AdvNLP/E Lecture 1

Lexical & Distributional Semantics

Dr Julie Weeds, Spring 2026

Lecture 1 Overview

PART 1

- Lexical semantics
  - word senses
  - semantic relationships
  - WordNet
  - semantic similarity measures based on WordNet
  - evaluation

PART 2

- Distributional Semantics
  - bootstrapping semantics from context
  - cosine similarity
  - (positive) pointwise mutual information
  - evaluation
  - word ambiguity
  - semantic relationships
  - sparsity

This might be revision if you have done Applied NLP!

Lexical Semantics

Lecture 1, part 1

Dr Julie Weeds, Spring 2026

Word senses

- Words are often **ambiguous**
- Words can have **multiple senses** (i.e., meanings)

I placed the book on the ***counter***.

vs

I placed my ***counter*** *on the gameboard*.

- How many more senses of ***counter*** can you think of?

Dictionaries

- **Lexicographers** produce dictionaries which:
  - enumerate the senses of all of the words in a language
  - provide definitions of different sense
  - provide examples of usage of different senses

WordNet online search:

http://wordnetweb.princeton.edu/perl/webwn

Oxford English Dictionary online search:

https://en.oxforddictionaries.com/?utm_source=od-panel&utm_campaign=en

How many different senses do words have?

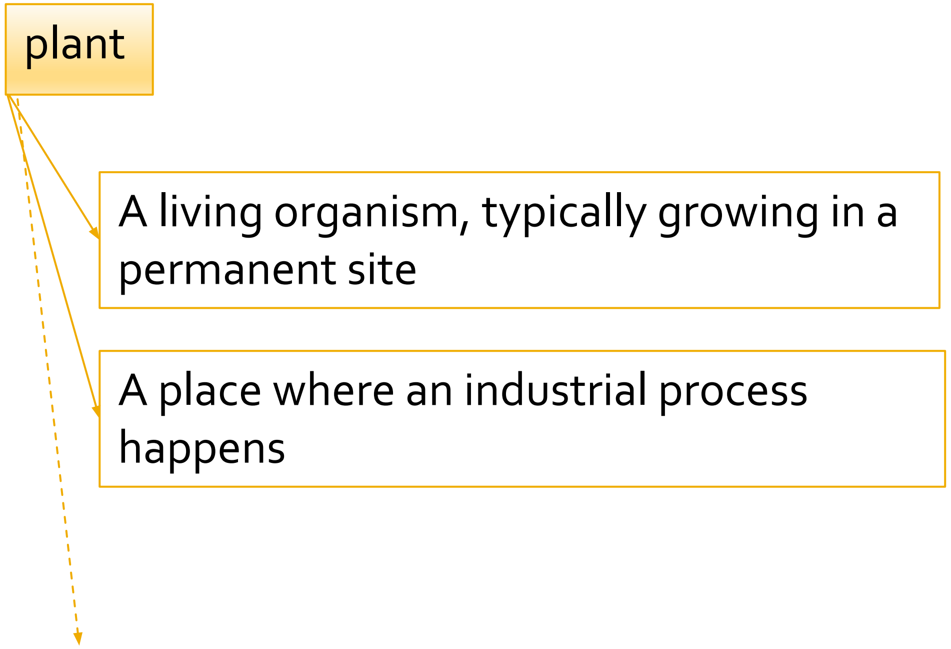
	WordNet	Oxford
plant	Noun:4, Verb:6	N:6, V:11
chicken	Noun:4, adJ:1	N:4, V:1, J: 1
book	Noun: 11, Verb:4	N:14,V:9
twig	Noun:1, Verb: 2	N:2
counter	Noun: 9, Verb:2, adJ:1, adveRb: 1	N:13, V: 3, J: 1, R: 1

Dictionaries do not always agree on this! Why is it so difficult?

Sense distinctions

HOMONYMY

- Broad distinctions



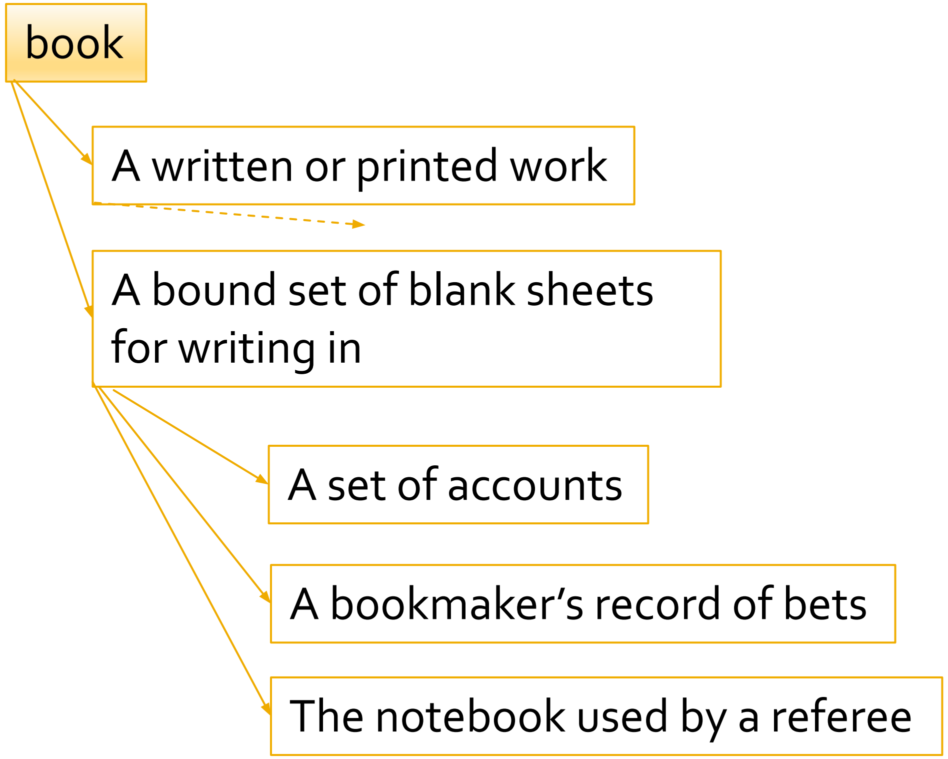
plant

A living organism, typically growing in a permanent site

A place where an industrial process happens

POLYSEMY

- fine-grained distinctions



book

A written or printed work

A bound set of blank sheets for writing in

A set of accounts

A bookmaker's record of bets

The notebook used by a referee

Lexical semantic relationships

- synonymy
- antonymy
- hyponymy / hypernymy
- meronymy / holonymy
- topical relatedness

