Semicompositionality

András Kornai
SZTAKI Computer Science Research Institute
and
Dept of Algebra, Budapest Institute of Technology

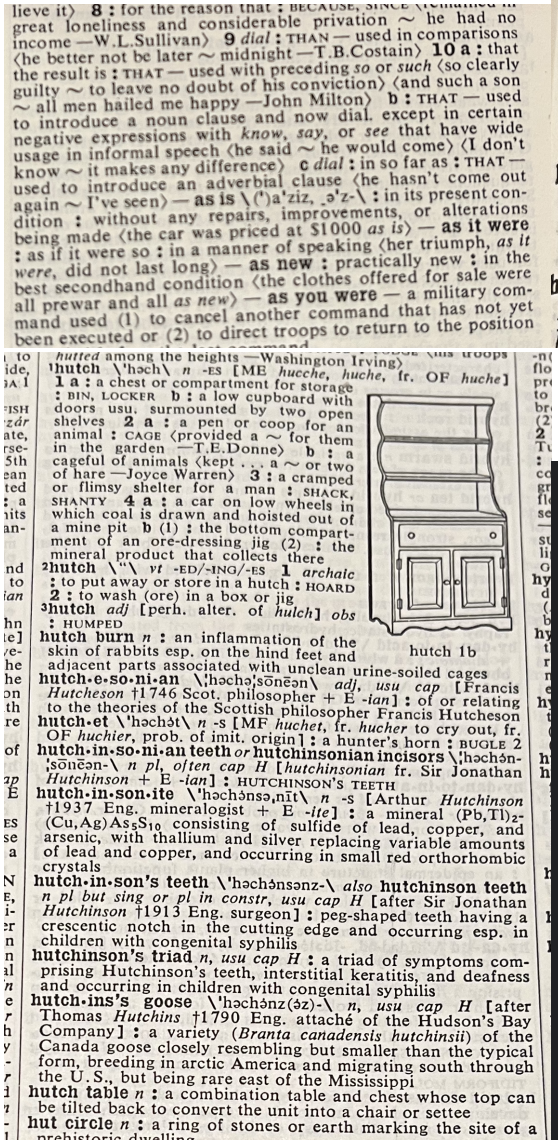
NASSLLI, June 25 2025

Outline

Lexicon or encyclopedia

- In many topics, technical vocabulary is key
- Proper names and named entities
- PER, LOC, ORG – hundreds of millions of entries in each category
- *hutch for sale, as is*

HUTCH, AS IS


lieve it) **8 :** for the reason that : BECAUSE, SINCE great loneliness and considerable privation ~ he had no income —W.L.Sullivan) **9** *dial* **:** THAN — used in comparisons (he better not be later ~ midnight —T.B.Costain) **10 a :** that the result is **:** THAT — used with preceding *so* or *such* (so clearly guilty ~ to leave no doubt of his conviction) (and such a son ~ all men hailed me happy —John Milton) **b :** THAT — used to introduce a noun clause and now dial. except in certain negative expressions with *know, say,* or *see* that have wide usage in informal speech (he said ~ he would come) (I don't know ~ it makes any difference) **c** *dial* **:** in so far as **:** THAT — used to introduce an adverbial clause (he hasn't come out again ~ I've seen) — **as is** \(')a'ziz, ˌə'z-\ **:** in its present condition **:** without any repairs, improvements, or alterations being made (the car was priced at $1000 *as is*) — **as it were :** as if it were so **:** in a manner of speaking (her triumph, *as it were,* did not last long) — **as new :** practically new **:** in the best secondhand condition (the clothes offered for sale were all prewar and all *as new*) — **as you were** — a military command used (1) to cancel another command that has not yet been executed or (2) to direct troops to return to the position

hutted among the heights —Washington Irving)

¹hutch \'həch\ *n* -ES [ME *huche, huche,* fr. OF *huche*] **1 a :** a chest or compartment for storage **:** BIN, LOCKER **b :** a low cupboard with doors usu. surmounted by two open shelves **2 a :** a pen or coop for an animal **:** CAGE (provided a ~ for them in the garden —T.E.Donne) **b :** a cageful of animals (kept . . . a ~ or two of hare —Joyce Warren) **3 :** a cramped or flimsy shelter for a man **:** SHACK, SHANTY **4 a :** a car on low wheels in which coal is drawn and hoisted out of a mine pit **b** (1) **:** the bottom compartment of an ore-dressing jig (2) **:** the mineral product that collects there

hutch 1b

²hutch \"\ *vt* -ED/-ING/-ES **1** *archaic* **:** to put away or store in a hutch **:** HOARD **2 :** to wash (ore) in a box or jig

³hutch *adj* [perh. alter. of *hulch*] *obs* **:** HUMPED

hutch burn *n* **:** an inflammation of the skin of rabbits esp. on the hind feet and adjacent parts associated with unclean urine-soiled cages

hutch·e·so·ni·an \ˌhəchəˈsōnēən\ *adj, usu cap* [Francis *Hutcheson* †1746 Scot. philosopher + E *-ian*] **:** of or relating to the theories of the Scottish philosopher Francis Hutcheson

hutch·et \'həchət\ *n* -S [MF *huchet,* fr. *hucher* to cry out, fr. OF *huchier,* prob. of imit. origin] **:** a hunter's horn **:** BUGLE 2

hutch·in·so·ni·an teeth *or* **hutchinsonian incisors** \ˌhəchən-ˈsōnēən-\ *n pl, often cap H* [*hutchinsonian* fr. Sir Jonathan *Hutchinson* + E *-ian*] **:** HUTCHINSON'S TEETH

hutch·in·son·ite \'həchənsəˌnīt\ *n* -S [Arthur *Hutchinson* †1937 Eng. mineralogist + E *-ite*] **:** a mineral $(Pb,Tl)_2(Cu,Ag)As_5S_{10}$ consisting of sulfide of lead, copper, and arsenic, with thallium and silver replacing variable amounts of lead and copper, and occurring in small red orthorhombic crystals

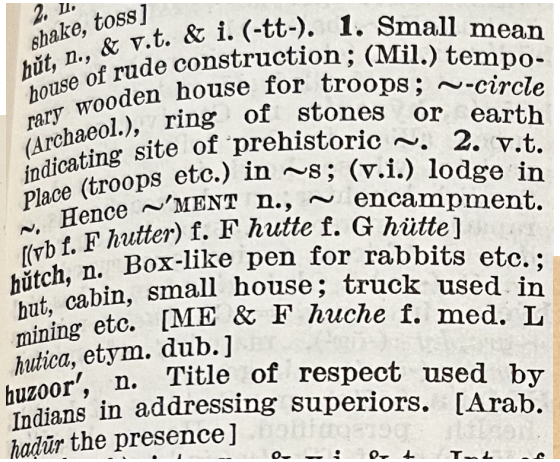
hutch·in·son's teeth \'həchənsənz-\ *also* **hutchinson teeth** *n pl but sing or pl in constr, usu cap H* [after Sir Jonathan *Hutchinson* †1913 Eng. surgeon] **:** peg-shaped teeth having a crescentic notch in the cutting edge and occurring esp. in children with congenital syphilis

hutchinson's triad *n, usu cap H* **:** a triad of symptoms comprising Hutchinson's teeth, interstitial keratitis, and deafness and occurring in children with congenital syphilis

hutch·ins's goose \'həchənz(ˌəz)-\ *n, usu cap H* [after Thomas *Hutchins* †1790 Eng. attaché of the Hudson's Bay Company] **:** a variety (*Branta canadensis hutchinsii*) of the Canada goose closely resembling but smaller than the typical form, breeding in arctic America and migrating south through the U.S., but being rare east of the Mississippi

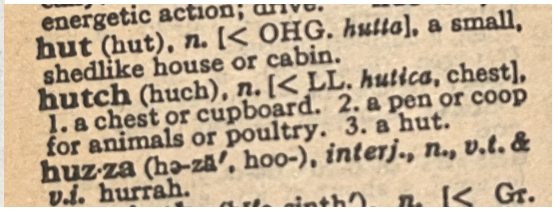
hutch table *n* **:** a combination table and chest whose top can be tilted back to convert the unit into a chair or settee

