ANLP

03 - Words (Part I)

David Schlangen
University of Potsdam, MSc Cognitive Systems
Winter 2019 / 2020

assignments

- assignment 1 will be out today
- make sure you have a python 3 environment installed, with Jupyter Notebook (we recommend the anaconda distribution)

recap prob. theory

- probability space: $(\Omega, \mathscr{F}, P)$
- random variables: $X: \Omega \rightarrow \mathbb{R}^n$
- probability mass function: $p(x) = P(X = x)$…
- …where $X=x$ is the set $\{\omega \in \Omega : X(\omega) = x\}$
- $P(X)$ is *distribution* over all events denotable by X
- $p(x)$ [or $P(X=x)$] is probability of this one event

recap prob. theory

- all relevant rules carry over to this notation:
  - $P(X|Y) = P(X,Y) / P(Y)$
  - $P(X,Y) = P(Y)\, P(X|Y)$
  - $P(X|Y) = P(Y|X)\, P(X) / P(Y)$
  - $P(X,Y) = P(X)\, P(Y)$ iff. X, Y *independent*
  - $P(X|Y) = P(X)$ iff. X, Y *independent*.
  - $P(X) = \sum_y P(X,Y=y)$, or $p_X(x) = \sum_y p(x,y)$
- can say that X behaves according to known distr. w/ parameter(s), e.g. $X \sim Be(p)$ [Bernoulli]
- can talk about *sampling from P(X)*, i.e., choosing a value for X at random, according to the probabilities specified by distribution P(X).

Example: model estimation

- Example: we flip a coin 100 times and observe H 61 times. Should we believe that it is a fair coin?
  - observation: 61x H, 39x T
  - model: assume rv X follows a *Bernoulli* distribution, i.e. X has two outcomes, and there is a value p such that $P(X = H) = p$ and $P(X = T) = 1 - p$.
  - want to estimate the *parameter* p of this model

Estimation of P

- Frequentist statistics
  - parametric methods
  - non-parametric (distribution-free)
- Bayesian statistics

Frequentist Statistics

- Relative frequency: proportion of times an outcome *u* occurs

$$f_u = C(u) / N$$

- C(u) is the number of times u occurs in N trials
- For N approaching infinity, the relative frequency tends to stabilize around some number: probability estimates

Non-Parametric Methods

- No assumption about the underlying distribution of the data
- For ex, simply estimate P empirically by counting a large number of random events is a distribution-free method
- Less prior information, more training data needed

Parametric Methods

- Assume that some phenomenon in language is acceptably modeled by one of the well-known families of distributions (such as binomial, normal)
- We have an explicit probabilistic model of the process by which the data was generated, and determining a particular probability distribution within the family requires only the specification of a few parameters (less training data)

Binomial Distribution

- Series of trials with only two outcomes, each trial being independent from all the others
- Number *r* of successes out of *n* trials given that the probability of success in any trial is *p*:

$$b(r;n,p) = \binom{n}{r} p^r (1-p)^{n-r}$$

Bin. Distribution – Examples

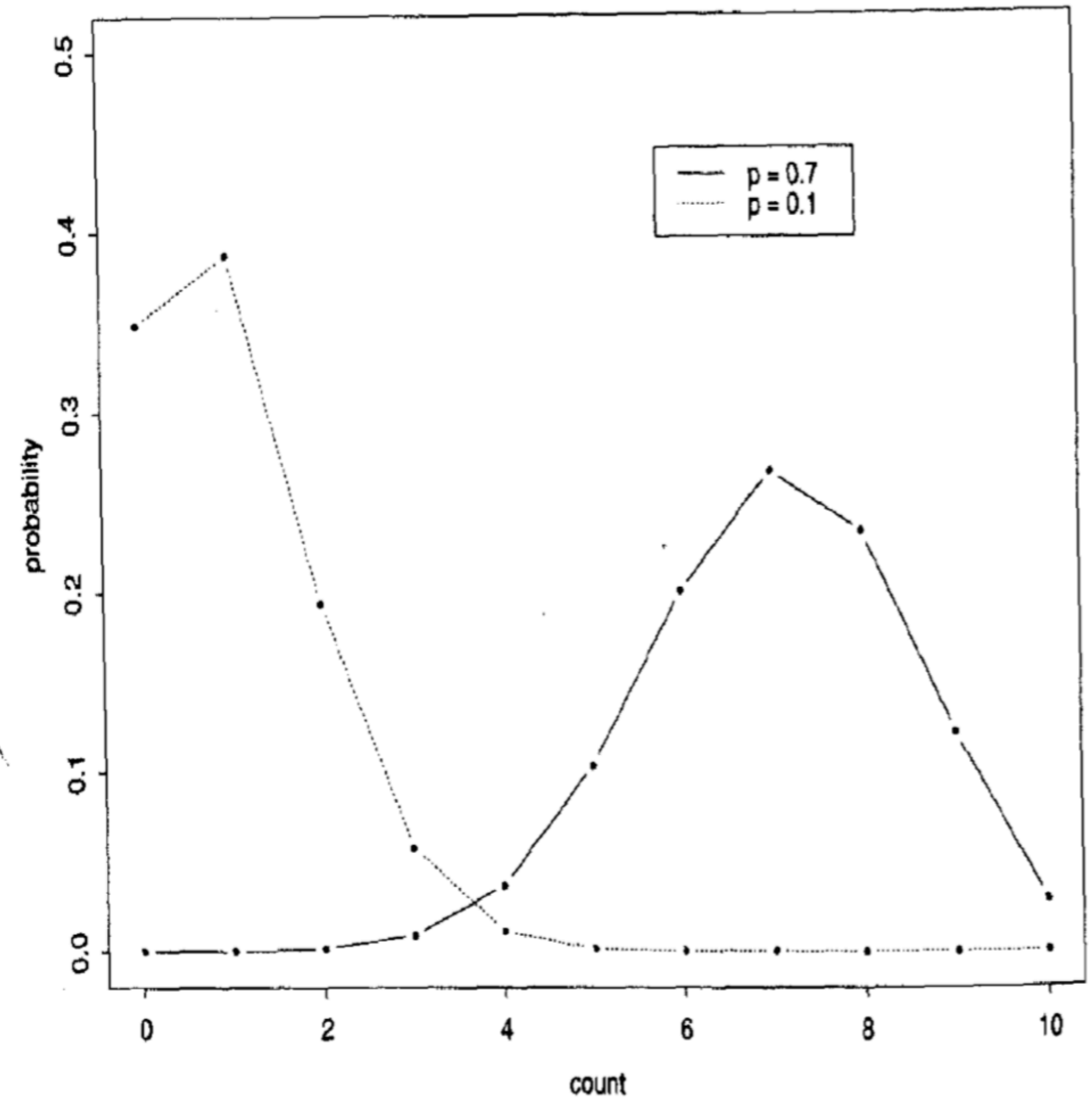


Figure 2.3 Two examples of binomial distributions: $b(r; 10, 0.7)$ and $b(r; 10, 0.1)$.

M/S, Fig. 2.3

Maximum Likelihood Estimation

- We want to estimate the parameters of our model from frequency observations. There are many ways to do this. For now, we focus on *maximum likelihood estimation*, MLE.
- *Likelihood* L(O ; p) is the probability of our model generating the observations O, given parameter values p.
- Goal: Find value for parameters that maximizes the likelihood.

