

CS5740: Natural Language Processing  
Spring 2017

# Lexical Semantics

Instructor: Yoav Artzi

Slides adapted from Dan Jurafsky, Chris Manning, Slav Petrov, Dipanjan Das,  
and David Weiss

# Overview

- Word sense disambiguation (WSD)
  - Wordnet
- Semantic role labeling (SRL)
- Continuous representations

# Lemma and Wordform

- A lemma or citation form
  - Basic part of the word, same stem, rough semantics
- A wordform
  - The “inflected” word as it appears in text

Wordform	Lemma
banks	bank
sung	sing
duermes	dormir

# Word Senses

- One lemma “bank” can have many meanings:

Sense 1: • ...a **bank**<sub>1</sub> can hold the investments in a custodial account...

Sense 2: • “...as agriculture burgeons on the east **bank**<sub>2</sub> the river will shrink even more”

- **Sense** (or **word sense**)
  - A discrete representation of an aspect of a word’s meaning.
- The lemma **bank** here has two senses

# Homonymy

**Homonyms:** words that share a form but have unrelated, distinct meanings:

$bank_1$ : financial institution,  $bank_2$ : sloping land

$bat_1$ : club for hitting a ball,  $bat_2$ : nocturnal flying mammal

1. Homographs (bank/bank, bat/bat)
2. Homophones:
  1. Write and right
  2. Piece and peace

# Homonymy in NLP

- Information retrieval
  - “bat care”
- Machine Translation
  - bat: [murciélago](#) (animal) or [bate](#) (for baseball)
- Text-to-Speech
  - **bass** (stringed instrument) vs. **bass** (fish)

# Quick Test for Multi Sense Words

- Zeugma
  - When a word applies to two others in different senses

Which flights **serve** breakfast?

Does Lufthansa **serve** Philadelphia?

Does Lufthansa serve breakfast and San Jose?

- The conjunction sounds “weird”
  - So we have two senses for *serve*

# Synonyms

- Word that have the same meaning in some or all contexts.
  - filbert / hazelnut
  - couch / sofa
  - big / large
  - automobile / car
  - vomit / throw up
  - Water / H<sub>2</sub>O
- Two words are synonyms if ...
  - ... they can be substituted for each other
- Very few (if any) examples of perfect synonymy
  - Often have different notions of politeness, slang, etc.



# Synonyms

- Perfect synonymy is rare
- Can we define it better in term of senses?
- Consider the words **big** and **large**
- Are they synonyms?
  - *How big is that plane?*
  - *Would I be flying on a large or small plane?*
- How about here:
  - *Miss Nelson became a kind of big sister to Benjamin.*
  - *Miss Nelson became a kind of large sister to Benjamin.*
- Why?
  - big has a sense that means being older, or grown up
  - large lacks this sense
- Synonymous relations must be defined between senses

# Antonyms

- Senses that are opposites with respect to one feature of meaning
- Otherwise, they are very similar!

<b>dark</b>	<b>short</b>	<b>fast</b>	<b>rise</b>	<b>hot</b>	<b>up</b>	<b>in</b>
<b>light</b>	<b>long</b>	<b>slow</b>	<b>fall</b>	<b>cold</b>	<b>down</b>	<b>out</b>

- Antonyms can
  - Define a binary opposition: **in/out**
  - Be at the opposite ends of a scale: **fast/slow**
  - Be reversives: **rise/fall**
- Very tricky to handle with some representations – remember for later!

# Hyponymy and Hypernymy

- One sense is a **hyponym** of another if the first sense is more specific, denoting a subclass of the other
  - *car* is a hyponym of *vehicle*
  - *mango* is a hyponym of *fruit*
- Conversely **hypernym/superordinate** (“hyper is super”)
  - *vehicle* is a hypernym of *car*
  - *fruit* is a hypernym of *mango*
- Usually transitive
  - (A hypo B and B hypo C entails A hypo C)

<b>Superordinate/hyper</b>	vehicle	fruit	furniture
<b>Subordinate/hyponym</b>	car	mango	chair

# WordNet

- A hierarchically organized lexical database
- On-line thesaurus + aspects of a dictionary
  - Word senses and sense relations
  - Some other languages available or under development (Arabic, Finnish, German, Portuguese...)

