Cs / Philo 372

Natural Language Understanding
week 6
Feb 27, 2006

Understanding Natural Language

- Parsing
  - Language theory
- Generating
- Understanding
  - Challenges for understanding

- Grammars
  - A set of rewrite rules
  - Written: A -> B
    - read “anywhere you see an A, substitute a B”

Language Theory

- Regular Grammars
  - Can represent a*b* but not A^nB^n
    - Equivalent to finite state machines
    - Left: one non-terminal, right: one terminal [1 non-terminal]
- Context free Grammars
  - Can represent A^nB^n but not $A^nB^nC^n$
    - Finite state machines with a single stack
    - Almost all of natural language fits here
    - Left: single non-terminal Right: anything
- Context Sensitive Grammars
    - Left: whatever, Right: at least as many symbols as left
- Recursively Enumerable Grammars
  - Turing equivalent

Parsing

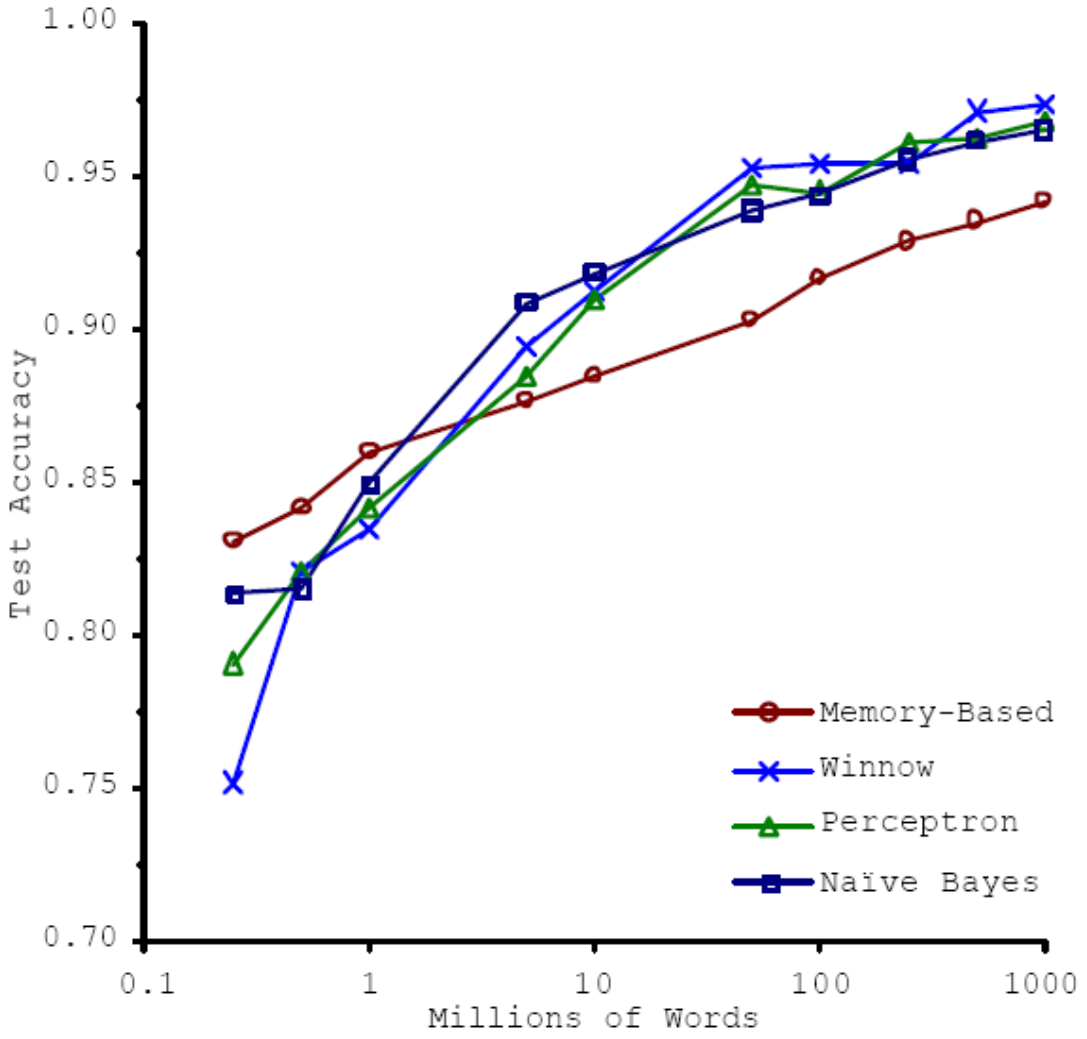
- Determine if a sentence is in a language
  - First step towards understanding
- Problem 1: POS Ambiguity
  - For example – garden path sentences
    - The man who hunts ducks out on weekends.
    - The old man the boat.
- Problem 2: Time
    - Lower on Chomsky hierarchy ==> faster
    - Computer languages are constructed for rapid parsing
    - There is more free text than computer text.
    - Chart parsers, etc

