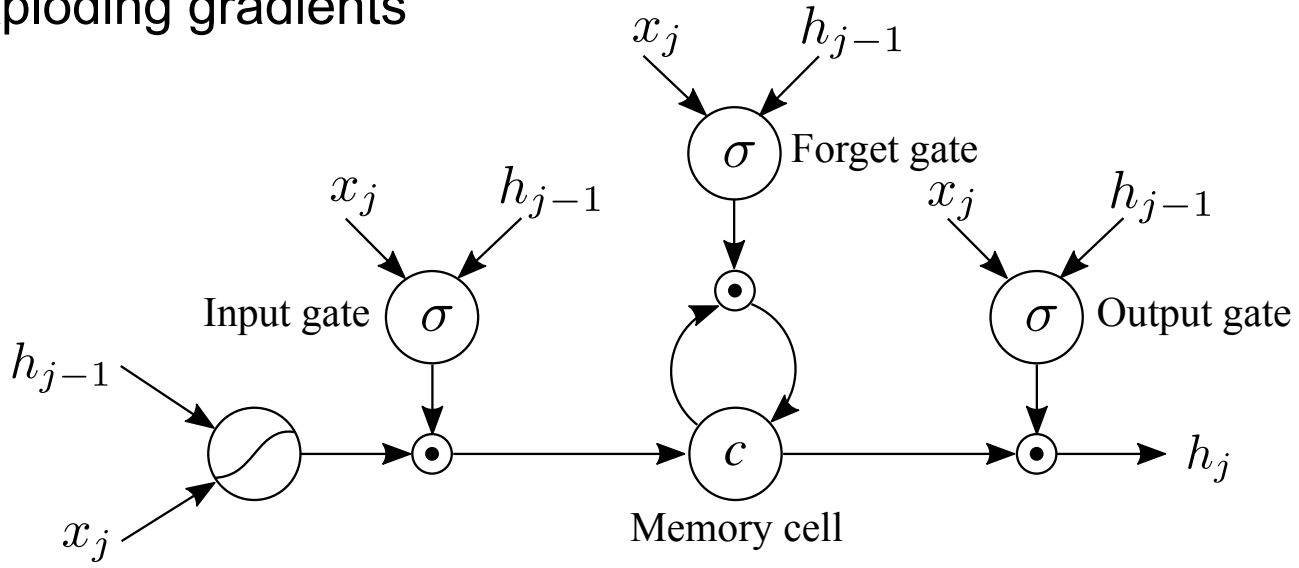
Recurrent NNs

- Recurrent NNs (RNNs) are non-tree, sequential versions of recursive RvNNs
- Weights tied together for each time step
- Loss function on identity of predicted word at each time step