ML Estimation

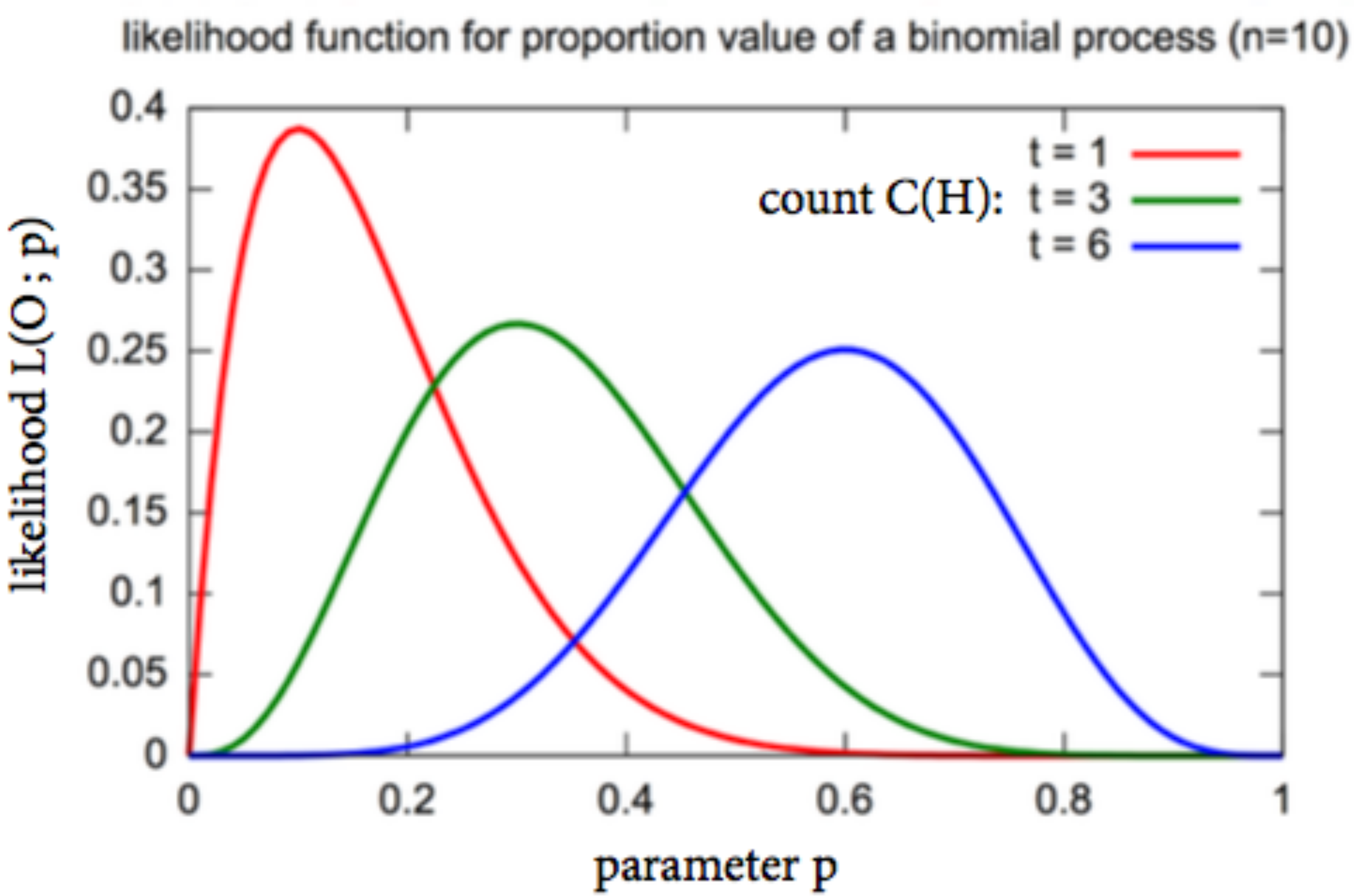
- For Bernoulli and multinomial models, it is extremely easy to estimate the parameters that maximize the likelihood:
  - P(X = a) = f(a)
  - in the coin example above, just take p = f(H)
- Why is this?

Bernoulli model

- Let's say we had training data C of size N, and we had N_H observations of H and N_T observations of T.

$$\text{likelihood } L(C) = \prod_{i=1}^{N} P(w_i \mid p) = \quad p^{N_H}(1-p)^{N_T}$$

Likelihood functions

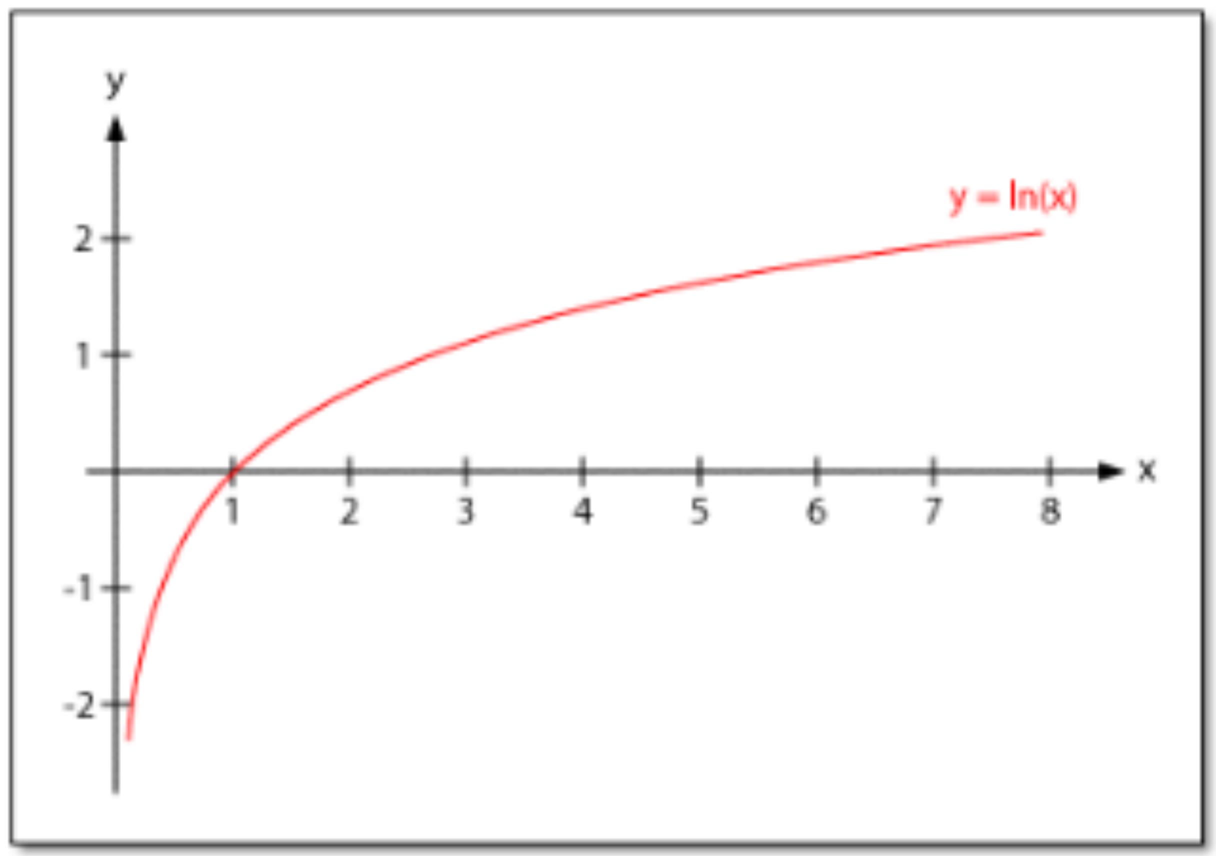


Bernoulli model

- Let's say we had training data C of size N, and we had N_H observations of H and N_T observations of T.

$$\text{likelihood} \quad L(C) = \prod_{i=1}^{N} P(w_i \mid p) = \quad p^{N_H}(1-p)^{N_T}$$

Logarithm is monotonic



- Observation: If $x_1 > x_2$, then $\ln(x_1) > \ln(x_2)$.
- Therefore, $\operatorname{argmax}_p L(C) = \operatorname{argmax}_p l(C)$

Bernoulli model

- Let's say we had training data C of size N, and we had N_H observations of H and N_T observations of T.

likelihood $L(C) = \prod_{i=1}^{N} P(w_i \mid p) = \quad p^{N_H}(1-p)^{N_T}$

log-likelihood

$$\ell(C) = \log L(C) =$$

Maximizing the log-likelihood

- Find maximum of function by setting derivative to zero:

$$\ell(C) = N_H \log p + N_T \log(1-p)$$

$$\frac{d\ell(C)}{dp} =$$

- Solution is $p = N_H / N = f(H)$.