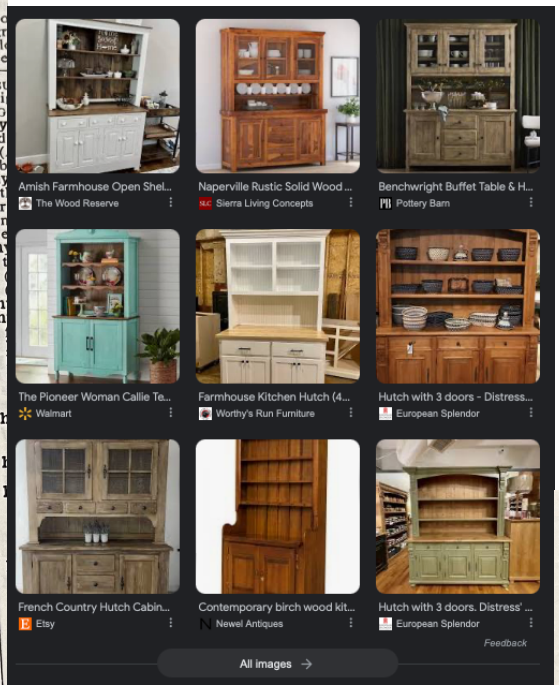
hut circle *n* **:** a ring of stones or earth marking the site of a prehistoric dwelling


2. n.
shake, toss]
hŭt, n., & v.t. & i. (-tt-). **1.** Small mean house of rude construction; (Mil.) temporary wooden house for troops; ~-*circle* (Archaeol.), ring of stones or earth indicating site of prehistoric ~. **2.** v.t. Place (troops etc.) in ~s; (v.i.) lodge in ~. Hence ~'MENT n., ~ encampment. [(vb f. F *hutter*) f. F *hutte* f. G *hütte*]

hŭtch, n. Box-like pen for rabbits etc.; hut, cabin, small house; truck used in mining etc. [ME & F *huche* f. med. L *hutica,* etym. dub.]

huzoor', n. Title of respect used by Indians in addressing superiors. [Arab. *hadūr* the presence]


energetic action; drive.
hut (hut), *n.* [< OHG. *hutta*], a small, shedlike house or cabin.
hutch (huch), *n.* [< LL. *hutica*, chest], 1. a chest or cupboard. 2. a pen or coop for animals or poultry. 3. a hut.
huz·za (hə-zä', hoo-), *interj., n., v.t. & v.i.* hurrah.



PROSTITUTE
hut /hʌt/ *n* a small building, often made of wood, esp. one used for living in or for shelter —compare SHED²
hutch /hʌtʃ/ *n* a small box or cage with one side made of wire netting, esp. one for keeping rabbits in
hut·ment /'hʌtmənt/ *n* a group of huts, esp. army huts for soldiers to camp in

General principles

- Universality – system should work the same for all languages
- Reductivity – can't define the simple by the more (or just equally) complex *speltz* 'any of several varieties of emmer'

  Suppose I make you a gift of a large sum of money saying you can collect it from Titius; Titius sends you to Caius; and Caius, to Maevius; if you continue to be sent like this from one person to another you will never receive anything (Leibniz, quoted in Wierzbicka (1985))

- No encyclopedic knowledge
- OK, but where to draw the line? We keep only *essential* properties

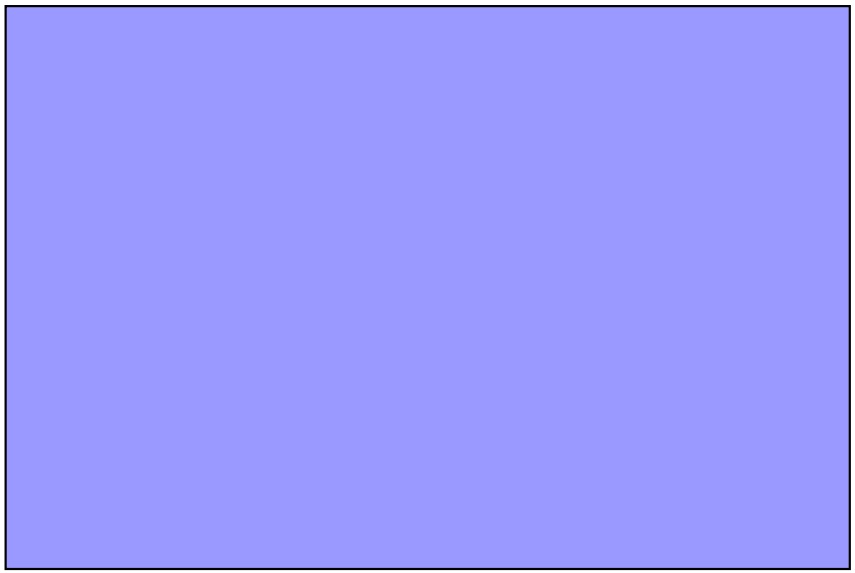
Lexical entries

- There are disjoint lexical entries (for words and morphemes) called *lexemes*
- These overwhelmingly correspond to traditional dictionary entries
- In dictionary databases, these used to be the *records*
- But these are not subdivided into *fields* as in typeset dictionaries or dictionary databases
- Rather, they are associative networks with *spreading activation* (Quillian, 1967; Collins and Loftus, 1975; Carroll, 1983)
- Phonology done by autosegmental representations (Goldsmith, 1976; McCarthy, 1988)
- Can be viewed as automata (Eilenberg machines)
- Can also be viewed as vectors

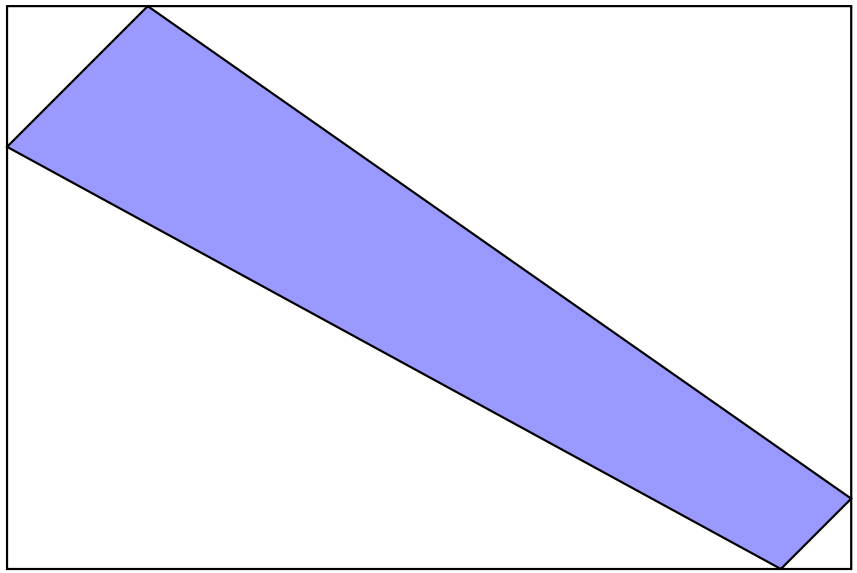
Lexical entries cont'd

- Stylistic and other labels by ultradense subspaces (Rothe, Ebert, and Schütze, 2016; Dufter and Schütze, 2019)
- We have the technology for etymology (diachronic phonological rules are just as easy by automata as synchronic rules) but kids don't have the data
- In addition to traditional lexemes (words, stems) we also have lexical entries for bound morphemes (roots, affixes)
- Morphology often has non-compositial semantics
- Lexicon also contains *conceptual schemas* (Schank and Abelson, 1977)
- OK, but what about syntax? We use *constructions* (Fillmore and Kay, 1997)
- Traditional concerns of syntacticians are addressed via a sparse system of linkers (thematic roles/deep cases/kārakas) (Kiparsky, 1987; Butt, 2006) → Wed

Semicompositionality as subdirect decomposition



Direct product



Subdirect product

(Figure from Kornai, 2023 Ch. 2.2, but the idea goes back at least to Kiparsky, 1982 on noun-noun compounding)