LSTM RNNs

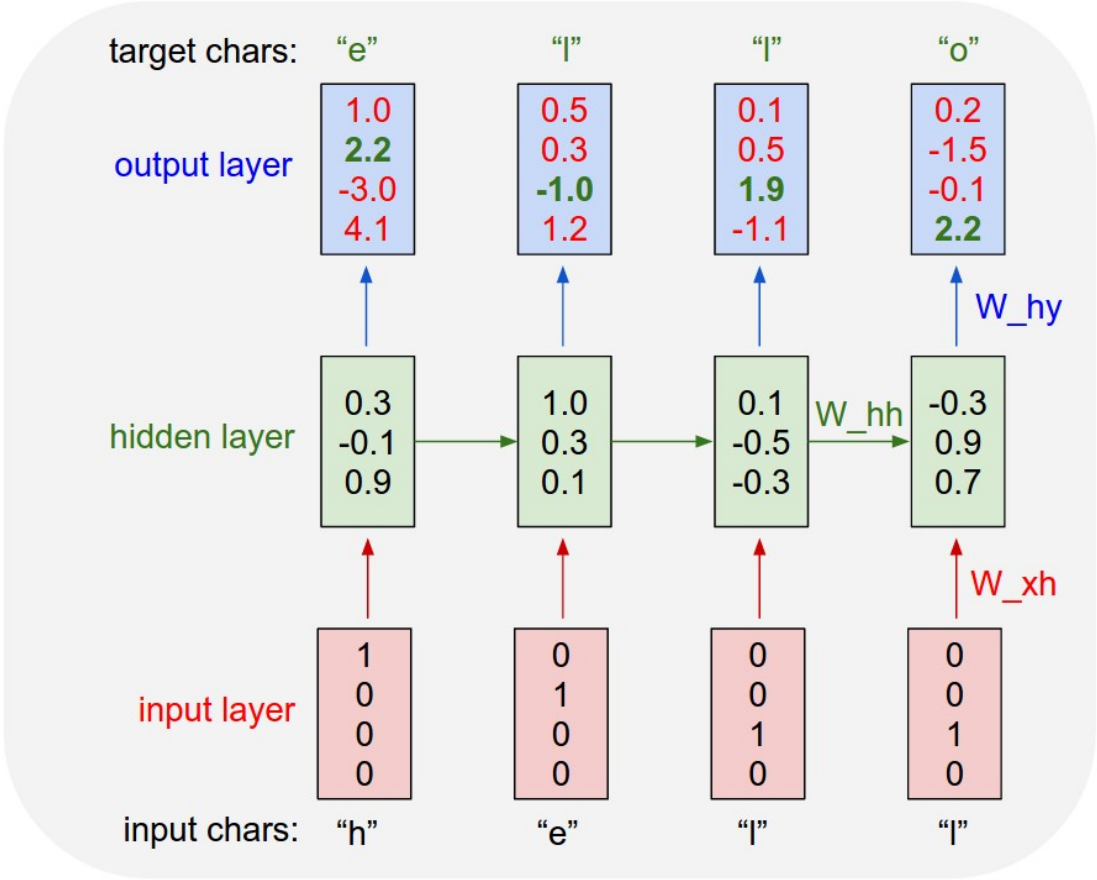
- LSTM (Long short term memory) RNNs have gates for forgetting, allowing learning of longer-term connections by avoiding vanishing/ exploding gradients



$$
\begin{aligned}
i_t &= \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right) \\
f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \\
c_t &= f_t c_{t-1} + i_t \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \\
o_t &= \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right) \\
h_t &= o_t \tanh(c_t)
\end{aligned}
$$

Character RNNs

- Can directly process each character as a unit!
- Helps learn prefixes, stems, suffixes (form vs. function, rare/ unseen words, etc.)



http://karpathy.github.io/2015/05/21/rnn-effectiveness/

RNN Generations

- Automatically generate Shakespeare from RNNs!

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.
```

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

RNN Generations

- Automatically generate Wikipedia-style text from RNNs!

```
Naturalism and decision for the majority of Arab countries' capitalide was grounded
by the Irish language by [[John Clair]], [[An Imperial Japanese Revolt]], associated
with Guangzham's sovereignty. His generals were the powerful ruler of the Portugal
in the [[Protestant Immineners]], which could be said to be directly in Cantonese
Communication, which followed a ceremony and set inspired prison, training. The
emperor travelled back to [[Antioch, Perth, October 25|21]] to note, the Kingdom
of Costa Rica, unsuccessful fashioned the [[Thrales]], [[Cynth's Dajoard]], known
in western [[Scotland]], near Italy to the conquest of India with the conflict.
Copyright was the succession of independence in the slop of Syrian influence that
was a famous German movement based on a more popular servicious, non-doctrinal
and sexual power post. Many governments recognize the military housing of the
[[Civil Liberalization and Infantry Resolution 265 National Party in Hungary]],
that is sympathetic to be to the [[Punjab Resolution]]
(PJS)[http://www.humah.yahoo.com/guardian.
cfm/7754800786d17551963s89.htm Official economics Adjoint for the Nazism, Montgomery
was swear to advance to the resources for those Socialism's rule,
was starting to signing a major tripad of aid exile.]]
```

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

RNN Generations

- Automatically generate source code from RNNs!

```
{ { cite journal | id=Cerling Nonforest Department|format=Newlymeslated|none } }
''www.e-complete''.