Category	Unique Strings
Noun	117,798
Verb	11,529
Adjective	22,479
Adverb	4,481

# WordNet

## WordNet Search - 3.1

- [WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations  
Display options for sense: (gloss) "an example sentence"

### Noun

- [S:](#) (n) **bass** (the lowest part of the musical range)
- [S:](#) (n) **bass**, [bass part](#) (the lowest part in polyphonic music)
- [S:](#) (n) **bass**, [basso](#) (an adult male singer with the lowest voice)
- [S:](#) (n) [sea bass](#), **bass** (the lean flesh of a saltwater fish of the family Serranidae)
- [S:](#) (n) [freshwater bass](#), **bass** (any of various North American freshwater fish with lean flesh (especially of the genus *Micropterus*))
- [S:](#) (n) **bass**, [bass voice](#), [basso](#) (the lowest adult male singing voice)
- [S:](#) (n) **bass** (the member with the lowest range of a family of musical instruments)
- [S:](#) (n) **bass** (nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes)

### Adjective

- [S:](#) (adj) **bass**, [deep](#) (having or denoting a low vocal or instrumental range) "*a deep voice*"; "*a bass voice is lower than a baritone voice*"; "*a bass clarinet*"

# WordNet

- **S: (n) bass, basso** (an adult male singer with the lowest voice)
  - **direct hypernym** / **inherited hypernym** / **sister term**
    - **S: (n) singer, vocalist, vocalizer, vocaliser** (a person who sings)
      - **S: (n) musician, instrumentalist, player** (someone who plays a musical instrument (as a profession))
      - **S: (n) performer, performing artist** (an entertainer who performs a dramatic or musical work for an audience)
      - **S: (n) entertainer** (a person who tries to please or amuse)
      - **S: (n) person, individual, someone, somebody, mortal, soul** (a human being) *"there was too much for one person to do"*
        - **S: (n) organism, being** (a living thing that has (or can develop) the ability to act or function independently)
          - **S: (n) living thing, animate thing** (a living (or once living) entity)
            - **S: (n) whole, unit** (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"*; *"the team is a unit"*
            - **S: (n) object, physical object** (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
            - **S: (n) physical entity** (an entity that has physical existence)
              - **S: (n) entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

# Senses and Synsets in WordNet

- Each word in WordNet has at least one sense
- The **synset** (synonym set), the set of near-synonyms, is a set of senses with a gloss
- Example: chump as a noun with the gloss:  
“a person who is gullible and easy to take advantage of”
- This sense of “chump” is shared by 9 words:  
chump<sup>1</sup>, fool<sup>2</sup>, gull<sup>1</sup>, mark<sup>9</sup>, patsy<sup>1</sup>, fall  
guy<sup>1</sup>, sucker<sup>1</sup>, soft touch<sup>1</sup>, mug<sup>2</sup>
- All these senses have the same gloss

# WordNet Noun Relations

Relation	Also called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	<i>breakfast</i> <sup>1</sup> → <i>meal</i> <sup>1</sup>
Hyponym	Subordinate	From concepts to subtypes	<i>meal</i> <sup>1</sup> → <i>lunch</i> <sup>1</sup>
Member Meronym	Has-Member	From groups to their members	<i>faculty</i> <sup>2</sup> → <i>professor</i> <sup>1</sup>
Has-Instance		From concepts to instances of the concept	<i>composer</i> <sup>1</sup> → <i>Bach</i> <sup>1</sup>
Instance		From instances to their concepts	<i>Austen</i> <sup>1</sup> → <i>author</i> <sup>1</sup>
Member Holonym	Member-Of	From members to their groups	<i>copilot</i> <sup>1</sup> → <i>crew</i> <sup>1</sup>
Part Meronym	Has-Part	From wholes to parts	<i>table</i> <sup>2</sup> → <i>leg</i> <sup>3</sup>
Part Holonym	Part-Of	From parts to wholes	<i>course</i> <sup>7</sup> → <i>meal</i> <sup>1</sup>
Antonym		Opposites	<i>leader</i> <sup>1</sup> → <i>follower</i> <sup>1</sup>



# WordNet 3.0

- Where it is:
  - <http://wordnetweb.princeton.edu/perl/webwn>
- Libraries
  - Python:
    - NLTK: <http://www.nltk.org/Home>
  - Java:
    - JWNL, extJWNL on sourceforge

