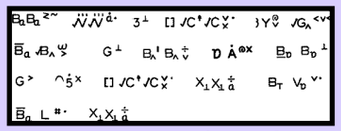
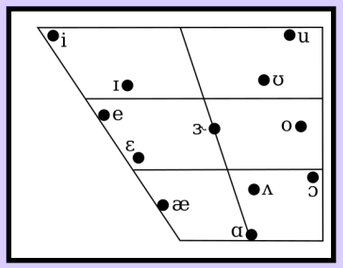
# Linguistics:
# Speech and Lexical Semantics


EECS 183/283a: Natural Language Processing

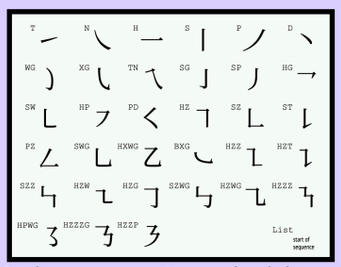
# Linguistic Units



*Stokoe notation of ASL*



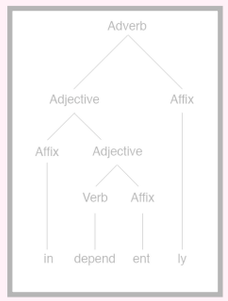

*GA English vowels*



*Chinese stroke primitives, from Jiang et al. 2024*

## Phonemes/ Graphemes

*ASL mouth morphemes Large (CHA) and Small (OO) learnhowtosign.com*




*Wikipedia (Annie Yang)*

fǎn
反 — "anti-"
fǎn kǒng
- 反 恐 [反恐] — "anti-terror"

fǎn jiàoquánde
- 反 教 权 的 [反教權的] — "anti-clerical"

fǎn fàxīsī
- 反 法西斯 [反法西斯] — "anti-fascist"

*Wikipedia*

## Morphemes /Lexemes

*"language" in ASL lifeprint.com*

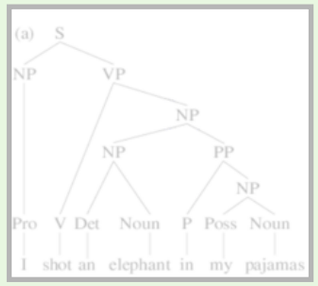
| | indicative | |
|---|---|---|
| | singular | plural |
| present | ich fege | wir **fegen** |
| | du fegst | ihr fegt |
| | er fegt | sie **fegen** |
| preterite | ich fegte | wir fegten |
| | du fegtest | ihr fegtet |
| | er fegte | sie fegten |
| imperative | feg (du)<br>fege (du) | fegt (ihr) |

*conjugation of German "fegen" (to sweep) (Wiktionary)*

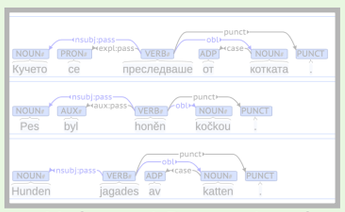
*ajar* = to teach
*ajari* = to teach (imperative, locative)
*ajarilah* = to teach (jussive, locative)
*ajarkan* = to teach (imperative, causative/applicative)
*ajarkanlah* = to teach (jussive, causative/applicative)
*ajarlah* = to teach (jussive, active)
*ajaran* = teachings
*belajar* = to learn (intransitive, active)
*diajar* = to be taught (intransitive)
*diajari* = to be taught (transitive, locative)
*diajarkan* = to be taught (transitive, causative/applicative)
*dipelajari* = to be studied (locative)
*dipelajarkan* = to be studied (causative/applicative)
*mempelajari* = to study (locative)

*Indonesian "ajar" (to teach) with affixes (Wikipedia)*

## Words



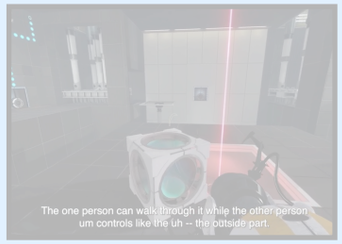

*syntactic ambiguity, from UBC CPSC522*




*Universal Dependencies*

## Constituents /Phrases


*靜靜養貓 (YouTube)*




*Portal 2*

Юная мышка Роза допела любимую песенку и теперь вслушивалась, как таяло эхо последних звуков. Роза была довольна собой: наконец-то она нашла место, откуда море отлично просматривалось. Вечерело, лучи солнца, заходящего за ее спиной, окрасили воду в золотистый и багровый цвета.

— Эгей, Роза, пожалуйте-ка ужинать. Не для того я старался, чтобы все это теперь простыло, значится... то есть отсырело. Нет уж, дудки!

Спутник Розы, крот Грумм, призывно помахал ей лапой. Мышка подошла к маленькому костерку, на котором он стряпал, и принюхалась:

— Ого, лепешки из дикого овса и суп из зелени!

Грумм улыбнулся, отчего его бархатная мордочка еморщилась, и помахал маленькой поварешкой, которую всегда носил за поясом наподобие меча:

— Садись, отведай супчику.

Роза взяла глубокую раковину морского гребешка, наполненную ароматным супом. Положив свою лепешку на плоский камень у костра, чтобы та не остыла, мышка отхлебнула супа и покачала головой:

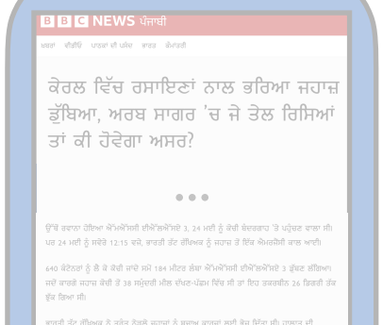
— Ты хуже старой няньки, Грумм Канавкинс. Бьюсь об заклад, если бы я позволила, ты бы меня и спать укладывал, и на ночь баюкал.

Перед носом Розы замаячила поварешка.

— Так что же тебе, спать, это самое... и не надобно вовсе, так, что ли? Предстань, ты батюшка твой екажет, если я тебя домой доставлю всю усталую такую да голодную, а?

*sentences in a Russian book (reddit)*

## Utterances/ Sentences




*BBC News in Punjabi*

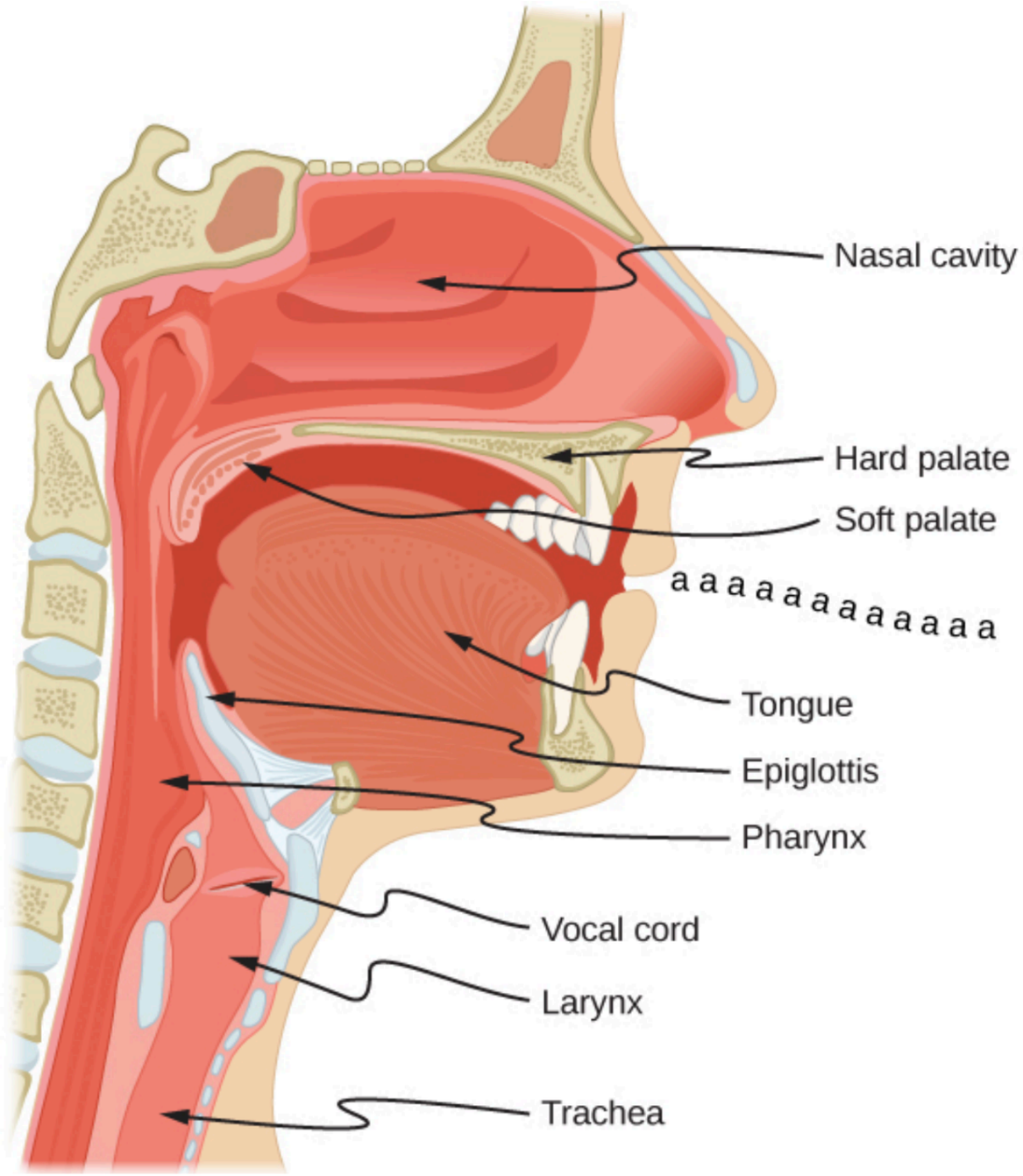
*Portal 2 dialogues*

12 CHAIR EID: Thank you. Does the Committee
13 have any questions? I do not see any. Thank you so
14 much for your presentation today.
15 MS. RESNIK: We appreciate your time and
16 that you enabled us all to offer comments. Many
17 thanks.
18 CHAIR EID: Thank you.
19 All right. We are now going to turn to
20 Carter Phillips, and we have now moved to Rule 29
21 comments.
22 MR. PHILLIPS: Judge Eid, can you see me and
23 hear me okay?
24 CHAIR EID: Yes, thank you.
25 MR. PHILLIPS: Okay. I apologize. It

*US Supreme Court*

## Discourse/ Dialogue

# Speech Production




Vocal articulators that produce speech

- Air passing through vocal articulators produces speech
- Vocal folds, tongue, jaw, lips, velum are both independently and jointly controlled to produce different sounds
  - E.g., vocal fold vibration causes voicing
- The output of vocal articulation is an acoustic pressure wave

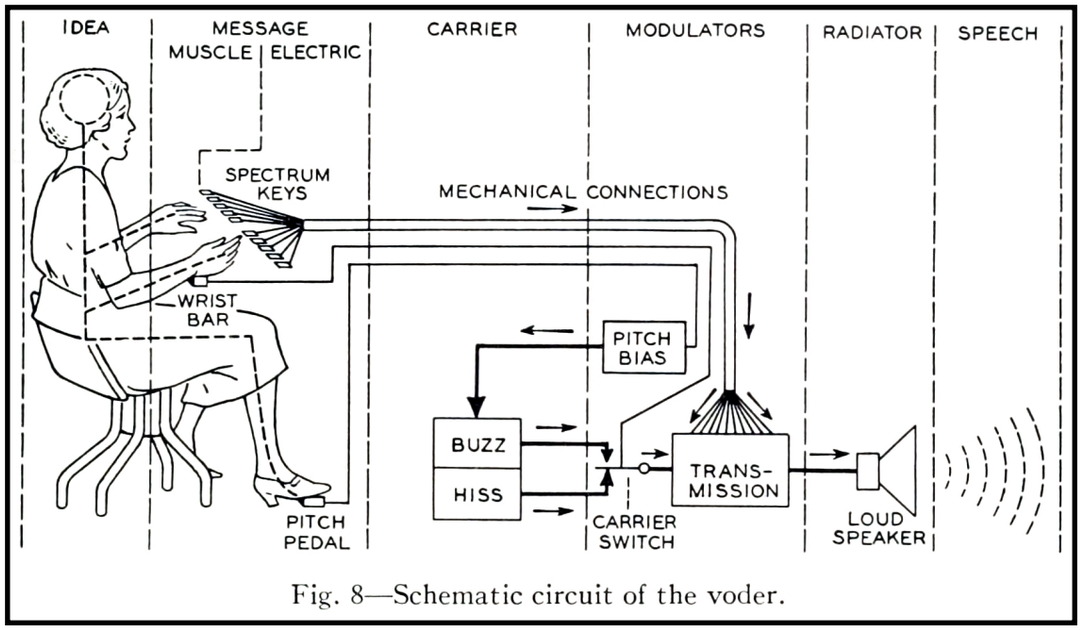
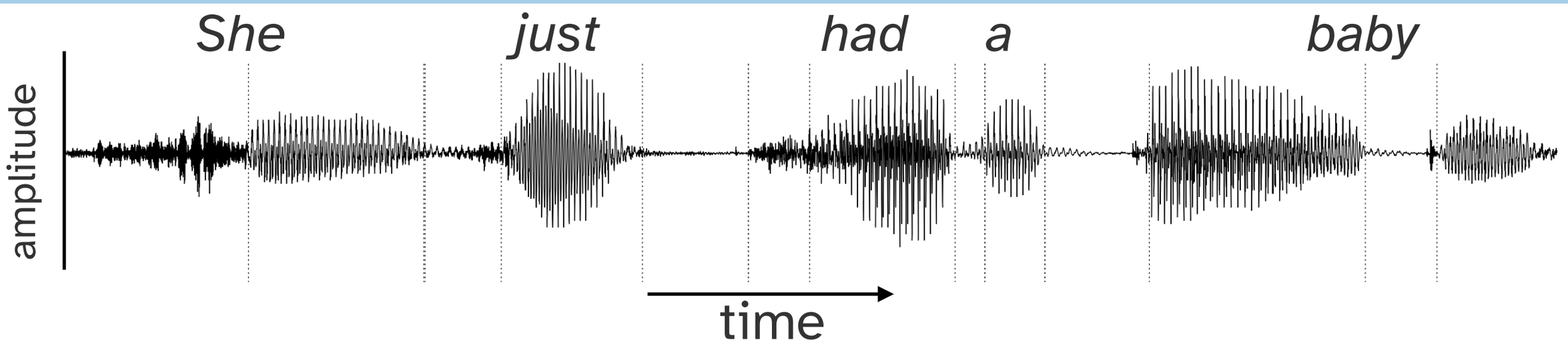



Fig. 8—Schematic circuit of the voder.

# Speech Representations



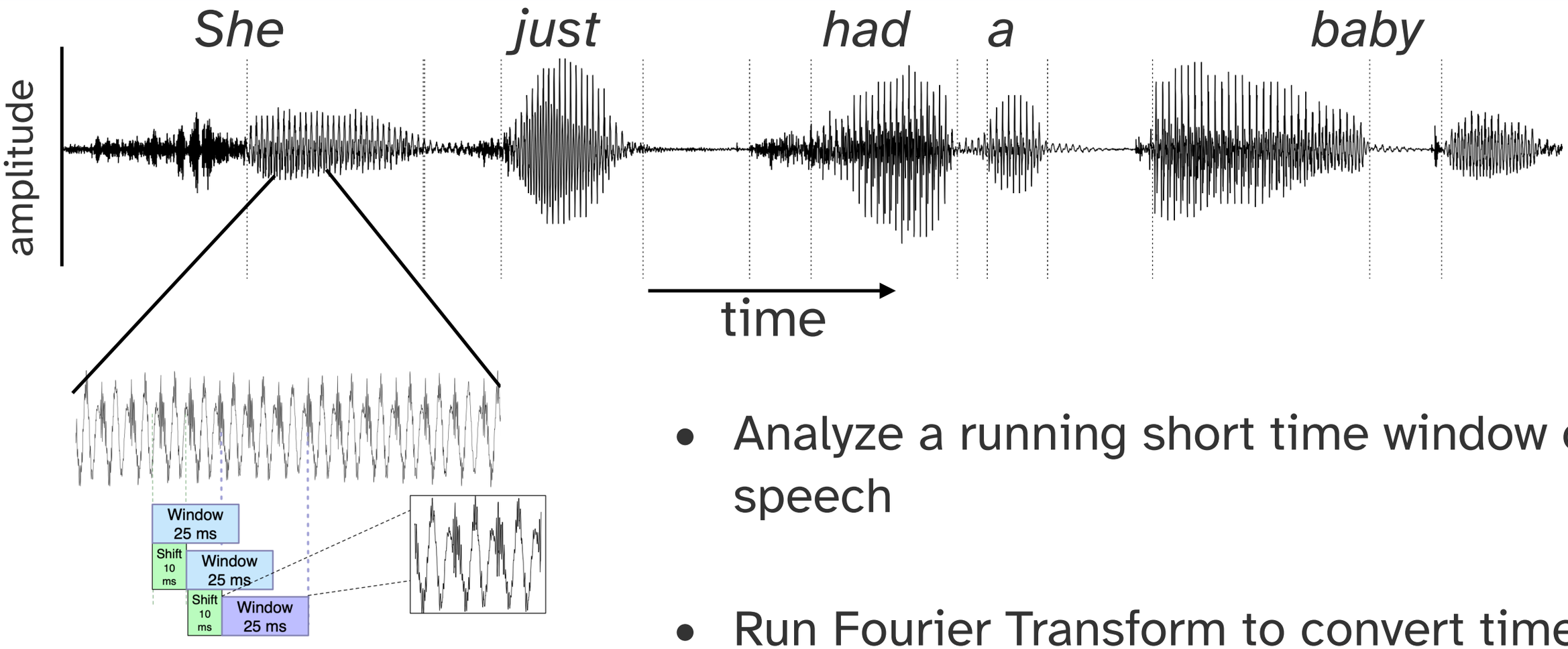

- The result of the vocal articulation is an acoustic pressure wave

- Speech can thus be represented as an **acoustic waveform**

- Waveforms are continuous time series cannot be easily analyzed or interpreted, or computed with

