

# What Makes Human Languages Interesting?

- ▶ Connecting minds: how one person's thoughts reach into another's
- ▶ Gender assignment to words, explicit in some languages
- ▶ Even in English, think of pronouns and names
  - ▶ Cat
  - ▶ Book
  - ▶ Faith
  - ▶ Hope

# What Makes Human Languages Challenging?

- ▶ Sarcasm
- ▶ Versus logic
  - ▶ No no
  - ▶ Yes yes

# Applications of NLP

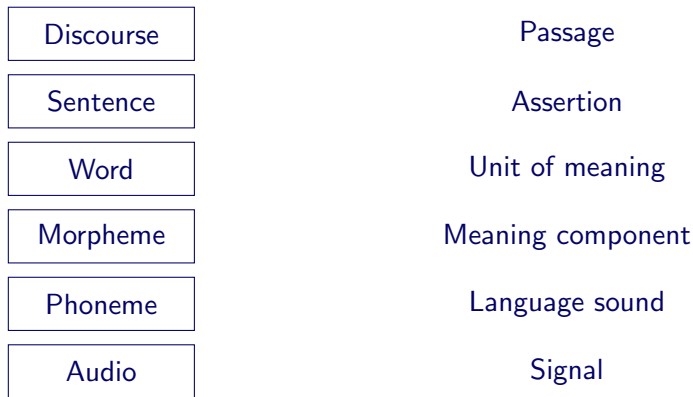
What makes NLP so valuable?

# Brief Historical Look

- ▶ Ad hoc
- ▶ Inspired by cognitive science
- ▶ Knowledge-based
- ▶ Statistical
- ▶ Speech

# Hierarchy of Language Concepts

Not to be taken too seriously



- ▶ How would you pronounce *project*?
- ▶ Verb vs. noun

# Language as a Symbolic System

Also called semiotics

Pragmatic

Meaning based on words and context

Semantics

Meaning based on words

Syntax

Structure of symbols

Symbol

Token (morpheme, phoneme, lexeme)

- ▶ Holy grail: to express meaning compositionally
  - ▶ Meaning of whole = combination of meanings of parts

# Text Normalization

- ▶ Tokenization
  - ▶ Punctuation
  - ▶ Abbreviations
  - ▶ Number, date, email address, ...
  - ▶ Clitics: not standalone, e.g., n't
  - ▶ Case to mark names, e.g., mark vs. Mark
  - ▶ Hyphenated words
- ▶ Normalization
  - ▶ Case folding
  - ▶ Stemming: remove affixes
  - ▶ Porter stemming: popular but heavy-handed application of rules
  - ▶ Lemmatization: standard root, even if superficially different, e.g., {am, is}  $\Rightarrow$  *be*
- ▶ Challenges
  - ▶ Scripts such as Chinese

# Minimum Edit Distance

## Illustration of dynamic programming

- ▶ Source string  $X[n]$ , prefixes  $X[1..i]$ ,  $i \in [1..n]$
- ▶ Target string  $Y[m]$ , prefixes  $Y[1..j]$ ,  $j \in [1..m]$
- ▶ Edit distance  $D(i,j)$  between  $X[1..i]$  and  $Y[1..j]$
- ▶  $D(0,0) = 0$ ; for  $i \in [1..n]$  and  $j \in [1..m]$ :

$$D(i,j) = \min \begin{cases} D(i-1,j) + \text{del-cost}(X[i]) \\ D(i,j-1) + \text{ins-cost}(Y[j]) \\ D(i-1,j-1) + \text{sub-cost}(X[i], Y[j]) \end{cases}$$

- ▶ Levenshtein values

$$D(i,j) = \min \begin{cases} D(i-1,j) + 1 \\ D(i,j-1) + 1 \\ D(i-1,j-1) + \begin{cases} 2 & X[i] \neq Y[j] \\ 0 & X[i] = Y[j] \end{cases} \end{cases}$$

- ▶  $D(n,m)$  is the answer; compute path from  $(n,m)$  back to  $(0,0)$



# Levenshtein Example

There (Source)  $\Rightarrow$  Their (Target)

		Target					
		0	1	2	3	4	5
Source		#	T	H	E	I	R
0	#						
1	T						
2	H						
3	E						
4	R						
5	E						