Synonymy

fast == quickly

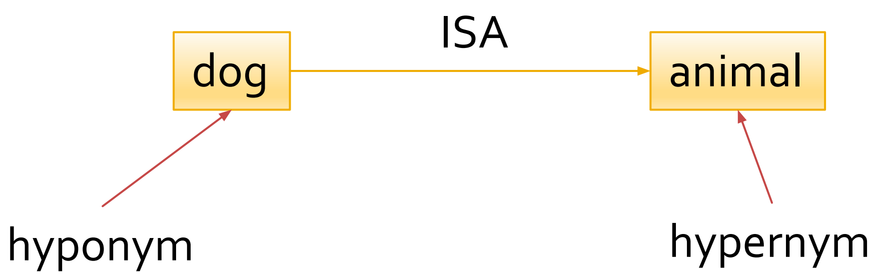
- Words which mean the same thing
- *Two words are **synonymous** if they can be substituted in all possible contexts without changing the meaning of the utterance.*
- True synonyms are very rare
- Choice of synonym usually gives us some extra information about the situation or speaker e.g., *car* vs *automobile*
- It is often defined as a relationship between word senses rather than between words. e.g., *plan*t == *spy* ?

Antonymy

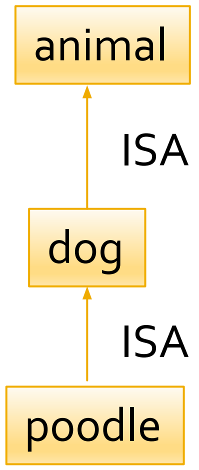
hot ≠ cold

- Words which are opposite in meaning
- Substituting one for the other would often cause a contradiction:
  - *The food is hot*.
  - *The food is cold*.
- Antonyms are actually very similar in meaning
  - *hot* and *cold* both describe the temperature of an object
  - *rise* and *fall* both describe an object which is moving in the vertical plane
- Most antonym pairs are adjectives, verbs or adverbs

Hyponymy and Hypernymy

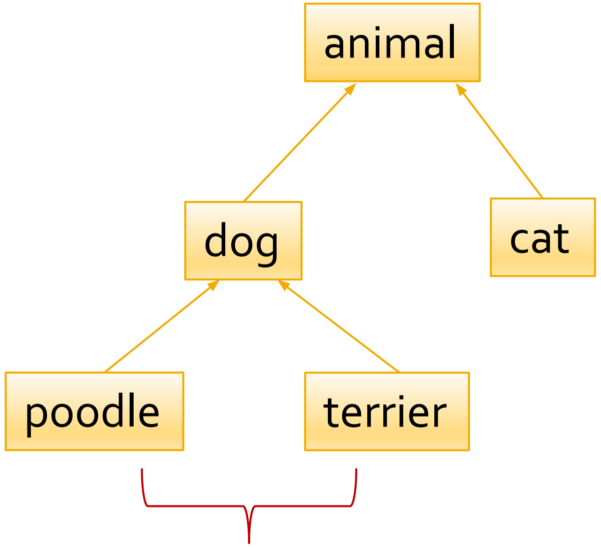


- Linguistic terms which capture the idea of class inclusion
- A *dog* is a type of *animal* so:
  - *dog* is a **hyponym** of *animal*
  - *animal* is a **hypernym** of *dog*
- It's a transitive relationship so
  - If *dog* is a hyponym of *animal*
  - And *poodle* is a hyponym of *dog*
  - *Poodle* is also a hyponym of *animal*



Hyponym Hierarchies

- The hyponymy relationship links together large numbers of concepts in a tree or hierarchy
- Most general superclass at the top
- Most specific types at the leaves



Words which share a common hypernym are called **co-hyponyms**

WordNet

- More than an electronic dictionary!
- See http://wordnet.princeton.edu for more general information
- Or see: Christiane Fellbaum (1998, ed.) *WordNet: An Electronic Lexical Database*. MIT Press.

WordNet

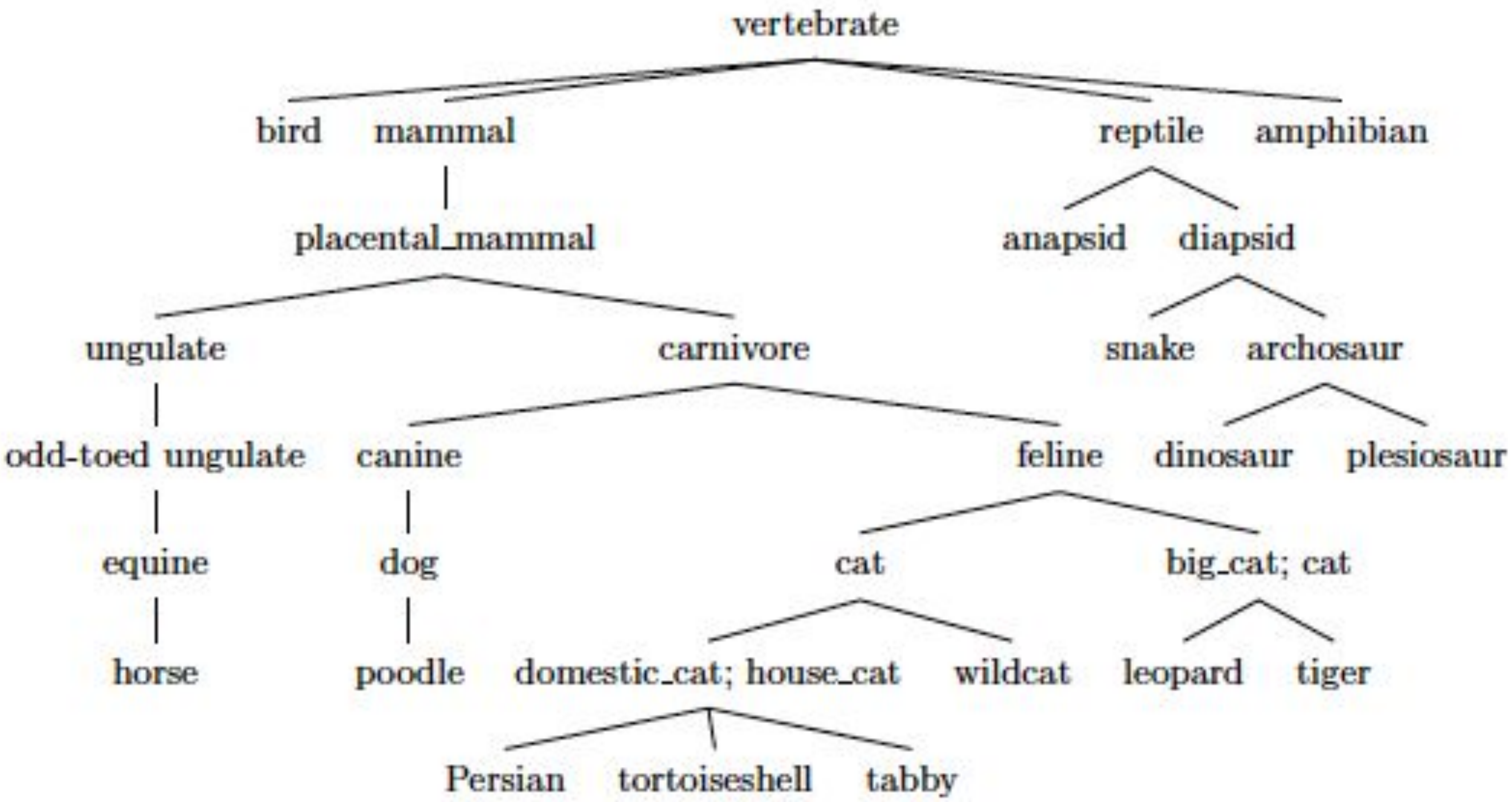
- A linguistic network organized around synonymy and hyponymy
- Core unit is the **synset**
  - a set of synonymous word senses
  - a set may contain a single word
  - synset items may be bigrams (e.g., “plant life”) as well as unigrams
  - each synset is also associated with a single definition
- Polysemous words appear in multiple synsets
  - One for each sense
- Synsets are then connected via hyponymy.....