The "commercial exchange" schema

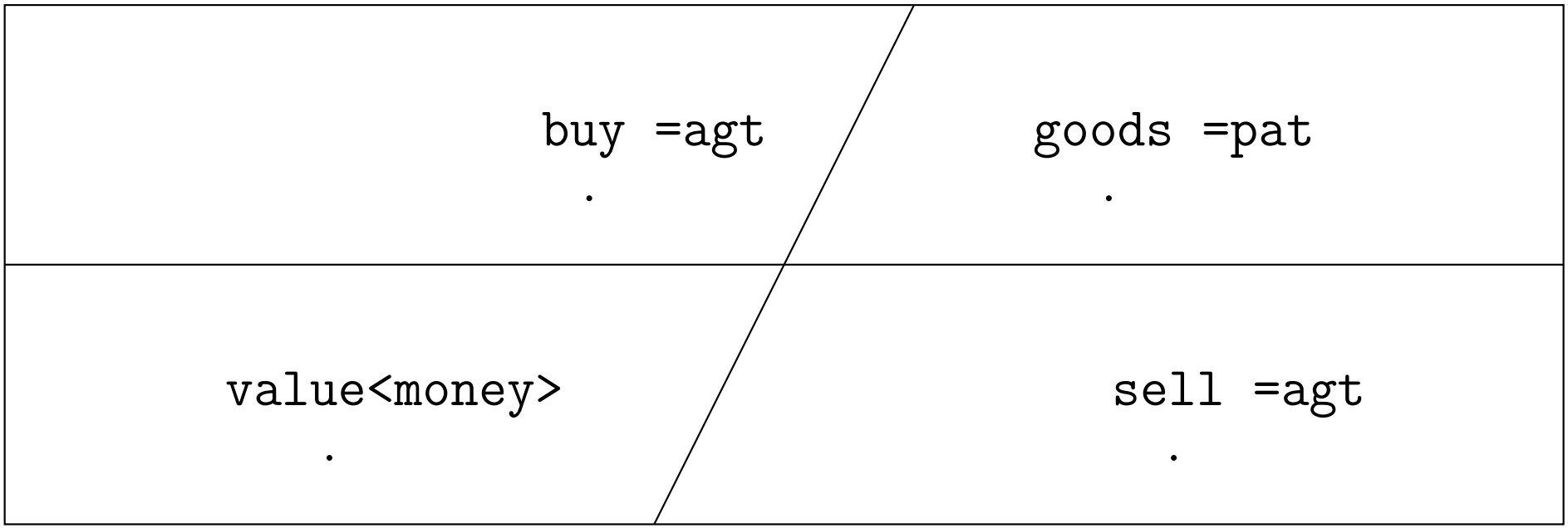


Figure: exchange_

Obl 4lang plug

- There are limitations in what we can do
- 4lang is not good for technical vocabulary
- Numbers are already a problem (this is a feature, not a bug)
- But we can do ordinary dictionary words
- First, we reduce large dictionaries to smaller ones (good computational project)
- Next we reduce these to a small defining vocabulary (we used LDV, 2,200 words)
- Next we looked for uroboros set in LDV (currently 770 entries)

Have we missed something?

- Input: any word in any language. First find English definition. Start with German *schlagfertig* and find translation *quick-witted* (as opposed to literal translation *ready to hit*)
- Reduce this definition to core vocabulary by repeated substitution *quick-witted* is clearly *quick.wit.ed* (note lack of **witted*) and the morphology will supply 'has quick(-)wit', cf. *triangle-shaped, bite-sized, able-bodied, baby-faced, big-hearted, well-intentioned, ...* (total of 168 candidates in LDOCE).
- In this case we are lucky: LDOCE already has *quick-witted* 'able to think and understand things quickly', but what if we are not so lucky?
- In that case, we have to work on *has*, *quick*, and *wit* separately. Of these, `has` and `quick` are already in 4lang, with definitions '=agt control =pat, =agt has =pat' and 'act in short(time)' respectively
- But *wit* is missing!

Substitution *salva veritate*

- By definition of *has*, we obtain '=agt control {quick wit}, agent has {quick wit}. Substituting the definition of quick, we obtain =agt control {wit, wit act in short(time)}, agent has {wit, wit act in short(time)}
- Unification is automatic (unless blocked by other). But we (a) haven't quite gotten rid of has (and we won't, it's a primitive!) and (b) still need to get away from *wit*.
- less surprisingly than for *quick-witted*, LDOCE also has *wit* (*quickwitted* is #2716299 on the Google frequency list, *wit* is #14661) 'the ability to say things that are clever and amusing'
- So now we substitute this to obtain =agt has {ability to say thing, thing is_a clever, thing is_a amusing, say in short(time)},...
- thing, short, say and time are in 4lang

Substitution con't

- We still need *clever* and *amusing*, but LDOCE has these, and uses only LDV in their definition

  `clever able to learn and understand things quickly`
  `clever able to use your intelligence to get what you want, especially in a slightly dishonest way`
  `clever skilful at doing a particular thing`
  `clever done or made in an unusual or interesting way that is very effective`
  `amusing funny and entertaining`
- So we can go on, getting things defined one by one until everything is in the uroboros core
- People can learn how to produce 4lang definitions surprisingly fast
- Machines have a guarantee of reductivity to the core

Is this hard?

- Everybody tries to build a basic list:
  `https://concepticon.clld.org` has 450+ sources
- I don't know of any other one that is actually reductive
- The best of breed is NSM (Natural Semantic Metalanguage)
  only 60+ primitives
- But the syntax is not fully defined, and no reductivity guarantee
- One would need to define all 4lang primitives by NSM primitives and they'd be done
- This is not a computational project: the person doing it has to be anointed in NSM

Elements

- Static elements (stored in the lexicon) are *roots* and *features*
- In Minimalism, they seem to differ only in multiplicity, but in traditional grammar we distinguish between *content* and *function* morphemes
- Some static elements are clearly contentful, others clearly functional, but they get fused early on e.g. noun stem + case marking
- Instead of (form, meaning) pairs, we may want to work on (form, category, meaning) triples extended signs as in Kracht, 2003

Data structures used by linguists

- Generally *trees*, but what kind? Rooted/unrooted, labeled/unlabeled, planar/nonplanar, connected or not (forests), directed or not, binary or more branching, unary nodes permitted or not, empty nodes permited or not.
- The big dividing line: weighted or unweighted
- Usually probability weights, but can be taken from any semiring
- A very relevant semiring is the `tropical semiring` approximating log probabilities: addition is max, multiplication is plus

Constituents

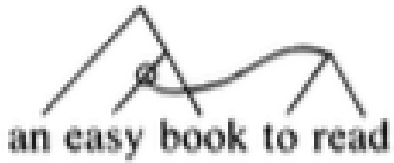
- Classic example Wells (1947): *The King of England opened Parliament*.
- We want to cut this in two parts that enjoy large combinatorial freedom: the best cutpoint is between the subject and the rest: (The K of E)(opened Parliament). (i) Both parts occur pretty freely elsewhere 'The K of E X(=did something)' and 'Y opened Parliament' (ii) both can be substituted by simpler (ideally, one word) material: (Joe)(slept).
- We do this recursively: *opened Parliament* is further analyzed as (opened)(Parliament) and *the King of England* as (the)(King of England), the latter as (king)(of England)
- This naturally gives trees
- Notice that the (important for grammar) notion *subject* is not used anywhere in the process

Now for some more complex examples

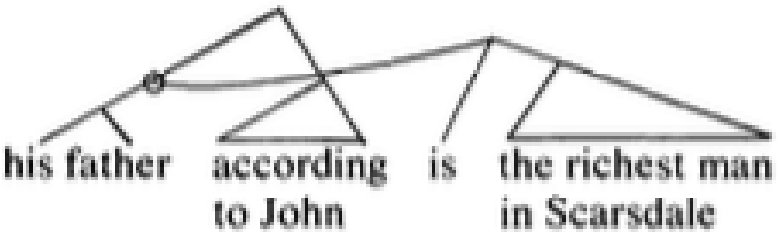
- *Úristen, mondtam, ez az én fiam, azonnal megismertem!*
  Lord god, I said, this is my son, I recognized him immediately
- The commas (which don't require a written form, they can be detected in the acoustic signal, (*comma intonation*, see Hetzron (1980) and Kornai and Kálmán (1988)) segment the material in four parts ABCD.
- There is no good cutpoint, the best parse is B+A..CD. Here B is called a *parenthetical*, and A..CD is called a *discontinuous constituent*. Examples from Wells (1947) via McCawley (1982):

(1) a.

b.



c.