'''See also''': [[List of ethical consent processing]]

== See also ==
*[[Iender dome of the ED]]
*[[Anti-autism]]

===[[Religion|Religion]]===
*[[French Writings]]
*[[Maria]]
*[[Revelation]]
*[[Mount Agamul]]

== External links==
* [http://www.biblegateway.nih.gov/entrepre/ Website of the World Festival. The labour

==External links==
* [http://www.romanology.com/ Constitution of the Netherlands and Hispanic Competition
```

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Various Applications of such RNNs

- Classification: Sentiment Analysis (discussed next)
- Language Modeling and Language Generation
- Conditioned Generation: End-to-end MT, Summarization
- Others: Parsing, Captioning, Q&A, Dialogue (some will be covered in future slides)

Supervised Sentence Embeddings

Supervised Sentence Embedding Models

- Just like word embeddings were supervised using lexicons, dictionaries, taxonomies (WordNet) etc., sentence embeddings also benefit greatly from supervision!

- 2 examples: supervision based on bidirectional sentence similarity (paraphrases) or directed similarity (entailment vs contradiction vs neutral)

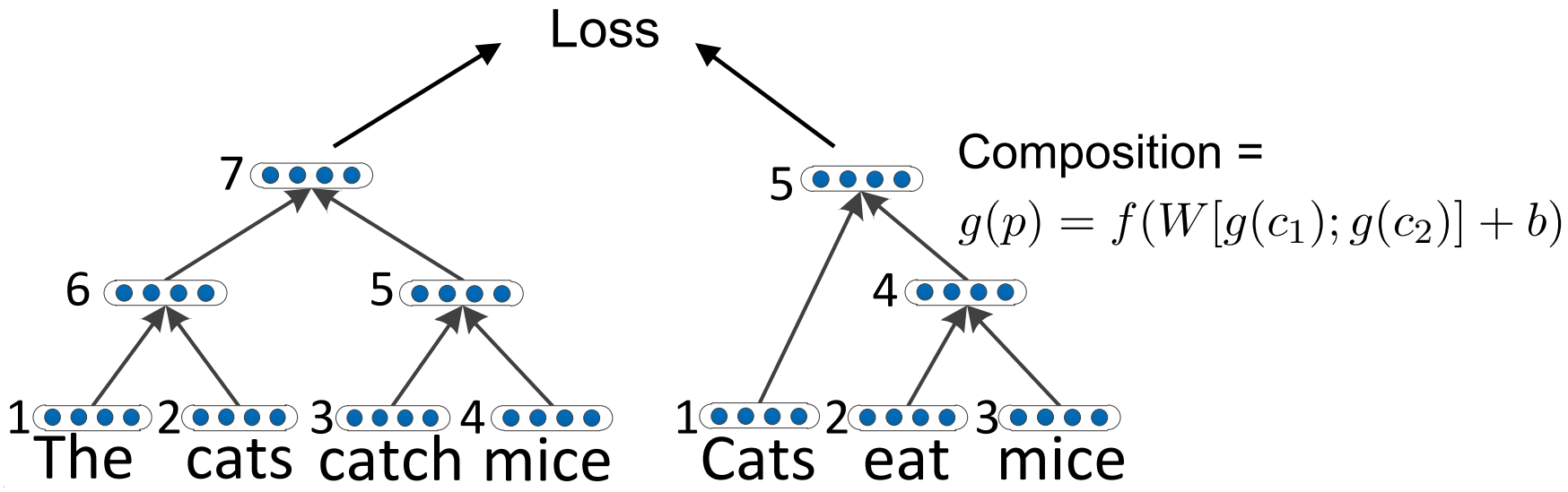
Paraphrase-based Sentence Embeddings

- Phrases that mean the same, are replaceable in context

main reason why	|||	*principal reason for*
informed about the outcome	|||	*notified of the results*