# Word Sense Disambiguation

I play **bass** in a Jazz band

musical\_instrument

She was grilling a **bass** on the stove top

freshwater\_fish

# Supervised WSD

- Given: a lexicon (e.g., WordNet) and a word in a sentence
- Goal: classify the sense of the word
- Linear model:

$$p(\text{sense} \mid \text{word}, \text{context}) \propto e^{\theta \cdot \phi(\text{sense}, \text{word}, \text{context})}$$

Feature function looking at the sense, word, and context

# Supervised WSD

- Given: a lexicon (e.g., WordNet) and a word in a sentence
- Goal: classify the sense of the word
- Linear model:

$$p(\text{sense} \mid \text{word}, \text{context}) = \frac{e^{\theta \cdot \phi(\text{sense}, \text{word}, \text{context})}}{\sum_{s'} e^{\theta \cdot \phi(s', \text{word}, \text{context})}}$$

Summing over all senses for the word (e.g., from WordNet)

# Unsupervised WSD

- Goal: induce the senses of each word and classify in context
  1. For each word in context, compute some features
  2. Cluster each instance using a clustering algorithm
  3. Cluster labels are word senses

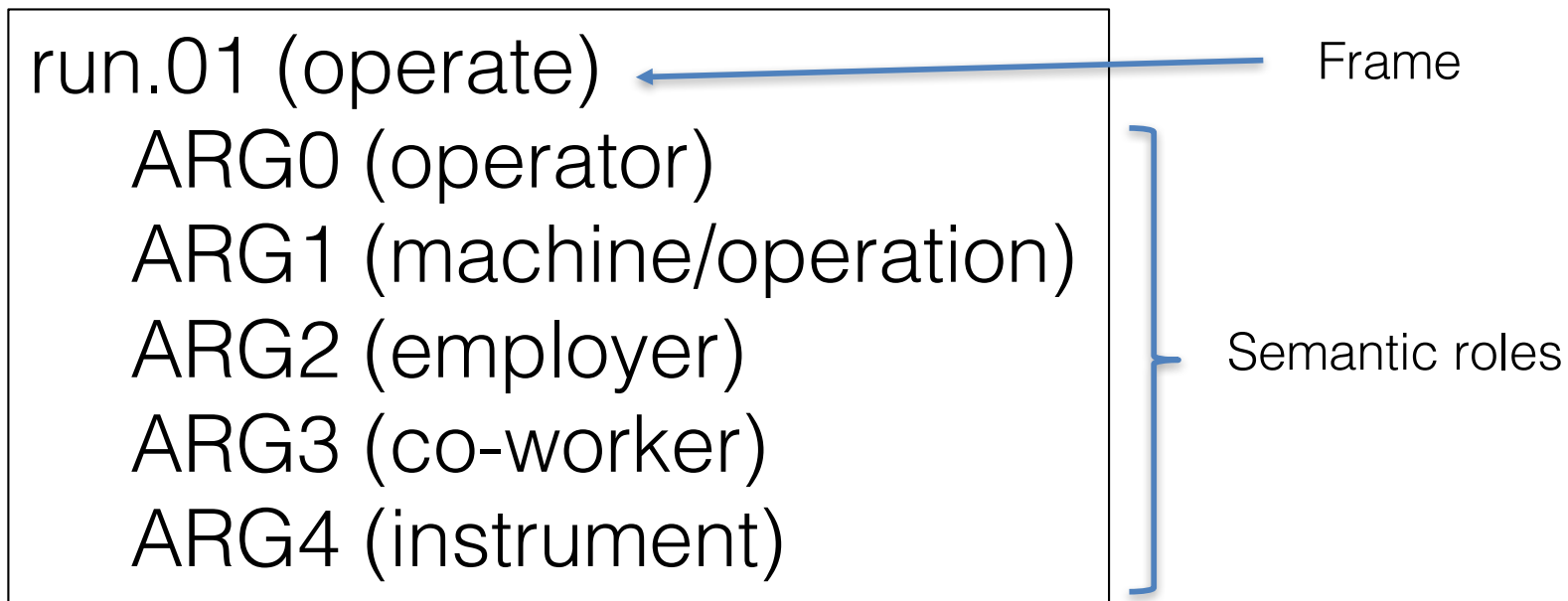
More reading: Section 20.10 of J&M

# Semantic Roles

- Some word senses (a.k.a. predicates) represent events
- Events have participants that have specific roles (as arguments)
- Predicate-argument structure at the type level can be stored in a lexicon

