6.863J Natural Language Processing
Lecture 1: Introduction

Instructor: Robert C. Berwick
berwick@ai.mit.edu

The Menu Bar

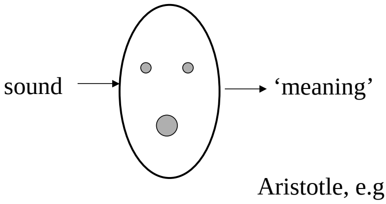
- Administrivia
  - All on web page: www.ai.mit.edu/courses/6.863
  - Stellar web site:

  http://stellar.mit.edu/S/course/6/sp03/6.863j/
- What this course is about
- Why NLP is hard, and interesting
- The ingredients of language
- Why language and computation?
- What you have to do in the course
- Till next time…

What is this course all about?

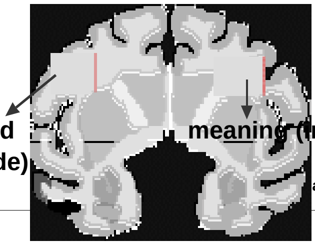
- Computational methods for working with natural (human) languages
- Applications of computer science & AI
- Linguistic theory
- Natural (psycholinguistics) or artificial computation (natural language processing, NLP)

Simple model: relation of sound-meaning



Aristotle, e.g., only 2500 years old…

Language = Pairing sound & meaning



sound (outside)

meaning (inside world)

ain'

Natural language at the heart of human intelligence

- The first Turing test:

"Rabbah Zoreh made a Gollum and brought it to Rabbah; he bid it to talk.

Rabbah replied: 'It cannot speak; return it unto the flames'

(Manhet Sahedrin, Babylonian Talmud, approx. 400 BCE)

Human language: special character

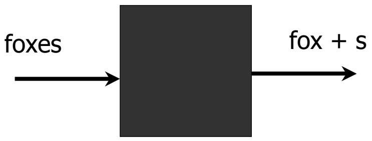
- Pop-quiz (multiple choice): who produced the following 'sentences' (Names changed to protect the innocent):
- (1) I see red one
- (2) P. want drink
- (3) P. open door
- (4) P. tickle S.
- (5) I go beach
- (6) P. forget this
- (7) P said no
- (8) P. want out
- (9) You play chicken
- Multiple choice: (a) Pidgin speakers; (b) apes (signing); (c) Feral child Genie; (d) ordinary children

Applications

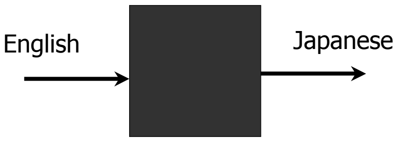
- Lightweight / AI-complete
- Line breakers, hyphenators, spell checkers, grammar & style checkers
- Information Retrieval (IR) / Question-answering systems
- Sentence/dialog understanding
- Document summarization
- Machine translation

But what’s inside the black box? Lightweight / AI-complete

Lightweight:



AI-complete



But the most important reason of all…

It’s the year 2003

Dave Bowman: Open the pod bay doors, HAL
HAL: I’m sorry Dave, I’m afraid I can’t do that.

Why study NLP? How the mind works: the Twain test

- How do people look up words?
- How do people parse sentences?
- How do people learn language?
- How does language evolve?

The human sentence processor

- Properties
  - Garden-path (Blind alleys)
    - Sue told the person that she hired a story
    - Sue told the person that she hired an assistant
  - Non-uniform processing cost
    - The reporter who the senator attacked admitted the error
    - The reporter who attacked the senator admitted the error

Sentence analysis – can be subtle – what is the knowledge?

- Representation of events
- John is too stubborn [to talk to]
  - Event(e) & agent(x, e)
- John is too stubborn [to talk to Mary]
- Who is the agent now?