with particular emphasis	|||	*with specific focus*
we 'll have a good time	|||	*we 're gonna have fun*
50 years ago	|||	*five decades ago*
that , according to	|||	*which , in accordance with*
program is aimed at	|||	*programme aims to*
are under the obligation	|||	*have a duty*
a critical component	|||	*an essential element*

Paraphrase Model

- 2 parse-based RvNNs with a hinge-based loss function



[Socher et al., 2011]

Paraphrase Model

- Loss: +ve pairs closer than -ve pairs with margin δ

Positive training pairs

Negative training pairs

$$\min_{W,b,W_w} \frac{1}{|X|} \left(\sum_{\langle x_1, x_2 \rangle \in X} \max(0, \delta - g(x_1) \cdot g(x_2) + g(x_1) \cdot g(t_1)) \right.$$

$$\left. + \max(0, \delta - g(x_1) \cdot g(x_2) + g(x_2) \cdot g(t_2)) \right)$$

Regularization terms

[Wieting, Bansal, Gimpel, Livescu, Roth, 2015]

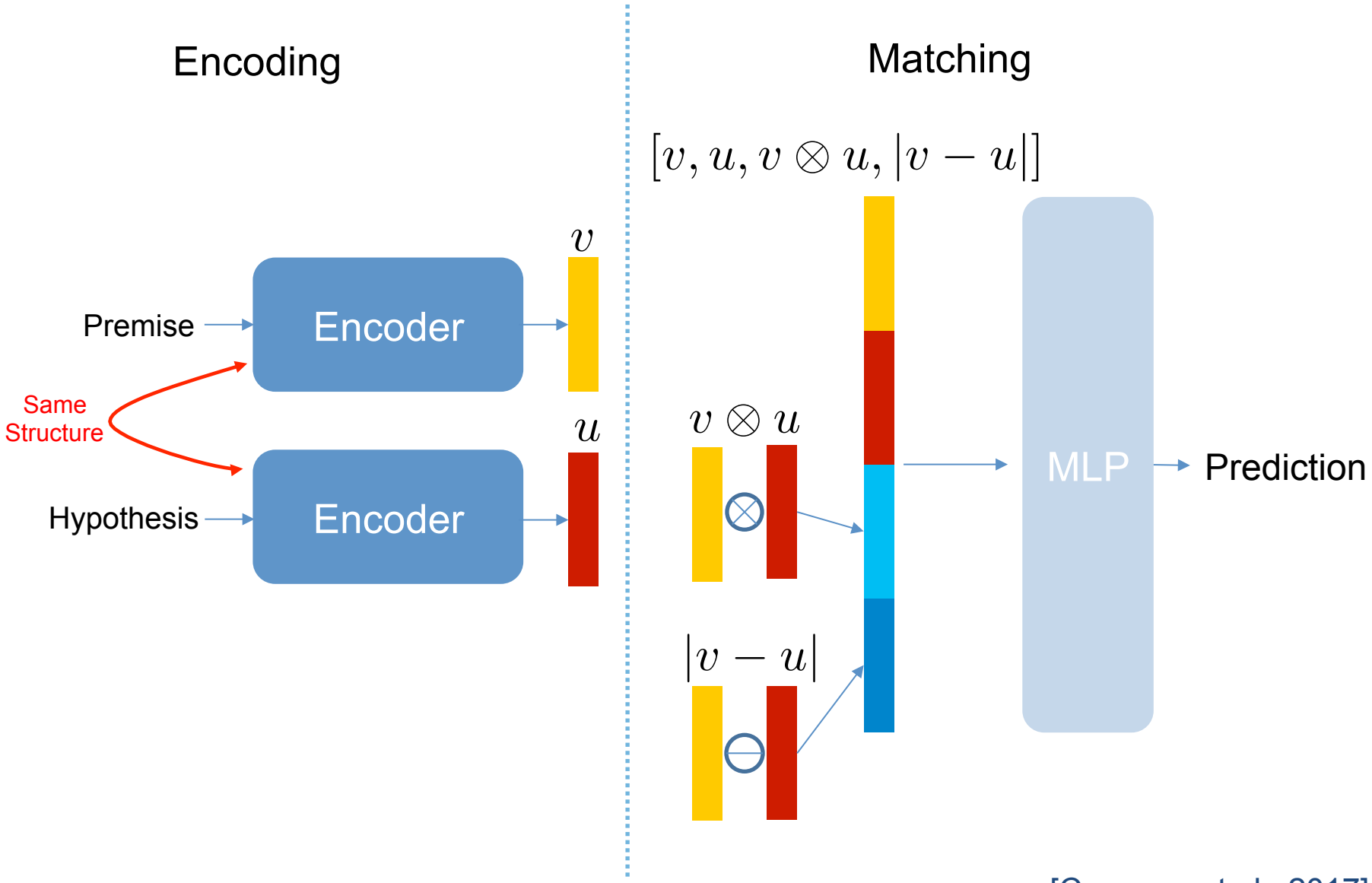
Entailment-based Embeddings

- SNLI and Multi-NLI corpora with sentence pairs of 3 relationships: entailment, contradiction, neutral/unrelated

Premise	Label	Hypothesis	Genre
The Old One always comforted Ca'daan, except today.	*neutral*	Ca'daan knew the Old One very well.	*Fiction*
Your gift is appreciated by each and every student who will benefit from your generosity.	*neutral*	Hundreds of students will benefit from your generosity.	*Letters*