{**plant, flora, plant life**} = a living organism lacking the power of locomotion

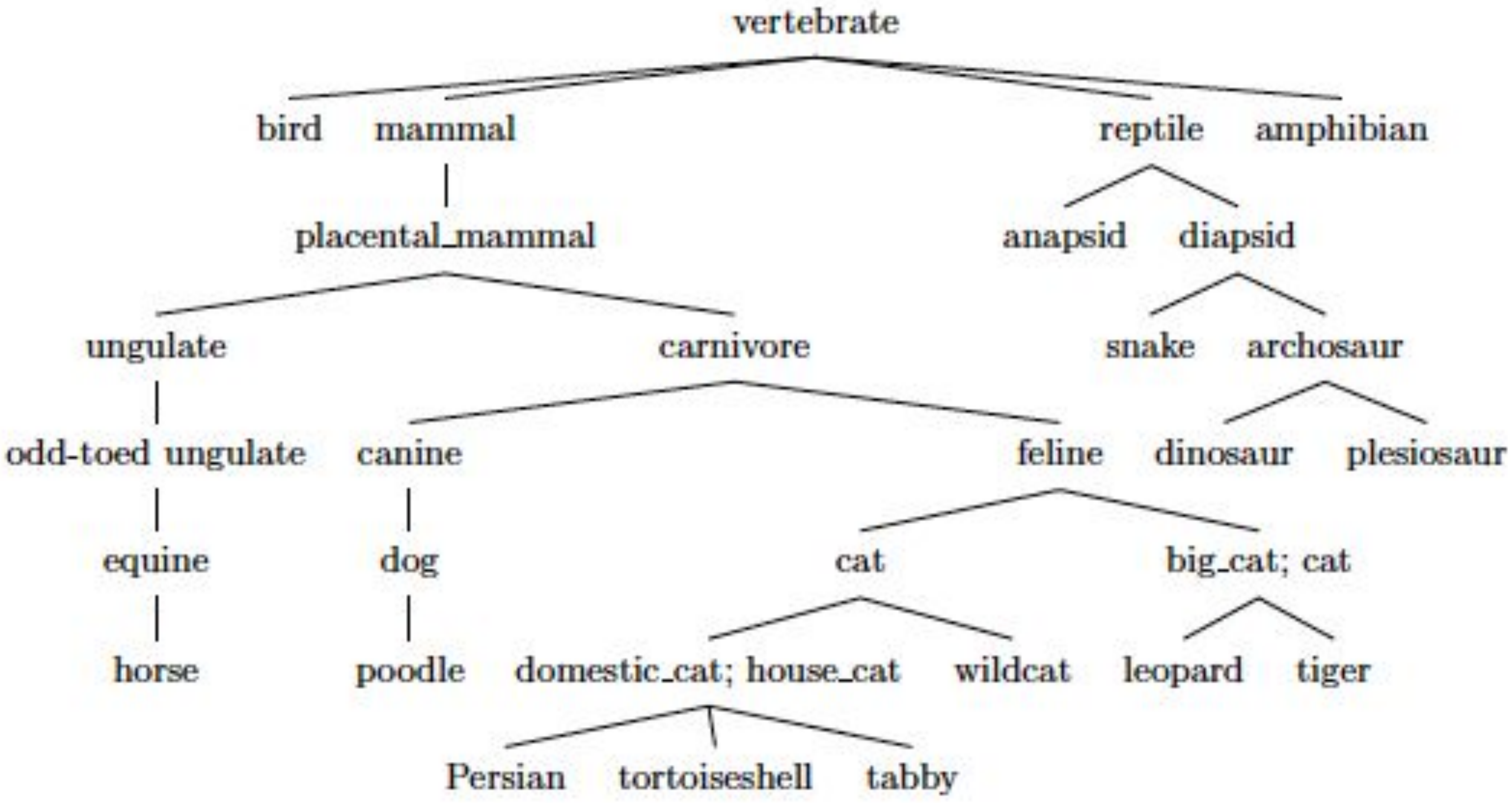
{plant} = something planted secretly for discovery by another

{**plant, works, industrial plant**} = buildings for carrying on industrial labour