What makes NLP interesting (and difficult)

- Complex phenomenon arising out of the interaction of many distinct *kinds* of knowledge
- *What* is this knowledge? (data structures - linguistics)
- *How* is it put to use? (algorithms)
- Example: "the dogs ate ice-cream"

Knowledge of language: What do we *know* about this sequence?

- *Do*.. begins a valid word of English, but no English word begins with *ptk*; the *s* on *dogs* marks it as plural
- Words must appear in a certain order: **Dogs ice-cream ate*
- Parts and divisions: *the dogs* is the Subject; *ate ice-cream,* the Predicate. Distinct parts or *constituents* (phrases)
- Who did what to whom: *the dogs* is the Agent of the action *ate,* while *ice-cream* is the Object

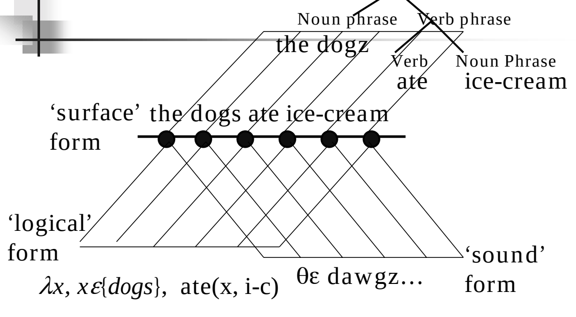
But wait, there's more… (you also get…)

- The two sentences *John claimed the dogs ate ice-cream* and *John denied the dogs ate ice-cream* are logically incompatible
- Sentence & the world: know whether the sentence is *true* or not - perhaps whether in some particular situation (possible world) the dogs did indeed eat ice-cream
- Know that it would sound fine if it were to follow *I had espresso this morning, but…*
- However, odder if it were to follow *John is intelligent*

The linguistic pipeline

- We need data representation (linguistic) primitives to represent sounds, sound pieces, words, word pieces, sentences, sentence pieces (compare to vision), so….
- Primitives only contain *partial* information, unique (proprietary) to their own "level", so **they must combine *in non-arbitrary ways***
- Levels must be connnected
- What is the knowledge in each level?

The "spiral notebook" model



'phrase' form

Sentence

Noun phrase

Verb phrase

the dogz

Verb

ate

Noun Phrase

ice-cream

'surface' form

the dogs ate ice-cream

'logical' form

$\lambda x,\ x\varepsilon\{dogs\}$, ate(x, i-c)

θε dawgz…

'sound' form

Sentence knowledge is subtle

- A book was given Mary
- Mary was given a book
- A book was given to Mary
- Mary was given a book to

Word knowledge is subtle

- He arrived at the station
- He chuckled at the station

- He arrived drunk
- He chuckled drunk

- He chuckled his way through the meeting
- He arrived his way through the meeting

Invisible knowledge

- I want to solve the problem
- I wanna solve the problem
- Displacement: I understand these students
- These students I understand
- I want these students to solve the problem
- These students I want [x] to solve the problem [x]= these students

Notice that contraction of want+to *is now blocked!*

What is the character of this knowledge?

- Some of it must be memorized (obviously so):
  - Singing-> Sing+ing; Bringing-> bring+ing

Duckling **->** ***?? Duckl +ing***

So, must know *duckl* is not a word

But it can’t all be memorized…
Because there is too much to know

Besides memory, what else do we need?

- English plural:
- Toy+s -> toyz ; add *z*
- Book+s -> books ; add *s*
- Church+s -> churchiz ; add *iz*
- Box+s-> boxiz ; add *iz*

- What if a *novel* word?
  - Bach's many cantatas
  - Which pronounciation is it? *S* or IZ ?

Bachs many cantatas NOT BachIZ despite Analogy/similarity to 'box' - why?

Conclusion: must be a rule system to generate/process infinite # of examples

- Insight of Panini (Sanskrit grammarians): circa 400BCE: system of morphological analysis, based on cascaded rules (we will see how to implement this later on)
- Nice to have whole book written to reveal this published in year 2000
- Still, have we made progress in the intervening two millenia…?