Maximizing the log-likelihood

- Find maximum of function by setting derivative to zero:

$$\ell(C) = N_H \log p + N_T \log(1-p)$$

$$\frac{d\ell(C)}{dp} = \frac{N_H}{p} - \frac{N_T}{1-p}$$

- Solution is $p = N_H / N = f(H)$.

More complex models

- Many, many models we use in NLP are *multinomial* probability distributions. More than two outcomes possible; think dice rolling.
- MLE result generalizes to multinomial models: $P(X = a) = f(a)$.
- Maximizing log-likelihood uses technique called *Lagrange multipliers* to ensure parameters sum to 1.
- If you want to see the details, see background material on the website.

Conclusion

- Probability theory is essential tool in modern NLP.
- Important concepts today:
  - random variable, probability distribution
  - joint and conditional probs; Bayes' rule; independence
  - expected values
  - statistical models; parameters; likelihood; MLE
- We will use all of these concepts again and again in this course. If you have questions, ask early.

Words

how many words do you see?

- *I haven't seen it yet. (I think. :-)*
- `I have n't seen it yet . ( I think :-)`
- One possibility, answer would be 11.
- Or just 10? Or just 7, 6?
- *To be or not to be ..* 6 words, or 4?
    - **Type**: Abstract concept.
    - **Token**: Concrete instantiation of concept.

tokenization

- *To be or not to be*
  - just split at whitespaces, easy!
- *I haven’t seen it yet. (I think. :-)*
  - “haven’t” “yet.” “(I”, “think.”
- other languages:

ลูกศิษย์วัดกระทิงยังยื้อปิดถนนทางขึ้นไปนมัสการพระบาทเขาคิชฌกูฏ หวิดปะทะ
กับเจ้าถิ่นที่ออกมาเผชิญหน้าเพราะเดือดร้อนสัญจรไม่ได้ ผวจ.เร่งทุกฝ่ายเจรจา
ก่อนที่ชื่อเสียงของจังหวัดจะเสียหายไปมากกว่านี้ พร้อมเสนอหยุดจัดงาน 15 วัน....

(example by
Noah Smith)

tokenization

- tokenization is the task of splitting a text into word tokens
- suprisingly non-trivial, even in languages (writing systems) that you’d think make it easy

one word token = one "meaning"?

uygarlaştıramadıklarımızdanmışsınızcasına
"(behaving) as if you are among those whom we could not civilize"

TIFGOSH ET HA-LELED BA-GAN
"you will meet the boy in the park"

unfriend, Obamacare, Manfuckinghattan

(examples by Noah Smith)

- 分かりました　　*I have understood*

one word type = one “meaning”?

I looked at the table in front of me…

… and got hungry.

Each trigonometric function in terms of the other five.[2]

in terms of	$\sin\theta$	$\cos\theta$	$\tan\theta$	$\csc\theta$	$\sec\theta$	$\cot\theta$
$\sin\theta =$	$\sin\theta$	$\pm\sqrt{1-\cos^2\theta}$	$\pm\frac{\tan\theta}{\sqrt{1+\tan^2\theta}}$	$\frac{1}{\csc\theta}$	$\pm\frac{\sqrt{\sec^2\theta-1}}{\sec\theta}$	$\pm\frac{1}{\sqrt{1+\cot^2\theta}}$
$\cos\theta =$	$\pm\sqrt{1-\sin^2\theta}$	$\cos\theta$	$\pm\frac{1}{\sqrt{1+\tan^2\theta}}$	$\pm\frac{\sqrt{\csc^2\theta-1}}{\csc\theta}$	$\frac{1}{\sec\theta}$	$\pm\frac{\cot\theta}{\sqrt{1+\cot^2\theta}}$
$\tan\theta =$	$\pm\frac{\sin\theta}{\sqrt{1-\sin^2\theta}}$	$\pm\frac{\sqrt{1-\cos^2\theta}}{\cos\theta}$	$\tan\theta$	$\pm\frac{1}{\sqrt{\csc^2\theta-1}}$	$\pm\sqrt{\sec^2\theta-1}$	$\frac{1}{\cot\theta}$
$\csc\theta =$	$\frac{1}{\sin\theta}$	$\pm\frac{1}{\sqrt{1-\cos^2\theta}}$	$\pm\frac{\sqrt{1+\tan^2\theta}}{\tan\theta}$	$\csc\theta$	$\pm\frac{\sec\theta}{\sqrt{\sec^2\theta-1}}$	$\pm\sqrt{1+\cot^2\theta}$
$\sec\theta =$	$\pm\frac{1}{\sqrt{1-\sin^2\theta}}$	$\frac{1}{\cos\theta}$	$\pm\sqrt{1+\tan^2\theta}$	$\pm\frac{\csc\theta}{\sqrt{\csc^2\theta-1}}$	$\sec\theta$	$\pm\frac{\sqrt{1+\cot^2\theta}}{\cot\theta}$
$\cot\theta =$	$\pm\frac{\sqrt{1-\sin^2\theta}}{\sin\theta}$	$\pm\frac{\cos\theta}{\sqrt{1-\cos^2\theta}}$	$\frac{1}{\tan\theta}$	$\pm\sqrt{\csc^2\theta-1}$	$\pm\frac{1}{\sqrt{\sec^2\theta-1}}$	$\cot\theta$

… and despaired.

working definition

- “word” as grammatically defined notion; stable re-currences of closely related forms fulfilling similar functions and making similar contributions in context
- substitutability tests, independence tests, etc.
- “fly” as noun or verb: different words
- “table” as on previous slide: same word, different *senses*

representing word types (part I)

- let’s assume that “fly” and “flies” are tokens of same type
- and “Haus”, “Hauses”, “Häuser”, “Hause”, ..
- and “walk”, “walking”, “walked”, …
- represent the *lexeme* with one single form (e.g., nominative singular; infinitive): *fly, Haus, to walk*
- in technical applications, this is not always done. *Lemmatisation* is not a trivial task.

how are word (types) distributed in corpora?

- corpus (pl. corpora) = collection of text
- vocabulary (w.r.t. a corpus) = the set of word types that occur in it
- are word types distributed uniformly? If I randomly grab a word from a text, are all words from vocab equally likely?
- *empirical law* (regularity that describes empirical finding):

  frequency is proportional to inverse rank. $f \propto 1/r$

  (second most frequent word will appear half as often as most freq., and so on.)

Zipf’s law

- power law
- described by George Kingsley Zipf in 1949
- ties it to Principle of Least Effort:
  - for speaker, few massively ambiguous words would be nice (little memory needed)
  - for hearer, lots of un-ambiguous words would be nice (little decoding needed)
  - let’s compromise!