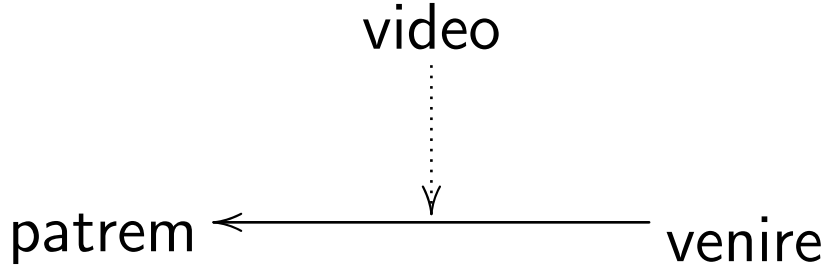
d.

Labels: positional or functional

- What is the replacement class of *lord god*?
- This is not a direct invocation of god (vocative) but an expression of the speaker's emotive state
- We can substitute *aw fuck* just as easily as *Jesus* but not *holy mackarel*
- Labels have complex internal structure (seen e.g. from agreement phenomena): *ez az én lányom, azonnal megismertem* 'this is my daughter, I recognized *him/her/*it immediately'

Hypernode graphs

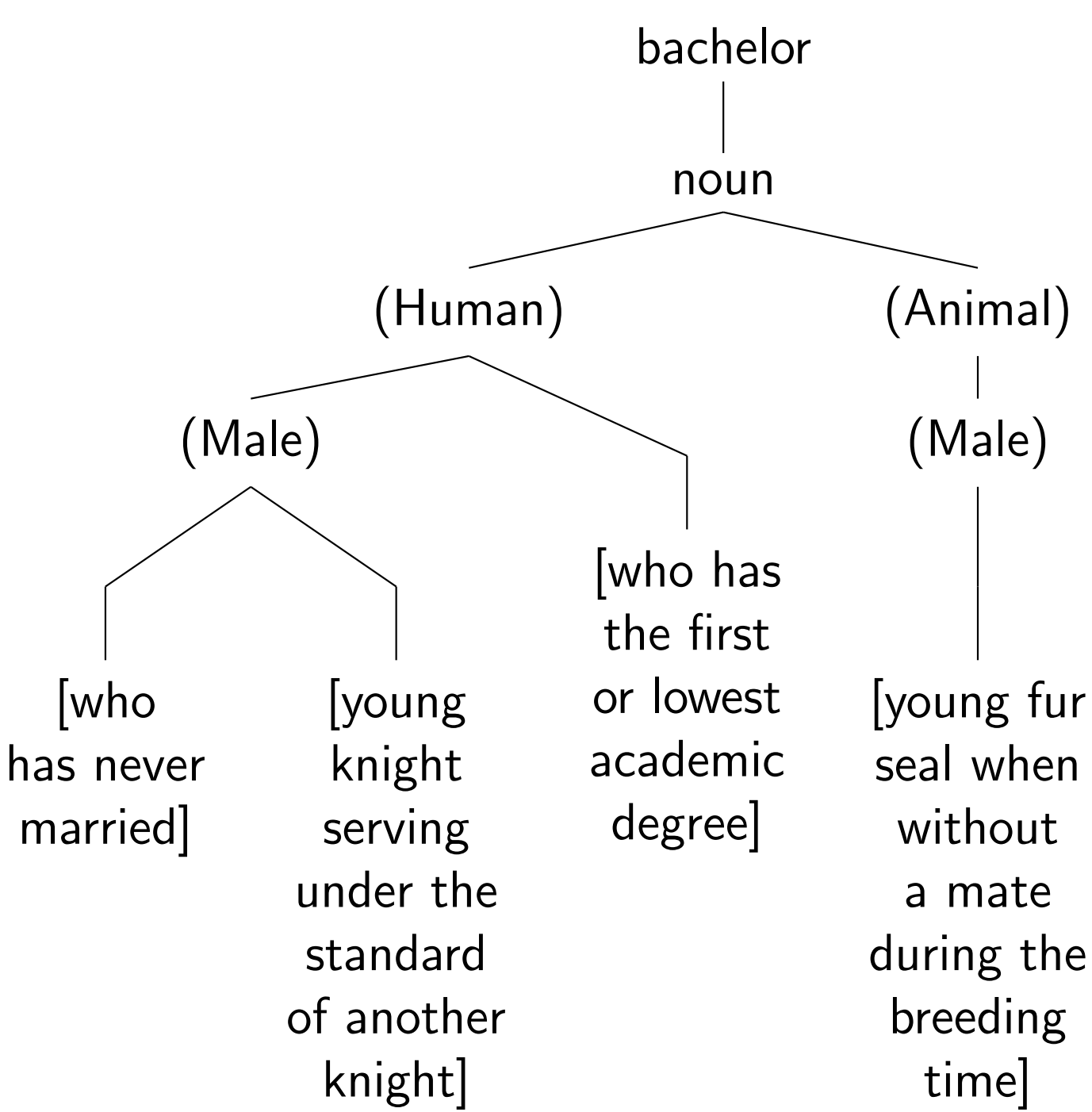
- Assume 'standard' SVO order (Subject-Verb-Object) as in *Kim chose Sandy*
- *Video patrem venire*



- ([Ego] video (patrem venire [null]))
- The first zero [ego] is substantive – we *know* from the conjugation not just that it is present tense, active, indicative but also that it is 1st person singular (cf. *tu vides, is videt, nos videmus, . . .*
- The second zero [null] is technical (signifies lack of object)

Traditional meaning decomposition

- Begins with the Tree of Porphiry ($\sim$ 300 CE by student of Plotinus)
- Standard through Middle Ages (Boethius, Albertus Magnus, . . .)
- Discrete, often binary decomposition, as in Prague School phonology
- Generative theory inherits the method from Katz and Fodor, 1963
- Relevant critique in Bolinger, 1965
- It's not the genera, it's the differentia specifica (K&F's 'distinguishers') that are problematic (very arbitrary)
- *bachelor, chrome, high*


bachelor
noun
(Human)
(Animal)
(Male)
(Male)
[who has never married]
[young knight serving under the standard of another knight]
[who has the first or lowest academic degree]
[young fur seal when without a mate during the breeding time]

Discrete (binary) feature vectors

- Main advantage of arranging these in trees: higher nodes mesh well with universal/grammatical features
- Main disadvantage: lower nodes haphazard
- Try *chrome*$_1$ 'hard and shiny metal'; *chrome*$_2$ 'eye-catching but ultimately useless ornamentation, especially for cars and software'; *chrome*$_3$ 'google browser'
- Now try *high*, as in *high price, high spirits, high opinion, high family, high note, high 'stoned', high and mighty, high mountain, high blood pressure, . . .*
- We will go in the exact opposite direction, *monosemy* (Ruhl, 1989)
- *bachelor* 'unfulfilled in a traditional male role' (Roman Jakobson)
- *high:* `top er_ gen, has top`
- *top:* `part, at position, vertical(position er_ part[other])`