Panini

- *Astadhyayi: (400-700BCE?)* Panini gives formal production rules and definitions to describe Sanskrit grammar. Starting with about 1700 basic elements like nouns, verbs, vowels, consonants he put them into classes. The construction of sentences, compound nouns etc. is explained as ordered rules operating on underlying structure

What's more…

- On the basis of just under 4000 sutras [rules expressed as aphorisms], he built virtually the whole structure of the Sanskrit language
- Uses a notation precisely as powerful as Backus normal form - an algebraic notation to represent numeral (and other patterns) by letters
- So, have we made progress?

The nature of levels (the projections)

- Each level consists of a set of primitives, plus a set of operations to glue them together
  - E.g., dog+s -> dogs
- Each level consists of a mapping relation to other levels (perhaps only 1 other level)
  - Model due to Chomsky (1951, 1955)
- *What* are the levels? *What* are the primitives?
- After this: *How* do we compute with them?

What and How: the What - Components of Language

1. *Sound structure:* includes *phonetics,* the actual pattern of speech sounds, and *phonology,* the sound pattern rules & regularities in a language
2. *Word structure:* or *morphology* and *morphophonemics* (after *morphos,* shape) analyzes how words are formed from minimal units of meaning, or *morphemes,* e.g., *dogs= dog+s.* Interacts w/ (1):

   you say potato, and I say potato,…
3. *Phrase structure:* or *syntax* (literally from the Greek syntaxis, σινταξιζ, "arranged together") describes possible word combinations or *phrases* & how these are arranged together hierarchically

Components of language, continued

4. *Thematic structure* casts sentences in a broad "who did what to whom" form, using notions like agent, theme, affected object - note how this ties to other levels: word properties and order affect what thematic structure we have
5. *Lexical-conceptual structure* looks at words in terms of simple, physical causal elements

- *The earth circles around the sun*
- *The earth revolves around the sun*
- *The earth circles the sun,* BUT
- *??The earth…*
- *(Why?)*

Components of language

6. *Semantic structure:* includes both (4) and (5), but also depends on what one takes 'meaning' to mean. One popular view: associates meaning of a sentences with precisely those conditions that make the sentence *true*- its *truth conditions* (Frege, Tarksi). Relation betw. meaning and the world
7. *Pragmatic* and *discourse structure* describe how language is used *across* sentences. (Reasoning about actions, beliefs, causes, intentions.) Line between language and thinking blurs…

It's cold in here ->

‘Parsing’ = mapping from <u>surface</u> to <u>underlying</u> representation

- What makes NLP hard: there is not a 1-1 mapping between *any* of these representations!
- We have to know the data structures and the algorithms to make this efficient, despite *exponential complexity* at every point
- What are the right models for this?
- (We might, and will, use different *representational* and *computational* models for different levels)

Good Engineering Demands Good Science

- *What* before *How*
- *How* before *How to do*
- Class will have several case studies
  - Morphology done by finite-state machines
  - Parsing done by context-free grammars
  - Learning word meanings by statistical means, without linguistic models

1-many mapping= ambiguity at *EVERY* level *AND* at every mapping between levels

- Multiple sound parsings: given ‘dogs’, multiple sound forms (well, is it dog+s plural, or dog+s, verb)
- Multiple word categories: given ‘dogs’, is it a noun or a verb?
- Multiple phrases: given ‘dogs eat ice-cream on the table’ is ‘on the table’ related to ice-cream or to eating?
- Multiple logical forms: which dogs?
- And so on..

Why is NLP hard?