# Sematic Roles

- PropBank: a semantic role lexicon



# Sematic Roles

- PropBank: a semantic role lexicon

run.01 (operate)  
ARG0 (operator)  
ARG1 (machine/operation)  
ARG2 (employer)  
ARG3 (co-worker)  
ARG4 (instrument)

run.02 (walk quickly)  
ARG0 (runner)  
ARG1 (course/race)  
ARG2 (opponent)

Also: FrameNet, an  
alternative role lexicon



# Semantic Role Labeling

- Task: given a sentence, disambiguate predicate frames and annotate semantic roles

Mr. Stromach wants to resume a more influential role in **running** the company.

ARG0

II. Role labeling

run.01

I. Frame identification

ARG1

# Role Identification

- Classification models similar to WSD

Mr. Stromach wants to resume a more influential role in **running** the company.

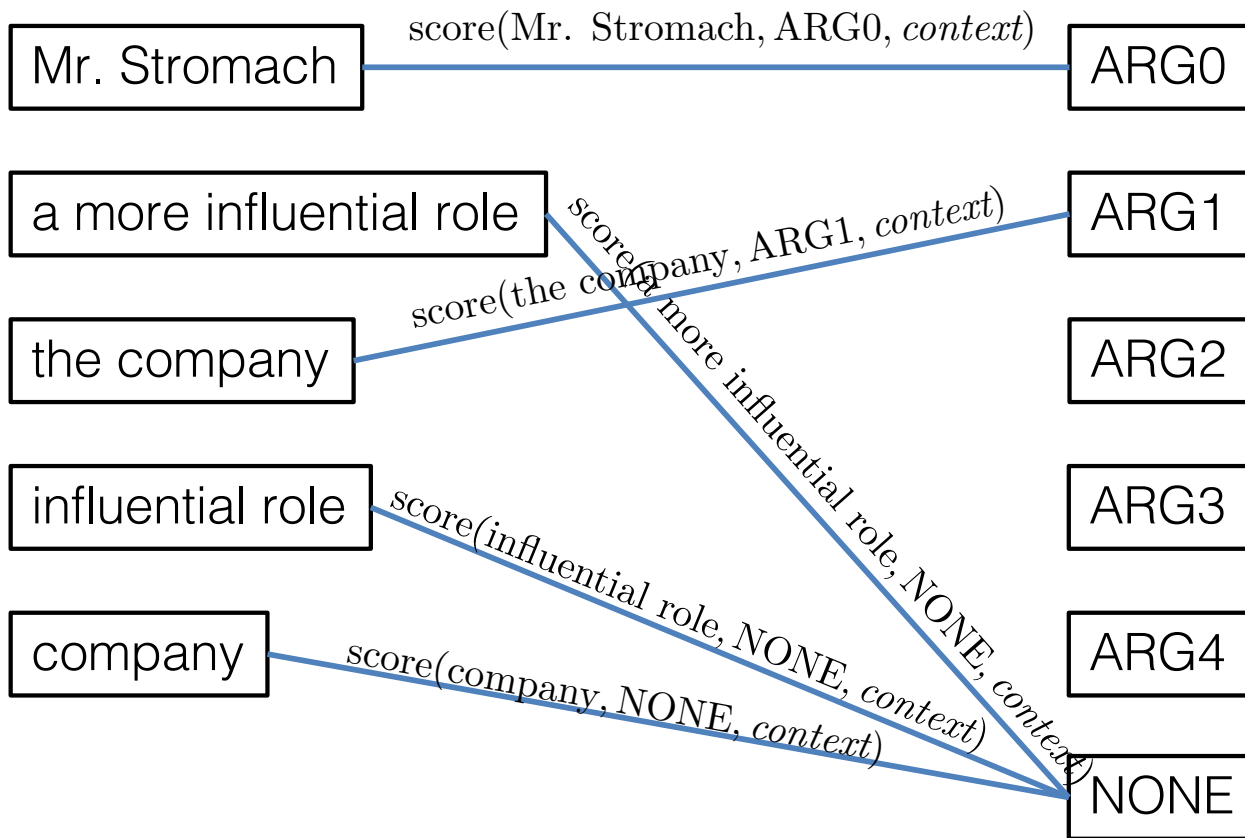
run.01

I. Frame identification

# Role Labeling

Sentence spans:

Potential roles:



Best matching  
between spans  
and roles

Score can come  
from any classifier  
(linear, SVM, NN)

# SRL

- [http://cogcomp.cs.illinois.edu/page/demo\\_view/srl](http://cogcomp.cs.illinois.edu/page/demo_view/srl)
- Example: The challenges facing Iraqi forces in Mosul include narrow streets, suicide bombers and small drones that the terror group has used to target soldiers.