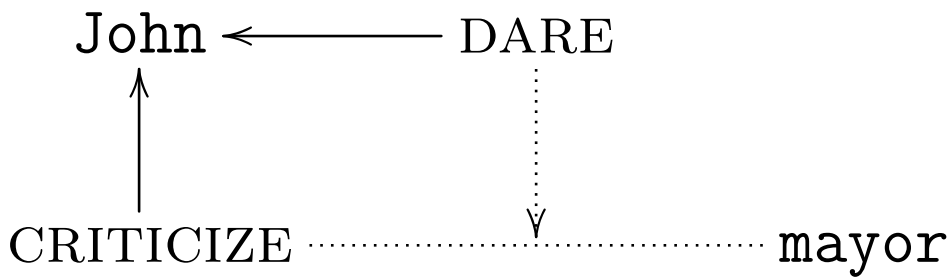
Compositionality since Frege and Montague

- $L \xrightarrow{d} D \xrightarrow{t} F \xrightarrow{I} \mathcal{M} \xrightarrow{g} W$ (mnemonics: *L*anguage; *D*isambiguated language; *F*ormula; $\mathcal{M}$odel; real *W*orld)
- Assume you have some mapping $t : A \to M$ from atomic units to meanings, are we done?
- No, we want to assign meanings to some (maybe not all) sequences in A^* in some representation space $M' \supset M$.
- Compositionality means one well-known thing (explicit since Frege) that we want meaning assignment to respect concatenation $t(uv) = f(t(u), t(v))$ where f is a fixed operation (function application, vector addition, 'merge', hypergraph substitution . . .)
- It also means one implicit thing: in an undifferentiated stream $x_1x_2x_3x_4 \ldots x_n$ finding the cutpoint so that $x_1 \ldots x_i = u$ and $x_{i+1} \ldots x_n = v$ *segmentation task*, see Kornai, 2024

Atomic units: hypernode graphs

- Graphs are defined by sets of nodes N and edges $E \subset N \times N$
- An ordinary graph G is a hypernode graph H
- We can think of these as being built from atomic nodes by adding edges labeled '1' and '2' among them
- Members of H can appear as nodes of hypergraphs
- H is the smallest set closed under conjunction and node substitution
- Triple notation: (A B C) means "an edge B runs from A to C". Hypernode graphs are finite lists of such triples built recursively at the two sides (the first and third elements can be triples, the center element is always atomic)
- Always read triples in SVO order (`Brutus kill Caesar`)
- Silent unification of atoms (unless blocked by `other`)

Hypernode graphs at work

- There are only two linkers: =agt (subject, nominative case) and =pat (object, accusative case) [Doing this right in ergative languages is not hard, but will not be discussed here]
- 

John dared to criticize the mayor
- Hypernodes are S-V-O triples as in RDF – becaue you don't need indirect objects, themes, goals, etc. (Kornai, 2012) you don't need hyperedges
- `John DARE {John CRITICIZE mayor}`
- Unification operates silently in the background to make sure the two *John*s are the same

Smooth interpenetration of syntax, semantics, pragmatics

- *Someone disrespected his brother, so he beat him to pulp, because blood is thicker than water*
- ∃ person p: p disrespected p's brother b `p disrespect b isa brother, q has brother`
- ∃ person p, person q: p disrespected q's brother b
- person r beat person s to pulp. Is r=p,q,b? Is s=p,q,b?
- Well, yes, the viscosity of blood is about 4 times higher than that of water, but what does this have to do with it? [Anti-Kripke rant omitted]
- Yet after this clause, it is evident to all speakers of English that p disrespected q's brother b, and it is q, rather than brother b, who beat p up (Kornai, 2012)

Syntax, semantics, pragmatics, who's responsible?

- *blood is thicker than water* 'said to emphasize that you believe that family connections are always more important than other types of relationship' https://dictionary.cambridge.org/dictionary/english/blood-is-thicker-than-water
- 'said to emphasize that you belive' – no. By this token, *the fries are great at McDonald's* is said to emphasize that you believe the fries to be great at McD.
- 'family connections are more important than other types of relationships' still too fluffy: `family er_ gen` will suffice
- We also need a bunch of commonsense implications, e.g. `disrespect isa attack, brother isa family` $\Rightarrow$ disrespecting brother is attack on family
- 'Attack provokes counterattack'

Too much real world knowledge

Principle of responsibility

The pragmatic wastebasket must remain empty at all times

- You could put in the encyclopedia 'if someone attacks your brother you should counterattack', but that would bring back Partee's Problem
- Having this as a derived rule from 'if someone attacks your family you should counterattack' and 'brother isa family' is already better
- For this we will need some very very general rules governing the interaction of is_a with all forms of predication, such as *transitivity:* if A isa B and B isa C $\Rightarrow$ A isa C; more generally *downward subject entailment:* if A isa B and B pred C $\Rightarrow$ A pred C
- In `4lang` these come for free because both `isa` and subjecthood are set-theoretical containment $\subset$

Laws of nature

- There are absolute, no-exceptions regularities, like the Law of Gravity 'unsupported things will fall'. Kornai, 2019 Ch. 3.5
- These are modeled by two-state finite automata, the `before` state containing the preconditions, and the `after` state holding the outcome
- *Mechanical causation:* result comes by inevitably, just by the passing of time
- *cause:* `before(=agt), after(=pat)`
- Not at all different from lexical entries of verbs, e.g. *rest:* `quiet, calm, before(tired), after(has energy)`
- Default logic in all cases, except when laws of nature break down we search for hidden factors *how come the balloon doesn't fall*

Mechanism, not policy

- We need mechanism for derivation chains (disrespect isa attack, beating to pulp isa counterattack)
- Key pieces are *pattern matching:* recognizing that a graph is a subgraph of some other graph
- *substitution:* via `=agt`, `=pat`
- *spreading activation:* over hypernode graphs
- The passing of time does everything
- Moral precepts/sociobiological observations are treated as Laws of Nature with modal force: attack *should be* followed by counterattack
- *Ceterus paribus* and default logic always lurking in the back

Further progress on Partee's Problem

- We do not want to posit 'if someone attacks your family you should counterattack' as readily available in memory (though it may be stored as such in the LTM of some people)
- What we want is an explanation that IF `family er gen` is made part of the common ground between the speaker and the hearer, how this helps resolve the anaphoric ambiguity
- Implication chain: `family er gen` $\Rightarrow$ `family er self` $\Rightarrow$ {`attack on family`} `er` {`attack on self`}
- This, combined with 'attack *should be* followed by counterattack' implies 'attack on family should be followed by counterattack by self' which is precisely what was needed
- Takeaway lesson: the world knowledge required for language understanding is the *deductive closure* of a much smaller set (lexical entries and laws of nature). The generating set must still be memorized, but this is less than 1MB. Small is beautiful

Mapping words or larger text to vectors

- CVS: continuous vector space (really $\mathbb{R}^n$, complex is rarely used)
- Continuity was emphasized because the preceeding standard was (partial) decomposition of meanings into finite bit vectors for example *brother* = '+sibling +male'
  sister = '+sibling –male'
- Continuous begins with Osgood, May, and Miron, 1975 who asked for judgements on a scale of -3 to +3 and performed PCA on the results
- Next big thing was Landauer and Dumais, 1997 who took term-document cooccurrence data and performed SVD "Latent semantic indexing" (see Kornai, 2019 Ch. 2.7)
- Today: term-term cooccurrence plus dimension reduction to assign some vector $\vec{word} \in \mathbb{R}^d$ to each word. This assignment is the *conceptual dictionary*.