- Why is NLP difficult?
- Multiple representations
  - many “words”, many “phenomena” --> many “rules”?
    - OED: 400k words; Finnish lexicon (of forms): ~$2 . 10^7$
    - sentences, clauses, phrases, constituents, coordination, negation, imperatives/questions, inflections, parts of speech, pronunciation, topic/focus, and much more!
    - irregularity (exceptions, exceptions to the exceptions, ...)
    - potato -> potato es (tomato, hero,...); photo -> photo s, **and even: both** mango -> mango s **or** -> mango es
    - **Adjective / Noun order:** new book, electrical engineering, general regulations, flower garden, garden flower, ...: **but** Governor General

NLP

- ambiguity
  - books: NOUN or VERB?
    - you need many books vs. she books her flights online
  - No left turn weekdays 4-6 pm / except transit vehicles (Charles Street at Park Station)
    - when may transit vehicles turn: Always? Never?
  - Thank you for not smoking, drinking, eating or playing radios without earphones. (MBTA bus)
    - Thank you for not eating without earphones??
    - or even: Thank you for not drinking without earphones!?
  - My neighbor’s hat was taken by wind. He tried to catch it.
    - ...catch the wind or ...catch the hat ?

How does it all factor together?

- Preferences:
  - clear cases: context clues: she books --> books is a verb
    - rule: if an ambiguous word (verb/nonverb) is preceded by a matching personal pronoun -> word is a verb
  - less clear cases: pronoun reference
    - she/he/it refers to the most recent noun or pronoun (?) (but maybe we can specify exceptions)
  - selectional:
    - catching hat >> catching wind (but why not?)
  - semantic:
    - never thank for drinking in a bus! (but what about the earphones?)

Phonetics/orthography

- Input:
  - acoustic signal (phonetics) / text (orthography)
- Output:
  - phonetic alphabet (phonetics) / text (orthography)
- Deals with:
  - Phonetics:
    - consonant & vowel (& others) formation in the vocal tract
    - classification of consonants, vowels, ... in relation to frequencies, shape & position of the tongue and various muscles in the vocal tract
    - intonation
  - Orthography: normalization, punctuation, etc.

Phonology

- Input:
  - sequence of phones/sounds (in a phonetic alphabet); or “normalized” text (sequence of (surface) letters in one language’s alphabet) [NB: phones vs. phonemes]
- Output:
  - sequence of phonemes (~ (lexical) letters; in an abstract alphabet)
- Deals with:
  - relation between sounds and phonemes (units which might have some function on the upper level)
  - e.g.: [u] ~ oo (as in book), [æ] ~ a (cat); i ~ y (flies)

Morphology (word ‘shape’)

- Input:
  - sequence of phonemes (~ (lexical) letters)
- Output:
  - sequence of pairs (lemma, (morphological) tag)
- Deals with:
  - composition of phonemes into word forms and their underlying lemmas (lexical units) + morphological categories (inflection, derivation, compounding)
  - e.g. quotations ~ quote/V + -ation(der.V->N) + NNS.

Surface syntax

- Input:
  - sequence of pairs (lemma, (morphological) tag)
- Output:
  - sentence structure (tree) with annotated nodes (all lemmas, (morphosyntactic) tags, functions), of various forms
- Deals with:
  - the relation between lemmas & morph. categories and sentence structure
  - uses syntactic categories such as Subject, Verb, Object,...
  - e.g.: I/PP1 see/VB a/DT dog/NN ~
    ((I/sg)SB ((see/pres)V (a/ind dog/sg)OBJ)VP)S

Meaning (semantics)

- Input:
  - sentence structure (tree) with annotated nodes (lemmas, (morphosyntactic) tags, surface functions)
- Output:
  - sentence structure (tree) with annotated nodes (semantic lemmas, (morphosyntactic) tags, functions)
- Deals with:
  - relation between categories such as “Subject”, “Object” and categories such as “Agent”, “Effect”; adds other cat’s