Extract from the WordNet noun hierarchy

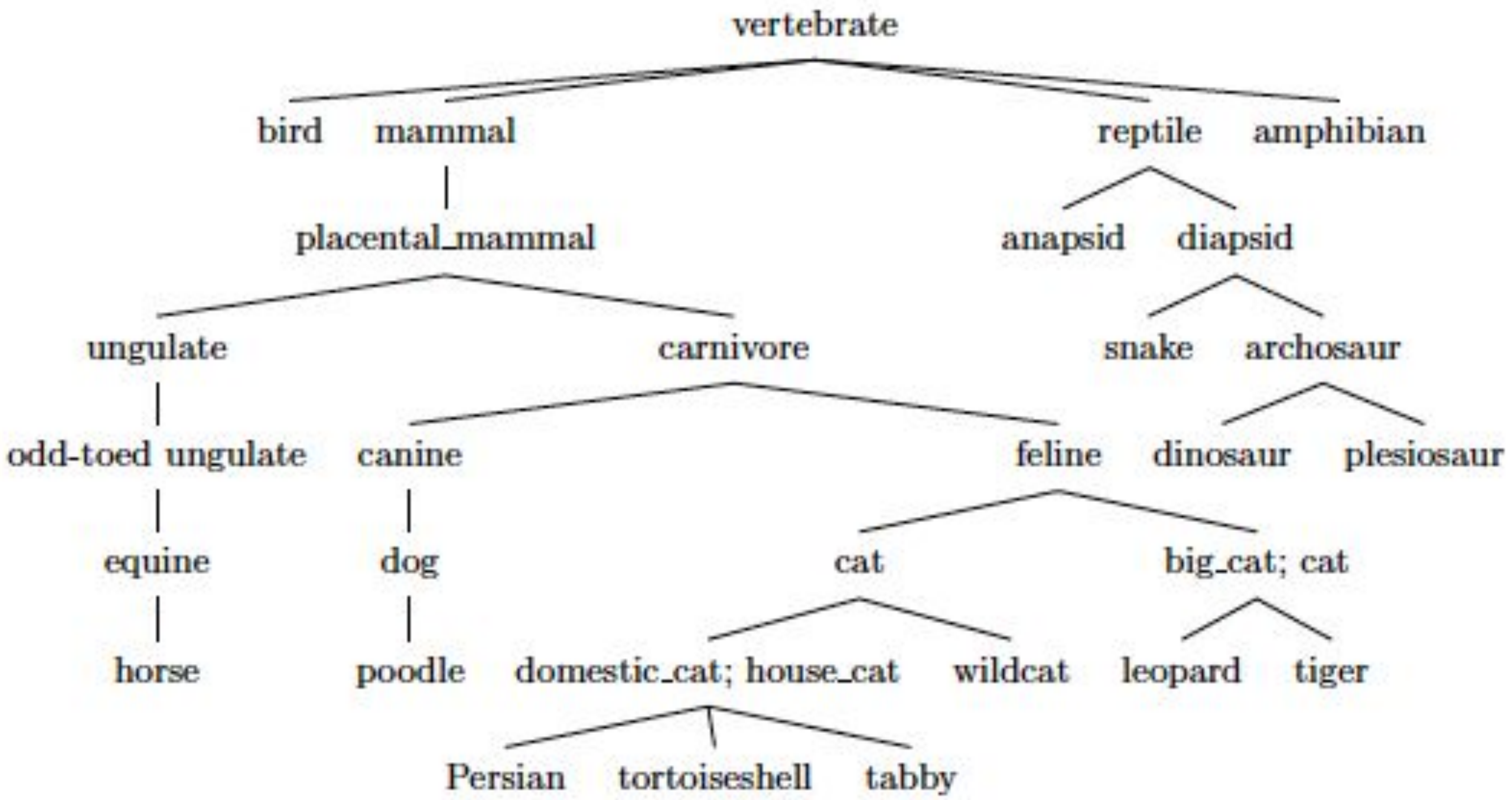


Semantic similarity based on WordNet



Intuition: More similar concepts are closer together in the hierarchy.

Path length: shorter path -> greater similarity

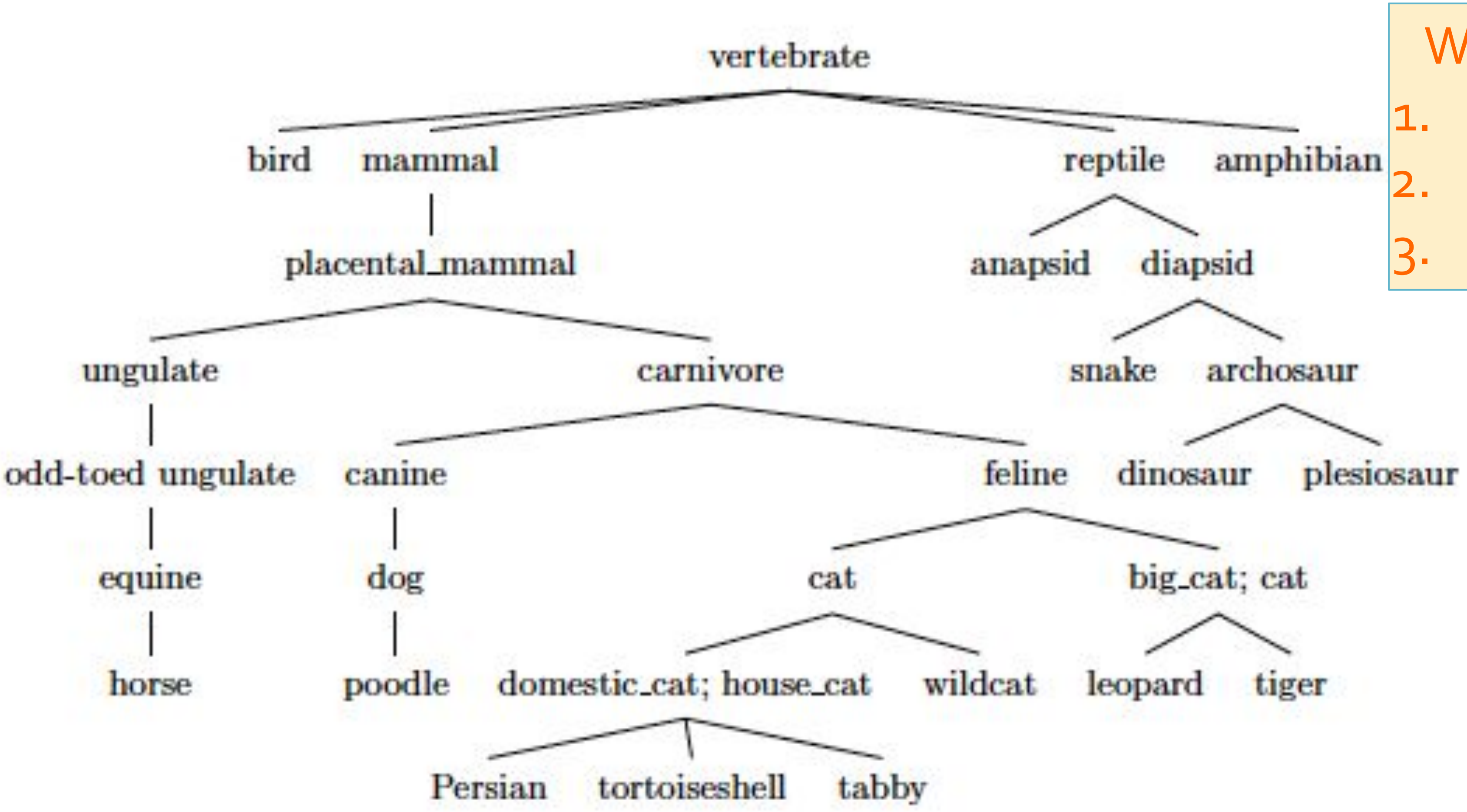


$$\text{sim}_{\text{path}}(c_1, c_2) = \frac{1}{1 + \text{pathlen}(c_1, c_2)}$$

Potential problems with pathlength

- Pathlength does not differentiate between different types of path e.g., *canine* □ … □ *vertebrate* vs *dog* □ … □ *cat*
- Intuitively, concepts (separated by same path length) are more dissimilar higher up the tree; but this is not captured by path length similarity measure
- Some parts of tree may be densely populated with rare terminology