	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> SRL	<input type="checkbox"/> Preposition	<input type="checkbox"/> Preposition
The	looker, facer [A0]	agent, entity causing some grouping [A0]										
challenges	V: face.01											
facing												
Iraqi	looked at, faced [A1]											
forces												
in												
Mosul												
include		V: include.01										
narrow												
streets												
,												
suicide		theme, thing being included in some group [A1]										
bombers												
and												
small												
drones												
that												
the												
terror												
group												
has												
used												
to												
target												
soldiers												
.												

[http://cogcomp.cs.illinois.edu/page/demo\\_view/srl](http://cogcomp.cs.illinois.edu/page/demo_view/srl)

# Word Similarity

- Task: given two words, predict how similar they are

The Distributional Hypothesis:

A word is characterized by the company it keeps  
(John Firth, 1957)



Know them by the company they keep!

# Word Similarity

A bottle of Tesgüino is on the table.  
Everybody likes tesgüino.  
Tesgüino makes you drunk.  
We make tesgüino out of corn.

- Occurs before *drunk*
- Occurs after *bottle*
- Is the direct object of *likes*
- ...



Similar to  
bear, wine,  
whiskey, ...

# Word Similarity

- Given a vocabulary of  $n$  words
- Represent a word  $w$  as:

$$\vec{w} = (f_1, f_2, f_3, \dots, f_n)$$

Binary (or count) features indicating the presence of the  $i^{\text{th}}$  word in the vocabulary in the word's context

- For example:

$$\text{Tsegüino} = (1, 1, 0, \dots)$$

corn

drunk

matrix



# Word Similarity

$$\vec{\text{Tsegüino}} = (1, 1, 0, \dots)$$

$$\vec{\text{beer}} = (0, 1, 0, \dots)$$

- Similarity can be measured using vector distance metrics
- For example, cosine similarity:

$$\text{similarity}(w, u) = \frac{w \cdot u}{\|w\| \|u\|} = \frac{\sum_{i=1}^n w_i u_i}{\sqrt{\sum_{i=1}^n w_i^2} \sqrt{\sum_{i=1}^n u_i^2}}$$

which gives values between -1 (completely different), 0 (orthogonal), and 1 (the same)

# Vector-Space Models

- Words represented by vectors
- In contrast to the discrete class representation of word senses
- Common method (and package):  
Word2Vec

# Word2Vec

- Open-source package for learning word vectors from raw text
- Widely used across academia/industry
  - Another common package: GloVe
- Goal: good word embeddings
  - Embedding are vectors in a low dimensional space
  - Similar words should be close to one another
- Two models:
  - Skip-gram (today)
  - CBOW (further reading: Mikolov et al. 2013)

# The Skip-Gram Model

- Given: Corpus  $D$  of pairs  $(w, c)$  where  $w$  is a **word** and  $c$  is **context**
- Context may be a single neighboring word (in window of size  $k$ )
  - Later: different definition

- Consider the parameterized probability

$$p(c|w; \theta)$$

- Goal: maximize the corpus probability

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(c|w; \theta)$$

- Wait! But we want good embeddings, so who cares about context pairs?