…and beyond

- Input:
  - sentence structure (tree): annotated nodes (autosemantic lemmas, (morphosyntactic) tags, deep functions)
- Output:
  - logical form, which can be evaluated (true/false)
- Deals with:
  - assignment of objects from the real world to the nodes of the sentence structure
  - e.g.: (I/Sg/Pat/t (see/Perf/Pred/t) Tom/Sg/Ag/f) ~

see(Mark-Twain[SSN:...],Tom-Sawyer[SSN:...])[Time:bef 99/9/27/14:15][Place:39ş19’40”N76ş37’10”W]

Phonology redux: from *what* to *how*

- (Surface ↔ Lexical) Correspondence
- "symbol-based" (no complex structures)
- Ex.: (stem-final change)
  - lexical: **b a b y + s** *(+ denotes start of ending)*
  - surface: **b a b i e s** *(phonetic-related:* **b🡅b🡥0s***)*
- Arabic: (interfixing, inside-stem doubling) (lit. 'read')
  - lexical: **kTb+uu+CVCCVC** *(CVCC...vowel/consonant pattern)*
  - surface: **kuttub**

Phonology examples

- German (umlaut) (satz ~ sentence)
  - lexical: **s A t z + e** *(A denotes "umlautable" a)*
  - surface: **s ä t z e** *(phonetic:* **zæc♏***, vs.* **zac***)*
- Turkish (vowel harmony)
  - lexical: **e v + l A r** (←houses) **b a š + l A r**
  - surface: **e v l e r** (heads→) **b a š l a r**
- Czech (e-insertion & palatalization)
  - lexical: **m a t E K + 0** (←mothers/gen.) **m a t E K + ě**
  - surface: **m a t e k** (mother/dat.→) **m a t _ c e**

Phonology-morphology interaction

- Fly+s -> flys -> flies (y->i rule)
- Duckling example
- Go-getter-> get+er; doer-> do+er; Beer->??
- So we start by asking *what knowledge* do we need
- Then we ask *how* do we want to represent it and *how to* compute with it?

What knowledge do we need?

- Knowledge of "stems" or "roots"
  - *Duck* is a possible root; but not *duckl*
  - Seems like we need a *dictionary* (a *lexicon*)
- Only some endings go on some words, not others
  - *Do+er* ok; (a class of verbs) but not following *be*
- In addition, *spelling change* rules that 'adjust' the *surface form* (what we hear/spell) vs. what the *underlying* dictionary (lexicon) form is:
  - Get+er-> double the *t* -> getter
  - Fox+s -> insert *e*-> foxes
  - Fly+s -> insert *e* -> flyes-> *Y* to *I* -> flies
  - Any others? (Turns out, about 5 rules do a lot)

Why not just put this all in a big dictionary? (lexicon)

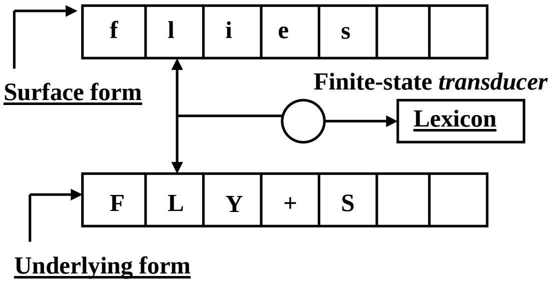
- Other languages: Turkish, approx. 600 x 10^6 forms
- Finnish: 10^7
- As we saw, always novel forms
- Actually, infinite even in English

antimissile

anti-anti-missile

anti-anti-anti-missile…
(can always up the anti)

Now the picture is a bit clearer…



f	l	i	e	s		

Surface form

Finite-state *transducer*

Lexicon

F	L	Y	+	S		

Underlying form

So, the question for next time…

- *Define* finite-state transducer formally
- *What* can we describe with this?
- *How* can we implement it?
- *What* are its strengths and limitations (given its representation) - what it can and cannot do well
- (Note: a *pure* model of concatenative morpho-phonology)
- You will do this for a more complicated language, using an implementation called PC-Kimmo (Turkish, Spanish, Yawlemani…)