- Signal processing can give more interpretable information

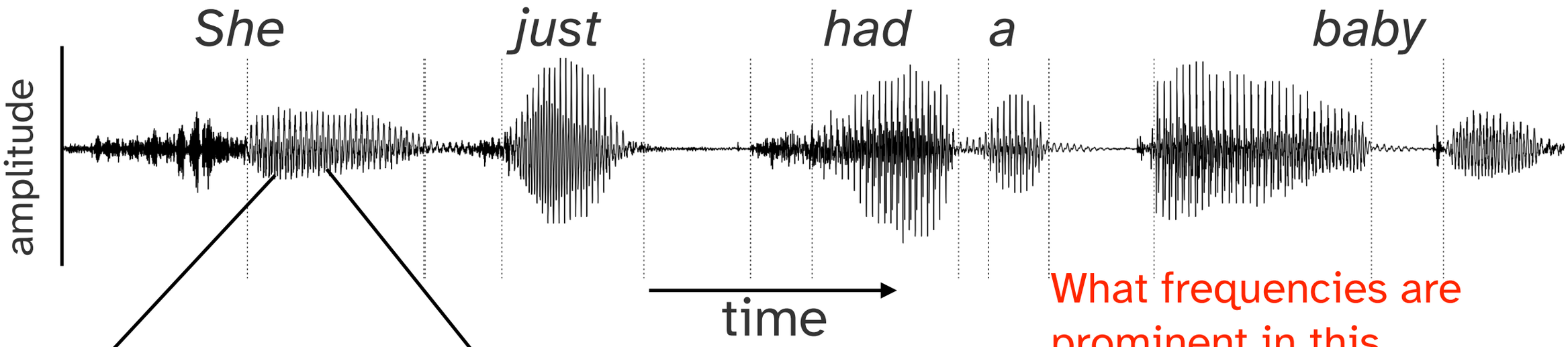
# Speech Waveform

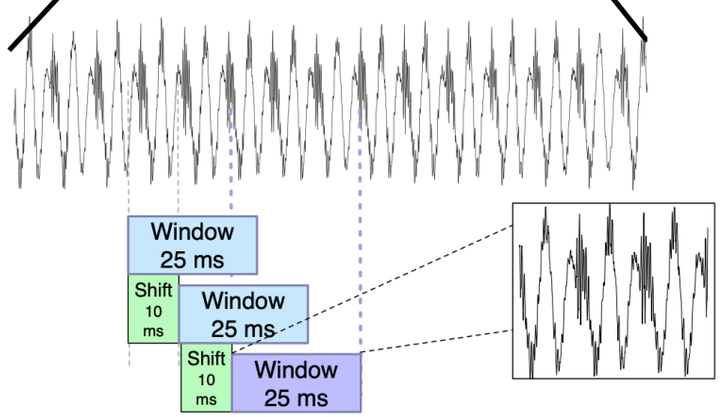



- Analyze a running short time window of speech

- Run Fourier Transform to convert time to time-frequency representation

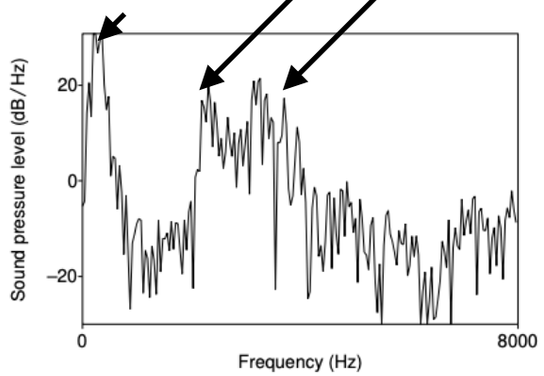
# Speech Waveform



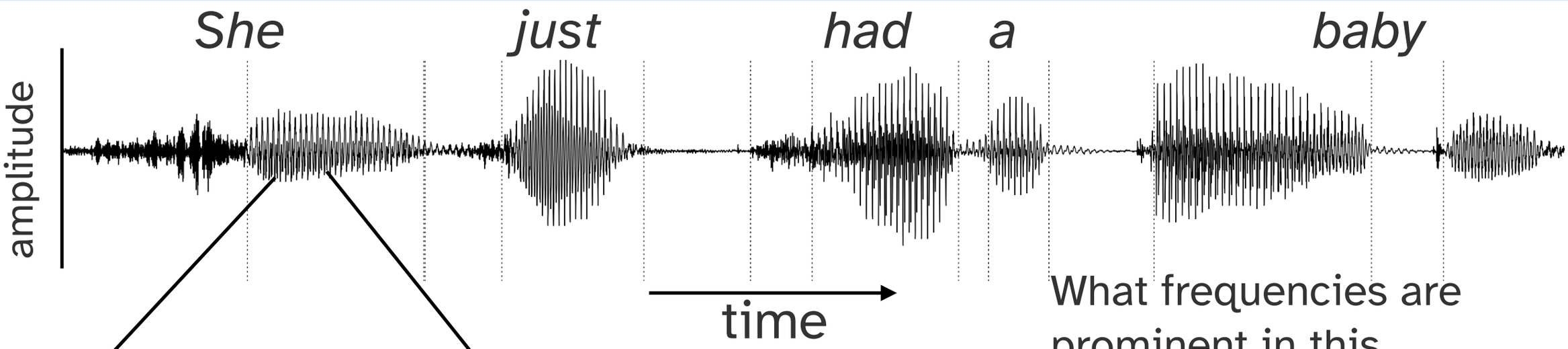



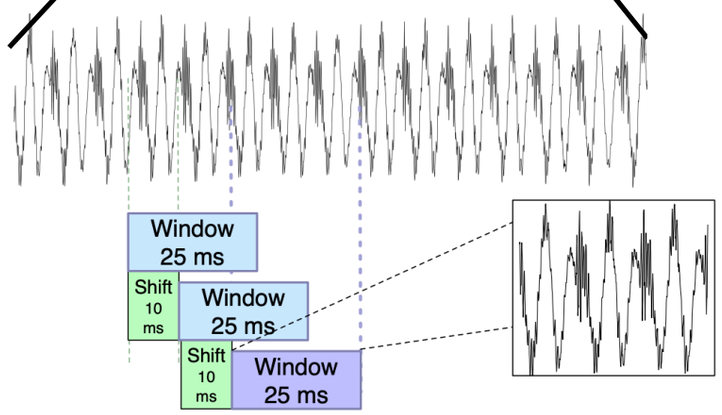

Discrete Fourier Transform

What frequencies are prominent in this sound chunk ?



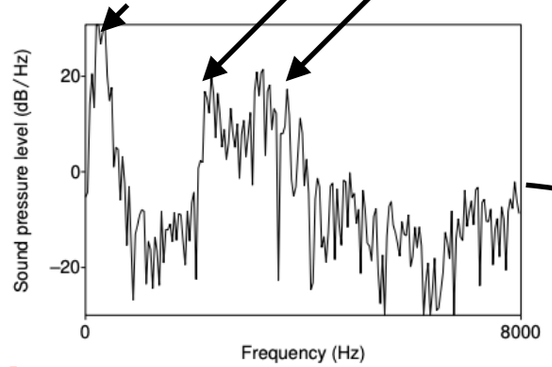
# Speech Spectrogram



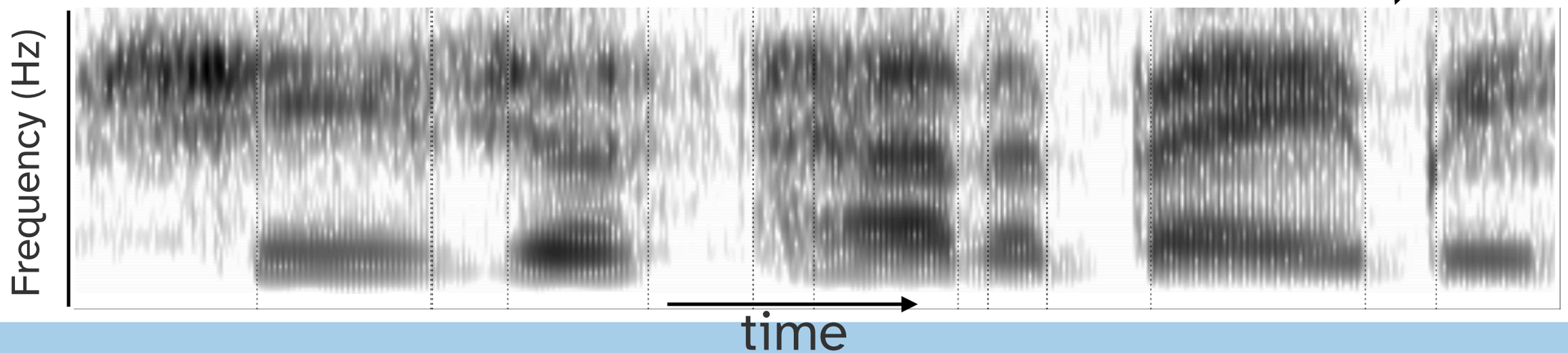

Discrete Fourier Transform

What frequencies are prominent in this sound chunk ?

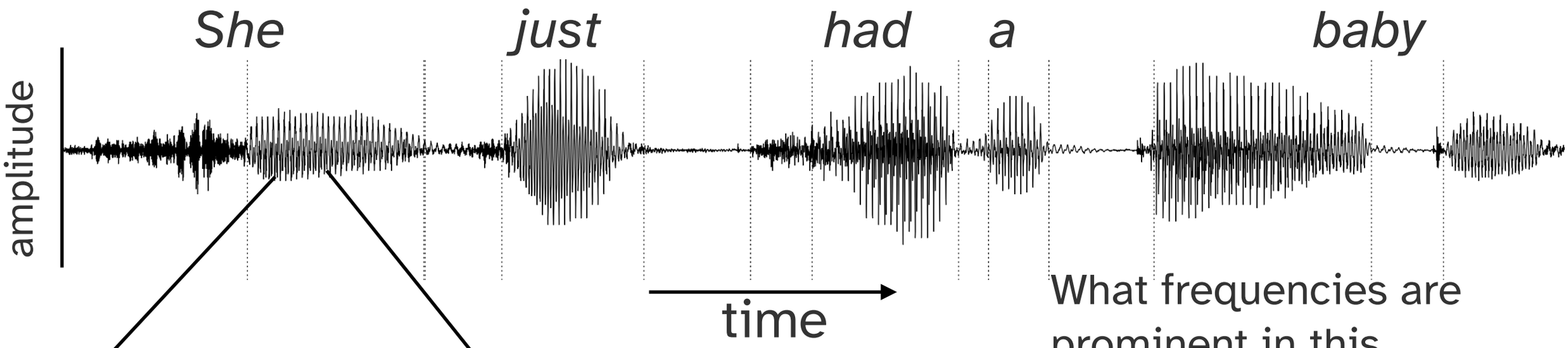


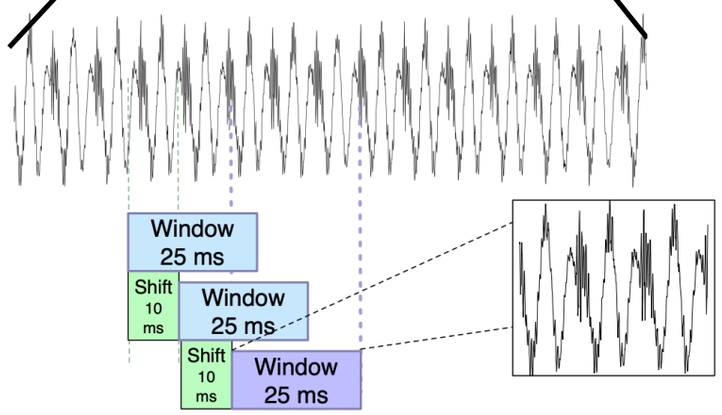

Flip axes and convert pressure into pixel intensity for entire waveform

## Speech Spectrogram



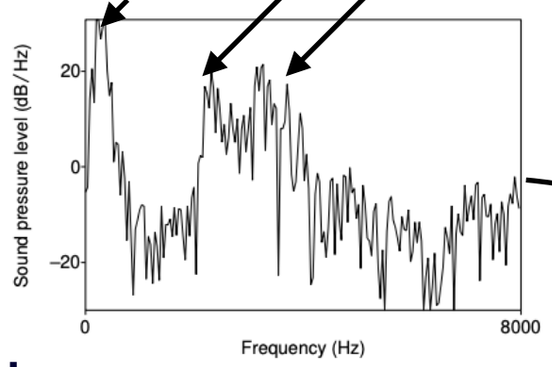
# Speech Spectrogram




Discrete Fourier Transform

What frequencies are prominent in this sound chunk ?

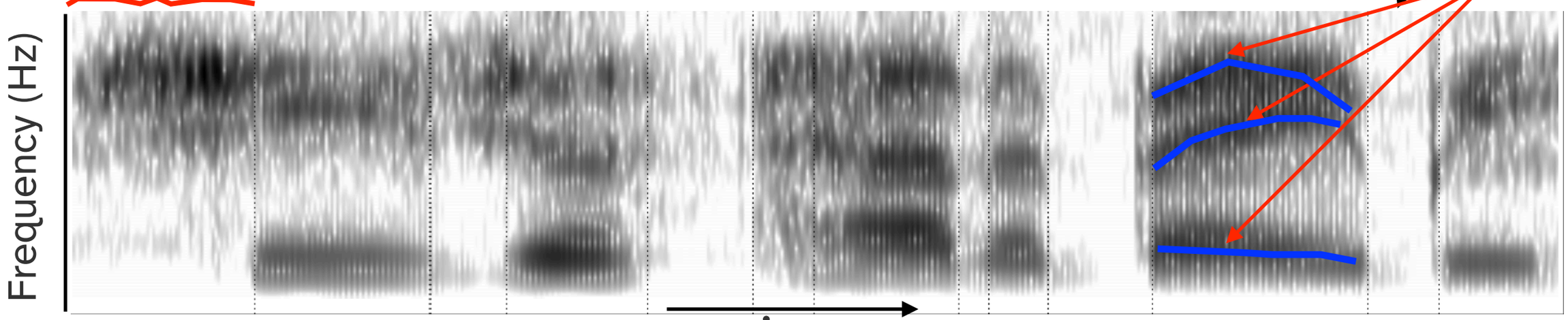



Flip axes and convert pressure into pixel intensity for entire waveform

## Speech Spectrogram

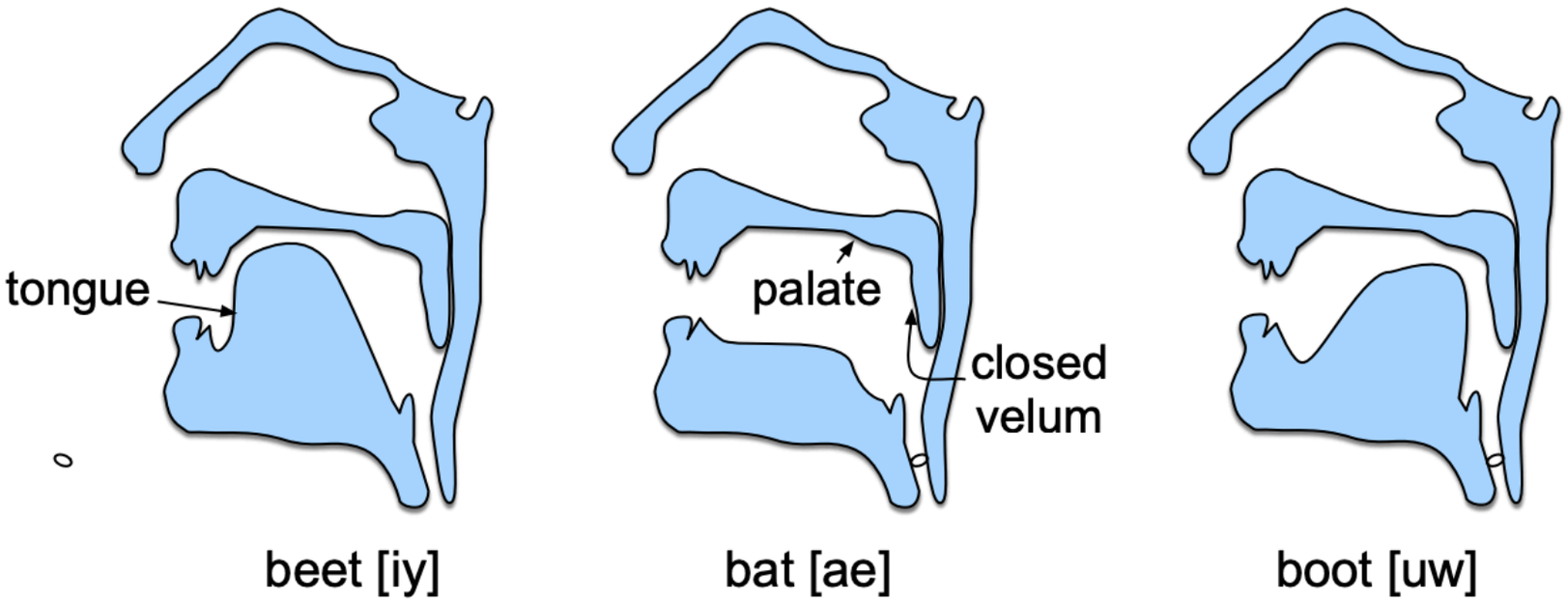
Unvoiced

Formants

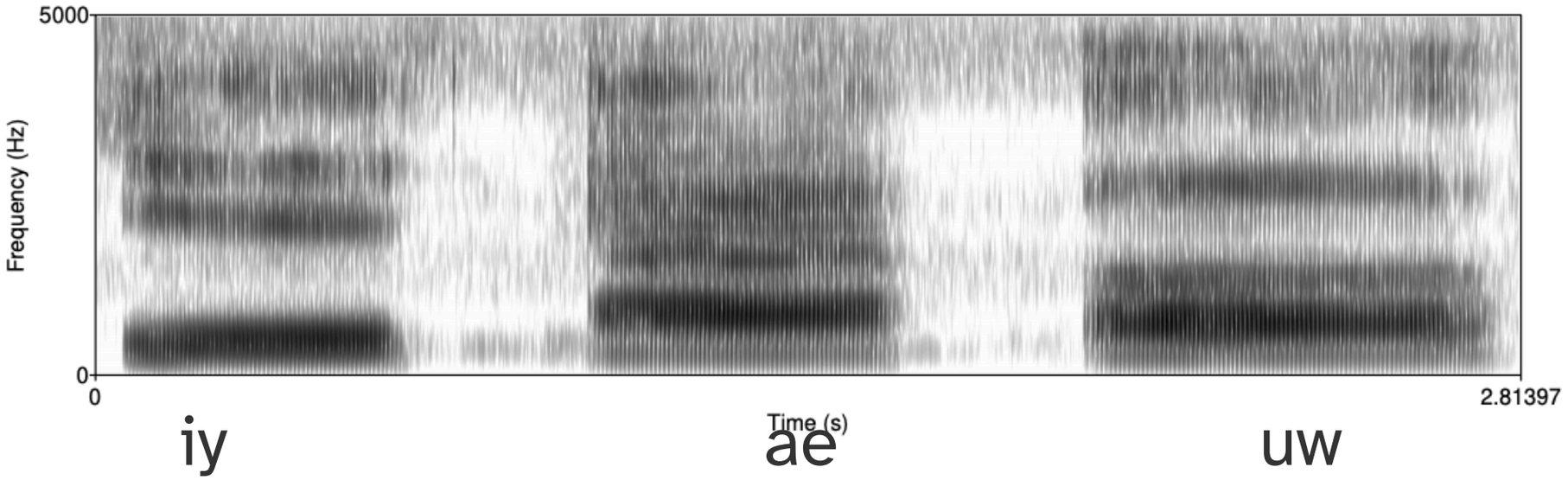


# Phonetics

Study of speech sounds — their physical production, spectral and perceptual properties