The general setup

- You have some concepts to learn e.g. natural kinds like *duck*
- Humans are incredibly good at this. Take a guided tour in a forest and you can learn, based on very few examples, the affordances of flora and fauna
- What you have are observable features (color, shape) which are not like the biologists' features (webbed feet)
- Each data point is a vector in feature space (can be squished to the unit cube)
- What you want is to characterize a set (probability distribution) in *sample space* by means of a *model* selected from the *hypothesis space*
- Standard choice: Gaussian Mixture Models
- Can be a single vector if the variance is negligible

Machine learning on one slide

- Strict separation (typically 80-10-10) of *train*, *dev* and *test* data
- Train is used for building the model, dev for finetuning, test typically hidden from the model builder
- A model optimizes some figure of merit (e.g. word error rate in speech recognition)
- Strong culture of shared tasks (each team working on the same data)
- Generally requires large datasets (gigaword is now typical)
- Supervised methods rule – unsupervised learning still in its infancy
- `aclwiki/POS_Tagging_(StateOfTheArt)`
- `HuggingFace LM leaderboard`

Osgood-May-Miron

- Method: just ask people to rank conceps on scales
- Collect results in array with subject i's response to question j about object k. Collapse j.k structure in a single vector, perform *means centering*, compute covariance matrix C. The *principal component* of the data is defined as the direction that maximizes the variance
- To find it, we need to solve

$$\frac{d}{d\vec{x}}\vec{x}^T C\vec{x} - \lambda\vec{x}^T\vec{x}, \tag{1}$$

  second term is a Lagrange multiplier that comes from the constraint of keeping the length of $\vec{x}$ fixed
- The critical points are obtained from solving $C\vec{x} = \lambda\vec{x}$, so the solutions λ_i are, by definition, the eigenvalues and the x_i are the corresponding eigenvectors

Latent Semantic Indexing

- Instead of asking people, look at cooccurrences
- For Osgood et al, inverting a 100x100 matrix was a big deal, Dumais et al., 1988; Deerwester, Dumais, and Harshman, 1990 could do 1000x1000 (not yet $10^4 \times 10^4$)
- Term-Document matrix tells you how often word i occurred in document j
- By PCA the the documents can be clustered
- Eventually we could move to Term-Term cooccurrence ($10^5 - 10^6$)

Success has many fathers

- Idea first suggested by Schütze, 1993
- First implementation that really worked Bengio et al., 2003
- NLP "almost from scratch" POS, CHUNK, NER, role labeling Collobert et al., 2011
- Has linear structure (king–queen=man–woman) Mikolov, Le, and Sutskever, 2013
- Why? Pennington, Socher, and Manning, 2014; Arora et al., 2015

How do we build the vectors?

- Originally: by gradient descent: we want to minimize the distance of vectors that appear in similar contexts, and maximize the distance of those appearing in dissimilar contexts. This is the `word2vec` algorithm
- How do we know two contexts are similar? They are made up of similar words!
- Wait, isn't this circular?
- No. What we need is the assumption that the semantics of a context is simply the sum of the vectors in it
- We will look at Mikolov et al., 2013; Pennington, Socher, and Manning, 2014; Levy and Goldberg, 2014
- Note some salient properties (linear structure, log frequency length)
- Modern tricks: word pieces and dynamic embeddings

Word2vec

- Feed-forward and RNN language models like (Bengio et al., 2003) learn rich vectors but incur heavy matrix multiplies. Mikolov et al., 2013 wanted to scale to 10^{11} tokens on one box. They abandoned hidden layers and predicted context words directly. Ideas that led to significant speedup included
- (1) Hierarchical Softmax $\rightarrow O(\log |V|)$
- (2) Negative Sampling $\rightarrow$ constant-time updates
- (3*) Subsampling frequent words $\rightarrow$ 2-10$\times$ speed-up and higher quality
- (4*) Overt detection of phrases
- Development doesn't quite fit the "Make it work, make it right, make it fast" model

Skip-gram objective

- Given center word w_t, maximize

$$\sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} \mid w_t)$$

  with context window c (Mikolov et al., 2013)
- Full soft-max:

$$p(w_O \mid w_I) = \frac{\exp\left(\mathbf{v}'^{\top}_{w_O} \mathbf{v}_{w_I}\right)}{\sum_{w \in V} \exp\left(\mathbf{v}'^{\top}_{w} \mathbf{v}_{w_I}\right)}$$

  costs $O(|V|)$ per update – impractical for 10^6+ vocab.
- Solution: approximate soft-max efficiently (next slide)

Cheap soft-max variants

- **Hierarchical Softmax** • Huffman binary tree over vocabulary • Path length $\approx \log_2 |V|$ Morin and Bengio, 2005 • Works well for rare words
- **Negative Sampling** Replace soft-max with logistic regression

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

  • Few ($k = 5 - 15$) negative samples; $P_n(w) \propto U(w)^{3/4}$.
  • Yields higher accuracy for frequent words and trains $> 2\times$ faster than HS
- Choose HS or NEG per task; Google News model used NEG-15.

Noise-Contrastive Estimation (NCE) & negative sampling

- NCE turns language modelling into binary **classification**: distinguish one true pair (w_I, w_O) from k noise words $w^{(n)} \sim P_n$ (Gutmann and Hyvärinen, 2010; Mnih and Kavukcuoglu, 2013)
- Objective for center word w_I:

$$\log \sigma(s_{w_I,w_O}) + \sum_{i=1}^{k} \left[\log \sigma\left(-s_{w_I,w_i^{(n)}}\right) - \log\left(kP_n(w_i^{(n)})\right) \right]$$

  where $s_{w_I,w} = \mathbf{v}'^{\top}_{w} \mathbf{v}_{w_I}$.
- As $k \to \infty$ the gradient equals that of maximum-likelihood soft-max; for small k it is biased but much cheaper
- **Negative sampling** (word2vec) = *simplified* NCE: drop the $\log(kP_n)$ term and set $P_n(w) \propto U(w)^{3/4}$; with $k = 5 - 15$ gives better vectors *and* faster training

Subsampling frequent words

- Frequent tokens ("the", "of") dominate updates yet add little information. Discard w_i with probability

$$P_{\text{drop}}(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \quad (t \approx 10^{-5})$$

- Effects: $2-10\times$ wall clock speed-up *and* cleaner vectors for rare words (less noise from stop words).
- Window size can be *expanded* dynamically when words are dropped—keeps number of training pairs stable
- (*) Idea already already present in GOFIR (Good Old-Fashioned Info Retrieval) which used *stopwords*

From words to phrases

- Many meanings are non-compositional ("Air Canada"). Detect candidate bigrams using PMI-style score $\frac{\text{count}(w_i w_j) - \delta}{\text{count}(w_i)\,\text{count}(w_j)}$ and merge iteratively.
- Treat each phrase as a single token; same Skip-gram training learns vectors for millions of phrases.
- Phrase vectors excel on new analogy set (e.g. "Montreal : Montreal Canadiens :: Toronto : ?") reaching 72 % accuracy at 1000 d with 33 B-token corpus
- (*) Idea already already present in GOFIR/GOFNLP where it was known as 'collocation detection', see Manning and Schütze, 1999 Ch. 5

Linear regularities

- Vector arithmetic captures analogical relations:
  $\mathbf{v}_{\text{Madrid}} - \mathbf{v}_{\text{Spain}} + \mathbf{v}_{\text{France}} \approx \mathbf{v}_{\text{Paris}}$
- Additive composition works for country + capital, adjective + ly, etc.; basis for analogy benchmark standard at the time
- Google News 300-d NEG-15 model (100 B tokens) trained in one day and became de facto off-the-shelf embedding for NLP
- Today we measure on downstream tasks, analogies are not seen as a good test of embedding quality
- But cross-linguistic rotation is still a big deal

GloVe

- Build a global matrix $X \in \mathbb{R}^{|V| \times |V|}$ where X_{ij} counts the number of times context j appears within a ± 10-word window of word i Pennington, Socher, and Manning, 2014.
- Turn counts into *probabilities* $P_{ij} = X_{ij} / \sum_k X_{ik}$ – probability that j shows up near i

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

- We therefore want a vector space where differences $\mathbf{w}_i - \mathbf{w}_j$ linearly reflect $\log\left(P_{ik}/P_{jk}\right)$

Deriving the GloVe loss

- Seek word vectors $\mathbf{w}_i, \tilde{\mathbf{w}}_j$ s.t. $\mathbf{w}_i^\top \tilde{\mathbf{w}}_j + b_i + \tilde{b}_j = \log X_{ij}$.
- Fit by weighted least squares

$$J = \sum_{i,j} f(X_{ij}) \left(\mathbf{w}_i^\top \tilde{\mathbf{w}}_j + b_i + \tilde{b}_j - \log X_{ij} \right)^2$$

  – convex in either set of vectors when the other is fixed
- $\log X_{ij}$ gives each factor an *additive* role (matching the desired linear offsets) and dampens the impact of collocation pairs
- Bias terms $b_i, \tilde{b}_j$ soak up corpus-wide frequency effects, letting the dot-product focus on *relative* information
- Optimize a global log-bilinear *weighted least-squares* loss on $\log X_{ij}$ for non-zero counts only.
- Weighting function $f(x) = \min\left((x/x_{\max})^{0.75}, 1 \right)$ with $x_{\max} = 100$ balances rare vs. frequent pairs.