Lowest common subsumer: similarity based on what two concepts share



What is the LCS of:

1. tabby and tiger?
2. poodle and carnivore?
3. poodle and tiger?

Information content

- Intuition: concepts which have the LCS *carnivore* are more similar than concepts which have the LCS *vertebrate*
- We gain more **information** when we are told two objects are both *carnivores* than when we are told they are both *vertebrates*.
- We capture this probabilistically via the information content (IC) of a concept
  - Annotate the hierarchy with the frequency of occurrence of each concept in some corpus
  - Remember that the occurrence of a concept implies the occurrence of all of its hypernyms (if something is a *dog*, it is also a *canine* and so on)

$$\boxed{P(c) = \frac{\text{freq}(c)}{\sum_c \text{freq}(c)}}$$

$$\boxed{\text{IC}(c) = -\log P(c)}$$

Question

How do we count the number of times a concept has occurred in a corpus?

WordNet similarity measures based on information content (IC)

$$\mathrm{IC}(c) = -\log P(c)$$

Information content in a concept

$$\mathrm{sim}_{\mathrm{res}}(c_1, c_2) = \mathrm{IC}(\mathrm{LCS}(c_1, c_2))$$

See Resnik, 1995

Information content in what the concepts share (their lowest common subsumer)

$$\mathrm{sim}_{\mathrm{lin}}(c_1, c_2) = \frac{2 \times \mathrm{sim}_{\mathrm{res}}(c_1, c_2))}{\mathrm{IC}(c_1) + \mathrm{IC}(c_2)}$$

See Lin 1998b

Ratio of shared information content to total information content

Word similarity

$$\mathrm{wordsim}(w_1, w_2) = \max_{\substack{c1 \in senses(w_1) \\ c2 \in senses(w_2)}} \mathrm{sim}(c_1, c_2)$$

- Can you write python code to implement this function?

Evaluation

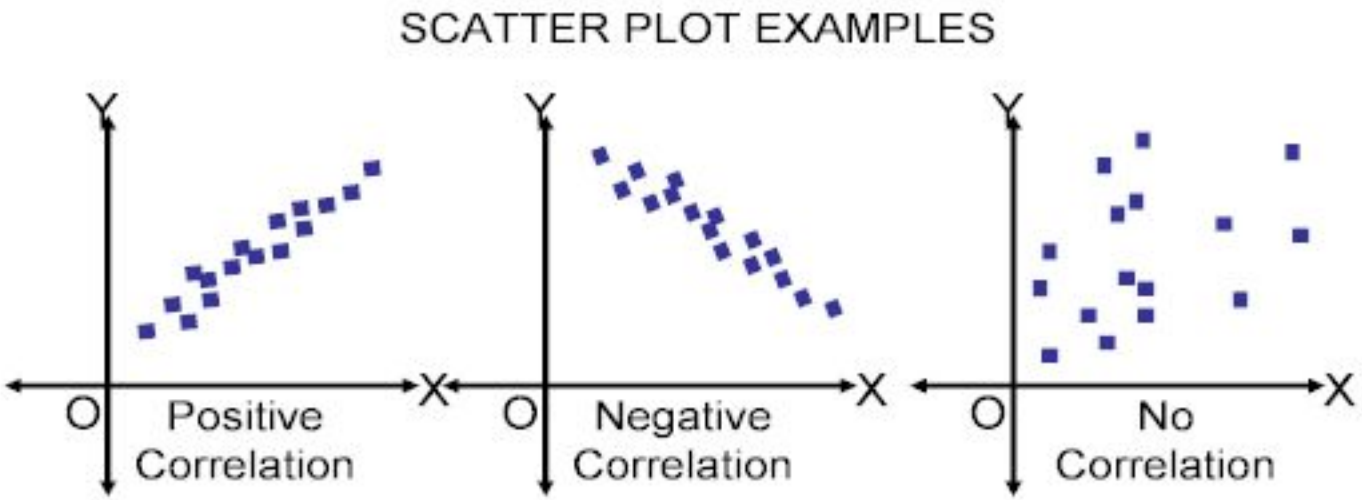
- How do we evaluate semantic similarity measures?
- What is the right answer?

Human synonymy judgements

- Rubenstein & Goodenough 1965 (65 pairs)
- Miller and Charles 1991 (30 pairs)
- WordSim-353 2002 (353 pairs)
- MEN dataset 2012 (3000 pairs)

	M&C	WN
car-automobile	3.92	1.0
magician-wizard	3.5	1.00
journey-car	1.16	0.0
coast-forest	0.42	0.15
noon-string	0.08	0.0

Correlation



- Pearson’s product-moment correlation coefficient ✗
- Spearman’s rank correlation coefficient ✓

Distributional Semantics

Lecture 1, Part 2

Dr Julie Weeds, Spring 2025

Distributional Semantics

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

Firth (1957)

The Distributional Hypothesis: "Words that occur in the same contexts tend to have similar meanings."

Harris (1954)

What does *tezguino* mean?

1. A bottle of *tezguino* is on the table.
2. Everyone likes *tezguino*.
3. *Tezguino* makes you drunk.
4. We make *tezguino* out of corn.

(Lin, 1998)

Bootstrapping the semantics of unknown words

- The **contexts** in which *tezguino* is used suggest that *tezguino* may be:
  - *A kind of alcoholic beverage made from corn mash*
- Similarity plays an important role in word acquisition (Gentner, 1982)
- Can we use corpora to infer similarity between words i.e., infer that *tezguino* is similar to *beer, wine, vodka* etc?