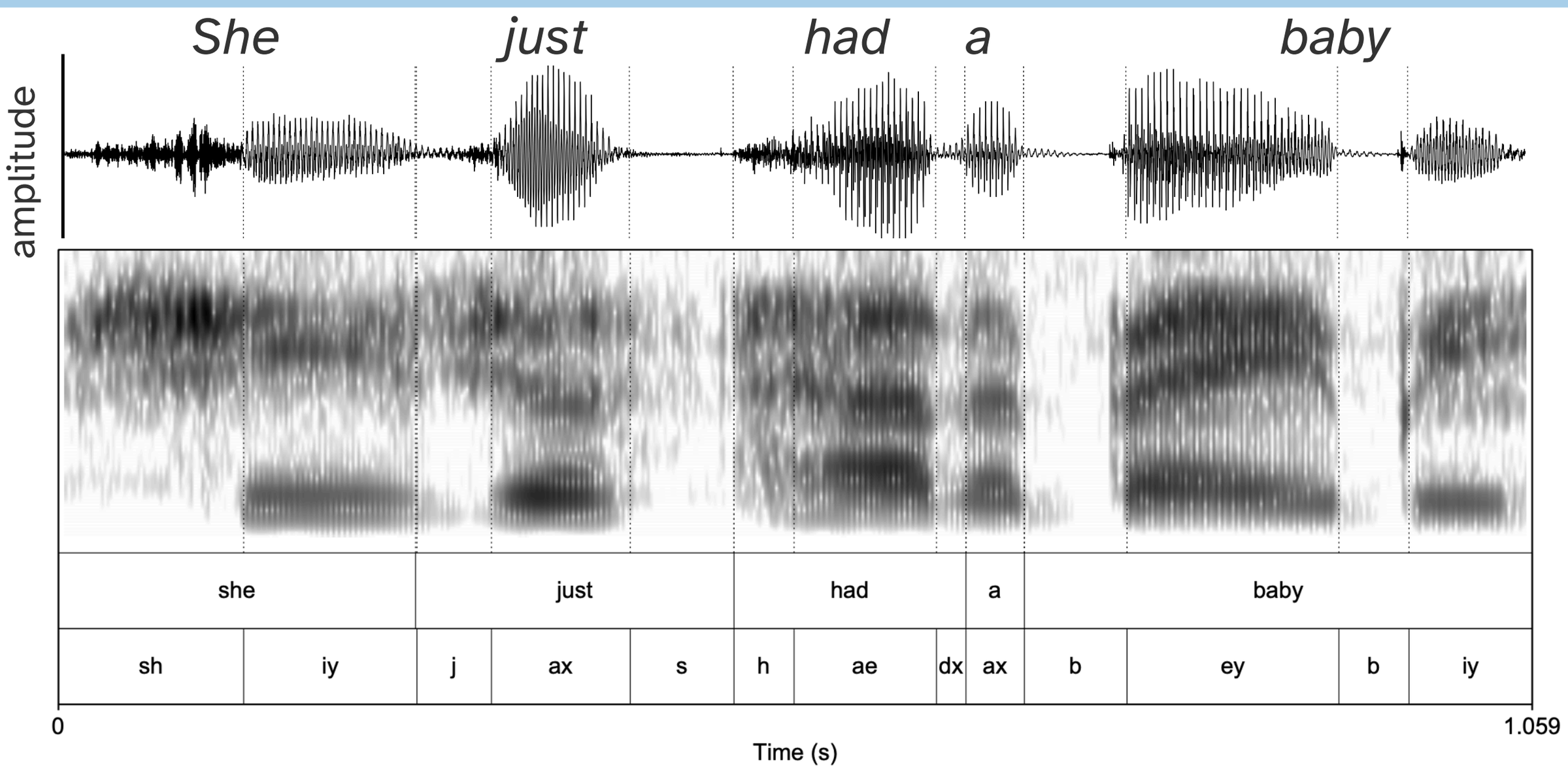



Articulatory Phonetics



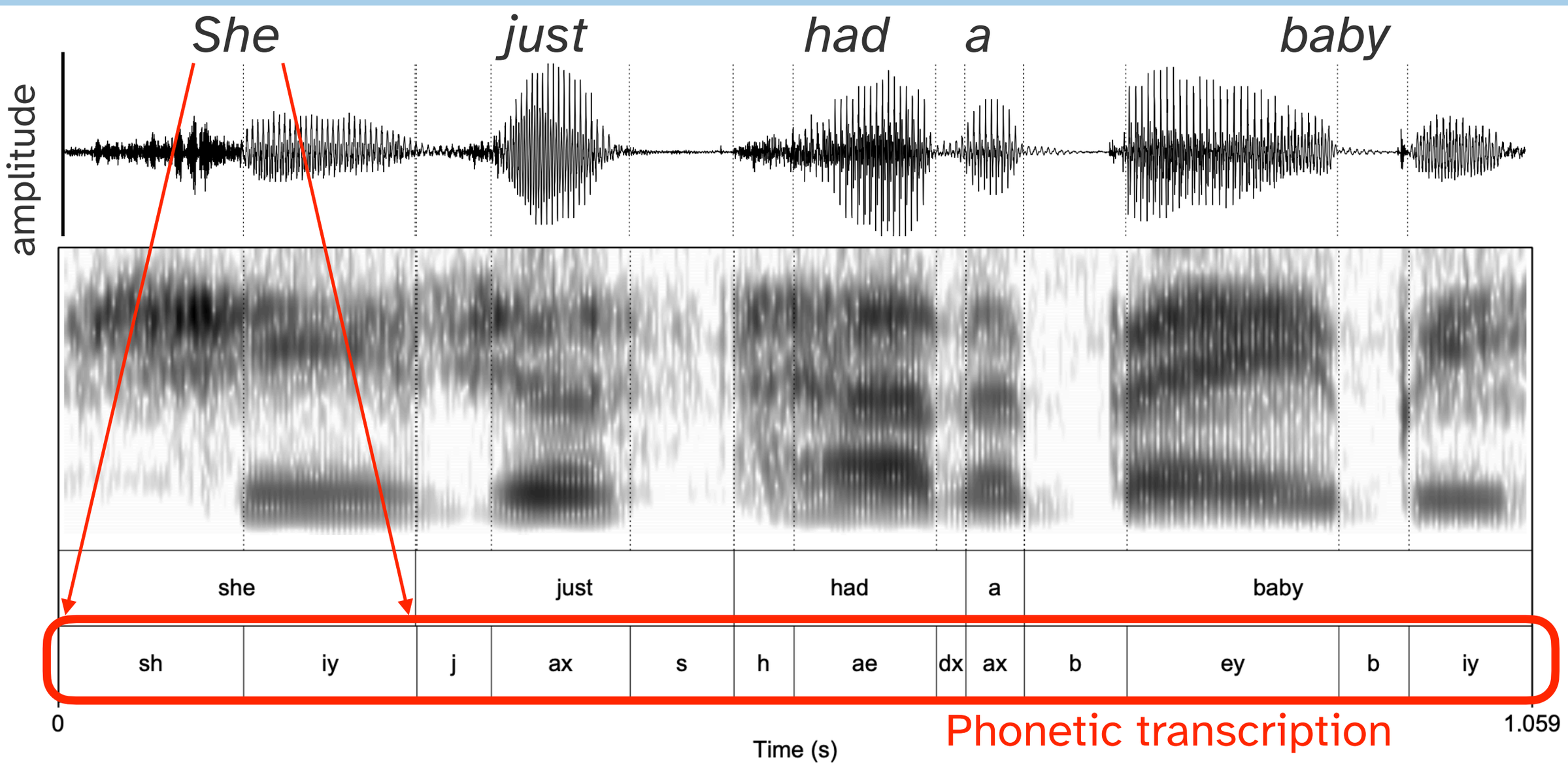

Acoustic Phonetics

# Acoustic Phonetics




- Spectrogram reveals some segmental structure with distinct properties
- These are phonemes — perceptually distinct speech sounds

# Acoustic Phonetics



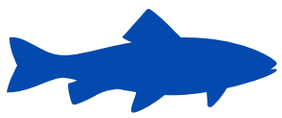

- Spectrogram reveals some segmental structure with distinct properties
- These are phonemes — perceptually distinct speech sounds

# International Phonetic Alphabet (IPA)

- Phoneticians compiled a common set of sounds used to codify different speech sounds (across languages)

- English spelling (aka orthography) is not phonetic: about 40 distinct phonemes represented by 26 graphemes
  - About 16 vowel sounds
  - About 24 consonant sounds

*women*



*ghoti*

*though* *nation*

THE INTERNATIONAL PHONETIC ALPHABET (revised to 2020)

CONSONANTS (PULMONIC) — © 2020 IPA

| | Bilabial | Labiodental | Dental | Alveolar | Postalveolar | Retroflex | Palatal | Velar | Uvular | Pharyngeal | Glottal |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Plosive | p b | | | t d | | ʈ ɖ | c ɟ | k g | q ɢ | | ʔ |
| Nasal | m | ɱ | | n | | ɳ | ɲ | ŋ | ɴ | | |
| Trill | ʙ | | | r | | | | | ʀ | | |
| Tap or Flap | | ⱱ | | ɾ | | ɽ | | | | | |
| Fricative | ɸ β | f v | θ ð | s z | ʃ ʒ | ʂ ʐ | ç ʝ | x ɣ | χ ʁ | ħ ʕ | h ɦ |
| Lateral fricative | | | | ɬ ɮ | | | | | | | |
| Approximant | | ʋ | | ɹ | | ɻ | j | ɰ | | | |
| Lateral approximant | | | | l | | ɭ | ʎ | ʟ | | | |

Symbols to the right in a cell are voiced, to the left are voiceless. Shaded areas denote articulations judged impossible.

CONSONANTS (NON-PULMONIC)

| Clicks | Voiced implosives | Ejectives |
|---|---|---|
| ʘ Bilabial | ɓ Bilabial | ’ Examples: |
| ǀ Dental | ɗ Dental/alveolar | pʼ Bilabial |
| ǃ (Post)alveolar | ʄ Palatal | tʼ Dental/alveolar |
| ǂ Palatoalveolar | ɠ Velar | kʼ Velar |
| ǁ Alveolar lateral | ʛ Uvular | sʼ Alveolar fricative |

OTHER SYMBOLS

- ʍ Voiceless labial-velar fricative
- w Voiced labial-velar approximant
- ɥ Voiced labial-palatal approximant
- ʜ Voiceless epiglottal fricative
- ʢ Voiced epiglottal fricative
- ʡ Epiglottal plosive
- ɕ ʑ Alveolo-palatal fricatives
- ɺ Voiced alveolar lateral flap
- ɧ Simultaneous ʃ and x

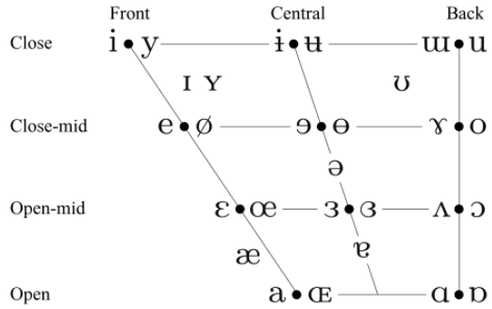
Affricates and double articulations can be represented by two symbols joined by a tie bar if necessary. t͡s k͡p

DIACRITICS

| | | | | | |
|---|---|---|---|---|---|
| ◌̥ Voiceless | n̥ d̥ | ◌̤ Breathy voiced | b̤ a̤ | ◌̪ Dental | t̪ d̪ |
| ◌̬ Voiced | s̬ t̬ | ◌̰ Creaky voiced | b̰ a̰ | ◌̺ Apical | t̺ d̺ |
| ◌ʰ Aspirated | tʰ dʰ | ◌̼ Linguolabial | t̼ d̼ | ◌̻ Laminal | t̻ d̻ |
| ◌̹ More rounded | ɔ̹ | ◌ʷ Labialized | tʷ dʷ | ◌̃ Nasalized | ẽ |
| ◌̜ Less rounded | ɔ̜ | ◌ʲ Palatalized | tʲ dʲ | ◌ⁿ Nasal release | dⁿ |
| ◌̟ Advanced | u̟ | ◌ˠ Velarized | tˠ dˠ | ◌ˡ Lateral release | dˡ |
| ◌̠ Retracted | e̠ | ◌ˤ Pharyngealized | tˤ dˤ | ◌̚ No audible release | d̚ |
| ◌̈ Centralized | ë | ~ Velarized or pharyngealized | ɫ | | |
| ◌̽ Mid-centralized | e̽ | ◌̝ Raised | e̝ (ɪ̝ = voiced alveolar fricative) | | |
| ◌̩ Syllabic | n̩ | ◌̞ Lowered | e̞ (β̞ = voiced bilabial approximant) | | |
| ◌̯ Non-syllabic | e̯ | ◌̘ Advanced Tongue Root | e̘ | | |
| ◌˞ Rhoticity | ɚ a˞ | ◌̙ Retracted Tongue Root | e̙ | | |

Some diacritics may be placed above a symbol with a descender, e.g. ŋ̊

VOWELS




Where symbols appear in pairs, the one to the right represents a rounded vowel.

SUPRASEGMENTALS

| | |
|---|---|
| ˈ Primary stress | ˌfoʊnəˈtɪʃən |
| ˌ Secondary stress | |
| ː Long | eː |
| ˑ Half-long | eˑ |
| ◌̆ Extra-short | ĕ |
| \| Minor (foot) group | |
| ‖ Major (intonation) group | |
| . Syllable break | ɹi.ækt |
| ‿ Linking (absence of a break) | |

TONES AND WORD ACCENTS

| LEVEL | | CONTOUR | |
|---|---|---|---|
| e̋ or ◌̋ | Extra high | ě or ◌̌ | Rising |
| é ◌́ | High | ê ◌̂ | Falling |
| ē ◌̄ | Mid | e᷄ ◌᷄ | High rising |
| è ◌̀ | Low | e᷅ ◌᷅ | Low rising |
| ȅ ◌̏ | Extra low | e᷈ ◌᷈ | Rising-falling |
| ↓ Downstep | | ↗ Global rise | |
| ↑ Upstep | | ↘ Global fall | |

# International Phonetic Alphabet (IPA)

- Phoneticians compiled a common set of sounds used to codify different speech sounds (across languages)

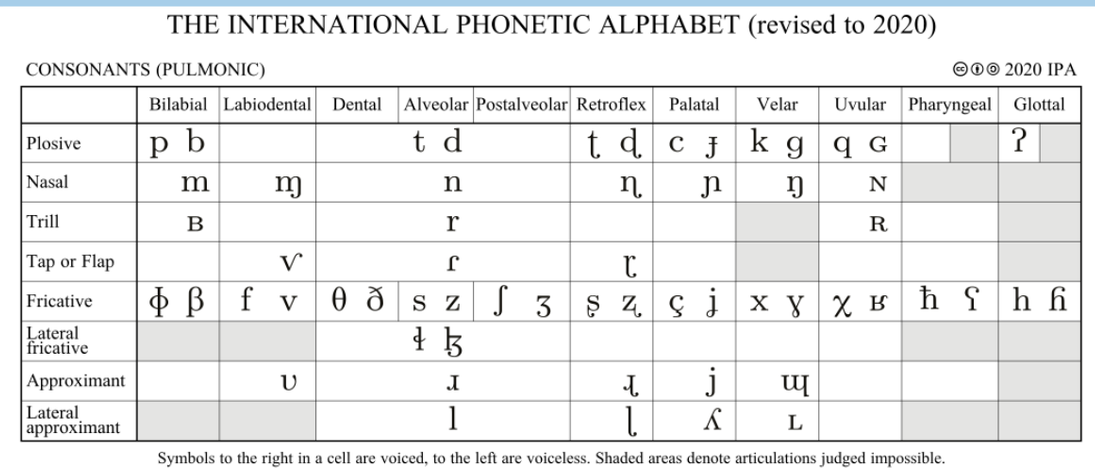
## Central Rotokas (Papua New Guinea): 12 consonants, 10 vowels

| | Bilabial | Alveolar | Velar |
|---|---|---|---|
| **Voiceless** | p | t | k |
| **Voiced** | b | d | g |

| | Front | Central | Back |
|---|---|---|---|
| **Close** | i (iː) | | u (uː) |
| **Close-mid** | e (eː) | | o (oː) |
| **Open** | | a (aː) | |

## Archi (Dagestan, Russia)

| | | | Labial | Dental | | (Post)-alveolar | | (Pre-)velar | | Uvular | | | | Epiglottal | Glottal |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | plain | lab. | plain | lab. | plain | lab. | plain | lab. | phar. | phar.+lab. | | |
| **Nasal** | | | m | n | | | | | | | | | | | |
| **Plosive** | **voiced** | | b | d | dʷ[2] | | | g | gʷ | | | | | | |
| | **voiceless** | | p | t | tʷ | | | k | kʷ | q | qʷ | qˤ | qˤʷ | ʔ | ʔ[1] |
| | **fortis** | | pː[1] | tː[1] | | | | kː[1] | kʷː[2] | qˤː[1] | | qˤː | | | |
| | **ejective** | | pʼ | tʼ | | | | kʼ | kʷʼ | qʼ | qʷʼ | qˤʼ | qˤʷʼ | | |
| **Affricate** | **voiceless** | **lenis** | | t͡s | t͡sʷ[2] | t͡ʃ | t͡ʃʷ | k͡ | k͡ʷ | | | | | | |
| | | **fortis** | | t͡sː[3] | | | | | | | | | | | |
| | **ejective** | **lenis** | | t͡sʼ | t͡sʷʼ | t͡ʃʼ | t͡ʃʷʼ | k͡ʼ | k͡ʷʼ | | | | | | |
| | | **fortis** | | t͡sˤː[1] | | t͡ʃˤː[2] | | | | | | | | | |
| **Fricative** | **voiceless** | **lenis** | | s | sʷ[2] | ʃ | ʃʷ | | ʷ | χ | χʷ | χˤ | χˤʷ | | h |
| | | **fortis** | | sː | sʷː[2] | ʃː | ʃʷː | ː | ʷː | χː | χʷː | χˤː | χˤʷː | | |
| | **voiced** | | | z | zʷ | ʒ | ʒʷ | ʟ[1] | | ʁ | ʁʷ[2] | ʁˤ | ʁˤʷ | | |
| **Trill** | | | | r | | | | | | | | | | ʜ | |
| **Approximant** | | | | l | | j | | | w | | | | | | |



# International Phonetic Alphabet (IPA)

- Vowels are characterized by jaw position and tongue shape
- Some vowels also use lips (eg. sound *uw* in *cool*)

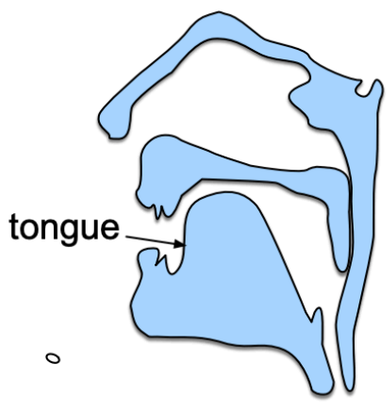
Tongue frontness

Jaw aperture

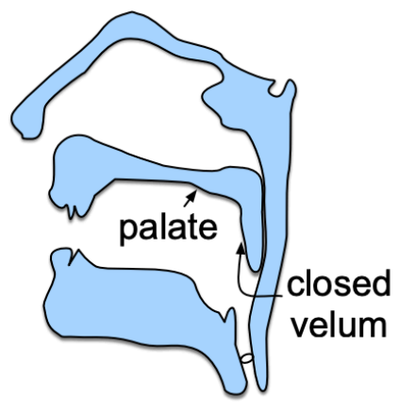
VOWELS

| | Front | Central | Back |
|---|---|---|---|
| Close | i • y | ɨ • ʉ | ɯ • u |
| | ɪ ʏ | | ʊ |
| Close-mid | e • ø | ɘ • ɵ | ɤ • o |
| | | ə | |
| Open-mid | ɛ • œ | ɜ • ɞ | ʌ • ɔ |
| | æ | ɐ | |
| Open | a • ɶ | | ɑ • ɒ |

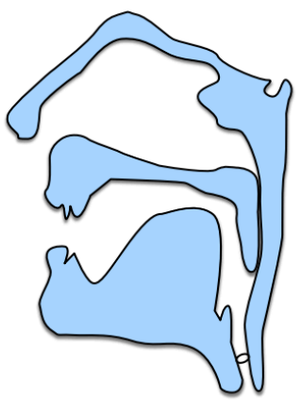
Where symbols appear in pairs, the one to the right represents a rounded vowel.




beet [iy]



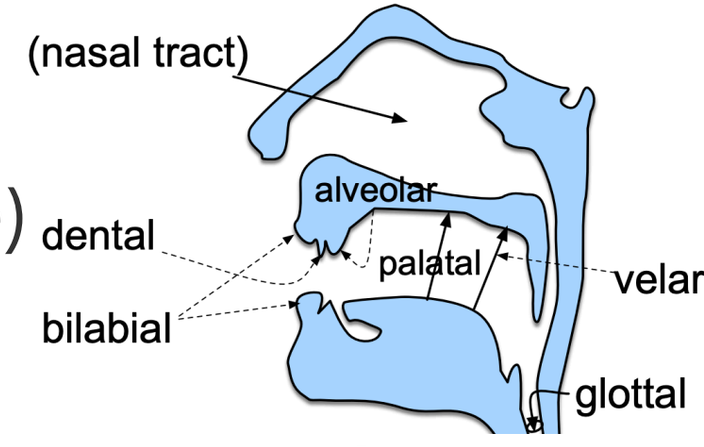

bat [ae]



boot [uw]

# International Phonetic Alphabet (IPA)

- Consonants are characterized by place and manner of articulation
  - /p/ is caused by constriction at lips (labial)
  - /p/ is caused by sudden release of air (plosive)




Place of articulation

Manner of articulation

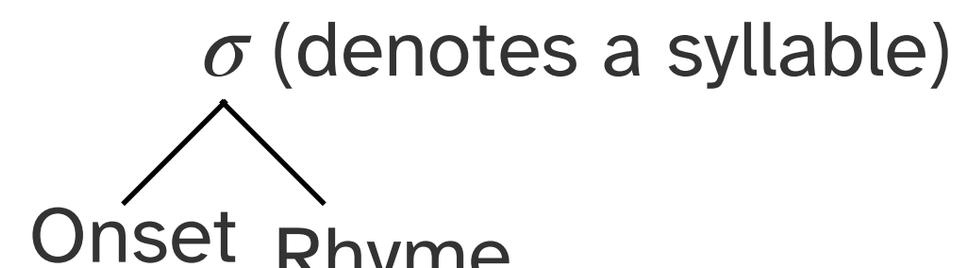
CONSONANTS (PULMONIC) ©⓪◎ 2020 IPA

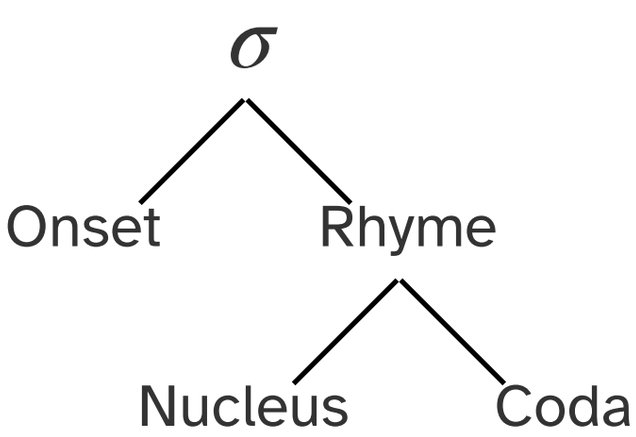
| | Bilabial | Labiodental | Dental | Alveolar | Postalveolar | Retroflex | Palatal | Velar | Uvular | Pharyngeal | Glottal |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Plosive | p b | | | t d | | ʈ ɖ | c ɟ | k g | q ɢ | | ʔ |
| Nasal | m | ɱ | | n | | ɳ | ɲ | ŋ | ɴ | | |
| Trill | ʙ | | | r | | | | | ʀ | | |
| Tap or Flap | | ⱱ | | ɾ | | ɽ | | | | | |
| Fricative | ɸ β | f v | θ ð | s z | ʃ ʒ | ʂ ʐ | ç ʝ | x ɣ | χ ʁ | ħ ʕ | h ɦ |
| Lateral fricative | | | | ɬ ɮ | | | | | | | |
| Approximant | | ʋ | | ɹ | | ɻ | j | ɰ | | | |
| Lateral approximant | | | | l | | ɭ | ʎ | ʟ | | | |

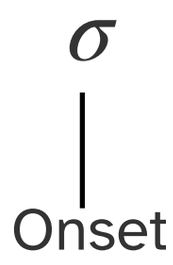
Symbols to the right in a cell are voiced, to the left are voiceless. Shaded areas denote articulations judged impossible.