Weighting rare vs. frequent pairs

- Unweighted LS would let common pairs dominate; down-weight big X_{ij} but *ignore* zero counts. Proposed heuristic:

$$f(x) = \begin{cases} (x/x_{\max})^{\alpha} & x < x_{\max} \\ 1 & x \geq x_{\max} \end{cases} \quad \text{with } \alpha = 0.75, x_{\max} = 100$$

- Empirically chosen $x_{\max} = 100$ balances stability and coverage; $\alpha = 0.75$ mirrors the Zipfian log-log slope of word frequencies.
- Objective now gives mid-frequency pairs the highest weight, learns something from rare pairs, and limits the gradient explosion of stop-word pairs
- Training efficiency: Complexity grows as $\mathcal{O}(|C|^{0.8})$, far below the naive $\mathcal{O}(|V|^2)$ of full count matrices. With stochastic gradient descent (AdaGrad, nowadays Adam is more popular) 100 iterations reached convergence on 6B token corpus in a few hours (400k vocab, 300d vectors)

Intrinsic evaluation

- 300-d GloVe (42 B tokens) scores 75% overall on the Mikolov analogy set (best published at the time)
- Outperforms skip-gram / CBOW at equal training cost
- New state-of-the-art on five word-similarity benchmarks
- Downstream NLP impact: Adding 50-d GloVe features to a CRF NER system boosts CoNLL-2003 F_1 from 85 to 88.3 (Turian, Ratinov, and Bengio, 2010), similar gains observed on ACE-2003 and MUC-7, confirming broad utility
- Others remained skeptical whether gains were really that great, but availability of code + pretrained vectors from `nlp.stanford.edu/projects/glove` was great service at the time

Why revisit word embeddings?

- Skip-gram with negative sampling (SGNS) delivers strong vectors but its objective was opaque
- Levy &Goldberg (2014) prove SGNS *implicitly* factorises a word–context matrix whose cells are $\mathrm{PMI}(w, c) - \log k$
- This unifies "predict" models (word2vec) with classical "count" models and opens new, simpler routes to embeddings
- SGNS maximises, for each observed pair (w, c), $\log \sigma(\mathbf{w} \cdot \mathbf{c}) + k\, \mathbb{E}_{c_N \sim P_D}[\log \sigma(-\mathbf{w} \cdot \mathbf{c}_N)]$
- With unlimited dimension the optimum for every pair is $\mathbf{w} \cdot \mathbf{c} = \log \frac{\#(w,c)\,|D|}{\#(w)\,\#(c)} - \log k = \mathrm{PMI}(w, c) - \log k$
- Hence SGNS performs a **weighted** low-rank factorisation of the *shifted PMI matrix* $M^{\mathrm{PMI}} - \log k$

Shifted Positive PMI (SPPMI)

- Raw PMI matrix is dense and has $-\infty$ for unseen pairs; apply **positive cut-off**:

$$\mathrm{SPPMI}_k(w, c) = \max\big(\mathrm{PMI}(w, c) - \log k, 0\big).$$

- SPPMI is sparse, consistent and — surprisingly — almost *optimises* the SGNS objective by itself
- Using SPPMI rows directly gives solid word similarity scores and removes stochastic training entirely

Spectral alternative: SVD over SPPMI

- Truncated SVD on the sparse matrix yields $M_d = U_d \Sigma_d V_d^\top$; use $W = U_d \Sigma_d^{1/2}$ for d-dimensional vectors
- Advantages: exact algebra, no learning rates, trains on aggregated counts, scales to very large corpora
- Empirically SVD matches or or beats SGNS on word *similarity* tasks but trails it on *analogies*, likely because SGNS's weighted loss favours frequent pairs
- SGNS weighted factorisation of PMI $-\log k$; weight $\propto$ pair frequency
- Sparse SPPMI gives a "cheap" high quality embedding; choosing k tunes performance
- Viewing prediction models as matrix factorisers bridges two lines of research

Segmentation

- Ideally you'd want morpheme vectors not word vectors
- Segmentation into morphemes is hard (and not just because of non-concatenative effects)
- It is fundamentally a global task, everything we (successfully) do is local
- The good algorithms like Morfessor (Virpioja et al., 2013) operate on an MDL basis
- Since LLMs are good for morphology (Ács et al., 2023) perhaps we could ask LLMs to do the segmentation for us, and use that as a basis
- But the mainstream approach is based on a workaround

Subword tokenisation: WordPiece & SentencePiece

- **WordPiece** Schuster and Nakajima, 2012; Wu et al., 2016: greedy byte-pair-like merges maximise likelihood of training text under a uni-gram LM; adds "##" continuation marker, enabling open-vocabulary while keeping $\leq$ 32k tokens
- **SentencePiece** Kudo, 2018: treats input as raw UTF-8 bytes, learns either BPE or uni-gram subword model; no tokenizer needed, portable across languages and whitespace conventions
- Advantages vs. word-level: •handles rare/novel words •reduces out-of-vocab to $< 0.01\%$ •balanced granularity improves translation and LM perplexity.
- Typical pipeline: train on $10^8 - 10^9$ chars, export vocab + deterministic encoder–decoder; same model serves both training and inference for reproducibility

From static to contextual word vectors

- **Static** (word2vec, GloVe): one fixed vector per type; ignores polysemy and syntax
- **Contextual** = vector is a *function* of the entire sentence: $\mathbf{e}(w_t|w_{1:n})$
- Technical leap (ELMo Peters et al., 2018): layered *bidirectional* LSTM language model; concatenate forward and backward hidden states at each position
- Task-specific linear combination $\mathbf{e}^{\text{task}} = \sum_\ell \alpha_\ell \mathbf{h}^{(\ell)}$ learned during fine-tuning; captures syntax in lower, semantics in higher layers

Transformers and masked-LM objectives

- **Transformer encoder** with self-attention Vaswani et al., 2017 replaces recurrence: direct access to any token at depth-one cost
- **WordPiece/SentencePiece** tokenisation yields open-vocab subword units; embeddings are summed with positional vectors
- **Masked language modelling** (BERT Devlin et al., 2019): randomly mask 15% of subwords, predict originals; forces bidirectional context use while keeping training symmetric
- **Autoregressive left-to-right** (GPT Radford et al., 2018): predict next token only; generates coherent text and yields contextual vectors after each self-attention block
- Fine-tune entire network or add lightweight adapters; same pre-trained parameters produce dynamic embeddings for classification, QA, generation, ...

Takeawys

- Apparently, attention is all you need
- We are still trying to understand the attention mechanism (query, key, value)
- Transformers acquired the syntax of human languages without any biologically pre-determined "Language Acquisition Device" (Piantadosi, 2024)
- They can do symbolic calculation (emulate production systems (Smolensky et al., 2024))
- They excel on System I tasks (Kahneman, 2011)

- Ács, Judit et al. (2023). "Morphosyntactic probing of multilingual BERT models". In: *Natural Language Engineering*, pp. 1–40. DOI: 10.1017/S1351324923000190.
- Arora, Sanjeev et al. (2015). "Random Walks on Context Spaces: Towards an Explanation of the Mysteries of Semantic Word Embeddings". In: *arXiv:1502.03520v1* 4, pp. 385–399. DOI: 10.1162/tacl_a_00106.
- Bengio, Yoshua et al. (2003). "A Neural Probabilistic Language Model". In: *Journal of Machine Learning Research* 3, pp. 1137–1155. DOI: 10.1162/tacl_a_00059. URL: http://www.jmlr.org/papers/v3/bengio03a.html.
- Bolinger, Dwight (1965). "The atomization of meaning". In: *Language* 41.4, pp. 555–573.
- Butt, Miriam (2006). *Theories of Case*. Cambridge University Press. DOI: 10.1017/CBO9781139164696.
- Carroll, John A. (1983). *An island parsing interpreter for the full augmented transition network formalism*. ACL Proceedings, First European Conference, pp. 101–105.