yes now you know if if everybody like in August when everybody's on vacation or something we can dress a little more casual or	*contradiction*	August is a black out month for vacations in the company.	*Telephone Speech*
At the other end of Pennsylvania Avenue, people began to line up for a White House tour.	*entailment*	People formed a line at the end of Pennsylvania Avenue.	*9/11 Report*
A black race car starts up in front of a crowd of people.	*contradiction*	A man is driving down a lonely road.	*SNLI*

[Bowman et al., 2015; Williams et al., 2017]

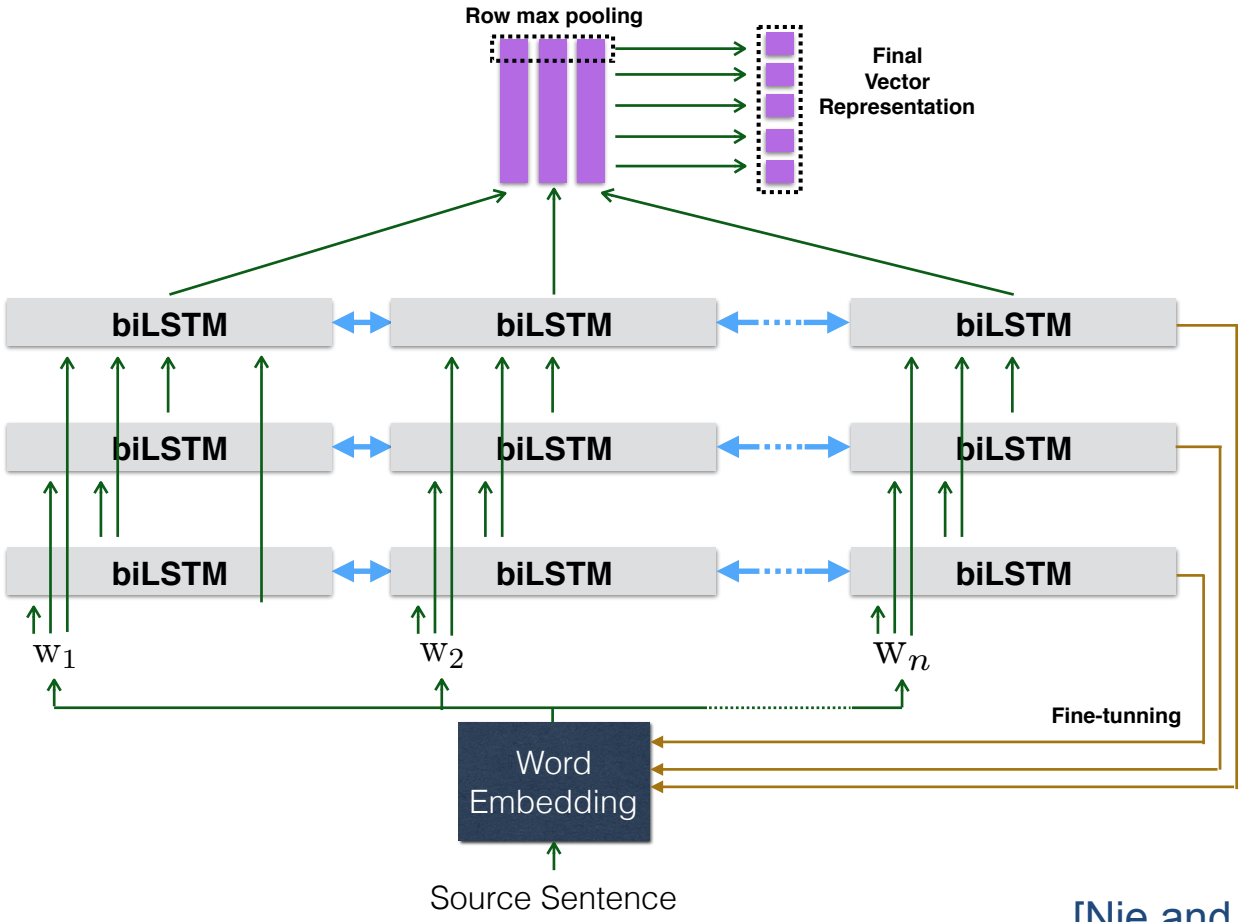
Entailment-based Embeddings



[Conneau et al., 2017]

Entailment-based Embeddings

- Improved Encoders: e.g., via shortcut-stacked RNNs (to help learn higher-level semantic features and to help sparse gradients from max-pooling to flow to lower layers



[Nie and Bansal, 2017]

Classification Tasks

Classification Tasks: Sentiment Analysis

Sentiment Analysis with Python NLTK Text Classification

This is a demonstration of **sentiment analysis** using a `NLTK 2.0.4` powered **text classification** process. It can tell you whether it thinks the text you enter below expresses **positive sentiment**, **negative sentiment**, or if it's **neutral**. Using **hierarchical classification**, *neutrality* is determined first, and *sentiment polarity* is determined second, but only if the text is not neutral.