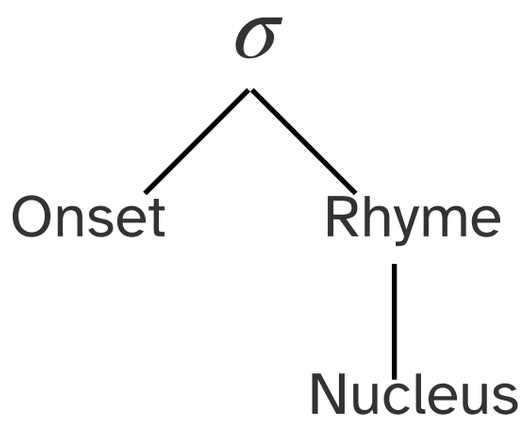
# Lexical Phonology

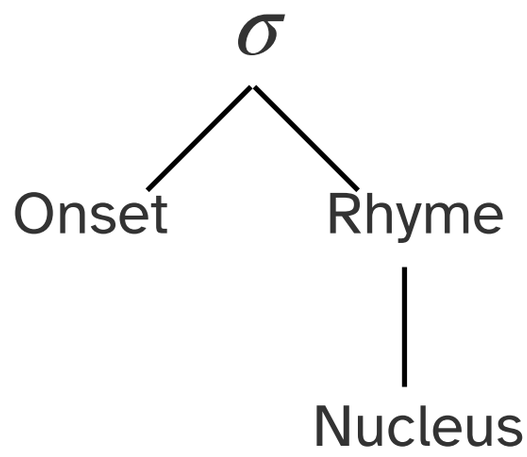
- Phonology is the study of rules that govern the organization of sounds in a language ( Phonemes → Syllables → Words)
- English syllable structure: (C3)V(C4)
  /strɛŋkθs/
- Hawaiian syllable structure: (C)V(C)
- Georgian syllable structure: (C8)V(C5)
  გვბრდღვნის /ˈgvbrdɣvnis/



# Letters to Sounds

- Pronunciation dictionaries (often made by linguists) give the syllables and phonemes within each word in vocabulary
  - CMU Phonetic Dictionary gives the syllabic and phonetic spellings for >110K words in English
  - ML based phonetizers are built on such phonetic dictionaries

| | | | | | |
|---|---|---|---|---|---|
| Graphemes | She | just | had | a | baby |
| IPA | ʃiː | ʤʌst | hæd | ə | ˈbeɪbi |
| Arpabet | sh iy | jh ah s t | h ae d | ah | b ey b iy |

Arpabet is an ASCII friendly representation of IPA

# Letters to Sounds

## The CMU Pronouncing Dictionary

query | phonemes | about | | Speech at

**Look up the pronunciation for a word or phrase in CMUdict (version 0.7b)**

Look It Up

Show Lexical Stress

BERKELEY
B ER1 K L IY0 .

## Berkeley

Entry Discussion Citations

### English [ edit ]

#### Alternative forms [ edit ]

- (*surname*): Barclay, Barclift, Barkley

#### Etymology [ edit ]

From Old English *beorc* (“birch”) + *leah* (“meadow, leigh”). Equivalent to *birch* +

#### Pronunciation [ edit ]

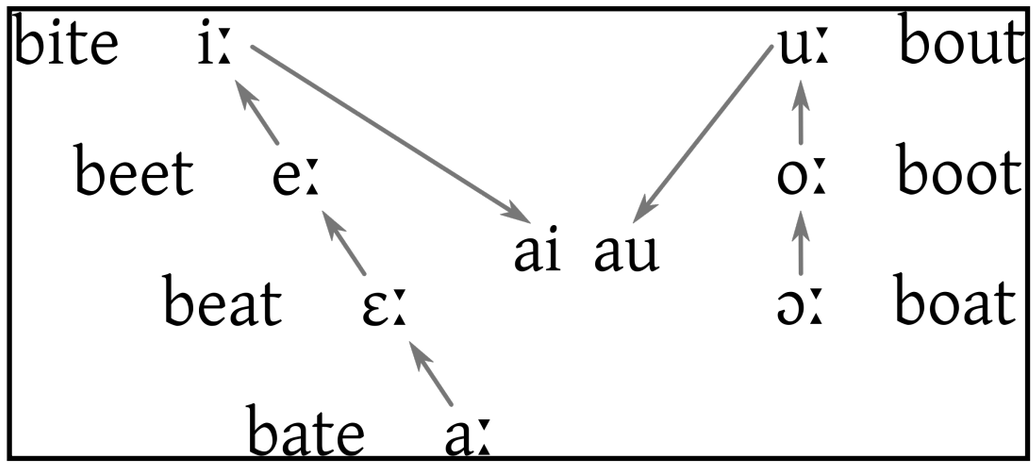
- (*Received Pronunciation*) IPA(key): (*English town, barony*) /ˈbɑːkli/, (*American city*) /ˈbɜːkli/
- (*General American*) IPA(key): /ˈbɝkli/
- Audio (*Southern England*): 0:01

| Variety | Location | 國 edit |
|---|---|---|
| Mandarin | Beijing | /kuo$^{35}$/ |
| | Harbin | /kuo$^{213}$/ |
| | Tianjin | /kuo$^{45}$/ |
| | Jinan | /kuə$^{55}$/ |
| | Qingdao | /kuə$^{55}$/ |
| | Zhengzhou | /kuo$^{24}$/ /kuɛ$^{24}$/ |
| | Xi'an | /kuei$^{21}$/ |
| | Xining | /kuɨ$^{44}$/ |
| | Yinchuan | /kuə$^{13}$/ |
| | Lanzhou | /kuə$^{13}$/ |
| | Ürümqi | /kuɤ$^{213}$/ |
| | Wuhan | /kuɤ$^{213}$/ |
| | Chengdu | /kue$^{31}$/ |
| | Guiyang | /kuɛ$^{21}$/ |
| | Kunming | /ko$^{31}$/ |
| | Nanjing | /kueʔ$^{5}$/ |
| | Hefei | /kuɐʔ$^{5}$/ |

Wiktionary
The free dictionary

# Sound Variation

- What might cause differences in pronunciation?
  - Dialectical differences
  - Native language when speaking a non-native language
  - Context — formality, noise levels, etc.
  - Language change over time




*Goran tek-en, Wikipedia*

- We learn sounds before we are born!

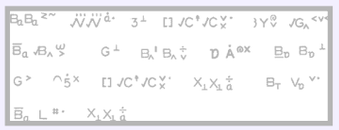
| Word | Vowel pronunciation | |
|---|---|---|
| | Late Middle English before the GVS | Modern English after the GVS |
| b*i*te | [iː] | [aɪ] |
| m*ee*t | [eː] | [iː] |
| m*ea*t | [ɛː] | |
| s*e*r*e*ne | | |
| m*a*te | [aː] | [eɪ] |
| *ou*t | [uː] | [aʊ] |
| b*oo*t | [oː] | [uː] |
| b*oa*t | [ɔː] | [oʊ] |
| st*o*ne | | |

*Moon et al. 2013*

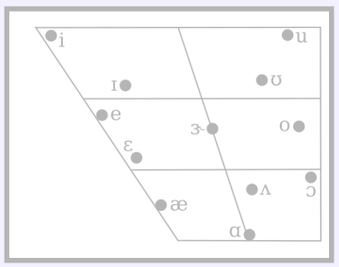
# Phonemes and Graphemes

- Words are composed of atomic units based on sound (for spoken languages)
- Sounds are a function of how we move our vocal tracts and mouth anatomy
- Languages have distinct sets of possible sounds (phonemic inventory)
- And “rules” governing which sound sequences are likely (syllable structure)

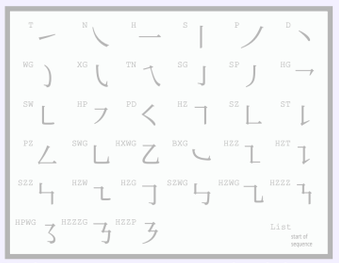
# Linguistic Units



Stokoe notation of ASL



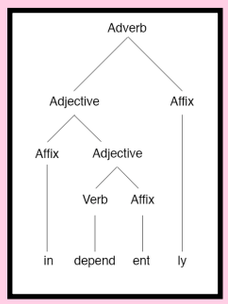

GA English vowels



Chinese stroke primitives, from Jiang et al. 2024

Phonemes/ Graphemes

ASL mouth morphemes Large (CHA) and Small (OO) learnhowtosign.com




Wikipedia (Annie Yang)

fǎn
反 — "anti-"
fǎn kǒng
- 反 恐 [反恐] — "anti-terror"

fǎn jiàoquánde
- 反 教 权 的 [反教權的] — "anti-clerical"

fǎn fàxīsī
- 反 法西斯 [反法西斯] — "anti-fascist"

Wikipedia

Morphemes /Lexemes

"language" in ASL lifeprint.com

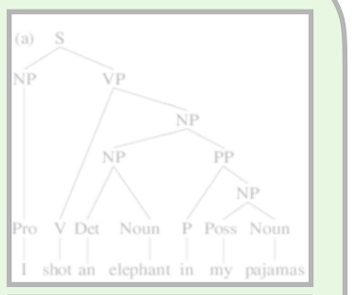
| | indicative | |
|---|---|---|
| | singular | plural |
| present | ich fege | wir **fegen** |
| | du fegst | ihr fegt |
| | er fegt | sie **fegen** |
| preterite | ich fegte | wir fegten |
| | du fegtest | ihr fegtet |
| | er fegte | sie fegten |
| imperative | feg (du)<br>fege (du) | fegt (ihr) |

conjugation of German "fegen" (to sweep) (Wiktionary)

***ajar*** = to teach
***ajari*** = to teach (imperative, locative)
***ajarilah*** = to teach (jussive, locative)
***ajarkan*** = to teach (imperative, causative/applicative)
***ajarkanlah*** = to teach (jussive, causative/applicative)
***ajarlah*** = to teach (jussive, active)
***ajaran*** = teachings
***belajar*** = to learn (intransitive, active)
***diajar*** = to be taught (intransitive)
***diajari*** = to be taught (transitive, locative)
***diajarkan*** = to be taught (transitive, causative/applicative)
***dipelajari*** = to be studied (locative)
***dipelajarkan*** = to be studied (causative/applicative)
***mempelajari*** = to study (locative)

Indonesian "ajar" (to teach) with affixes (Wikipedia)

Words

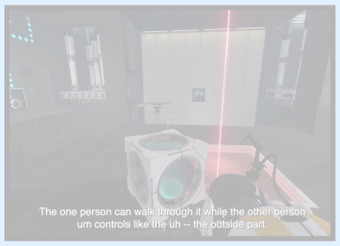



syntactic ambiguity, from UBC CPSC522

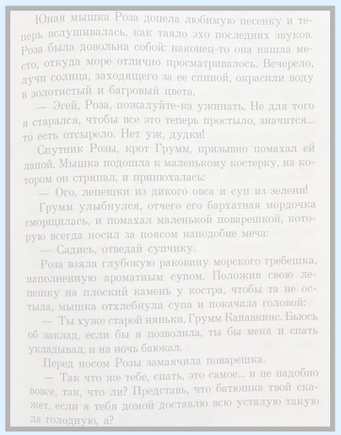

Universal Dependencies

Constituents /Phrases


靜靜養貓 (YouTube)




Portal 2

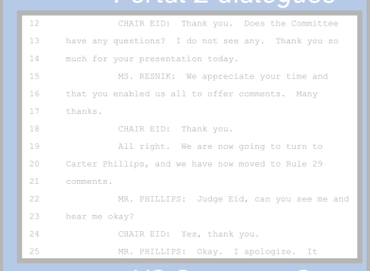


sentences in a Russian book (reddit)

Utterances/ Sentences


BBC News in Punjabi

Portal 2 dialogues

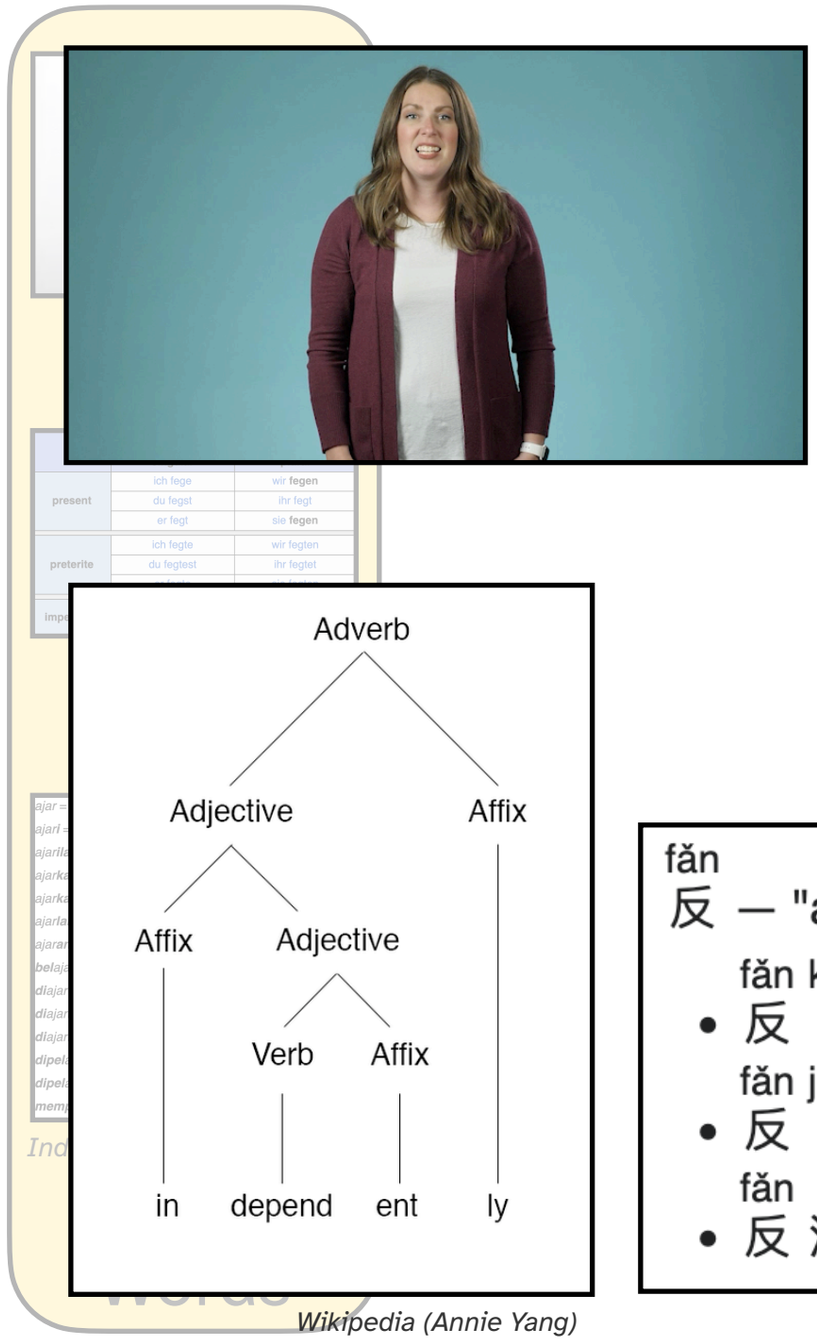



US Supreme Court

Discourse/ Dialogue

# Linguistic Units

Morphemes /Lexemes




*Wikipedia (Annie Yang)*

*ASL mouth morphemes*
*Large (CHA) and Small (OO)*
*learnhowtosign.com*

fǎn
反 — "anti-"

fǎn kǒng
- 反 恐 [反恐] — "anti-terror"

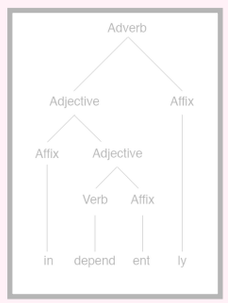
fǎn jiàoquánde
- 反 教 权 的 [反教權的] — "anti-clerical"

fǎn fàxīsī
- 反 法西斯 [反法西斯] — "anti-fascist"

*Wikipedia*

# Linguistic Units

*ASL mouth morphemes*
*Large (CHA) and Small (OO)*
*learnhowtosign.com*




*Wikipedia (Annie Yang)*

fǎn
反 — "anti-"
fǎn kǒng
- 反 恐 [反恐] — "anti-terror"

fǎn jiàoquánde
- 反 教 权 的 [反教權的] — "anti-clerical"

fǎn fàxīsī
- 反 法西斯 [反法西斯] — "anti-fascist"

*Wikipedia*

Morphemes
/Lexemes

Words

*“language” in ASL*
*lifeprint.com*

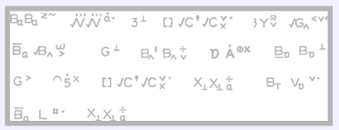
***ajar*** = to teach
***ajari*** = to teach (imperative, locative)
***ajarilah*** = to teach (jussive, locative)
***ajarkan*** = to teach (imperative, causative/applicative)
***ajarkanlah*** = to teach (jussive, causative/applicative)
***ajarlah*** = to teach (jussive, active)
***ajaran*** = teachings
***belajar*** = to learn (intransitive, active)
***diajar*** = to be taught (intransitive)
***diajari*** = to be taught (transitive, locative)
***diajarkan*** = to be taught (transitive, causative/applicative)
***dipelajari*** = to be studied (locative)
***dipelajarkan*** = to be studied (causative/applicative)
***mempelajari*** = to study (locative)

*Indonesian “ajar” (to teach)*
*with affixes*
*(Wikipedia)*

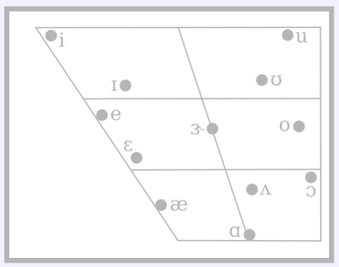
| | indicative | |
|---|---|---|
| | **singular** | **plural** |
| **present** | ich fege | wir **fegen** |
| | du fegst | ihr fegt |
| | er fegt | sie **fegen** |
| **preterite** | ich fegte | wir fegten |
| | du fegtest | ihr fegtet |
| | er fegte | sie fegten |
| **imperative** | feg (du)<br>fege (du) | fegt (ihr) |

*conjugation of German*
*“fegen” (to sweep)*
*(Wiktionary)*

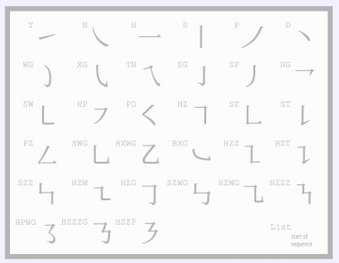
# Linguistic Units



Stokoe notation of ASL



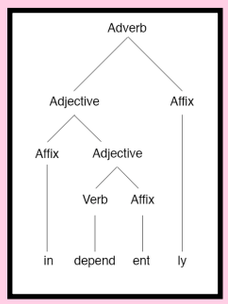

GA English vowels



Chinese stroke primitives, from Jiang et al. 2024

Phonemes/ Graphemes

ASL mouth morphemes Large (CHA) and Small (OO) learnhowtosign.com




Wikipedia (Annie Yang)

fǎn
反 — "anti-"
fǎn kǒng
- 反 恐 [反恐] — "anti-terror"

fǎn jiàoquánde
- 反 教 权 的 [反教權的] — "anti-clerical"

fǎn fàxīsī
- 反 法西斯 [反法西斯] — "anti-fascist"

Wikipedia

Morphemes /Lexemes

"language" in ASL lifeprint.com

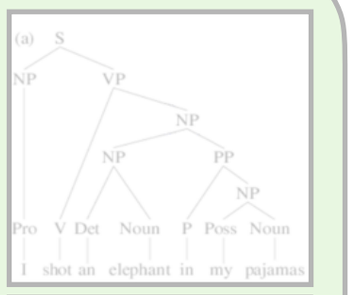
| | indicative | |
|---|---|---|
| | singular | plural |
| present | ich fege | wir **fegen** |
| | du fegst | ihr fegt |
| | er fegt | sie **fegen** |
| preterite | ich fegte | wir fegten |
| | du fegtest | ihr fegtet |
| | er fegte | sie fegten |
| imperative | feg (du)<br>fege (du) | fegt (ihr) |

conjugation of German "fegen" (to sweep) (Wiktionary)

***ajar*** = to teach
***ajari*** = to teach (imperative, locative)
***ajarilah*** = to teach (jussive, locative)
***ajarkan*** = to teach (imperative, causative/applicative)
***ajarkanlah*** = to teach (jussive, causative/applicative)
***ajarlah*** = to teach (jussive, active)
***ajaran*** = teachings
***belajar*** = to learn (intransitive, active)
***diajar*** = to be taught (intransitive)
***diajari*** = to be taught (transitive, locative)
***diajarkan*** = to be taught (transitive, causative/applicative)
***dipelajari*** = to be studied (locative)
***dipelajarkan*** = to be studied (causative/applicative)
***mempelajari*** = to study (locative)