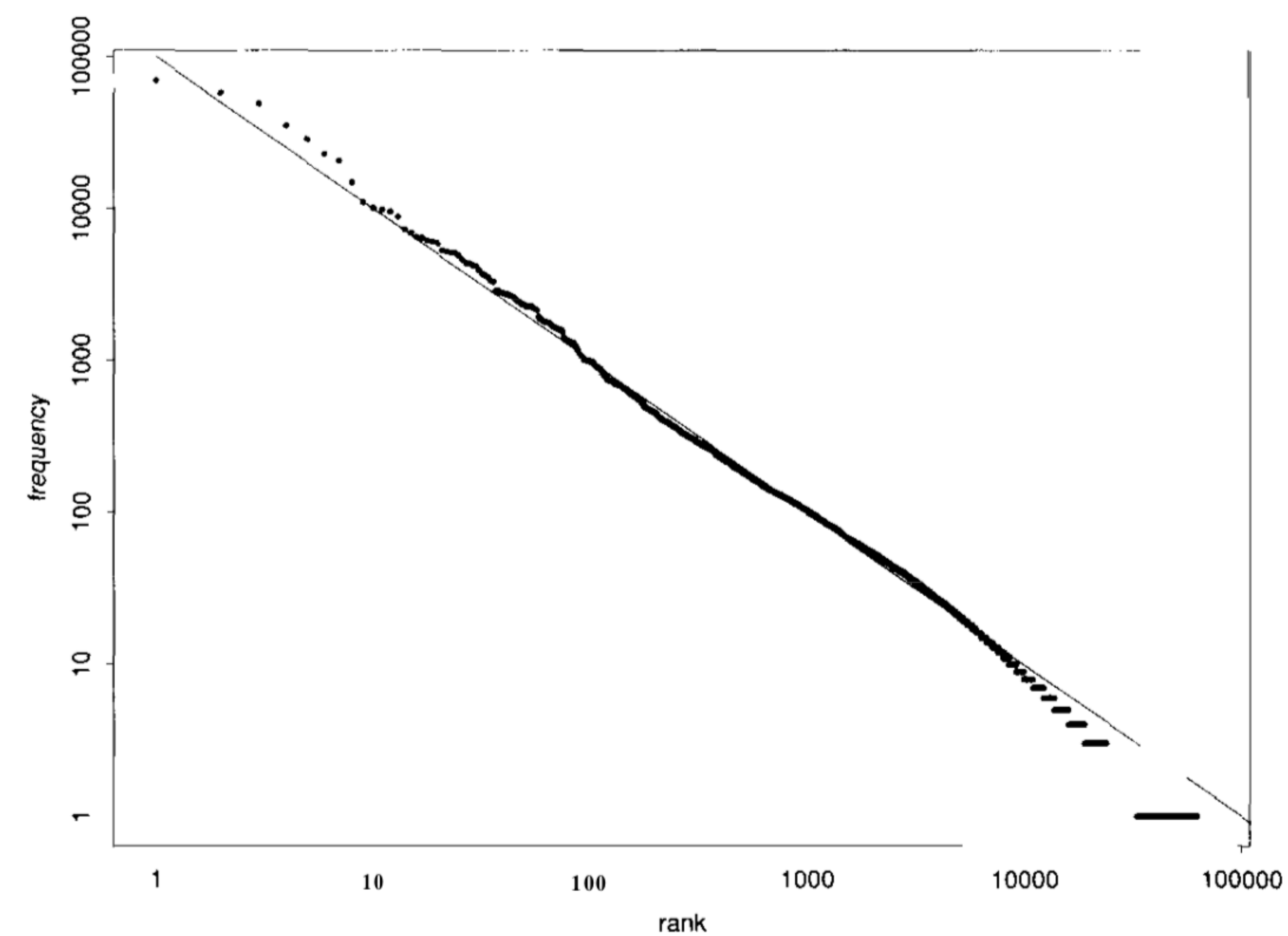


Figure 1.1 Zipf’s law. The graph shows rank on the X-axis versus frequency on the Y-axis, using logarithmic scales. The points correspond to the ranks and frequencies of the words in one corpus (the Brown corpus). The line is the relationship between rank and frequency predicted by Zipf for k = 100,000, that is f x r = 100,000.

challenges

- it’s not always easy to split texts into tokens. (there may not even be one single right way.)
- it’s not always easy to split tokens into meaning-parts
- even the same word form parts can have different meanings
- you will encounter most words only rarely

lexical knowledge

- what is *linguistic* knowledge of a word?
  - sound (pronunciation), orthography
  - morphology: how to build plural, declination / conjugation, etc.
  - syntax: its part of speech, other things like preposition selection (*proud of* / **in*)
  - other selectional restrictions:
    strong tea / *?powerful tea*
    powerful computer / *?strong computer*
- what is *not* linguistic knowledge about a word?

what's the size of the vocabulary?

- depends on size of corpus… large corpus, easily 100s of 1000s.
  - and most will have been *hapax legomena*
- what's the size of *your* vocabular?
  - Jean Aitchison (2003): 60,000 words by end of high-school

representing words (cont.)

- just store the string of the lemma?
  - how much space? is space used well?
- determine vocabulary, and store as index into vocab: `vocab2index[“fly”]`
- “this is a text, is what this is.” —> [387, 23, 11, 4096, 23, 9544, 387, 23]
- or, {11, 23, 387, 4096, 9544} ? *set of words* representation
  [11, 11, 23, 23, 23, 387, 387, 4096, 9544] *bag of words* repr.
- reversible?

representing words (cont.)

- N.B.: Zipf's law tells us that no matter how large our corpus, there will always be words that we didn't encounter

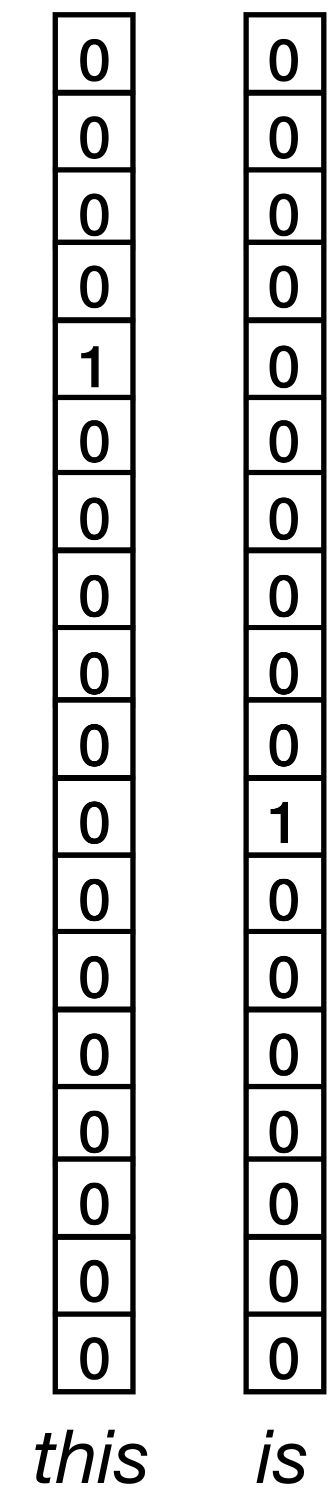
- so need to reserve index position for `UNK`, the unknown word, and map words we encounter that are not in index to that

representing words (cont.)

- remember: the job of a representation is to be *useful*
- what can we do with these index representations? can we do any operations on numbers that we know and have them make sense?
- “this is a text, is what this is.” —> +([387, 23, 11, 4096, 23, 9544, 387, 23]) = 14,494
- ???

representing words (cont.)

- a different encoding: *one-hot*
- f : V → $R^{|V|}$,
  f(w) = **x**, x_i = 1 iff `vocab2index[w]` = i,
  else 0
- now I have vectors… any improvement?



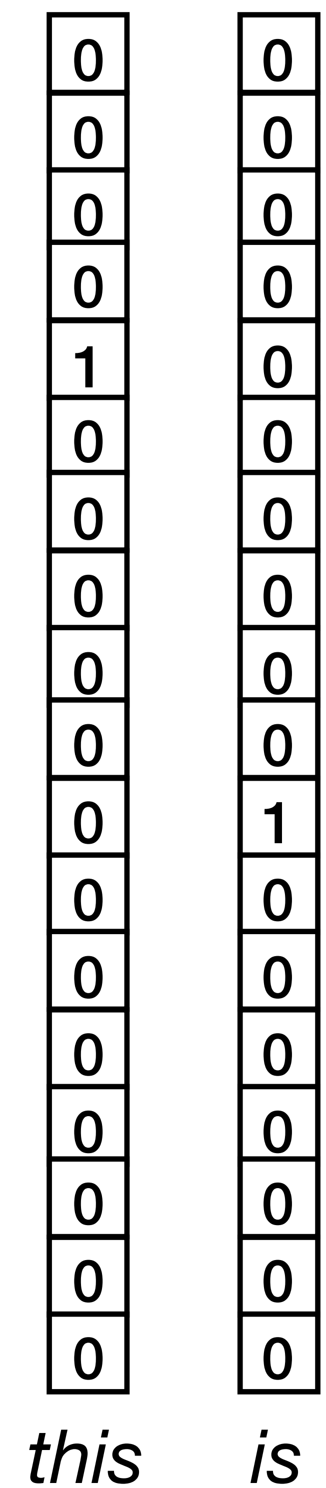
representing words (cont.)