# The Skip-Gram Model

- Goal: maximize the corpus probability

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(c|w; \theta)$$

where:

$$p(c|w; \theta) = \frac{e^{v_c \cdot v_w}}{\sum_{c' \in C} e^{v_{c'} \cdot v_w}}$$

if  $d$  is the dimensionality of the vectors, we have  $d \times |V| + d \times |C|$  parameters

# The Skip-Gram Model

- Goal: maximize the corpus probability

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(c|w; \theta)$$

- The log of the objective is:

$$\arg \max_{\theta} \sum_{(w,c) \in D} (\log e^{v_c \cdot v_w} - \log \sum_{c'} e^{v_{c'} \cdot v_w})$$

- Not tractable in practice
  - Sum over all context – intractable
  - Approximated via negative sampling

# Negative Sampling

- Efficient way of deriving word embeddings
- Consider a word-context pair  $(w, c)$
- Let the probability that this pair was observed:

$$p(D = 1 | w, c)$$

- Therefore, the probability that it was not observed is:

$$1 - p(D = 1 | w, c)$$

# Negative Sampling

- Parameterization:

$$p(D = 1|w, c) = \frac{1}{1 + e^{-v_c \cdot v_w}}$$

- New learning objective:

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(D = 1|w, c) \prod_{(w,c) \in D'} p(D = 0|w, c)$$

- Need to get  $D'$



# Negative Sampling

- For a given  $\mathbf{k}$ , the size of  $\mathbf{D}'$  is  $\mathbf{k}$ -times bigger than  $\mathbf{D}$
- Each context  $\mathbf{c}$  is a word
- For each observed word-context pair,  $\mathbf{k}$  samples are generated based on unigram distribution

# Negative Sampling

- New learning objective:

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(D = 1 | w, c) \prod_{(w,c) \in D'} p(D = 0 | w, c)$$

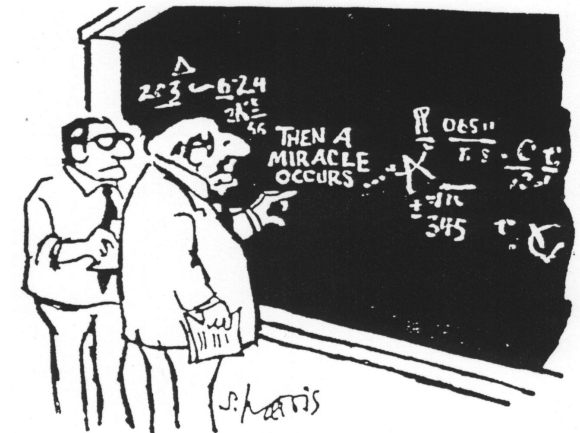
- Original learning objective:

$$\arg \max_{\theta} \prod_{(w,c) \in D} p(c | w; \theta)$$

- How does the new objective approximate the original one?

# The Skip-Gram Model

- Optimized for word-context pairs
- To get word embedding, take the vectors of the words  $v_w$
- But why does it work?
- Intuitively: words that share that many contexts will be similar
- Formal:
  - *Neural Word Embedding as Implicit Matrix Factorization / Levy and Goldberg 2014*
  - *A Latent Variable Model Approach to PMI-based Word Embeddings / Arora et al. 2016*



I think you should be a little more specific, here in Step 2

# Word Galaxy

- Word Galaxy
  - <http://anthonygarvan.github.io/wordgalaxy/>
- Embeddings for word substitution
  - <http://ghostweather.com/files/word2vecpride/>

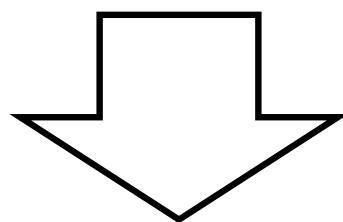
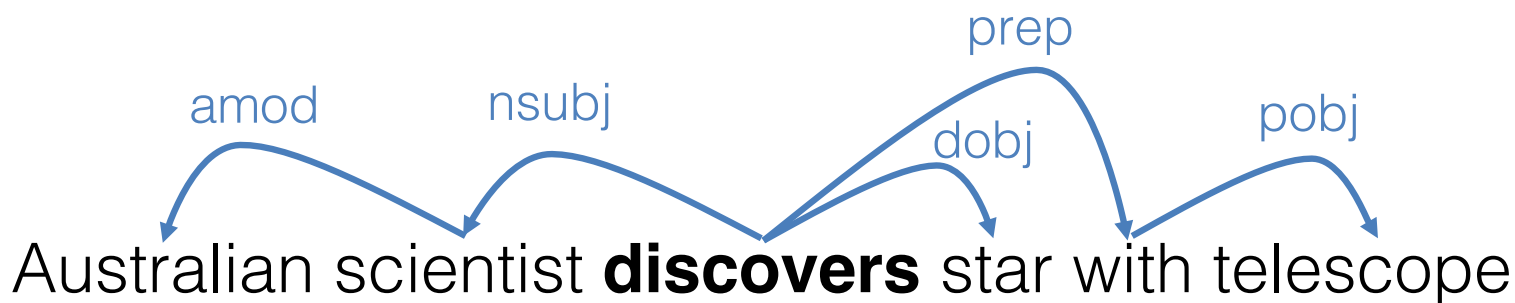
# Structured Contexts