Indonesian "ajar" (to teach) with affixes (Wikipedia)

Words

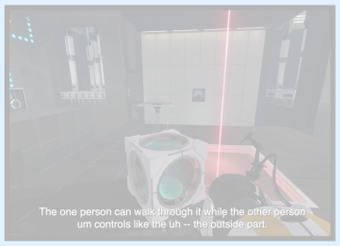



syntactic ambiguity, from UBC CPSC522

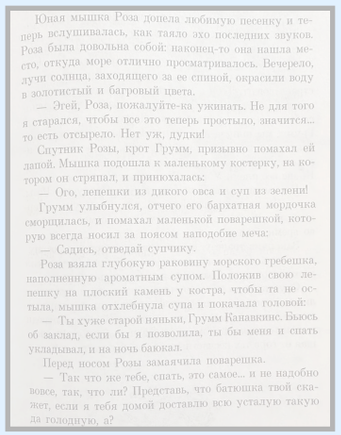

Universal Dependencies

Constituents /Phrases


靜靜養貓 (YouTube)




Portal 2

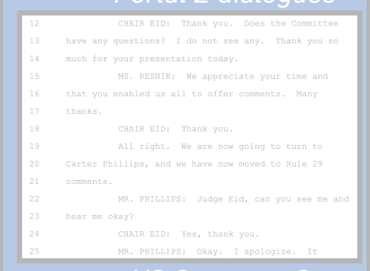


sentences in a Russian book (reddit)

Utterances/ Sentences


BBC News in Punjabi

Portal 2 dialogues




US Supreme Court

Discourse/ Dialogue

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Text data can be viewed as a sequence of words
- First step in building a language technology: building a function that maps from arbitrary text data to that sequence

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Type-token distinction:
  - Type: a unique word in a text corpus
  - Token: an *instance* of a word type, appearing in a particular context

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

instances (tokens) of wordtype `'.'`

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

instances (tokens) of wordtype `','`

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

instances (tokens) of wordtype `'Every'`

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

instances (tokens) of wordtype `'and'`

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
{'.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young'}
```

wordtypes (vocabulary)

instances (tokens) of wordtype `'language'`

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

tokenized text

# Word Types and Tokens

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

```
vocabulary =
['.', ',', ':', 'Every', 'The', 'a', 'adults', 'and', 'are', 'basic',
'can', 'children', 'conversation', 'dialogue', 'even', 'far', 'form',
'from', 'hold', 'illiterate', 'including', 'is', 'language', 'listening',
'most', 'natural', 'of', 'preparing', 'reading', 'skills', 'speeches',
'to', 'universal', 'use', 'user', 'whereas', 'writing', 'young']
```

vocabulary as a look-up table

```
tokenized_text = ['The', 'most', ... , 'skills, '.']
tokenized_indices = [vocabulary.index(token) for token in tokenized_text]

tokenized_indices: [4, 24, 25, 7, 9, 16, 26, 22, 33, 21, 13, 2, 3, 22,
34, 1, 20, 37, 11, 7, 19, 6, 1, 10, 18, 5, 12, 1, 35, 28, 1, 36, 1, 27,
30, 7, 14, 23, 31, 30, 8, 15, 17, 32, 29, 0]
```

text as a sequence of wordtype indices

# Tokenize on Spaces

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004,*
*"A mechanistic psychology of dialogue"*

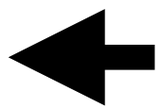
- Simplest tokenizer (for English): splitting on spaces

```
tokenized = s.split(' ')
```

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue:', 'Every', 'language', 'user,', 'including',
'young', 'children', 'and', 'illiterate', 'adults,', 'can', 'hold', 'a',
'conversation,', 'whereas', 'reading,', 'writing,', 'preparing',
'speeches', 'and', 'even', 'listening', 'to', 'speeches', 'are', 'far',
'from', 'universal', 'skills.']
```

- But this gets us some weird wordtypes:

```
'dialogue:'
'user,'
'skills.'
```



Not really words different from *dialogue*, *user*, *skills*

# Rule-Based Tokenization

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- nltk tokenizers, with special rules for punctuation

```
import nltk
tokenized = nltk.word_tokenize(s)
```

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

- But this still loses similarity between wordtypes

```
'skills'
'reading'
'speeches'
```

Lexically similar to, but morphologically distinct from *skill, read, speech*

# Rule-Based Tokenization

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- nltk tokenizers, with special rules for punctuation

```
import nltk
tokenized = nltk.word_tokenize(s)
```

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

- And doesn't work for all languages

**โดยที่การยอมรับนับถือเกียรติศักดิ์ประจำตัว และสิทธิเท่าเทียมกันและโอนมิได้ของบรรดา สมาชิก ทั้ง หลายแห่งครอบครัว มนุษย์เป็นหลักมูลเหตุแห่งอิสรภาพ ความยุติธรรม และสันติภาพในโลก**

*Universal Declaration of Human Rights in Thai*

# Rule-Based Tokenization

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- nltk tokenizers, with special rules for punctuation

```
import nltk
tokenized = nltk.word_tokenize(s)
```

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

- Once you've "trained" your tokenizer, you're stuck with it

```
vocab = ['.', ',', ':', 'Every', ... , 'writing', 'young']
vocab.index('ChatGPT') → not found!
```

# Rule-Based Tokenization

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- nltk tokenizers, with special rules for punctuation

```
import nltk
tokenized = nltk.word_tokenize(s)
```

```
['The', 'most', 'natural', 'and', 'basic', 'form', 'of', 'language',
'use', 'is', 'dialogue', ':', 'Every', 'language', 'user', ',',
'including', 'young', 'children', 'and', 'illiterate', 'adults', ',',
'can', 'hold', 'a', 'conversation', ',', 'whereas', 'reading', ',',
'writing', ',', 'preparing', 'speeches', 'and', 'even', 'listening',
'to', 'speeches', 'are', 'far', 'from', 'universal', 'skills, '.']
```

- Once you've "trained" your tokenizer, you're stuck with it

```
vocab = ['.', ',', ':', 'Every', ... , 'writing', 'young', '<UNK>']
tokenized_indices =
    [vocabulary.index(token) for token in tokenized_text
     if token in vocabulary
     else vocabulary.index('<UNK>')]
```

**add a special token for unknown words**

# Bytes as Wordtypes

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

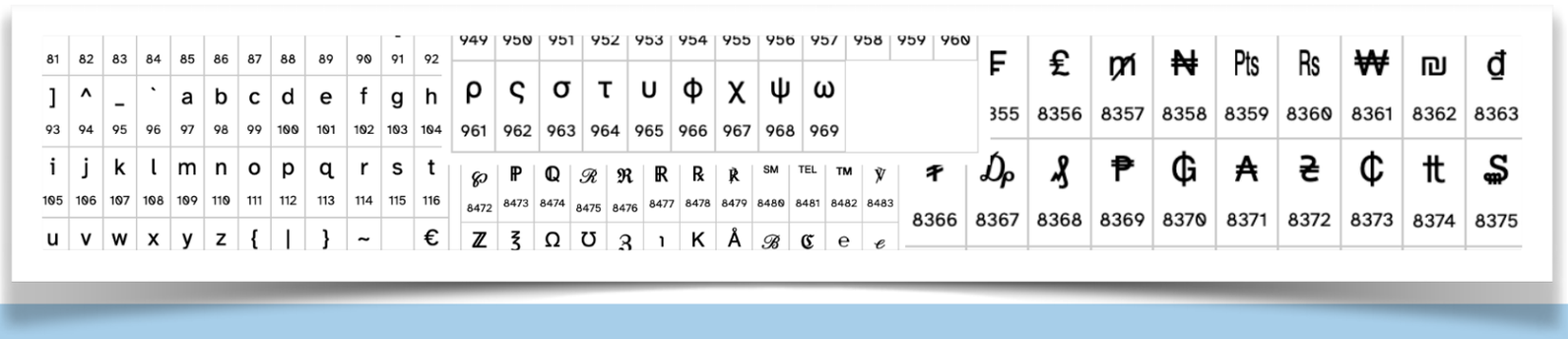
*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Strings are sequences of characters (bytes)!

```
tokenized = s.encode()
```

```
54 68 65 20 6d 6f 73 74 20 6e 61 74 75 72 61 6c 20 61 6e 64 20 62 61 73 69 63 20 66 6f 72 6d 20 6f 66 20
6c 61 6e 67 75 61 67 65 20 75 73 65 20 69 73 20 64 69 61 6c 6f 67 75 65 3a 20 45 76 65 72 79 20 6c 61 6e
67 75 61 67 65 20 75 73 65 72 2c 20 69 6e 63 6c 75 64 69 6e 67 20 79 6f 75 6e 67 20 63 68 69 6c 64 72 65
6e 20 61 6e 64 20 69 6c 6c 69 74 65 72 61 74 65 20 61 64 75 6c 74 73 2c 20 63 61 6e 20 68 6f 6c 64 20 61
20 63 6f 6e 76 65 72 73 61 74 69 6f 6e 2c 20 77 68 65 72 65 61 73 20 72 65 61 64 69 6e 67 2c 20 77 72 69
74 69 6e 67 2c 20 70 72 65 70 61 72 69 6e 67 20 73 70 65 65 63 68 65 73 20 61 6e 64 20 65 76 65 6e 20 6c
69 73 74 65 6e 69 6e 67 20 74 6f 20 73 70 65 65 63 68 65 73 20 61 72 65 20 66 61 72 20 66 72 6f 6d 20 75
6e 69 76 65 72 73 61 6c 20 73 6b 69 6c 6c 73 2e
```

- Now our vocabulary is a fixed size (all possible Unicode characters)



# Bytes as Wordtypes

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

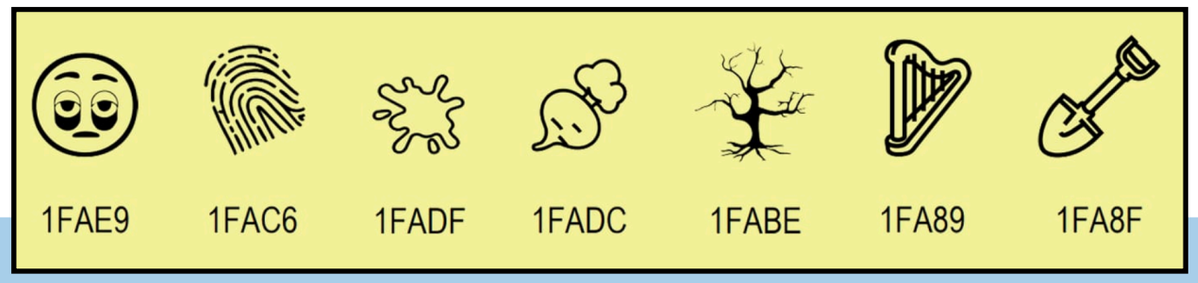
*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Strings are sequences of characters (bytes)!

```
tokenized = s.encode()
```

```
54 68 65 20 6d 6f 73 74 20 6e 61 74 75 72 61 6c 20 61 6e 64 20 62 61 73 69 63 20 66 6f 72 6d 20 6f 66 20
6c 61 6e 67 75 61 67 65 20 75 73 65 20 69 73 20 64 69 61 6c 6f 67 75 65 3a 20 45 76 65 72 79 20 6c 61 6e
67 75 61 67 65 20 75 73 65 72 2c 20 69 6e 63 6c 75 64 69 6e 67 20 79 6f 75 6e 67 20 63 68 69 6c 64 72 65
6e 20 61 6e 64 20 69 6c 6c 69 74 65 72 61 74 65 20 61 64 75 6c 74 73 2c 20 63 61 6e 20 68 6f 6c 64 20 61
20 63 6f 6e 76 65 72 73 61 74 69 6f 6e 2c 20 77 68 65 72 65 61 73 20 72 65 61 64 69 6e 67 2c 20 77 72 69
74 69 6e 67 2c 20 70 72 65 70 61 72 69 6e 67 20 73 70 65 65 63 68 65 73 20 61 6e 64 20 65 76 65 6e 20 6c
69 73 74 65 6e 69 6e 67 20 74 6f 20 73 70 65 65 63 68 65 73 20 61 72 65 20 66 61 72 20 66 72 6f 6d 20 75
6e 69 76 65 72 73 61 6c 20 73 6b 69 6c 6c 73 2e
```

- And we don't have any unknown wordtypes (until the Unicode Consortium adds new emoji)



# Bytes as Wordtypes

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Strings are sequences of characters (bytes)!

```
tokenized = s.encode()
```

```
54 68 65 20 6d 6f 73 74 20 6e 61 74 75 72 61 6c 20 61 6e 64 20 62 61 73 69 63 20 66 6f 72 6d 20 6f 66 20
6c 61 6e 67 75 61 67 65 20 75 73 65 20 69 73 20 64 69 61 6c 6f 67 75 65 3a 20 45 76 65 72 79 20 6c 61 6e
67 75 61 67 65 20 75 73 65 72 2c 20 69 6e 63 6c 75 64 69 6e 67 20 79 6f 75 6e 67 20 63 68 69 6c 64 72 65
6e 20 61 6e 64 20 69 6c 6c 69 74 65 72 61 74 65 20 61 64 75 6c 74 73 2c 20 63 61 6e 20 68 6f 6c 64 20 61
20 63 6f 6e 76 65 72 73 61 74 69 6f 6e 2c 20 77 68 65 72 65 61 73 20 72 65 61 64 69 6e 67 2c 20 77 72 69
74 69 6e 67 2c 20 70 72 65 70 61 72 69 6e 67 20 73 70 65 65 63 68 65 73 20 61 6e 64 20 65 76 65 6e 20 6c
69 73 74 65 6e 69 6e 67 20 74 6f 20 73 70 65 65 63 68 65 73 20 61 72 65 20 66 61 72 20 66 72 6f 6d 20 75
6e 69 76 65 72 73 61 6c 20 73 6b 69 6c 6c 73 2e
```

- But: individual characters are not meaningful

```
['.', ',', ':', 'Every', ... , 'user', 'whereas', 'writing', 'young']
```

**vs.**

```
['.', ',', ':', 'A', 'B', 'C', ... , 'a', 'b', 'c', ... , '7', '8', '9']
```

# Bytes as Wordtypes

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Strings are sequences of characters (bytes)!

```
tokenized = s.encode()
```

```
54 68 65 20 6d 6f 73 74 20 6e 61 74 75 72 61 6c 20 61 6e 64 20 62 61 73 69 63 20 66 6f 72 6d 20 6f 66 20
6c 61 6e 67 75 61 67 65 20 75 73 65 20 69 73 20 64 69 61 6c 6f 67 75 65 3a 20 45 76 65 72 79 20 6c 61 6e
67 75 61 67 65 20 75 73 65 72 2c 20 69 6e 63 6c 75 64 69 6e 67 20 79 6f 75 6e 67 20 63 68 69 6c 64 72 65
6e 20 61 6e 64 20 69 6c 6c 69 74 65 72 61 74 65 20 61 64 75 6c 74 73 2c 20 63 61 6e 20 68 6f 6c 64 20 61
20 63 6f 6e 76 65 72 73 61 74 69 6f 6e 2c 20 77 68 65 72 65 61 73 20 72 65 61 64 69 6e 67 2c 20 77 72 69
74 69 6e 67 2c 20 70 72 65 70 61 72 69 6e 67 20 73 70 65 65 63 68 65 73 20 61 6e 64 20 65 76 65 6e 20 6c
69 73 74 65 6e 69 6e 67 20 74 6f 20 73 70 65 65 63 68 65 73 20 61 72 65 20 66 61 72 20 66 72 6f 6d 20 75
6e 69 76 65 72 73 61 6c 20 73 6b 69 6c 6c 73 2e
```

- It's now the ML model's job to learn to compose words from scratch

# Bytes as Wordtypes

*The most natural and basic form of language use is dialogue: Every language user, including young children and illiterate adults, can hold a conversation, whereas reading, writing, preparing speeches and even listening to speeches are far from universal skills.*

*from Pickering and Garrod 2004, "A mechanistic psychology of dialogue"*

- Strings are sequences of characters (bytes)!

```
tokenized = s.encode()
```

```
54 68 65 20 6d 6f 73 74 20 6e 61 74 75 72 61 6c 20 61 6e 64 20 62 61 73 69 63 20 66 6f 72 6d 20 6f 66 20
6c 61 6e 67 75 61 67 65 20 75 73 65 20 69 73 20 64 69 61 6c 6f 67 75 65 3a 20 45 76 65 72 79 20 6c 61 6e
67 75 61 67 65 20 75 73 65 72 2c 20 69 6e 63 6c 75 64 69 6e 67 20 79 6f 75 6e 67 20 63 68 69 6c 64 72 65
6e 20 61 6e 64 20 69 6c 6c 69 74 65 72 61 74 65 20 61 64 75 6c 74 73 2c 20 63 61 6e 20 68 6f 6c 64 20 61
20 63 6f 6e 76 65 72 73 61 74 69 6f 6e 2c 20 77 68 65 72 65 61 73 20 72 65 61 64 69 6e 67 2c 20 77 72 69
74 69 6e 67 2c 20 70 72 65 70 61 72 69 6e 67 20 73 70 65 65 63 68 65 73 20 61 6e 64 20 65 76 65 6e 20 6c
69 73 74 65 6e 69 6e 67 20 74 6f 20 73 70 65 65 63 68 65 73 20 61 72 65 20 66 61 72 20 66 72 6f 6d 20 75
6e 69 76 65 72 73 61 6c 20 73 6b 69 6c 6c 73 2e
```

- But: input sequences are much longer

```
'language'
```

**vs.**

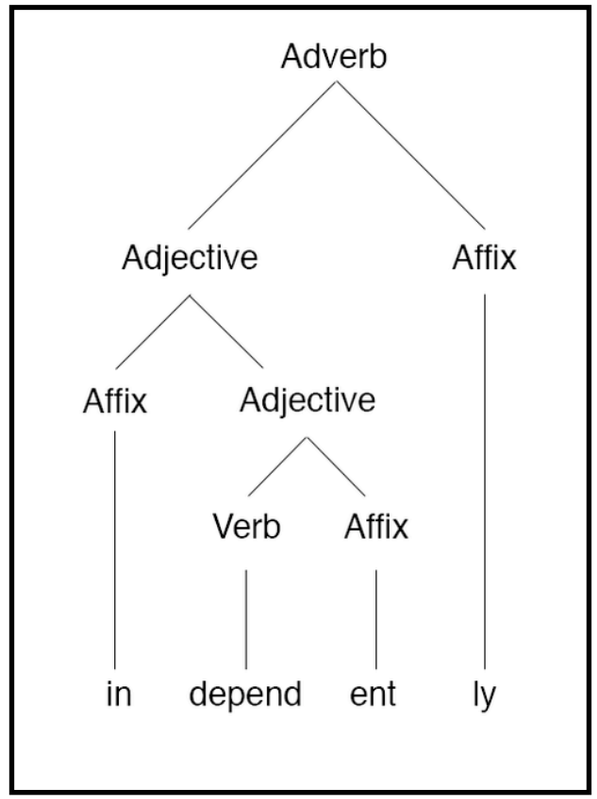
```
'l', 'a', 'n', 'g', 'u', 'a', 'g', 'e'
```

# A Compromise: Subword Tokenization

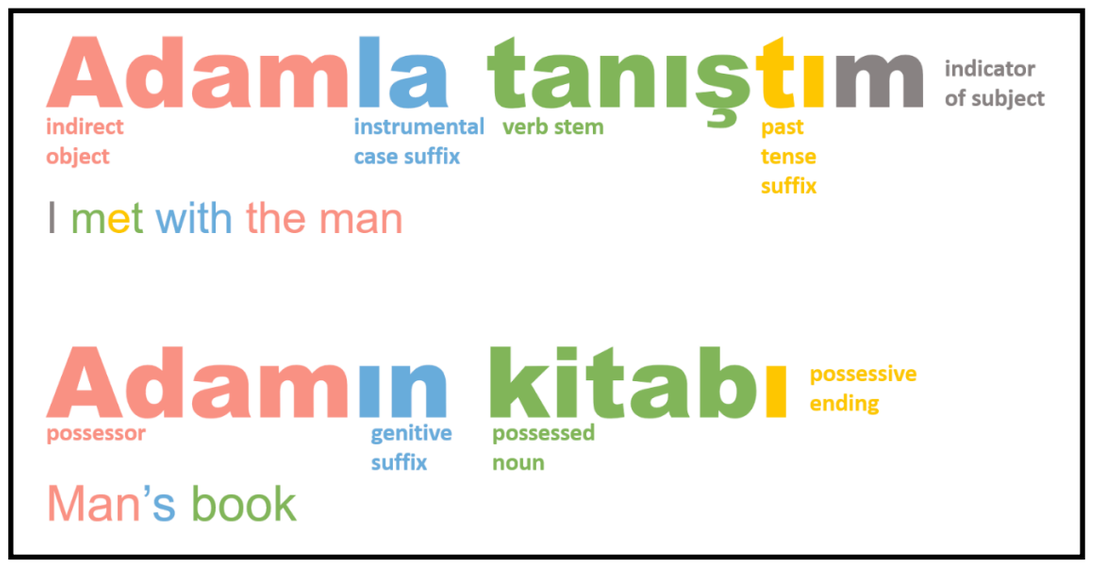
- Main principle: words are (often) composed of subparts (morphemes)

- Our vocabulary should have entries for frequent words kept whole, because we have a lot of data about those words

- But it should also have entries for parts of less-frequent words, so our ML models can learn how to compose parts of words into whole words (especially unfamiliar words!)