Applications of distributional semantics

- Automatic thesaurus construction
  - For any language, genre, domain … where we have a corpus
- Overcoming data sparseness in models which require labelled training data

Distributional semantics in document classification

- Imagine we have built a Naïve Bayes document relevancy classifier using a relatively small training sample (e.g., 500 documents)
- A test document contains the word *tezguino* which has not been seen in the training sample
  - so it cannot contribute to the relevancy classification
- But by applying distributional semantics to a very large unlabeled corpus (e.g., the web), we know that *tezguino* is very similar to *beer*
  - *beer* has been seen in the training sample
  - Assume **P(*tezguino*|class) ≈ P(*beer*|class)**

Facets of meaning

- Tigers **eat** meat.
- The monkey **ate** a banana.
- X17 likes to **eat** falafel.
- My son does not **eat** courgettes.
- The machine **ate** my credit card.

From these examples we can learn:

- What can be **eaten**?
- What **eats** things?
- *Meat*, *banana*, *falafel*, *courgettes* and *credit card* all share 1 facet of meaning – that they can be eaten
- *Tigers*, *monkey*, *X17*, *son* and *machine* all share 1 facet of meaning – that they eat things

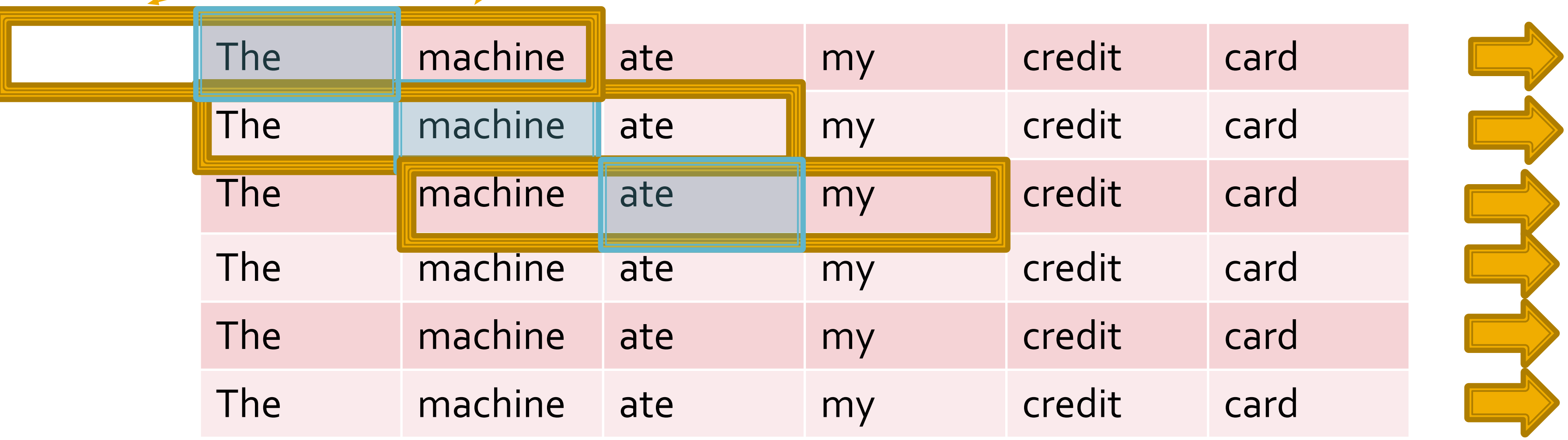
Features to capture facets of meaning

- Dependency relationships between words:
  - “is subject of *eat*”
  - “is object of *eat*”
- Proximity between words
  - “occurs within a **window of +- m words** either side of the word *eat*”
- Feature values can be Boolean but are usually real-valued
  - strength of association
- Dependency parsing is difficult
- Windows are easy to construct
- Window size can be varied to capture different types of semantic relationships

Context windows

window size around target word = +-1

Features added per target word



The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card

the: {machine: 1}

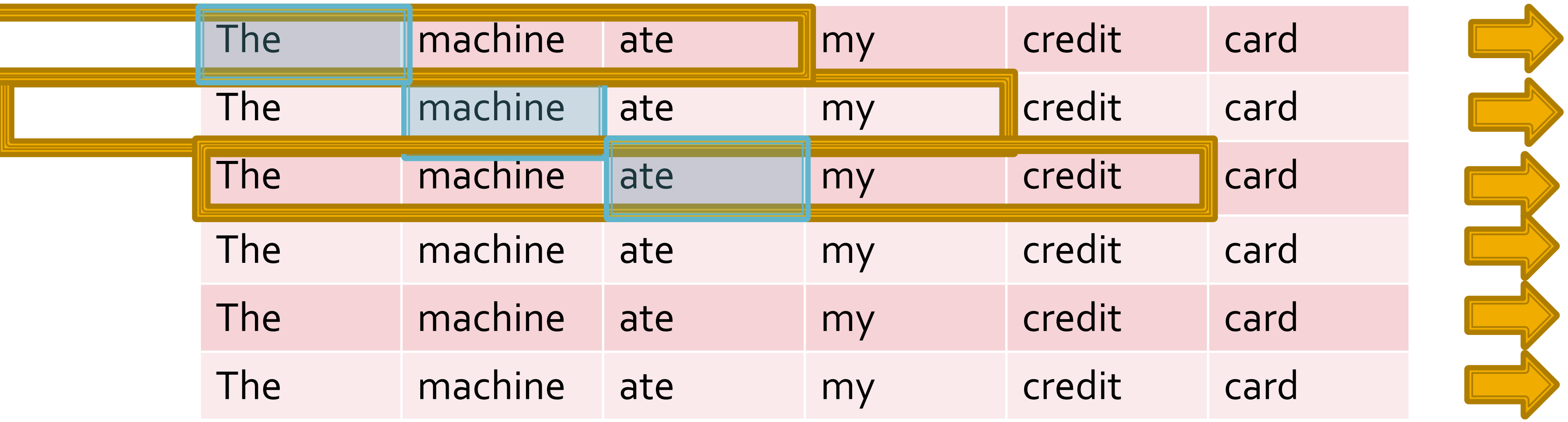
machine: {the: 1, ate: 1}

ate: {machine: 1, my: 1}

What features will be added for *my*, *credit* and *card*?