- a different encoding: *one-hot*
- $f: V \rightarrow R^{|V|}$,
  $f(w) = \mathbf{x}$, $x_i = 1$ iff `vocab2index[w]` = i, else 0
- now I have vectors… any improvement?
- addition now realises *bag of words* representation for texts!

$$\underset{\textit{this}}{\begin{bmatrix}0\\0\\0\\0\\1\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\end{bmatrix}} + \underset{\textit{is}}{\begin{bmatrix}0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\1\\0\\0\\0\\0\\0\\0\\0\end{bmatrix}} = \begin{bmatrix}0\\0\\0\\0\\1\\0\\0\\0\\0\\0\\1\\0\\0\\0\\0\\0\\0\\0\end{bmatrix}$$

representing words (cont.)

- this represents word *identities*
- can we represent word *meanings*?



representing word meanings

- let's call the word meanings *concepts*
- can we represent concepts?
- classic attempt: represent them via *definitions*
    - *bachelor:* $\forall x$ (bachelor(x) $\rightarrow$ male(x) & unmarried(x))

Classical ("Aristotelian") Theory of Concepts

The meaning of a word:

a concept defined by **necessary** and **sufficient** conditions

A **necessary** condition for being an X is a condition C that X must satisfy in order for it to be an X.

- If not C, then not X
- "Having four sides" is necessary to be a square.

A **sufficient** condition for being an X is condition such that if something satisfies condition C, then it must be an X.

- If and only if C, then X
- The following necessary conditions, jointly, are sufficient to be a square
  - x has (exactly) four sides
  - each of x's sides is straight
  - x is a closed figure
  - x lies in a plane
  - each of x's sides is equal in length to each of the others
  - each of x's interior angles is equal to the others (right angles)
  - the sides of x are joined at their ends

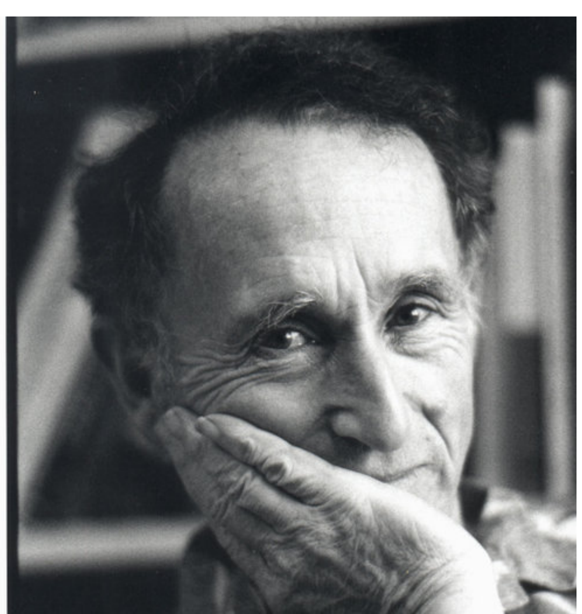
Example from Norman Swartz, SFU

Problem 1: The features are complex and may be context-dependent

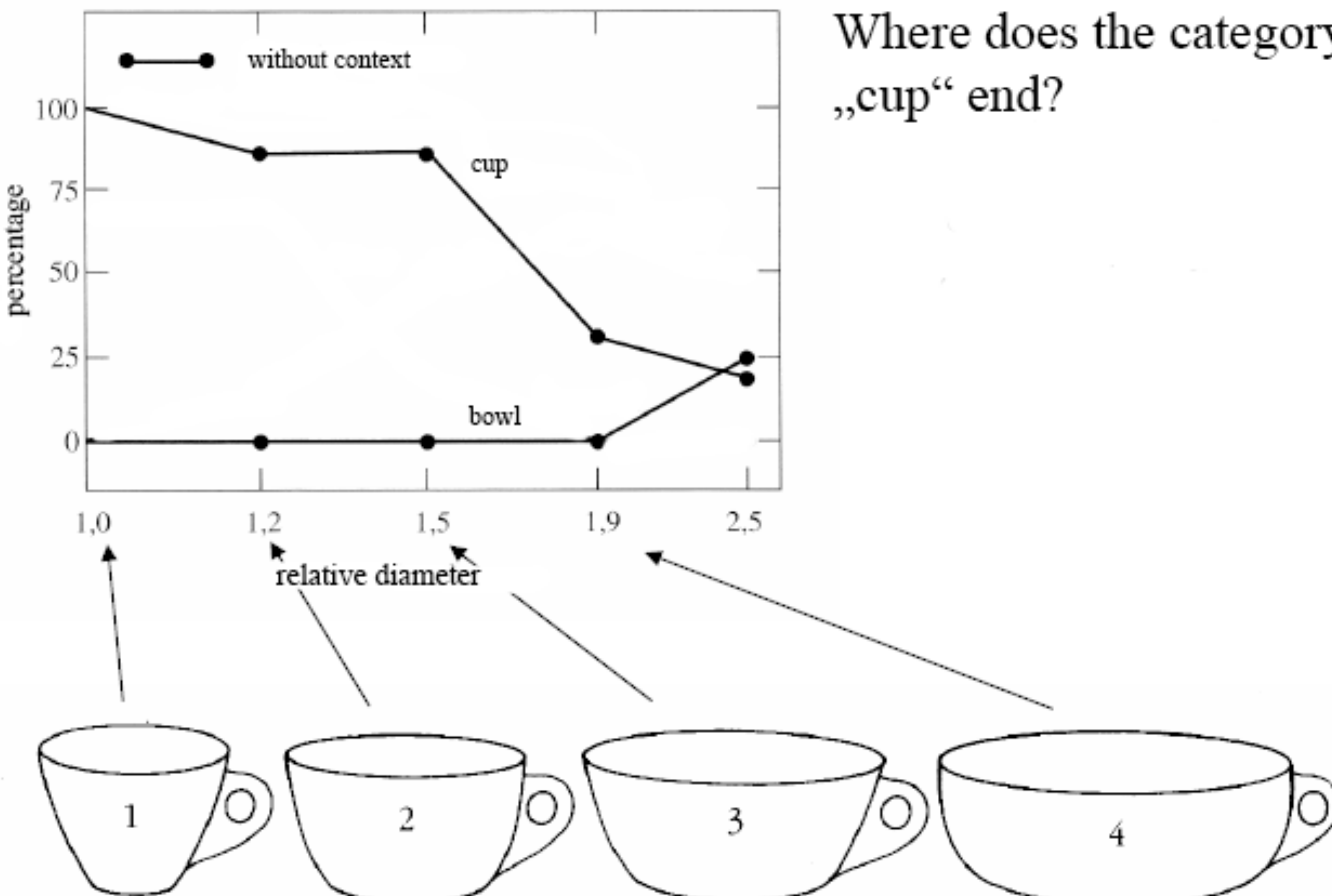
William Labov. 1975

What are these?

Cup or bowl?