Analyze Sentiment

Language

english

Enter text

It always amazes me how Universal never cares to create anything remotely clever when it comes to their animations, and so once again they come up with a harmless little story that wants to be cute and funny (which it is sometimes) but is only bound to be quickly forgotten.

Enter up to 50000 characters

Analyze

Sentiment Analysis Results

The text is **neg**.

The final sentiment is determined by looking at the classification probabilities below.

Subjectivity

- neutral: 0.3
- **polar: 0.7**

Polarity

- pos: 0.2
- **neg: 0.8**

Sentiment Analysis

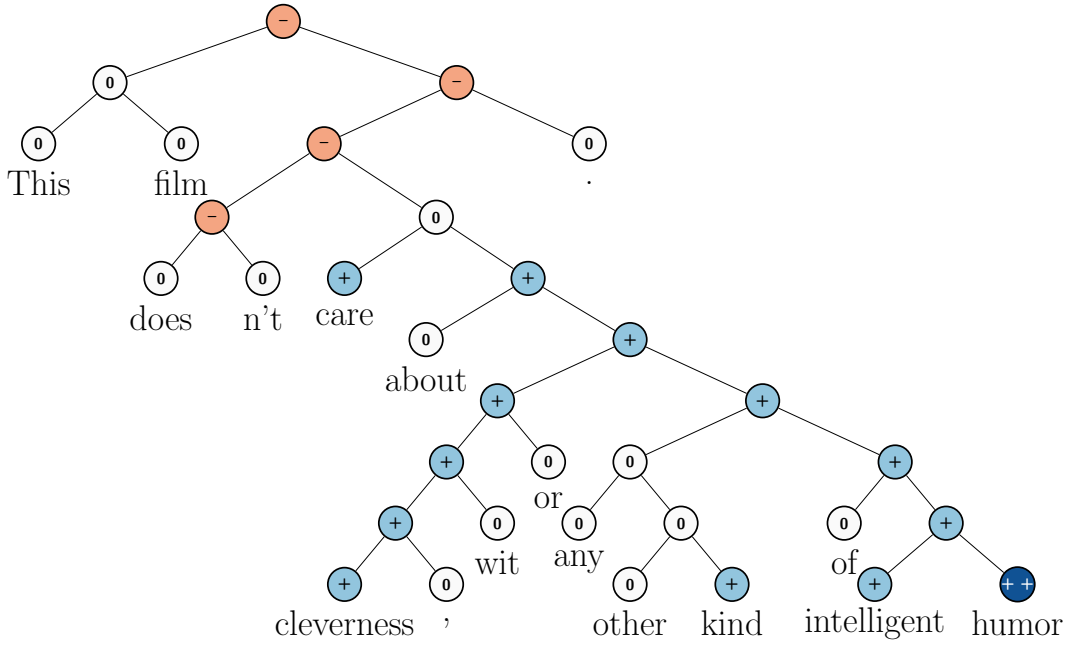
- Earlier methods used bag of words, e.g., lexicons of positive and negative words and phrases
- Cannot distinguish tricky cases like:

\+ *white blood cells destroying an infection*
– *an infection destroying white blood cells*

\+ *There are slow and repetitive parts but it has just enough spice to keep it interesting.*
– *Stealing Harvard doesn't care about cleverness, wit or any other kind of intelligent humor.*

Sentiment Analysis

- Even simpler issues like negation hard to understand
- Socher et al., 2013b present new compositional training data and new composition model



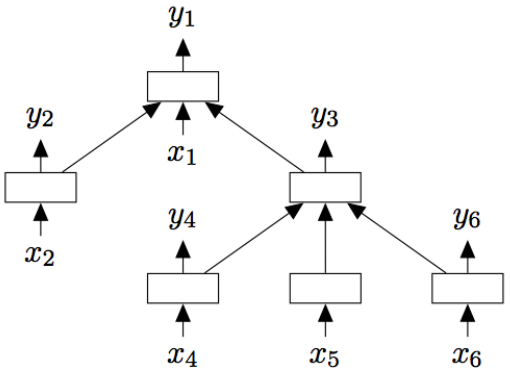