*Wikipedia (Annie Yang)*



# Byte Pair Encoding

- Modern standard for building a tokenizer
- Inputs: collection of texts and target vocabulary size
- Initial vocabulary is the set of all bytes (characters) across the texts
- Until the target vocabulary size is reached, repeat the following:
  - Tokenize all of the texts using the current vocabulary
  - Find the most common bigram in the tokenized texts, then add it to the vocabulary as a new wordtype

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')`

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')`

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

```
'h' 'u'    15
```

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` →

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

```
'h' 'u'    15
'u' 'g'    20
```

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')`

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

```
'h' 'u'    15
'u' 'g'    20
'p' 'u'    17
```

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')`

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

```
'h' 'u'    15
'u' 'g'    20
'p' 'u'    17
'u' 'n'    16
```

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` →

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

| | | |
|---|---|---|
| 'h' | 'u' | 15 |
| 'u' | 'g' | 20 |
| 'p' | 'u' | 17 |
| 'u' | 'n' | 16 |
| 'b' | 'u' | 4 |

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')`

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

```
'h' 'u'    15
'u' 'g'    20
'p' 'u'    17
'u' 'n'    16
'b' 'u'     4
'g' 's'     5
```

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` →

*tokenized texts*

`('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

*bigrams + frequencies*

| | |
|---|---|
| 'h' 'u' | 15 |
| **'u' 'g'** | **20** |
| 'p' 'u' | 17 |
| 'u' 'n' | 16 |
| 'b' 'u' | 4 |
| 'g' 's' | 5 |

most frequent bigram; add to vocabulary

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary* *tokenized texts*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` → `('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'ug' 's', 5)`

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary* *tokenized texts*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` → `('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'ug' 's', 5)`

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary* *tokenized texts*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` → `('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'ug' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug', 'un')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'un', 12), ('b' 'un', 4), ('h' 'ug' 's', 5)`

*Example from HuggingFace*

# Byte Pair Encoding

**Documents + frequencies:** `('hug', 10), ('pug', 5), ('pun', 12), ('bun', 4), ('hugs', 5)`

*vocabulary* *tokenized texts*

`('h', 'u', 'g', 'p', 'n', 'b', 's')` → `('h' 'u' 'g', 10), ('p' 'u' 'g', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'u' 'g' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'u' 'n', 12), ('b' 'u' 'n', 4), ('h' 'ug' 's', 5)`

`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug', 'un')` → `('h' 'ug', 10), ('p' 'ug', 5), ('p' 'un', 12), ('b' 'un', 4), ('h' 'ug' 's', 5)`

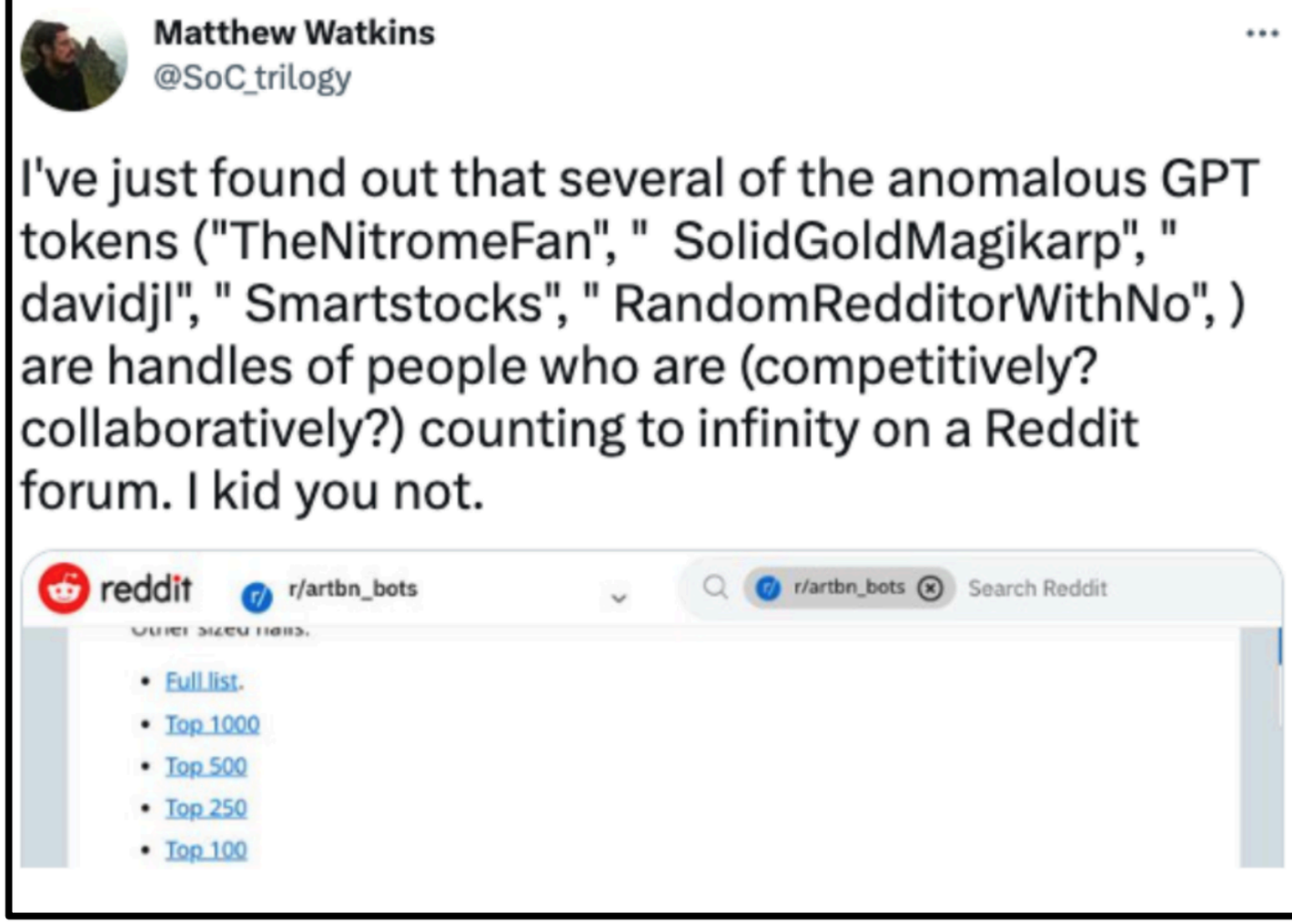
`('h', 'u', 'g', 'p', 'n', 'b', 's', 'ug', 'un', 'hug')` → `('hug', 10), ('p' 'ug', 5), ('p' 'un', 12), ('b' 'un', 4), ('hug' 's', 5)`

**New word: “puns”**

`'p' 'u' 'n' 's'` → `'p' 'un' 's'` ✓

*Example from HuggingFace*

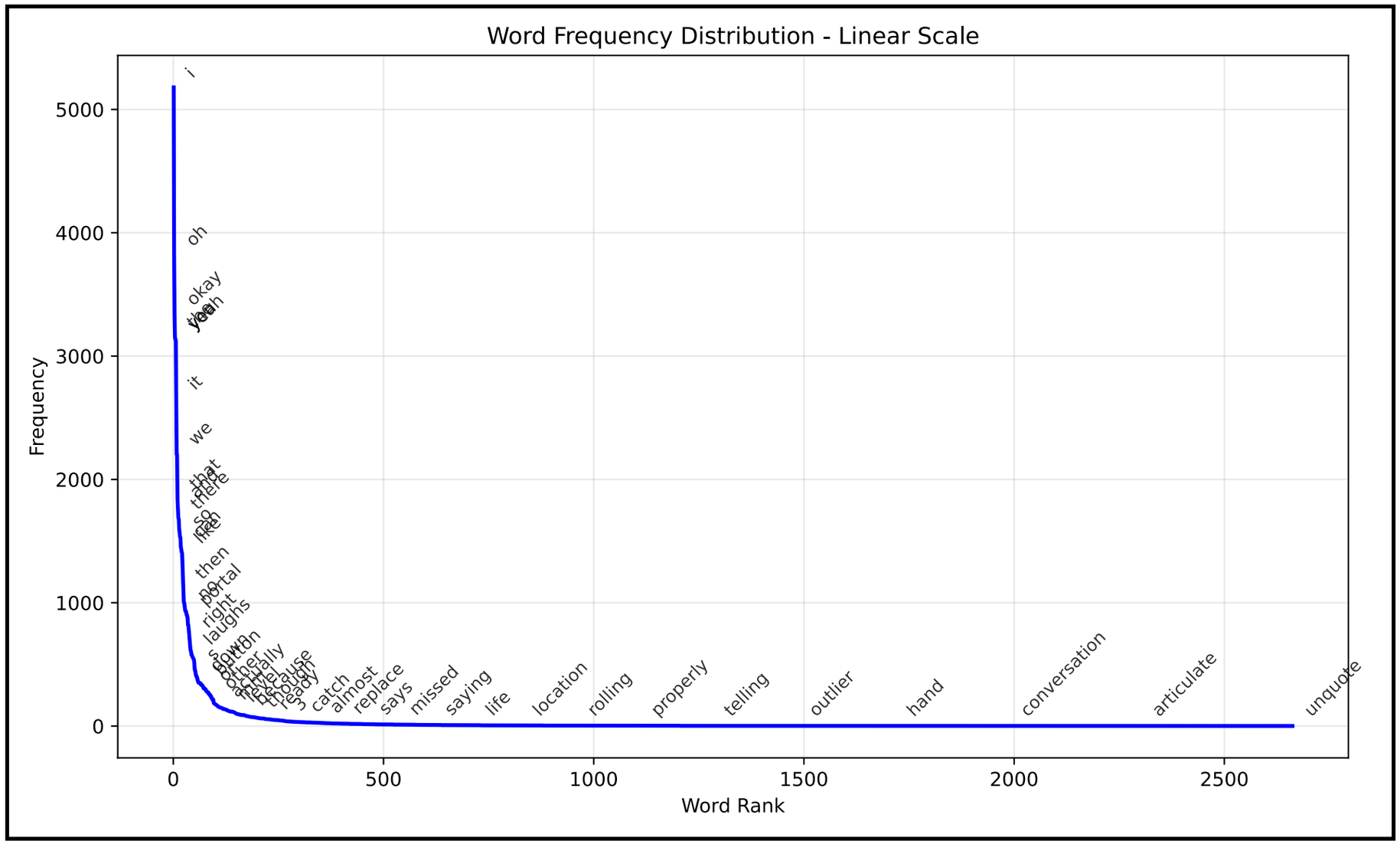
# Vocabularies




| Rank | User | Counts |
|---|---|---|
| 1 | /u/davidjl123 | 163477 |
| 2 | /u/Smartstocks | 113829 |
| 3 | /u/atomicimploder | 103178 |
| 4 | /u/TheNitromeFan | 84581 |
| 5 | /u/SolidGoldMagikarp | 65753 |
| 6 | /u/RandomRedditorWithNo | 63434 |
| 7 | /u/rideride | 59024 |
| 8 | /u/Mooraell | 57785 |
| 9 | /u/Removedpixel | 55080 |
| 10 | /u/Adinida | 48415 |
| 11 | /u/rschaosid | 47132 |

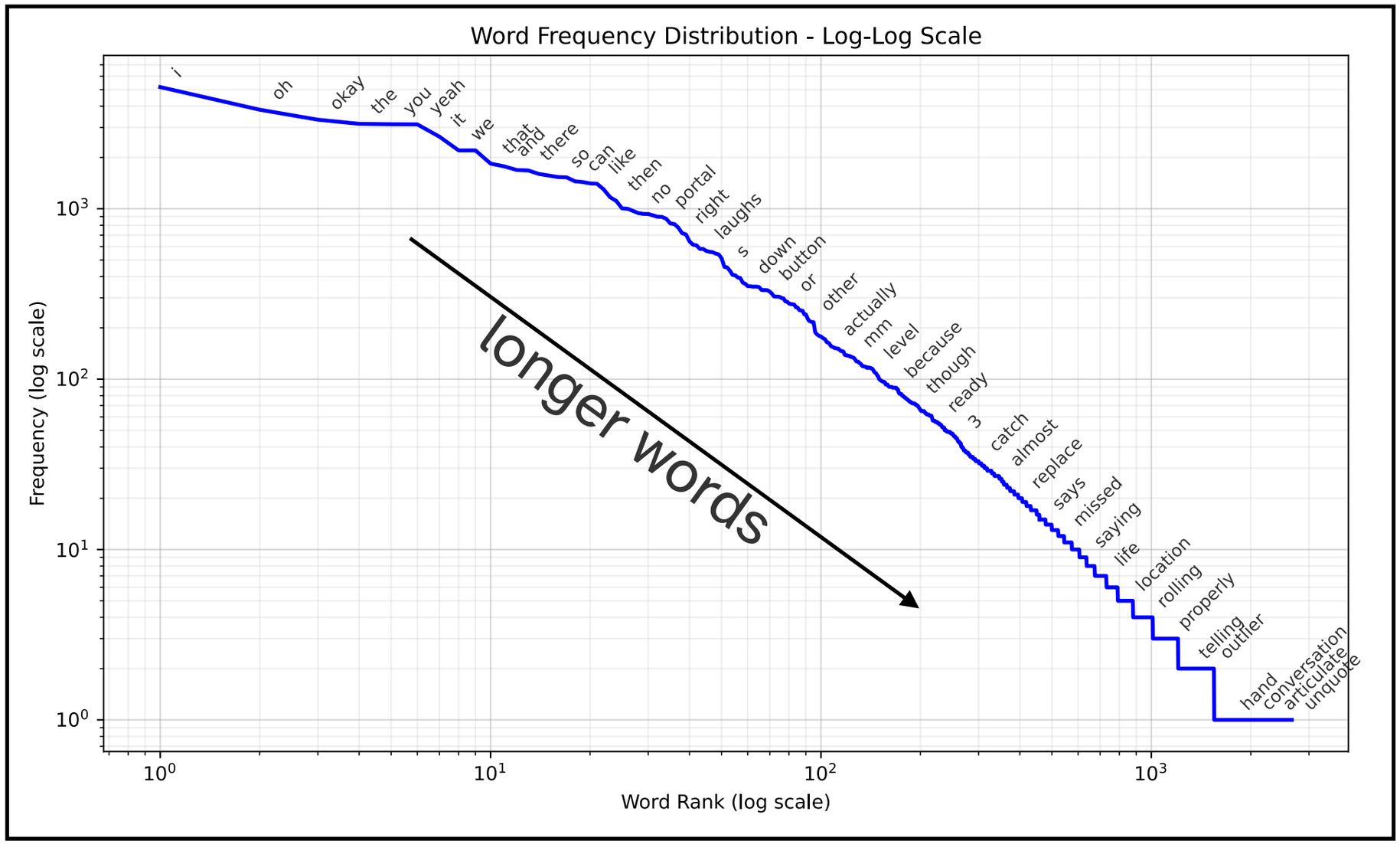
# Vocabularies

## **Long-tail (“Zipfian”) distribution of wordtypes**
(from Portal 2 dialogues)



# Vocabularies

## **Long-tail (“Zipfian”) distribution of wordtypes**
(from Portal 2 dialogues)



# Meaning and Representation

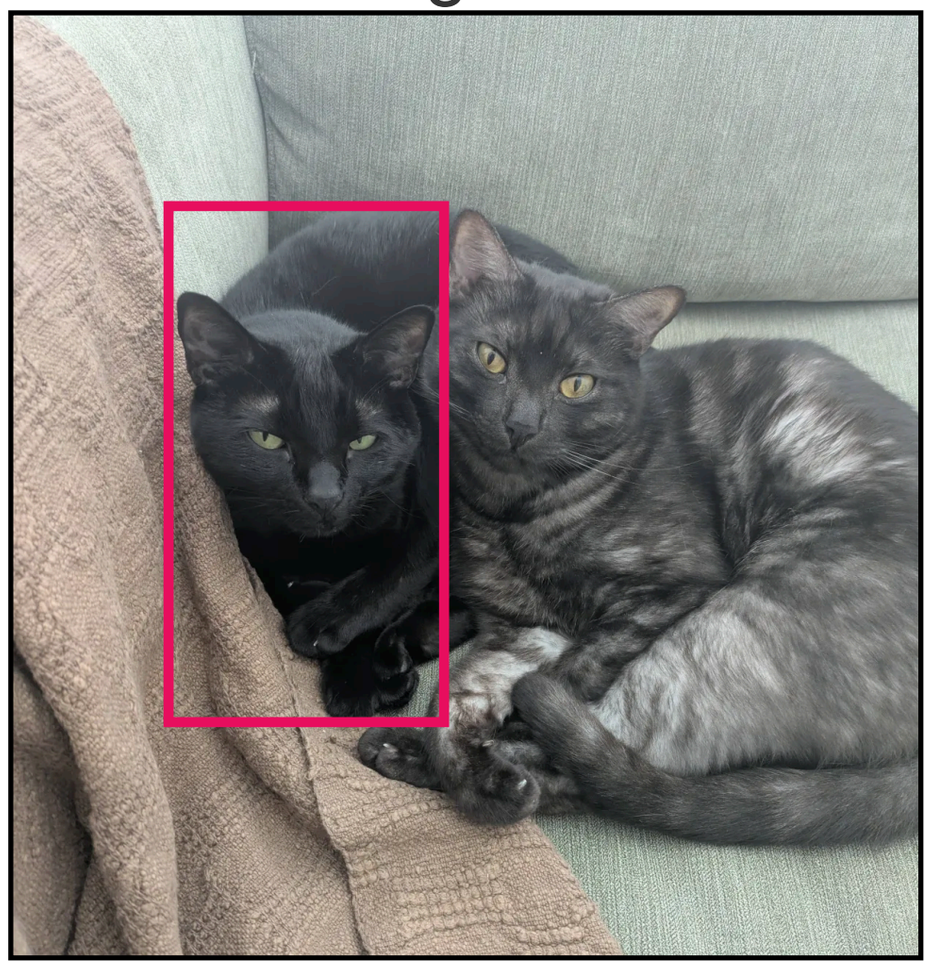
- Words are themselves not actual things, they only refer to things
  - Kind of like a pointer
  - Even if they aren’t actual things, we can still do things to and know things about them
  - E.g., we know cats are a type of pet
- Core question: if a machine learning system is processing text, how should words be represented as input to and within the ML system?

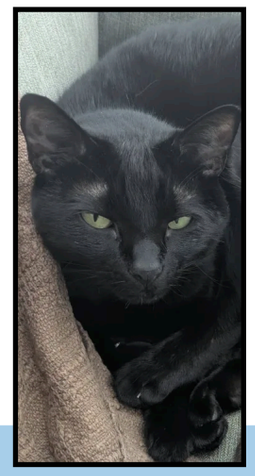
# Word Meaning: Denotational Semantics

What color is Pepper?