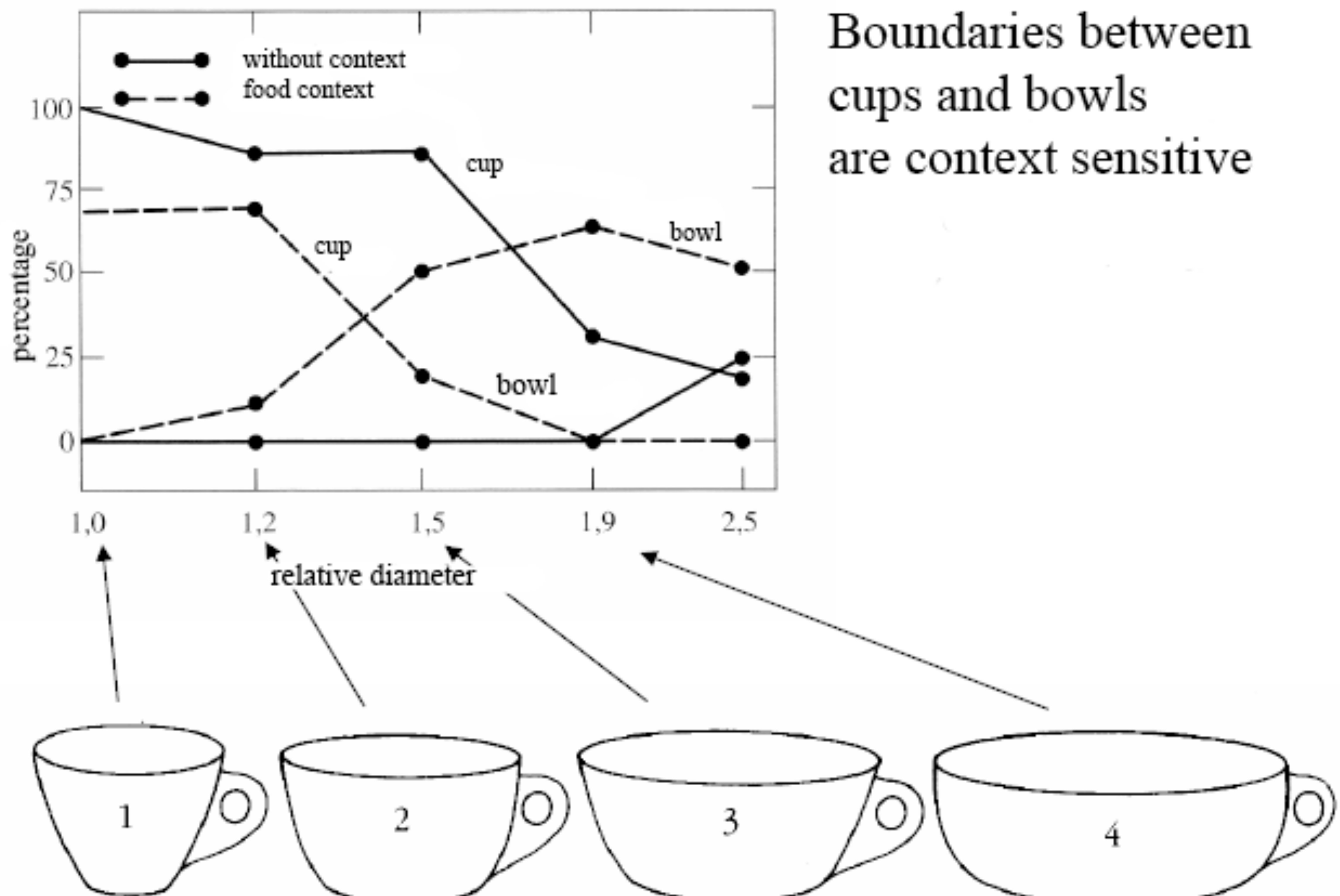


The category depends on complex features of the object (diameter, etc)



Where does the category „cup" end?

The category depends on the context! (If there is food in it, it's a bowl)



Boundaries between cups and bowls are context sensitive

Labov's definition of cup

The term *cup* is used to denote round containers with a ratio of depth to width of $1\pm r$ where $r \leq r_b$, and $r_b = \alpha_1 + \alpha_2 + \ldots \alpha_v$ and α_1 is a positive quality when the feature i is present and 0 otherwise.

feature 1 = with one handle
2 = made of opaque vitreous material
3 = used for consumption of food
4 = used for the consumption of liquid food
5 = used for consumption of hot liquid food
6 = with a saucer
7 = tapering
8 = circular in cross-section

Cup is used variably to denote such containers with ratios width to depth $1\pm r$ where $r_b \leq r \leq r_1$ with a probability of $r_1 - r/r_t - r_b$. The quantity $1\pm r_b$ expresses the distance from the modal value of width to height.

representing word meanings

- let’s call the word meanings *concepts*
- can we represent concepts?
- classic attempt: represent them via *definitions*
- problems:
  - all definitions *leak*
  - circularity
  - context-dependence

representing word meanings

- let's try something else:
  - a cup is kind of like a mug or a bowl, or a glass (all *containers*); and also like a plate (*crockery)*…
- *relations* to other concepts
- remember what we said about representations, want them to capture relations between objects
- how is "cup" in one-hot representation related to "mug", and to "car"?

meaning relations

- *polysemy*: one word, several (related) senses. “read a book” / “bought a book” (meaning / sense relation)

- *homonymy*: one word, several (unrelated) senses. (“bank” / “bank”, “table” / “table”?) .. Zeugma test: “Today, I went to and sat on a bank”. “Lufthansa serves Frankfurt and lunch.”

- *synonymy*: several words, same sense.
    - filbert / hazelnut
    - couch / sofa
    - big / large
    - automobile / car
    - vomit / throw up
    - Water / H20

Relation: Synonymity

Note that there are probably no examples of perfect synonymy.

- Even if many aspects of meaning are identical
- Still may not preserve the acceptability based on notions of politeness, slang, register, genre, etc.

The Linguistic Principle of Contrast:

- Difference in form -> difference in meaning

Relation: Synonymity?

Water/H_20

Big/large

Brave/courageous

Relation: Antonymy

Senses that are opposites with respect to one feature of meaning

Otherwise, they are very similar!

`dark/light` `short/long` `fast/slow` `rise/fall`
`hot/cold` `up/down` `in/out`

More formally: antonyms can

- define a binary opposition
  or be at opposite ends of a scale
  - `long/short, fast/slow`
- Be *reversives*:
  - `rise/fall, up/down`

Relation: Similarity

Words with similar meanings. Not synonyms, but sharing some element of meaning

```
car, bicycle
cow, horse
```

Relation: Word relatedness

Also called "word association"

Words be related in any way, perhaps via a semantic frame or field

- `car, bicycle`: **similar**
- `car, gasoline`: **related**, not similar

Semantic field

Words that

- cover a particular semantic domain
- bear structured relations with each other.

hospitals

surgeon, *scalpel*, *nurse*, *anaesthetic*, *hospital*

restaurants

waiter, *menu*, *plate*, *food*, *menu*, *chef*),

houses

door, *roof*, *kitchen*, *family*, *bed*

Relation: Superordinate (hyperonym) / subordinate (hyponym)

One sense is a **subordinate** of another if the first sense is more specific, denoting a subclass of the other