Parsing
Resolving POS Ambiguity

- Unigram Tagging -- ~90%
  - Just tag as with most likely category
- Trigram Tagger -- ~93%
  - Use mostly likely POS of final word in tri-gram
- “transformation tagger” -- ~94%
  - Start with unigram then apply rules to fix mistakes
- Max Entropy -- ~95%
  - Similar to transformation, but uses rules differently
- Clever combining & big training -- 98.9%

Results from Brill & Wu, 1998 and Brill & Banko, 2001

Learning Curves on 1 billion labeled words



Text Generation

- Basic idea – run a parser backwards
  - Problem – Overgeneration
- A bunch of issues
  - e.g., subjective vs objective pronouns
    - Him threw it to I.
- Solution
  - Augment words and rules.
    - np(Case,Y) :- pronoun(case, X), append([X], [], Y).
    - np(Case,Y) :- det(Z), noun(Q), append([Z], [Q], Y).
      - Note that this rule does not use Case on left
    - Pronoun(object, me).
    - pronoun(subject, I).

Text Generation

- Augmented Grammars
  - Chomsky – context sensitive grammar
  - Note that in “np(Case,Y) :- det(Z), noun(Q), append([Z], [Q], Y).” the variable “Case” is free.
    - This is not really correct. Might need another augmentation for nouns “animate”
      - The ball threw
  - Verb categories?
  - Adjectives, adverbs?
- Colorless green ideas sleep furiously

Understanding Text

- Clearly parsing is not understanding
- Generation can be done without understanding
- Understanding without meaning?
  - Symbol grounding problem
    - Dictionaries do not ground symbols
    - Cyc?
    - Can computers understand “red”.
      - Try to explain “red” to a person blind since birth.
      - Try to explain a dog to a person who has never seen one.

Philo of understanding text

- “Grue Emeralds”
  - Goodman, 1965
    - All emeralds you have seen to now are green, therefore all emeralds are green. But you could as easily conclude that all emeralds are “grue”. So, which concept of emeralds does the other person have (irregardless of correctness)?
- Private Languages
  - Wittgenstein, 1953 & Kripke, 1982
    - Suppose I “internally” use “quus” rather than “plus”
    - Quss XqY==X+Y when X,Y<57, else XqY=5
      - “Each new application we make is a leap in the dark; any present intention could be interpreted so as to accord with anything we may choose to do. So there can be neither accord, nor conflict”
      - Linked to COLT and antagonist learners
- Conclusions???