Context windows

window size around target word = +-2



The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card
The	machine	ate	my	credit	card

Features added per target word

the: {machine: 1, ate: 1}

machine: {the: 1, ate: 1, my: 1}

ate: {the: 1, machine: 1, my: 1, credit: 1}

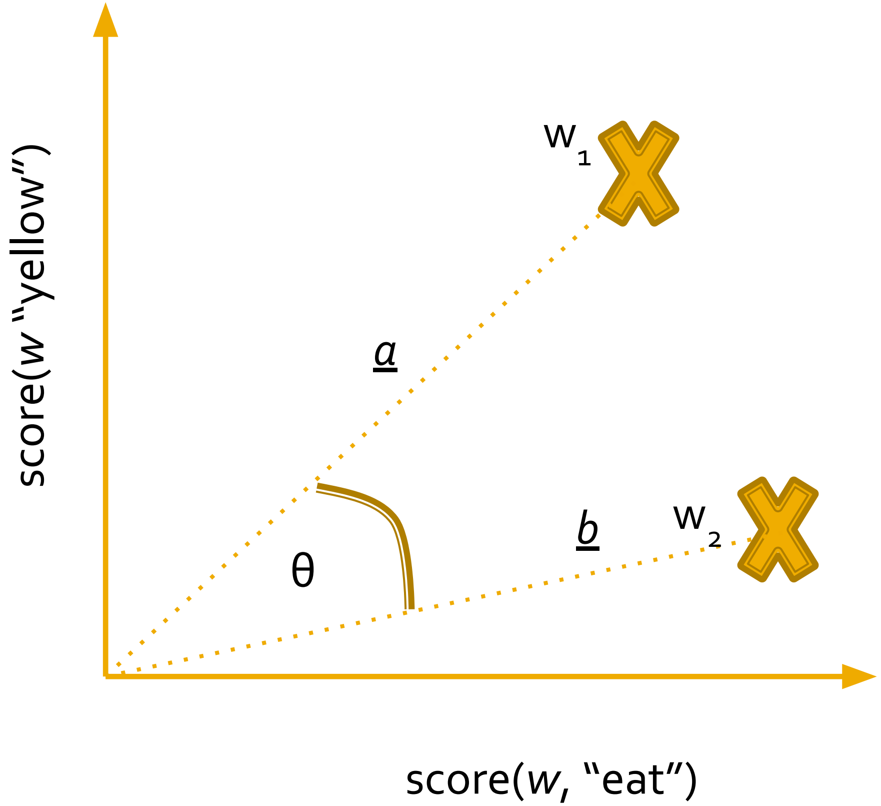
What features will be added for *my*, *credit* and *card*?

Distributional Representations

Use windowing to extract and count features for all words in a large corpus i.e., distributional representations or vectors

feature	**banana**	**meat**	**credit**	**Total**
yellow	10	2	3	15
red	2	14	19	35
eat	20	9	1	30
spend	1	2	27	30
card	3	2	50	55
the	25	25	50	100
is	20	20	40	80
tiger	3	17	0	20
man	6	9	10	25
monkey	10	0	0	10
Total	100	100	200	400

Cosine similarity



- The more similar two words are, the smaller the angle θ between their vectors will be.
- So:

$$sim(w_1, w_2) = \cos(\theta)$$

dot product

$$= \frac{\underline{a}.\underline{b}}{\sqrt{\underline{a}.\underline{a}\times\underline{b}.\underline{b}}}$$

Where:

m=number of dimensions

$$\underline{a}.\underline{b} = \sum_{i}^{m} a_i b_i$$

Calculating cosine

feature	**banana**	**meat**	***a.b***	**a.a**	**b.b**
yellow	10	2	20	100	4
red	2	14	28	4	196
eat	20	9	180	400	81
spend	1	2	2	1	4
card	3	2	6	9	4
the	25	25	625	625	625
is	20	20	400	400	400
tiger	3	17	51	9	289
man	6	9	54	36	81
monkey	10	0	0	100	0
Total	100	100	1366	1684	1684

$$\cos(banana, meat) = \frac{1366}{1684} = 0.81$$

Pointwise Mutual information (PMI)

- Frequency and/or simple conditional probability do not capture the intuition that some features are more informative than others
- *the* and *is* appear relatively frequently with all of the words
  - so their contribution to similarity should be smaller
- PMI measures the amount of information gained by seeing a word and a feature together
- A feature which co-occurs with a target word more than we would expect (if words and features occurred independently) has more weight in the similarity calculation

calculating PMI

$$I(w,f) = \log \frac{P(f \mid w)}{P(f)} \qquad = \log \frac{P(f \cap w)}{P(f) \times P(w)}$$

$$I(w,f) = \log \frac{\text{freq}(f,w) \times \overbrace{\text{freq}(*,*)}^{\text{grand total}}}{\underbrace{\text{freq}(*,w)}_{\text{row total}} \times \underbrace{\text{freq}(f,*)}_{\text{column total}}}$$

Representations based on PMI

feature	**banana**	**meat**	**credit**	**Total**
yellow	10	2	3	15
red	2	14	19	35
eat	20	9	1	30
spend	1	2	27	30
card	3	2	50	55
the	25	25	50	100
is	20	20	40	80
tiger	3	17	0	20
man	6	9	10	25
monkey	10	0	0	10
Total	100	100	200	400

$$log\frac{10\times400}{100\times15}$$

feature	banana	meat	credit
yellow	1.42		
red			
eat			
spend			
card			
the			
is			
tiger			
man			
monkey			

Positive PMI (PPMI)

- What happens when frequency of co-occurrence is 0?
- PMI = negative infinity!!!
- positive PMI avoids this problem
  - similarity is then also based on shared features rather than the sharing of absent features

$$\boxed{\mathrm{PPMI}(w,f) = \begin{cases} I(w,f) & I(w,f) > 0 \\ 0 & \text{otherwise} \end{cases}}$$

Representations based on PPMI

feature	**banana**	**meat**	**credit**	**Total**
yellow	10	2	3	15
red	2	14	19	35
eat	20	9	1	30
spend	1	2	27	30
card	3	2	50	55
the	25	25	50	100
is	20	20	40	80
tiger	3	17	0	20
man	6	9	10	25
monkey	10	0	0	10
Total	100	100	200	400

$$log \frac{10\times400}{100\times15}$$

feature	banana	meat	credit
yellow	1.42	**0**	**0**
red	**0**	0.68	0.12
eat	1.42	0.26	**0**
spend			
card			
the			
is			
tiger			**0**
man			
monkey		**0**	**0**

Automatic thesaurus generation

- Extract feature representations based on corpus co-occurrence frequencies
- Convert representations to PPMI
- Calculate cosine similarities for all pairs of words
  - computationally very expensive
  - may want to reduce the number of words considered in vocab
  - e.g., top 10,000 words
- Find nearest neighbours of each word

Evaluation

- Difficult – why?
- Intrinsic evaluation
  - human synonymy judgements
  - manually compiled thesauruses
- Extrinsic evaluation
  - performance gain in an application