- *car* is a subordinate of *vehicle*
- *mango* is a subordinate of *fruit*

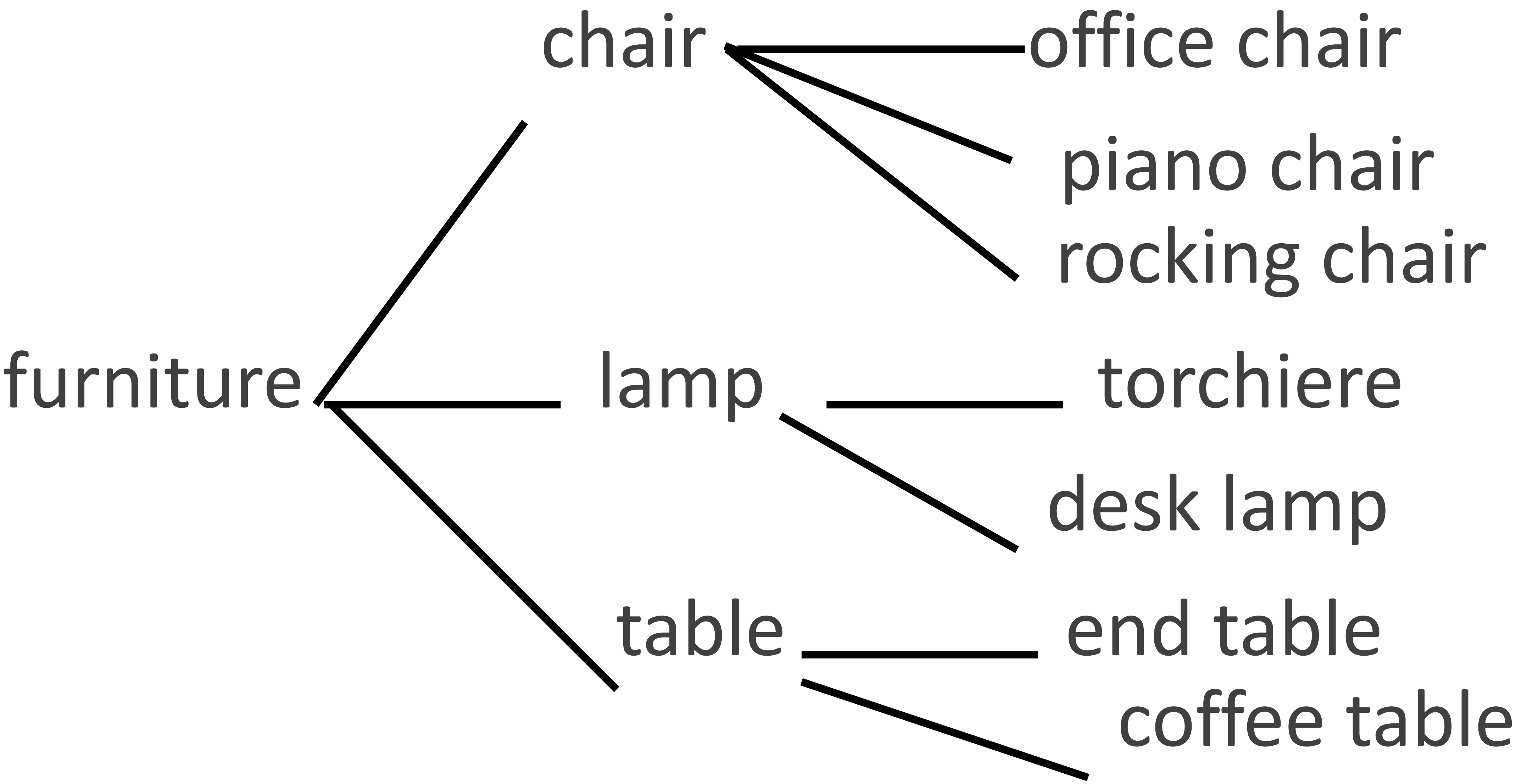
Conversely **superordinate**

- *vehicle* is a superordinate of *car*
- *fruit* is a subodinate of *mango*

Superordinate	vehicle	fruit	furniture
Subordinate	car	mango	chair

Name these items

Superordinate **Basic** **Subordinate**



Cluster of Interactional Properties

Basic level things are “human-sized”

Consider chairs

- We know how to interact with a chair (sitting)
- Not so clear for superordinate categories like furniture
  - “Imagine a furniture without thinking of a bed/table/chair/specific basic-level category”

The basic level

Is the level of distinctive actions

Is the level which is learned earliest and at which things are first named

It is the level at which names are shortest and used most frequently

Connotation

Words have **affective** meanings

positive connotations (*happy*)

negative connotations (*sad*)

positive evaluation (*great*, *love*)

negative evaluation (*terrible*, *hate*).

So far

Concepts or word senses

- Have a complex many-to-many association with **words** (homonymy, multiple senses)

Have relations with each other

- Synonymy
- Antonymy
- Similarity
- Relatedness
- Superordinate/subordinate
- Connotation

representing relations

- WordNet (Fellbaum *et al.* 1998): a lexical database of nouns, verbs, adjectives, and their relations
- manually curated
- originally built to support psycholinguistic research
- word senses represented as *synsets*
- relations expressed as typed edges in graph

Relations

Relation	Also Called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	$breakfast^1 \rightarrow meal^1$
Hyponym	Subordinate	From concepts to subtypes	$meal^1 \rightarrow lunch^1$
Instance Hypernym	Instance	From instances to their concepts	$Austen^1 \rightarrow author^1$
Instance Hyponym	Has-Instance	From concepts to concept instances	$composer^1 \rightarrow Bach^1$
Member Meronym	Has-Member	From groups to their members	$faculty^2 \rightarrow professor^1$
Member Holonym	Member-Of	From members to their groups	$copilot^1 \rightarrow crew^1$
Part Meronym	Has-Part	From wholes to parts	$table^2 \rightarrow leg^3$
Part Holonym	Part-Of	From parts to wholes	$course^7 \rightarrow meal^1$
Substance Meronym		From substances to their subparts	$water^1 \rightarrow oxygen^1$
Substance Holonym		From parts of substances to wholes	$gin^1 \rightarrow martini^1$
Antonym		Semantic opposition between lemmas	$leader^1 \Longleftrightarrow follower^1$
Derivationally Related Form		Lemmas w/same morphological root	$destruction^1 \Longleftrightarrow destroy^1$

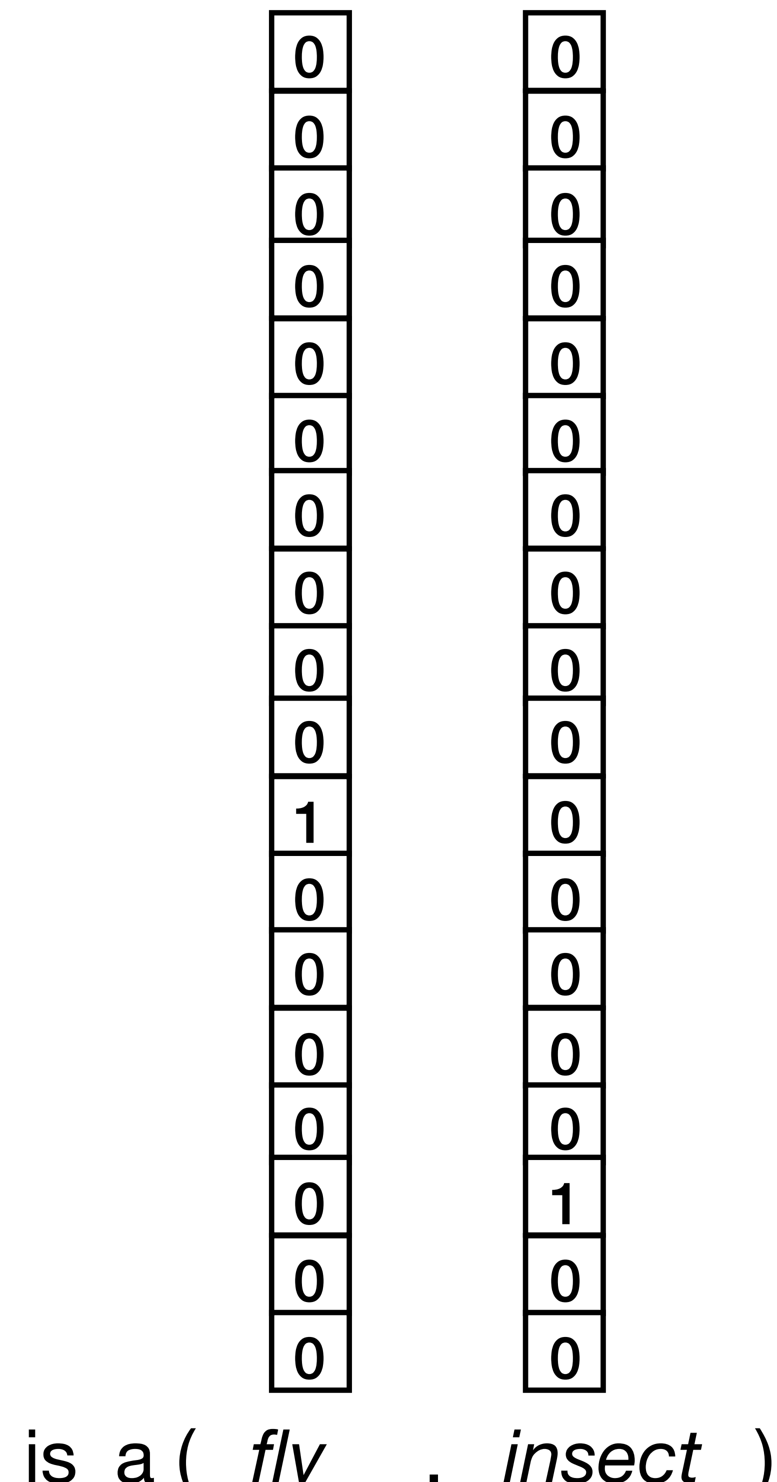
Figure 17.2 Noun relations in WordNet.