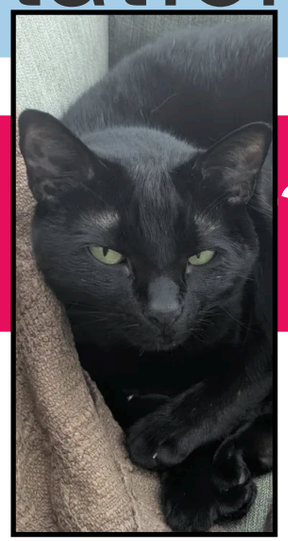
- Let's start from the bottom up: what does each word mean?
  - What does "Pepper" mean?
- **Denotational semantics:** the symbol *refers to* something in the context in which the language is used

Image $i$



$$[\![\text{Pepper}]\!]^i = \{$$  $$\}$$

# Word Meaning: Denotational Semantics

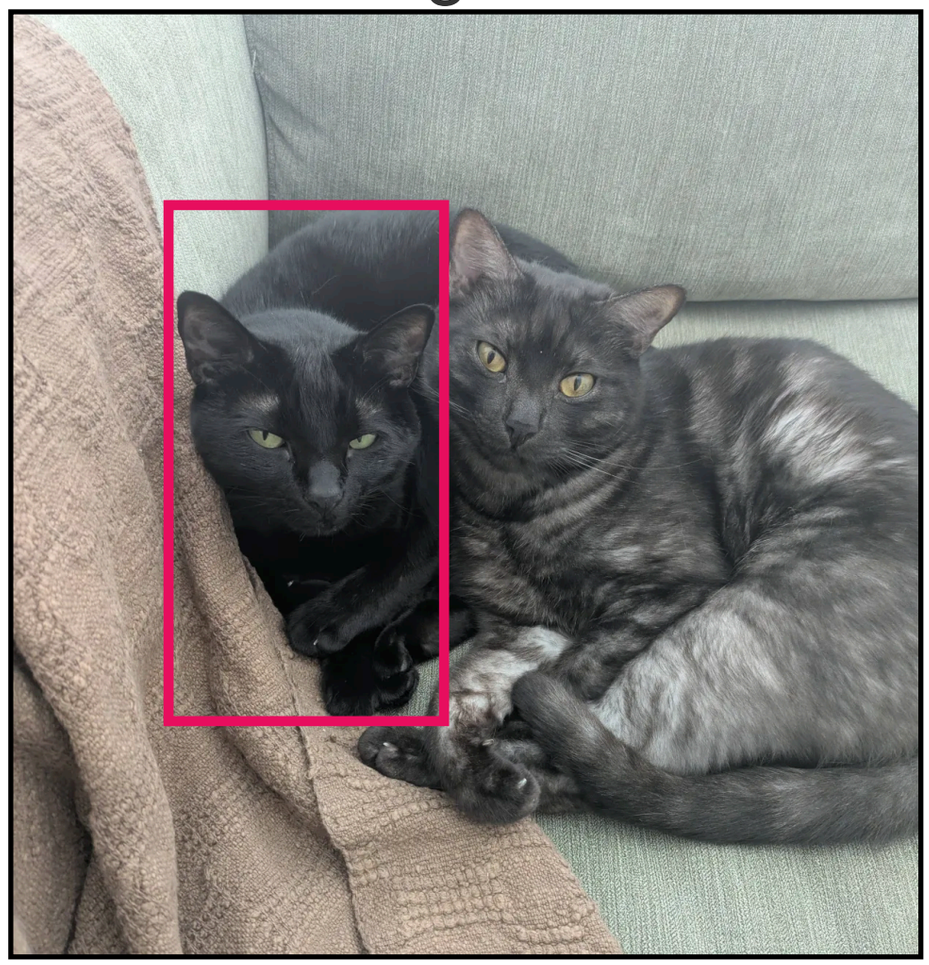


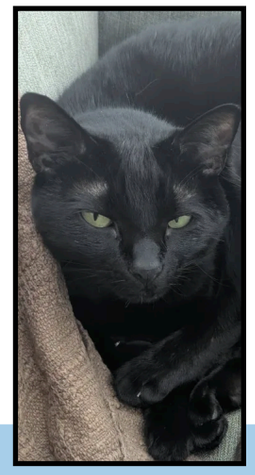
What color is ?

Black with green eyes

- Denotational semantics allows us to *ground* language to a world outside of language use
- But what happens when we don't have an available outside world?

Image $i$



$$[\![\text{Pepper}]\!]^i = \{$$  $$\}$$

# Word Meaning: Ontologies

## Cats

### Lucy's Cats

Toast

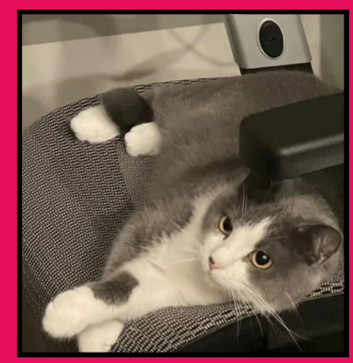
### Nick's Cats

Coco

### Téa's Cats

Marceline

### Zineng's Cats



Shaomai

### Alane's Cats

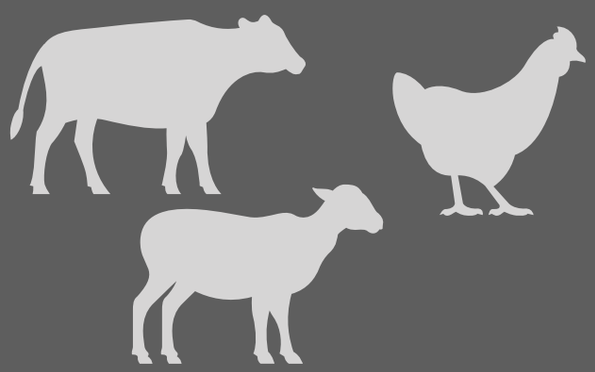
Patrick

Pepper

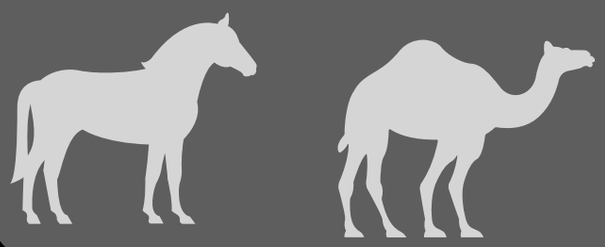
# Word Meaning: Ontologies

## Domesticated Animals

### Livestock



### Draught Animals

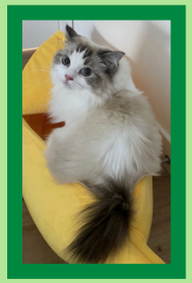


### Pets

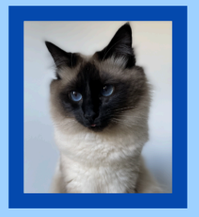
#### Cats

Alane's Cats

Lucy's Cats

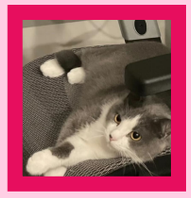


Nick's Cats



Téa's Cats

Zineng's Cats



#### Dogs

Mer's Dogs

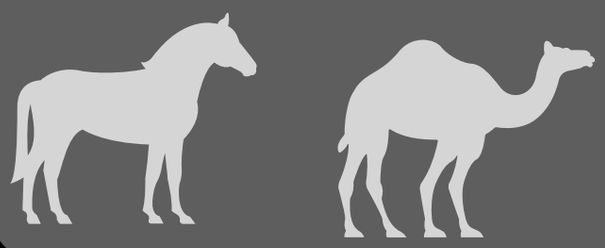
# Word Meaning: Ontologies

## Domesticated Animals

**Hyponym / hypernym relation:**

“pet” is a hypernym of “dog”;
“draught animal” is a hyponym of “domesticated animals”

Draught Animals

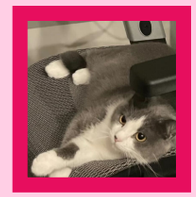


Pets

Dogs

Téa’s Cats

Zineng’s Cats



Mer’s Dogs

**Synsets:**

{“draught animal”, “beast of burden”, “draft animal”}

# Word Meaning: Ontologies

What color is Pepper?

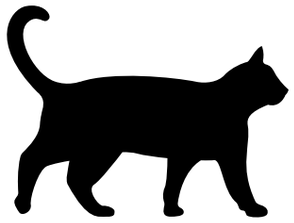
- From our ontology, we know
  - **`isa(Pepper, Alane's cat)`**
  - **`isa(Alane's cat, cat)`**

← (hypernym relation)

- We might know something about cats, too:
  - Cats are typically not blue, pink, yellow, etc.
  - So the answer is probably not one of these

# Word Meaning: Ontologies

What color is pepper?




- Missing context-dependence (polysemy)

# Word Meaning: Ontologies

- Missing context-dependence (polysemy)
- Missing meaning of new words

In cryptography, a **pepper** is a secret added to an input such as a password during hashing with a cryptographic hash function. This value differs from a salt in that it is not stored alongside a password hash, but rather the pepper is kept separate in some other medium, such as a Hardware Security Module.[1] Note that the National Institute of Standards and Technology refers to this value as a **secret key** rather than a **pepper**. A pepper is similar in concept to a salt or an encryption key. It is like a salt in that it is a randomized value that is added to a password hash, and it is similar to an encryption key in that it should be kept secret.

Webster, Craig (2009-08-03). "Securing Passwords with Salt, Pepper and Rainbows". *Barking Iguana*. Retrieved 2020-11-11.

*https://en.wikipedia.org/wiki/Pepper_(cryptography)*

# Word Meaning: Ontologies

*Paullassiter, Wikipedia*

*Falcodigiada, Wikipedia*

*Hankwang, Wikipedia*

*Daniel Risacher, Wikipedia*

*Endwints, Wikipedia*

- Missing context-dependence (polysemy)
- Missing meaning of new words
- Requires human labor

There are perhaps fifty thousand *Capsicum* cultivars grown worldwide.[1]

- *Capsicum annuum*, which includes bell peppers, cayennes, friggitello, jalapeños, paprika, and serrano.
  - *Capsicum annuum* '*New Mexico Group*', common name Hatch or Anaheim, which includes Big Jim, Chimayó, and Sandia peppers.
- *Capsicum baccatum*, which includes the South American varieties, such as ají amarillo, ají limón, and criolla sella.
- *Capsicum chinense*, which includes all of the habaneros,[5] Scotch bonnets, Trinidad Scorpions, the Bhut Jolokia, and the Carolina Reaper.
- *Capsicum frutescens*, which includes the Tabasco pepper and many of the peppers grown in India;[6] sometimes not distinguished as a species separate from *C. annuum*.[7][8]
- *Capsicum pubescens*, which includes the rocoto and manzano pepper, are distinctive plants, having violet flowers, black seeds, and hairy dark green leaves, and grow as a large, multi-stemmed vine up to 5 meters long.

*https://en.wikipedia.org/wiki/List_of_Capsicum_cultivars*

# Word Meaning: Ontologies

- Missing context-dependence (polysemy)
- Missing meaning of new words
- Requires human labor
- Subjective and culturally dependent





*Image credits: Ragesoss; Kham Tran; Takeaway; Matt K; Lemmikkipuu; Soerfm*

# Word Meaning: Discrete Symbols

- Machine learning models expect numerical inputs

- Represent each word as a unique **one-hot vector** (vector with values 0, except for one value of 1)

$\phi$(cat) = [ 0 0 0 0 1 0 0 0 0 ]
$\phi$(kitten) = [ 0 0 1 0 0 0 0 0 0 ]

- Vector dimension: the same size as the vocabulary

- Problem: no notion of similarity

How can I make my house safe for a new kitten?

*How can I make my house safe for a new cat?*

- (Is there some way we could use WordNet to get more informative numerical representations of words?)

# Word Meaning: Continuous Vectors

- Instead: represent words as continuous vectors

$\phi(\text{cat}) = [\ -0.6\ \ 1.3\ \ -0.1\ \ 0.3\ ]$

$\phi(\text{kitten}) = [\ -0.7\ \ -1.5\ \ 0.0\ \ 0.3\ ]$

- Implicitly provides notions of similarity:

$$||\phi(\text{cat}), \phi(\text{kitten})|| < ||\phi(\text{cat}), \phi(\text{dog})||$$

- Similarity is learnable from text at scale:

*… feeding time in multiple cat households can often be …*
*… is to relieve the cat from the stress of …*
*… effect on the feral cat population in rural areas …*

*… So I wrapped the kitten up in a towel …*
*… they noticed that the kitten population was getting high …*
*… were looking for a kitten , hoping that someone …*

# Word Embeddings

- How do we get these vectors?
- Core principle: **distributional hypothesis**
  - Words that are used in similar contexts have similar meanings
  - Context: typically, other words in a text, but really anything can be context!

# Distributional Hypothesis

A bottle of tesgüino is on the table.

Everybody likes tesgüino.

Tesgüino makes you drunk.

We make tesgüino out of corn.

- Found in bottles
- People like it
- It makes you drunk
- It’s made of corn
- **Maybe:** similar to beer, wine, whiskey, etc.

# Word Co-Occurrence

- Co-occurrence matrix, counting number of texts in which both words occur

- Word similarity ≈ cosine similarity of vector representations $\phi$

- But embedding is huge: the size of the vocabulary!

| | tesgüino | beer | wine | whiskey |
|---|---|---|---|---|
| drunk | 3 | 400 | 450 | 600 |
| bottle | 10 | 600 | 1000 | 600 |
| corn | 15 | 0 | 0 | 400 |
| hops | 0 | 450 | 0 | 0 |
| grapes | 0 | 0 | 1000 | 0 |
| dairy | 0 | 0 | 0 | 0 |

# Word2Vec: Skip-Gram

- Input: corpus $D$ including pairs $(w,\ c)$ where $w$ is a word and $c$ is a context, e.g.:
  - *"Everybody likes tesgüino"*
    ($w$ = *"tesgüino"*, $c$ = "likes")
    ($w$ = *"tesgüino"*, $c$ = "everybody")

e.g., context is every word in the sentence

- We want to model: given that we observe word $w$, what's likely in the context $c$?

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

- Our objective: find parameters that maximize the corpus probability

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

- (There are other methods for word2vec (e.g., GloVe, CBOW), but we will only look at Skip-Gram in this class)

# Word2Vec: Skip-Gram

**Objective: find**

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

**where**

$$p(c \mid w;\theta) = \frac{\exp(\text{score}(c,w))}{\sum_{c'\in\mathcal{C}} \exp(\text{score}(c',w))}$$

score: similarity of representations of word and context

*we use cosine similarity because, in contrast to dot product, it is unbiased towards vector magnitude:* $\text{score}(c,w) = \cos(\phi(c),\phi(w)) = \frac{\phi(c)\cdot\phi(w)}{||\phi(c)||\ ||\phi(w)||}$

*Slides adapted from Yoav Artzi's LM-Class; Skip-Gram from Mikolov et al. 2013*

# Word2Vec: Skip-Gram

**Objective: find**

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

**where**

$$p(c \mid w;\theta) = \frac{\exp(\text{score}(c,w))}{\sum_{c'\in\mathcal{C}} \exp(\text{score}(c',w))}$$

softmax function:
normalizes categorical
scores to a smooth
probability distribution

# Word2Vec: Skip-Gram

**Objective: find**

$$\arg\max_{\theta} \prod_{(w,c)\in\mathcal{D}} \frac{\exp(\text{score}(c,w))}{\sum_{c'\in\mathcal{C}} \exp(\text{score}(c',w))}$$

Problem: intractable to sum over all contexts

*Slides adapted from Yoav Artzi's LM-Class; Skip-Gram from Mikolov et al. 2013*

# Word2Vec: Skip-Gram

* Solution: negative sampling
  * Instead of summing over all possibly contexts in which a word could appear
  * Approximate via learning from contexts in which a word *doesn't* appear

* Model whether or not a word-context pair is likely to exist in the dataset

$$p((w,c) \in \mathcal{D} = \frac{1}{1+\exp(-\text{score}(c,w))}$$

similarity of representations of word and context

# Word2Vec: Skip-Gram

* Solution: negative sampling
  * Instead of summing over all possibly contexts in which a word could appear
  * Approximate via learning from contexts in which a word *doesn't* appear

* Model whether or not a word-context pair is likely to exist in the dataset

$$p((w,c) \in \mathcal{D} = \frac{1}{1+\exp(-\text{score}(c,w))}$$

sigmoid function: normalizes a scalar into a probability

# Word2Vec: Skip-Gram

**NEW objective: find**

$$\arg\max_{\theta} \prod_{(w,c)\in\mathcal{D}} p((w,c) \in \mathcal{D}) \prod_{(w,c)\in\mathcal{D}'} p((w,c) \notin \mathcal{D})$$

What is $D'$?

**where**

$$p((w,c) \in \mathcal{D} = \frac{1}{1+\exp(-\text{score}(c,w))}$$

$$p((w,c) \notin \mathcal{D}) = 1 - p((w,c) \in \mathcal{D})$$

# Word2Vec: Skip-Gram

**NEW objective: find**

$$\arg\max_{\theta} \prod_{(w,c)\in\mathcal{D}} p((w,c)\in\mathcal{D}) \prod_{(w,c)\in\mathcal{D}'} p((w,c)\notin\mathcal{D})$$

* Negative samples: sample and train on $l * |D|$ pairs $(w',\ c')$ where

$$w' \sim p(W) \qquad c' \sim p(C)$$

unigram prior over words

unigram prior over contexts

* How to train?
  * Gradient descent
  * In practice, we work with log probabilities, not direct probabilities, to avoid float underflow

# Word2Vec: Skip-Gram

What counts as context? Basically anything you want

A bottle of tesgüino is on the table.

Everybody likes tesgüino.

Tesgüino makes you drunk.

We make tesgüino out of corn.

| | *w* = tesgüino |
|---|---|
| of *w* | 1 |
| bottle of *w* | 1 |
| bottle __ *w* | 1 |
| cup __ *w* | 0 |
| *w* is | 1 |
| *w* is on | 1 |
| *w* is on the table | 1 |
| *w* __ __ __ table | 1 |
| {corn, *w*} | 1 |
| make *w* out | 1 |

# Word Vectors

* In Skip-Gram, the parameters of our models are the vectors for words and the vectors for contexts

$$p((w, c) \in \mathcal{D} = \frac{1}{1 + \exp(-\cos(\phi(c), \phi(w))}$$

context vectors word vectors

* Once we've trained a model, we can use these vectors for whatever we'd like!

# Vector Arithmetic

$$\text{a} : \text{b} :: \text{c} : ? \equiv d = \arg\max_{i \in \mathcal{W}} \frac{(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)}{||(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)||}$$

vector representing the
relationship between a and b

# Vector Arithmetic

$$\text{a} : \text{b} :: \text{c} : ? \equiv d = \arg\max_{i \in \mathcal{W}} \frac{(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)}{||(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)||}$$

apply this relationship
to a new word c

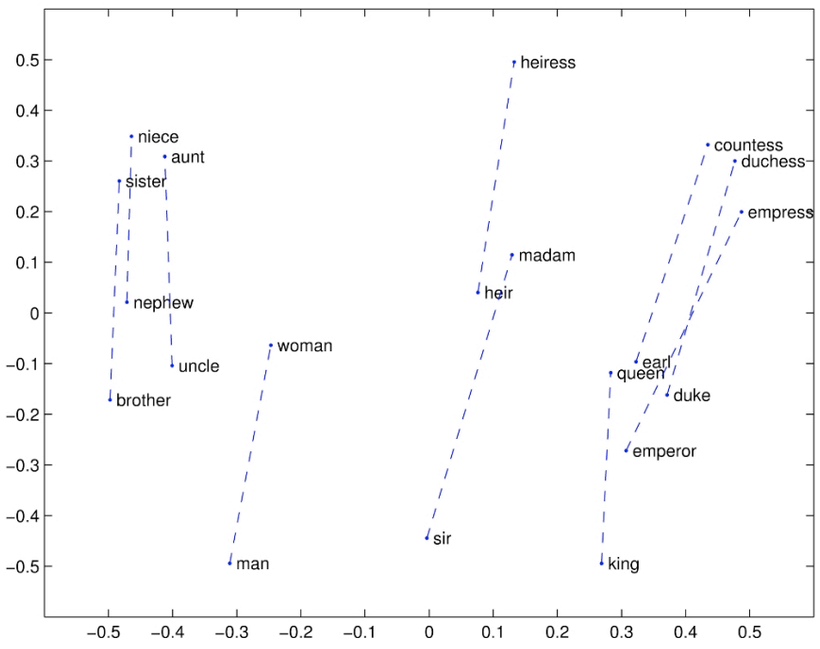
# Vector Arithmetic

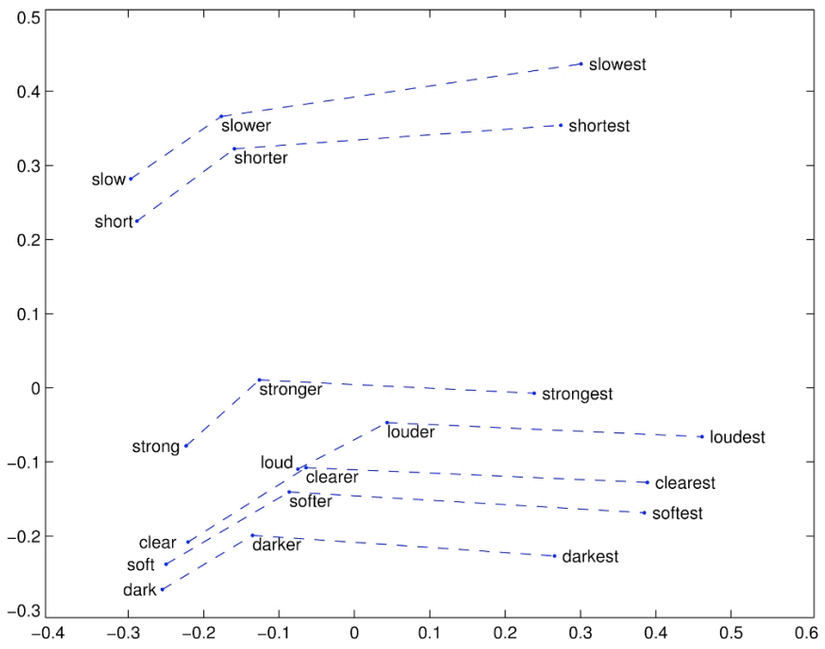
$$\text{a} : \text{b} :: \text{c} : ? \equiv d = \arg\max_{i \in \mathcal{W}} \frac{(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)}{||(\phi(b) - \phi(a) + \phi(c)) \cdot \phi(i)||}$$