Australian scientist **discovers** star with telescope

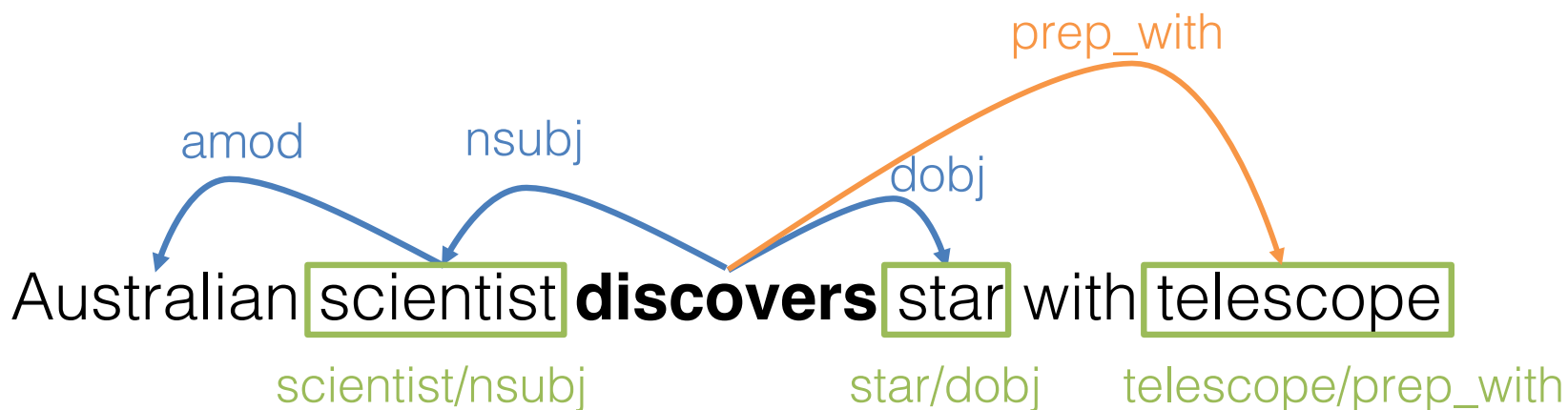
Skip-Gram context with  $n=2$

- Just looking at neighboring words, often doesn't capture arguments and modifiers
- Can we use anything except adjacency to get context?

# Structured Contexts



Collapsing “prep” links



# Structured Context

Target Word	BOW5	BOW2	DEPS
batman	nightwing aquaman catwoman superman manhunter	superman superboy aquaman catwoman batgirl	superman superboy supergirl catwoman aquaman
hogwarts	dumbledore hallows half-blood malfoy snape	evernight sunnydale garderobe blandings collinwood	sunnydale collinwood calarts greendale millfield
turing	nondeterministic non-deterministic computability deterministic finite-state	non-deterministic finite-state nondeterministic buchi primality	pauling hotelling heting lessing hamming
florida	gainesville fla jacksonville tampa lauderdale	fla alabama gainesville tallahassee texas	texas louisiana georgia california carolina
object-oriented	aspect-oriented smalltalk event-driven prolog domain-specific	aspect-oriented event-driven objective-c dataflow 4gl	event-driven domain-specific rule-based data-driven human-centered
dancing	singing dance dances dancers tap-dancing	singing dance dances breakdancing clowning	singing rapping breakdancing miming busking


Table 1: Target words and their 5 most similar words, as induced by different embeddings.

# Word Embeddings vs. Sparse Vectors

- Count vectors: sparse and large
- Embedded vectors: small dense
- One advantage: dimensionality
- More contested advantage: better generalization
  - See Levy et al. 2015 (Improving Distributional Similarity with Lessons Learned from Word Embeddings) for detailed analysis



# Applications

- Word vectors are often input to various end applications
  - Parsing, co-reference resolution, named-entity recognition (maybe even for ) , semantic role labeling, etc.
- Input to sentence models, including recurrent and recursive architectures