Word ambiguity

Here is the distributional thesaurus entry for the noun *bow* (derived using nltk.lin_thesaurus)

bow	
ribbon	0.09
machete	0.07
spear	0.07
hull	0.07
sword	0.07
knife	0.07
arrow	0.06
scarf	0.06
rope	0.06
streamer	0.06

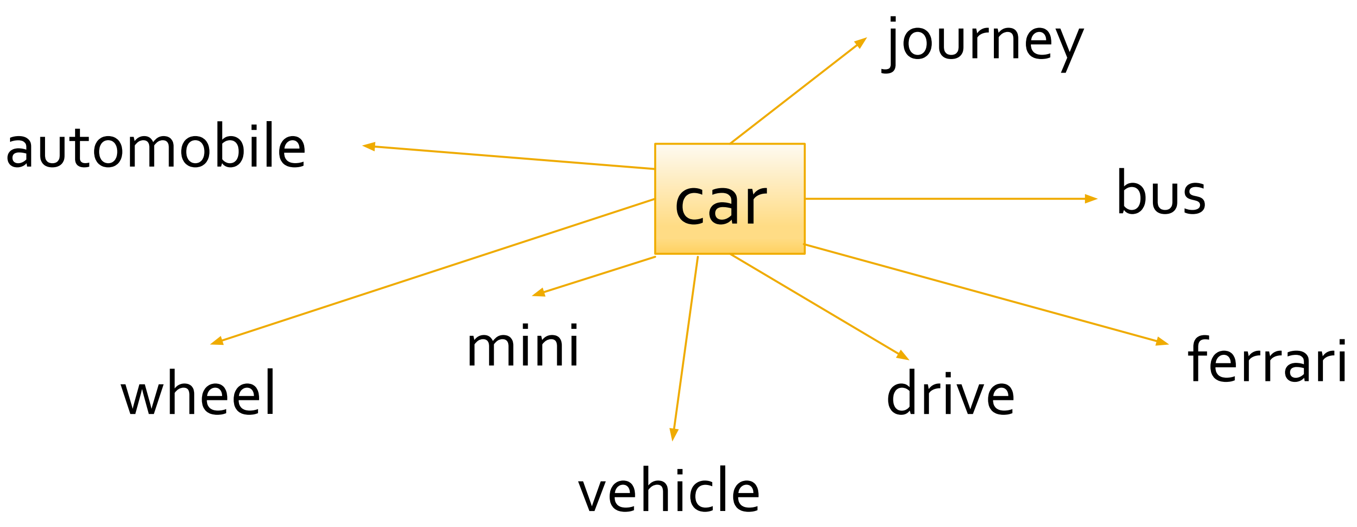
- What different senses of the word bow do you think are captured by the thesaurus entry?
- Are the neighbours distributed evenly between the senses or do some senses have more neighbours than others?
- Why do you think this is?

Senses in Distributional Semantics

- Distributional representations are of words not senses
  - mixture of senses in distributional neighbourhoods
  - this can be a problem in some applications. ….?
  - possible solutions: carry out WSD
    - before finding distributional neighbours
    - after finding distributional neighbours
- Distributional neighbours tend to reflect predominant sense of word
  - how could this be useful?

Semantic relationships

- Similar words are not necessarily synonyms
- Neighbourhoods typically contain:
  - synonyms, antonyms, hypernyms, hyponyms, co-hyponyms, meronyms, topically related words ….



car

journey

automobile

bus

wheel

mini

vehicle

drive

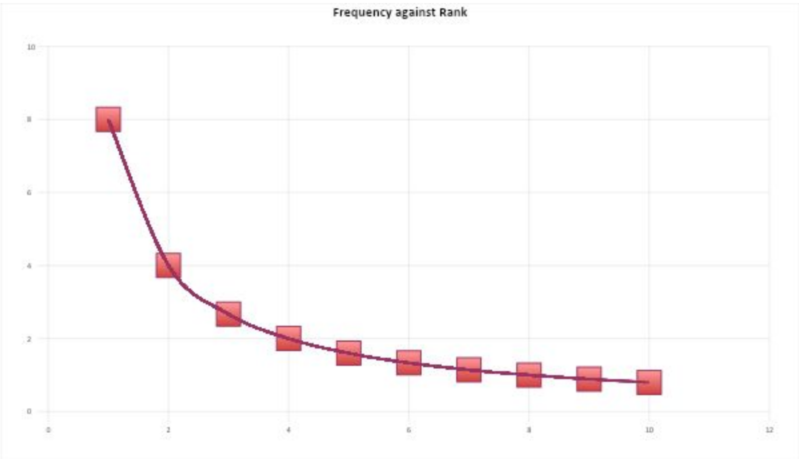
ferrari

The nearest neighbour of a word is often an antonym (or co-hyponym). Why might this be a problem?

Sparsity

- Zipf’s Law: **“The product of the frequency of a word and its rank is approximately constant.”**

Rank	1/Rank	Freq
1	8	8
2	4	4
3	2.667	3
4	2	2
5	1.6	2
6	1.333	1
7	1.143	1
8	1	1
9	0.889	1
10	0.8	1
	23.43	24



Hapax Legomena : words which only occur once. However large the corpus, these make up approximate half the vocabulary.

Consequences of Zipf's Law

- 100k dimensional co-occurrence vectors will be very sparse (lots of zeros)
- difficult to compare vectors because of all of this unseen stuff
- What can we do?

Coming up

- Solutions to this problem (week 4):
  - Smoothing
  - Dimensionality reduction
  - Language models with fixed dimensionality e.g., recurrent neural network language models (RNNLMs)

- Probabilistic language models (week 2)
  - n-gram modelling
  - evaluation and perplexity
  - generalization and smoothing

Reading

- Week 1 seminar:
  - Pedersen (2010): Information Content Measures of Semantic Similarity Perform Better without Sense Tagged Text

References

1. Christiane Fellbaum (1998, ed.) WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
2. Lin, D. (1998a) Automatic retrieval and clustering of similar words. In Proceedings of COLING/ACL.
3. Lin, D (1998b) An information-theoretic definition of similarity. In ICML 1998, San Francisco, pp 296-304
4. Mikolov, Yih and Zweig 2013 – Linguistic Regularities in Continuous Space Word Representations, (NAACL-HCT 2013)
5. Pedersen, T. (2010). Information Content Measures of Semantic Similarity Perform Better without Sense Tagged Text
6. Resnik, P (1995) Using information content to evaluate semantic similarity in a taxonomy. In IJCAI-95, pp. 448-424
7. Zweig, G. and Burges, A. 2011. The Microsoft Research Sentence Completion Challenge. Microsoft Technical Report