- Demos: http://nlp.stanford.edu:8080/sentiment/rntnDemo.html

Sentiment Analysis

- Better Models: Tree-based LSTM-RNNs

Tree LSTMs

- We can use those ideas in grammatical tree structures!
- Paper: Tai et al. 2015:
  Improved Semantic Representations From
  Tree-Structured Long Short-Term Memory Networks



- Idea: Sum the child vectors in a tree structure

- Each child has its own forget gate

- Same softmax on h

$$\tilde{h}_j = \sum_{k \in C(j)} h_k,$$

$$i_j = \sigma\left(W^{(i)} x_j + U^{(i)} \tilde{h}_j + b^{(i)}\right),$$

$$f_{jk} = \sigma\left(W^{(f)} x_j + U^{(f)} h_k + b^{(f)}\right),$$

$$o_j = \sigma\left(W^{(o)} x_j + U^{(o)} \tilde{h}_j + b^{(o)}\right),$$

$$u_j = \tanh\left(W^{(u)} x_j + U^{(u)} \tilde{h}_j + b^{(u)}\right),$$

$$c_j = i_j \odot u_j + \sum_{k \in C(j)} f_{jk} \odot c_k,$$

$$h_j = o_j \odot \tanh(c_j),$$

Other Classification Tasks

- Sentence similarity
- Entailment classification
- Spam detection
- Document topic classification
- Others: humor, rumor, sarcasm detection, etc.

SemEval has great new tasks every year with novel datasets in many cases! Some recent years:
http://alt.qcri.org/semeval2018/index.php?id=tasks
http://alt.qcri.org/semeval2017/index.php?id=tasks
http://alt.qcri.org/semeval2016/index.php?id=tasks

End of Lecture-1

Lecture-2 starts after 30-min break!