- Collins, A.M. and E.F. Loftus (1975). "A spreading-activation theory of semantic processing". In: *Psychological Review* 82, pp. 407–428. DOI: 10.1037/0033-295X.82.6.407.
- Collobert, Ronan et al. (2011). "Natural Language Processing (Almost) from Scratch". In: *Journal of Machine Learning Research (JMLR)*.
- Deerwester, Scott C., Susan T Dumais, and Richard A. Harshman (1990). "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6, pp. 391–407. DOI: 10.1002/(SICI)1097-4571(199009)41:6<3C391::AID-ASI1>3E3.0.CO;2-9.
- Devlin, Jacob et al. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proc. of NAACL*.
- Dufter, Philipp and Hinrich Schütze (2019). *Analytical Methods for Interpretable Ultradense Word Embeddings*. arXiv: 1904.08654. URL: https://arxiv.org/pdf/1904.08654.pdf.

- Dumais, Susan T et al. (1988). "Using latent semantic analysis to improve access to textual information". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 281–285. DOI: 10.1145/57167.57214.
- Fillmore, Charles and Paul Kay (1997). *Berkeley Construction Grammar*. URL: http://www.icsi.berkeley.edu/%5C~%7B%7Dkay/bcg/ConGram.html.
- Goldsmith, John A. (1976). *Autosegmental Phonology*. PhD thesis MIT.
- Gutmann, Michael and Aapo Hyvärinen (2010). In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*.
- Hetzron, Robert (1980). "Hungarian Tonosyntax". In: *Nyelvtudományi Értekezések* 104.
- Kahneman, Daniel (2011). *Thinking, fast and slow*. Farrar, Straus, and Giroux.

- Katz, Jerrold J. and Jerry A. Fodor (1963). "The structure of a semantic theory". In: *Language* 39, pp. 170–210. DOI: 10.2307/411200.
- Kiparsky, Paul (1982). "From cyclic phonology to lexical phonology". In: *The structure of phonological representations, I*. Ed. by H. van der Hulst and N. Smith. Dordrecht: Foris, pp. 131–175.
- — (1987). *Morphosyntax*. Stanford University: ms.
- Kornai, András (2012). "Eliminating ditransitives". In: *Revised and Selected Papers from the 15th and 16th Formal Grammar Conferences*. Ed. by Ph. de Groote and M-J Nederhof. LNCS 7395. Springer, pp. 243–261. DOI: 10.1007/978-3-642-32024-8_16.
- — (2019). *Semantics*. Springer Verlag. ISBN: 978-3-319-65644-1. DOI: 10.1007/978-3-319-65645-8. URL: http://kornai.com/Drafts/sem.pdf.

Kornai, András (2023). *Vector semantics*. Springer Verlag. DOI: 10.1007/978-981-19-5607-2. URL: http://kornai.com/Drafts/advsem.pdf.

— (2024). "What is the simplest semantics imaginable?" In: *From fieldwork to linguistic theory: A tribute to Dan Everett*. Ed. by Edward Gibson and Moshe Poliak. Language Science Press, pp. 247–259. URL: https://langsci-press.org/catalog/book/434.

Kornai, András and László Kálmán (1988). "Hungarian sentence intonation". In: *Autosegmental studies on pitch accent*. Ed. by Harry van der Hulst and Norval Smith. Dordrecht: Foris, pp. 183–195.

Kracht, Marcus (2003). *The Mathematics of Language*. Berlin: Mouton de Gruyter.

Kudo, Taku (July 2018). "Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers). Melbourne, Australia: Association for Computational Linguistics, pp. 66–75. DOI: 10.18653/v1/P18-1007. URL: https://www.aclweb.org/anthology/P18-1007.


Landauer, Thomas K and Susan T Dumais (1997). "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.". In: *Psychological review* 104.2, p. 211. DOI: 10.1037/0033-295X.104.2.211.


Levy, Omer and Yoav Goldberg (2014). "Neural Word Embedding as Implicit Matrix Factorization". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al., pp. 2177–2185.


Manning, Christopher and Hinrich Schütze (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.


McCarthy, John J. (1988). "Feature geometry and dependency: A review". In: *Phonetica* 45.2–4, pp. 84–108.


McCawley, James D. (1982). "Parantheticals and discontinuous constituents". In: *Linqistic Inqiury* 13.1, pp. 91–106.

- Mikolov, Tomas, Quoc V Le, and Ilya Sutskever (2013). "Exploiting similarities among languages for machine translation". arXiv preprint arXiv:1309.4168.
- Mikolov, Tomas et al. (May 2013). "Efficient Estimation of Word Representations in Vector Space". In: *1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. arXiv: 1301.3781 [cs.CL]. URL: http://arxiv.org/abs/1301.3781.
- Mnih, Andriy and Koray Kavukcuoglu (2013). "Learning word embeddings efficiently with noise-contrastive estimation". In: *Advances in Neural Information Processing Systems 26 (NIPS 2013)*.
- Morin, Frederic and Yoshua Bengio (2005). "Hierarchical Probabilistic Neural Network Language Model". In: *Aistats*. Vol. 5. Citeseer, pp. 246–252.
- Osgood, Charles E., William S. May, and Murray S. Miron (1975). *Cross Cultural Universals of Affective Meaning*. University of Illinois Press.


Pennington, Jeffrey, Richard Socher, and Christopher Manning (2014). "Glove: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: http://www.aclweb.org/anthology/D14-1162.


Peters, Matthew et al. (2018). "Deep Contextualized Word Representations". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: http://aclweb.org/anthology/N18-1202.


Piantadosi, Steven (2024). "Modern language models refute Chomsky's approach to language". In: ed. by Edward Gibson and Moshe Poliak. URL: https://langsci-press.org/catalog/book/434.

- Quillian, M. Ross (1967). "Semantic memory". In: *Semantic information processing*. Ed. by Minsky. Cambridge: MIT Press, pp. 227–270.
- Radford, Alec et al. (2018). "Improving language understanding by generative pre-training". https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.
- Rothe, Sascha, Sebastian Ebert, and Hinrich Schütze (June 2016). "Ultradense Word Embeddings by Orthogonal Transformation". In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, pp. 767–777. arXiv: 1602.07572 [cs.CL]. URL: http://www.aclweb.org/anthology/N16-1091.
- Ruhl, C. (1989). *On monosemy: a study in lingusitic semantics*. State University of New York Press.

- Schank, Roger C. and Robert P. Abelson (1977). *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Hillsdale, NJ: Lawrence Erlbaum.
- Schuster, Mike and Kaisuke Nakajima (2012). "Japanese and Korean voice search". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 5149–5152.
- Schütze, Hinrich (1993). "Word Space". In: *Advances in Neural Information Processing Systems 5*. Ed. by SJ Hanson, JD Cowan, and CL Giles. Morgan Kaufmann, pp. 895–902.
- Smolensky, Paul et al. (2024). *Mechanisms of Symbol Processing for In-Context Learning in Transformer Networks*. arXiv: 2410.17498 [cs.AI]. URL: https://arxiv.org/abs/2410.17498.
- Turian, Joseph, Lev-Arie Ratinov, and Yoshua Bengio (2010). "Word Representations: A Simple and General Method for Semi-Supervised Learning". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Uppsala, Sweden: Association for Computational Linguistics, pp. 384–394.


Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., pp. 5998–6008. arXiv: 1706.03762 [cs.CL]. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.


Virpioja, Sami et al. (2013). "Morfessor 2.0: Python implementation and extensions for Morfessor Baseline". In.


Wells, Roulon S. (1947). "Immediate constituents". In: *Language* 23, pp. 321–343.


Wierzbicka, Anna (1985). *Lexicography and conceptual analysis*. Ann Arbor: Karoma.


Wu, Yonghui et al. (2016). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. URL: http://arxiv.org/abs/1609.08144.