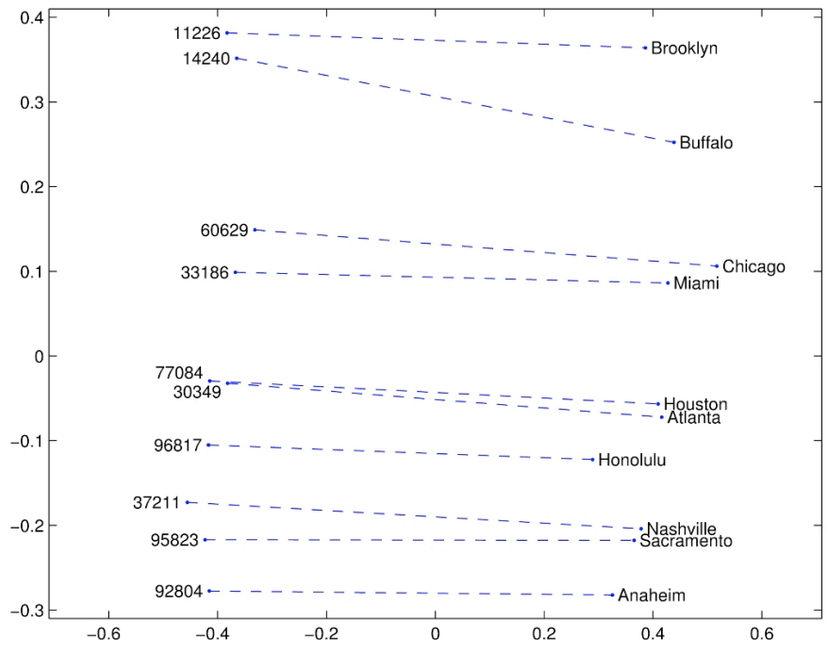
cosine similarity of

the expected representation of the analogy word

and some other word i



# Vector Arithmetic

Pairs $(x,\ y)$ that maximize $\cos(\phi(\text{he}) - \phi(\text{she}), \phi(x) - \phi(y))$

**Gender stereotype *she-he* analogies.**

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

**Gender appropriate *she-he* analogies.**

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

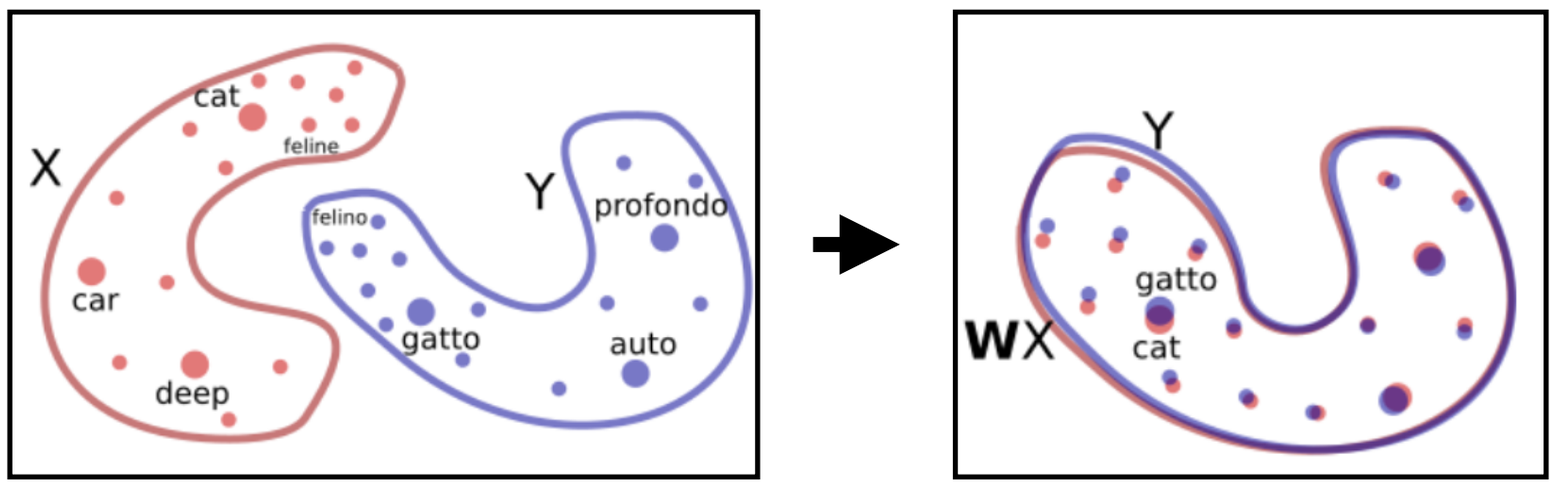
- Stereotypes are embedded in text data

- One way to “debias”: identify category subspace (e.g., “gender” subspace), project words onto that subspace, then subtract those projections from the original word

*Bolukbasi et al. 2016*

# Vector Arithmetic

- What if we have embeddings in different languages and want to align them? Given:
  - $\phi^A$ : language A's embedding function
  - $\phi^B$ : language B's embedding function
  - $\mathcal{D} = \{w_i^A, w_i^B\}_{i=1}^M$ : dataset pairing *M* word translations
- Find a matrix *W* such that $\min_W \sum_{i=1}^{M} ||W\phi(w_i^A) - \phi(w_i^B)||^2$




*Mikolov et al. 2013, Conneau et al. 2017*

# Challenge: Polysemy

## Verb [ edit ]

**run** (*third-person singular simple present* **runs**, *present participle* **running**, *simple past* **ran**, *past participle* **run** *or* (*nonstandard, colloquial*) **ran**)

1. To move swiftly.
    1. (*intransitive*) To move forward quickly upon two feet by alternately making a short jump off either foot. [coordinate term ▲] [quotations ▼]

        Coordinate term: walk

        ***Run**, and you might still catch the train!*
    2. (*intransitive*) To go at a fast pace; to move quickly. [quotations ▼]

        *I have been **running** all over the building looking for him.*

        *Sorry, I've got to **run**; my house is on fire!*
    3. (*transitive*) To cover (a course or a distance) by running.

        *I can **run** a mile, but I can't **run** the cross-country course.*
    4. (*transitive*) To complete a running course or event in (a given time).

        *I was hoping to make the team, but I didn't **run** the qualifying time.*
    5. (*intransitive*) To move briskly or smoothly with a motion of sliding, rolling, sweeping etc.

        *The shuttle **runs** back and forth on these rollers.*

        *As its name suggests, the monorail **runs** on a single rail.*

        *I felt her fingers **running** over my cheek.*

    • • •

    31. To tend, as to an effect or consequence; to incline. [quotations ▼]
    32. To have a legal course; to be attached; to continue in force, effect, or operation; to follow; to go in company. [qu

        *Certain covenants **run** with the land.*
    33. To encounter or suffer (a particular, usually bad, fate or misfortune). [quotations ▼]
    34. (*golf*) To strike (the ball) in such a way as to cause it to run along the ground, as when approaching a hole.
    35. (*video games*, *rare*) To speedrun.
    36. (*sports*, *especially baseball*) To eject from a game or match.

        *Jackson got himself **run** in the top of the sixth for arguing a borderline strike three call.*
    37. To press (a bank, etc.) with immediate demands for payment.

## Noun [ edit ]

**run** (*plural* **runs**)

1. Act or instance of running, of moving rapidly using the feet. [quotations ▼]

    *I just got back from my morning **run**.*
2. Act or instance of hurrying (to or from a place) (*not necessarily on foot*); dash or errand, trip. [quotations ▼]

    *I need to make a **run** to the store.*
3. A pleasure trip. [quotations ▼]

    *Let's go for a **run** in the car.*
4. Flight, instance or period of fleeing. [quotations ▼]
5. Migration of fish.
6. A group of fish that migrate, or ascend a river for the purpose of spawning.
7. A literal or figurative path or course for movement relating to:
    1. A (regular) trip or route. [quotations ▼]

        *The bus on the Cherry Street **run** is always crowded.*
    2. The route taken while running or skiing.

        *Which **run** did you do today?*

    • • •

    21. A line of knit stitches that have unravelled, particularly in a nylon stocking. [quotations ▼]

        *I have a **run** in my stocking.*
    22. (*nautical*) The stern of the underwater body of a ship from where it begins to curve upward and inward.
    23. (*mining*) The horizontal distance to which a drift may be carried, either by licence of the proprietor of a mine or by the nature of the formation; also, the direction which a vein of ore or other substance takes.
    24. A pair or set of millstones.
    25. One's gait while running; the way one runs.

        *I think they only have a weird **run** because their leg hurts.*

## Adjective [ edit ]

**run** (*not comparable*)

1. In a liquid state; melted or molten. [quotations ▼]

    *Put some **run** butter on the vegetables.*
2. Cast in a mould. [quotations ▼]
3. Exhausted; depleted (*especially with "down" or "out"*).
4. (*of a zoology*) Travelled, migrated; having made a migration or a spawning run. [quotations ▼]
5. Smuggled.

    ***run** brandy*

# Challenge: Polysemy

- Word embedding now has to somehow encode the meanings of all 67 senses of “run”…
  - Why is this hard?
  - What could we do instead?

- Contextualized word embeddings: compute representation of individual words dependent on the context in which it appears
  - The resulting architectures were some of the main catalysts for the design of modern LLMs!

*ELMo; Peters et al. 2018*

# Opportunity: Sentence Representations

- If we have representations for parts of sentences (wordtypes), then can we get a representation of the whole sentence?

- One option: bag-of-words representation

$$\phi(\overline{x}) = \frac{1}{|\overline{x}|} \sum_{i=0}^{|\overline{x}|} \phi(x_i)$$

- What's missing?

# Word Embeddings in the Age of LLMs

- Building a language technology pre-LLMs, for a target task:
  - Download some pre-trained word embeddings from the Internet
  - Initialize your language model with these word embeddings (all other parameters randomized)
  - Use your task-specific data to fine-tune the other parameters (possibly also fine-tuning the word embeddings)
  - Having a good starting point via word embeddings is **really** important, especially with little task-specific data

- But we don’t do this anymore
  - Language models still learn embeddings specific to each wordtype
  - But we don’t download them from the Internet — we download the whole language model

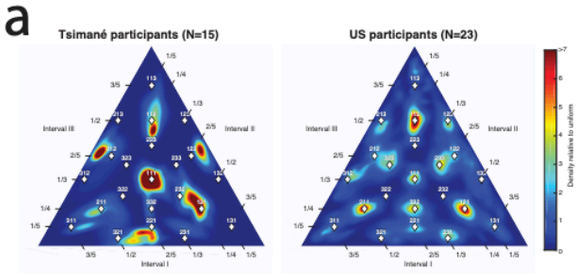
# Word Embeddings in the Age of LLMs

## So why might we still care about word embeddings?

Measuring

### Cognitive Science

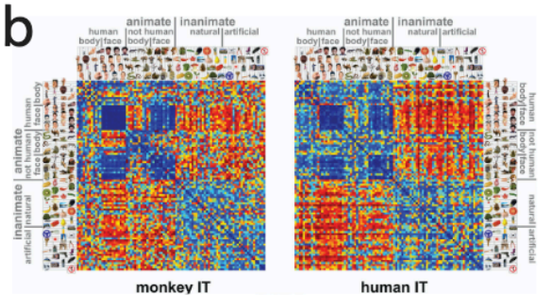
a




Jacoby & McDermott 2017

### Neuroscience

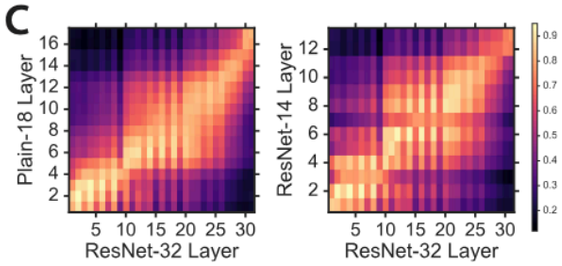
b




Kriegeskorte et al. 2008

### Machine Learning

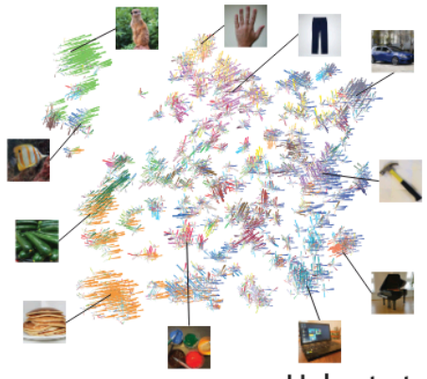
c




Kornblith et al. 2019

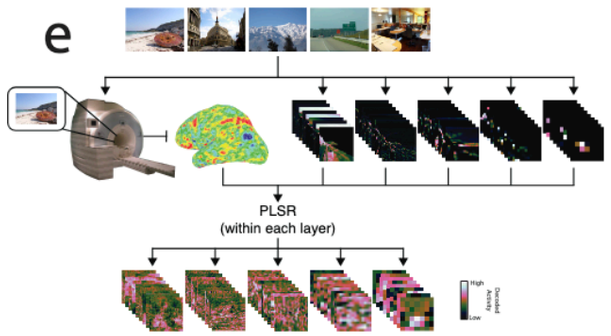
Bridging

d



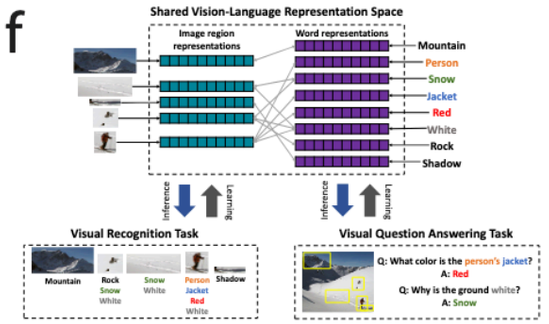
Hebart et al. 2020

e




O'Connell & Chun 2018

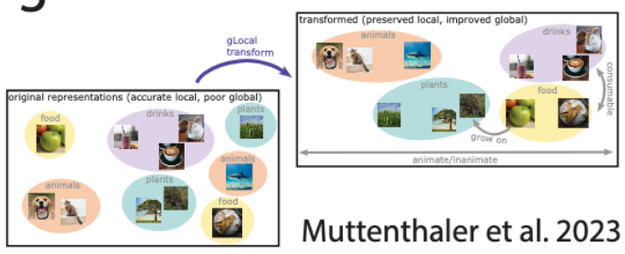
f




Gupta et al. 2017

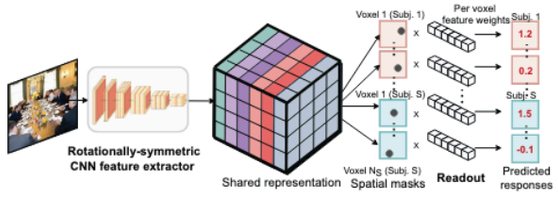
Increasing

g



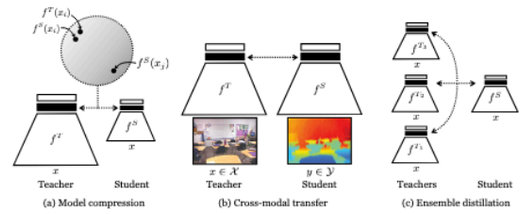

Muttenthaler et al. 2023

h




Khosla et al. 2023

i



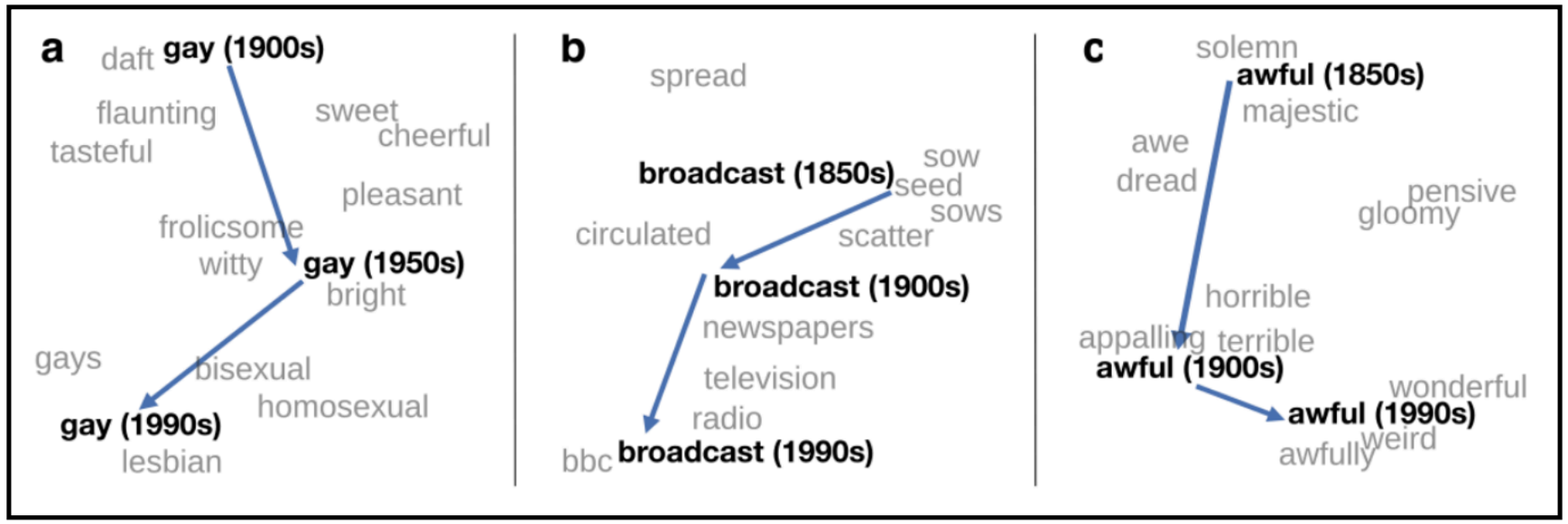

Tian et al. 2020

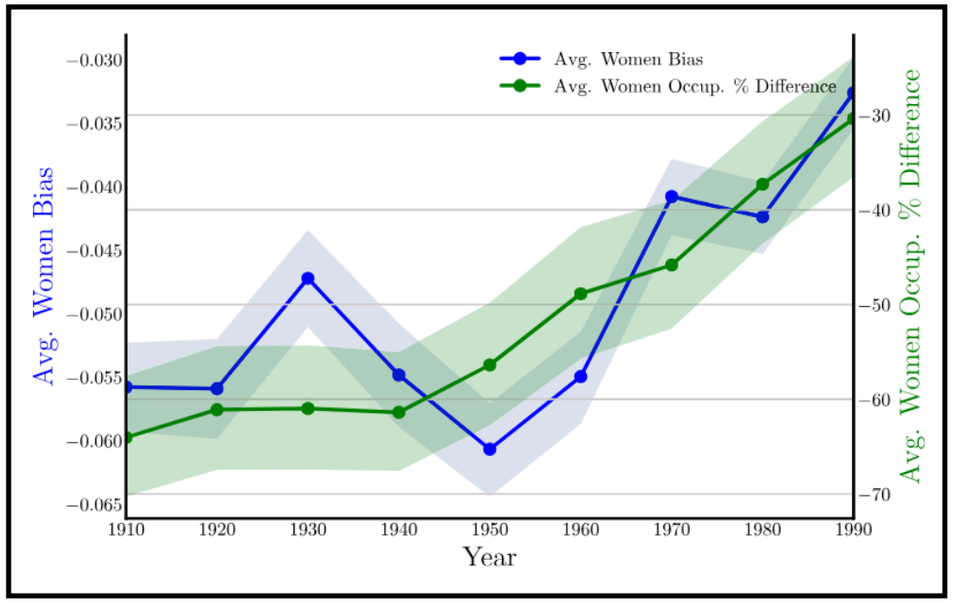
## Evaluating how well an LLM's learned concept space correlates with human concept spaces

*Sucholutsky et al. 2023*

# Word Embeddings in the Age of LLMs

So why might we still care about word embeddings?




| 1910 | 1950 | 1990 |
|---|---|---|
| Charming | Delicate | Maternal |
| Placid | Sweet | Morbid |
| Delicate | Charming | Artificial |
| Passionate | Transparent | Physical |
| Sweet | Placid | Caring |
| Dreamy | Childish | Emotional |
| Indulgent | Soft | Protective |
| Playful | Colorless | Attractive |
| Mellow | Tasteless | Soft |
| Sentimental | Agreeable | Tidy |

Analyze how language changes over time

*Hamilton et al. 2018, Garg et al. 2018*