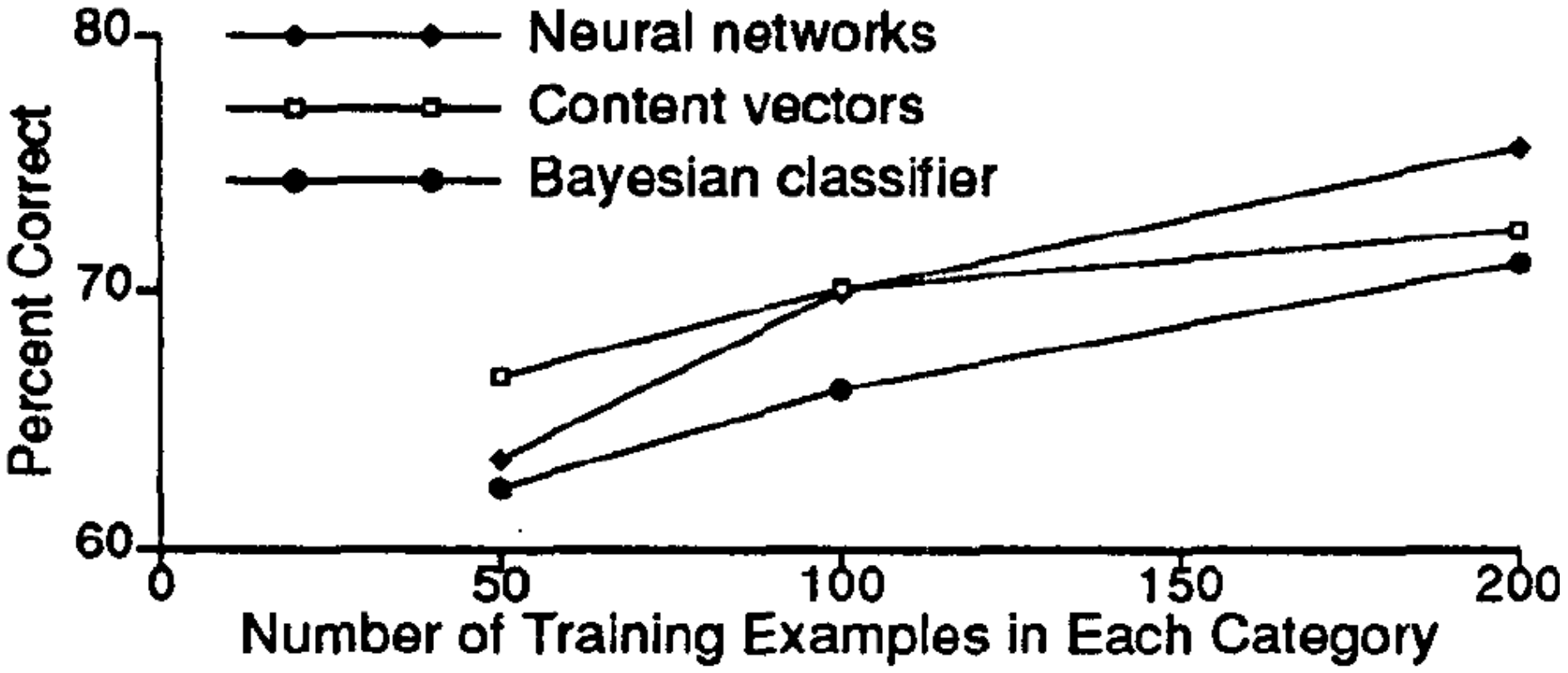
NL Understanding -- more

- Understanding is not about single sentences
  - Else cyc + parse might be sufficient
  - E.g.,
    - “John had a book. He gave the book to me”
    - “John had a book. He gave it to me.”
  - So understanding covers ALL of a conversational thread – possibly over all time.

Semantic Ambiguity

- Even knowing POS words have multiple meanings
- Understanding requires knowing which meaning
  - Consider “line”
  - 30 meanings in wordnet (wordnet.princeton.edu)
  - Leacock et al (1996)
    - Learning from examples to pick the meaning
    - From among 6 meanings get about 80% correct.
    - Note skew with Brill POS work

Disambiguating Word Meanings



Prolog – more lists

- Take a multilevel list and flatten it
  - fltten([[[[a]]]], Res).
    - Res=[a]
  - fltten([a[b[c,d],e]], Res).
    - Res= [a,b,c,d,e]
- Base case
  - fltten([], []).
- Calls
  - fltten([A|T], X) :- atom(A), fltten(T,Y), append([A],Y,X). % note the use of “atom”
  - fltten([A|T], X) :- append([A],Y,X), fltten(T, Y).
- What happens if reverse append and fltten in last rule?

Prolog and Math

- Compute the Nth Fibonacci number
- Fib(9, N).
  - N=34
- fib(N,Result) :- ...
- ??
- Fib(0, A, B, Result) :- plus(A,B,Result).

List Membership

- Determine if something is a member of a list
- Maybe flatten it first?
- membr(a, [d,f,g,h,a,s,d])
  - Yes
- membr(a,[d f g]).
  - No
- membr(X, List) :- membrf(X, Flt), fltten(List, Flt).
- Base case: membrf(A, []) :- ???.