representing relations

- WordNet is a useful resource that has been used in a lot of computational research as well
- play around with it online , also see assignment 1
- problems:
  - manually assembled, only grows when more work is done
  - sometimes dubious, or at least debateable decisions… ontology building (hypernym relation) is hard!
  - some parts of graph are very dense, others rather spare
  - needs relatively complex computational machinery (graph algorithms)
  - this is *added* to representation of word identity

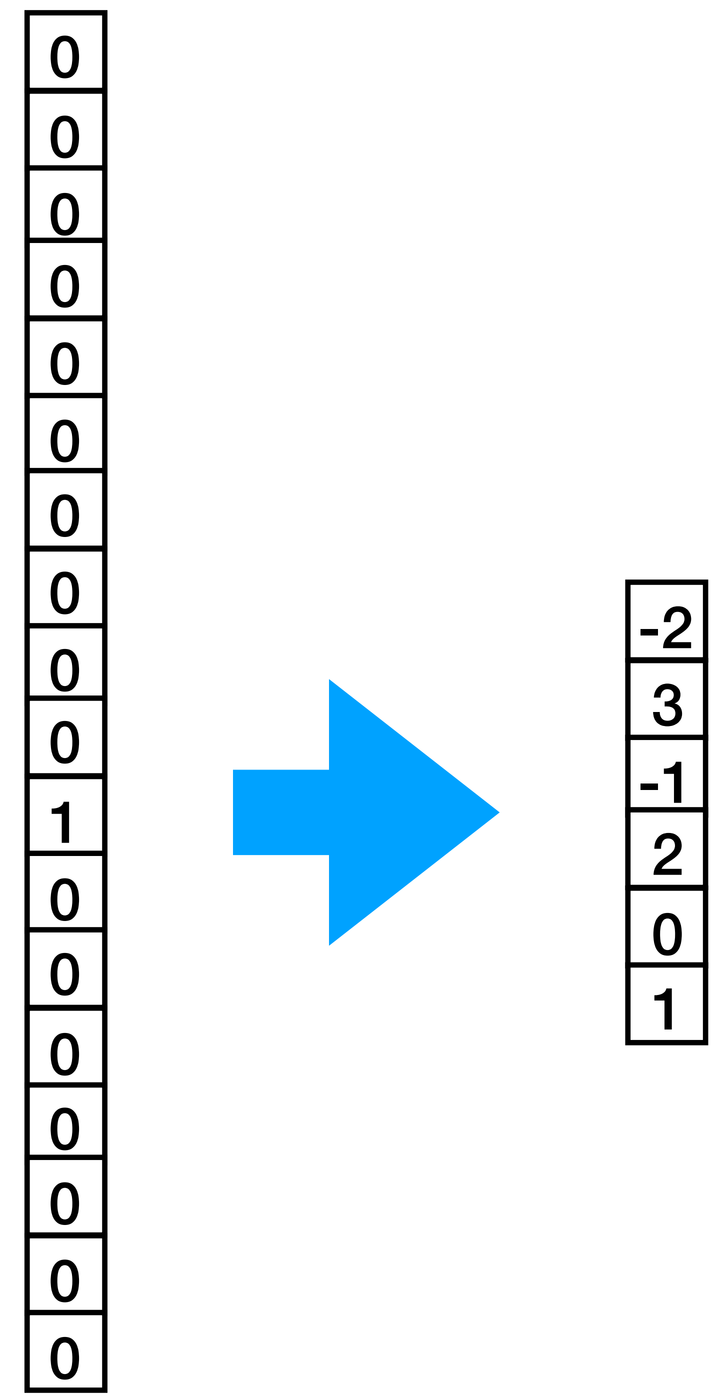
taking stock

- to represent words to machine, we now have one-hot vector (of size |V+1|), plus relations (= pairs of such vectors)…
- could now represent pair of words through 2 vectors + one hot vector over relation types
- but wouldn't it be nice to represent word identity and word meaning in the same way?



outlook

- we'll get this from *distributed representations*
- Hinton (1984): "Each entity is represented by a pattern of activity distributed over many computing elements, and each computing element is involved in representing many different entities"
- the inherent *similarity* between vectors will (magically?) represent similarity between words!

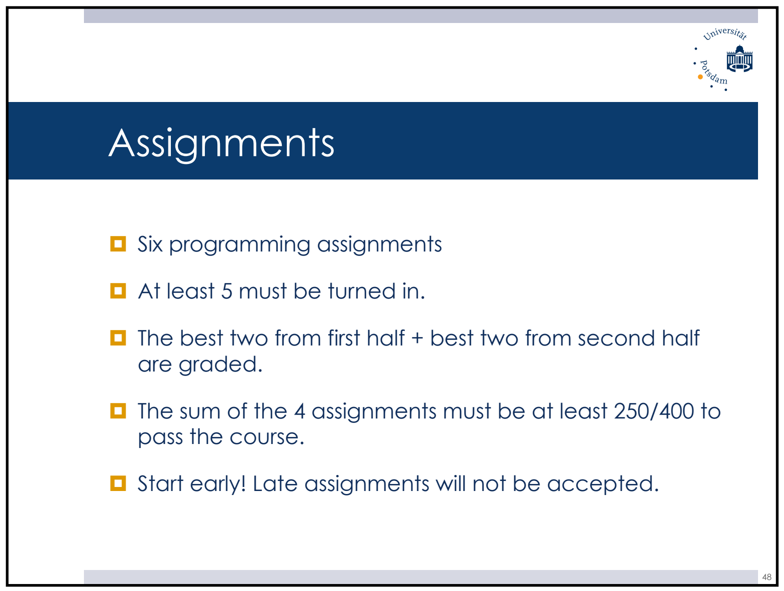


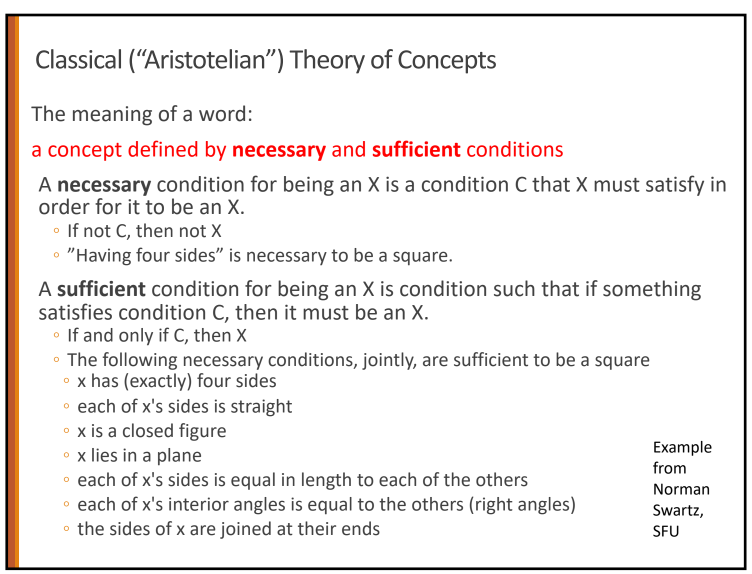
fly

Questions, Queries, Comments?

slide credits

slides that look like this





come from

earlier editions of this class (ANLP), given by Tatjana Scheffler and Alexander Koller

Dan Jurafsky's slide deck for J&M

and their use is gratefully acknowledged. I try to make any modifications obvious, but if there are errors on a slide, assume that I added them.