We’re going to study what goes into getting computers to perform useful and interesting tasks involving human languages.

We are also concerned with the insights that such computational work gives us into human processing of language.
Why Should You Care?

1. An enormous amount of knowledge is now available in machine readable form as natural language text
2. Conversational agents are becoming an important form of human-computer communication
3. Much of human-human communication is now mediated by computers

Commercial World

- Lot’s of exciting stuff going on, e.g.
  - Google
  - Microsoft
  - IBM
  - Powerset
  - Yahoo!
Web Q/A

Deep Q/A to Jeopardy Winner
Weblog Analytics

- Data-mining of Weblogs, discussion forums, message boards, user groups, and other forms of user generated media
  - Product marketing information
  - Political opinion tracking
  - Social network analysis
  - Buzz analysis (what’s hot, what topics are people talking about right now).

Major Topics (from Textbook)

1. Words
2. Syntax
3. Semantics
4. Pragmatics

5. Applications exploiting each
Applications

• First, what makes an application a *language processing application* (as opposed to any other piece of software)?
  - An application that requires the use of knowledge about human languages
    - Example: Is Unix wc (word count) an example of a language processing application?

Big Applications

• Question answering
• Conversational agents
• Summarization
• Machine translation
Conversational Agents

TURING TEST EXTRA CREDIT:
CONVINCE THE EXAMINER
THAT HE’S A COMPUTER.

YOU KNOW, YOU MAKE
SOME REALLY GOOD POINTS.
I’M ... NOT EVEN SURE
WHO I AM ANYMORE.

Big Applications

• These kinds of applications require a
tremendous amount of knowledge of
language.

• Consider the following interaction with
HAL the computer from 2001: A Space
Odyssey
HAL from 2001

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

What’s needed?

- Speech recognition and synthesis
- Knowledge of the English words involved
  - What they mean
- How groups of words clump
  - What the clumps mean
What’s needed?

• Dialog
  ◆ It is polite to respond, even if you’re planning to kill someone.
  ◆ It is polite to pretend to want to be cooperative (I’m afraid, I can’t...)

Caveat

NLP has an AI aspect to it.
  ◆ We’re often dealing with ill-defined problems
  ◆ We don’t often come up with exact solutions/algorithms
  ◆ We can’t let either of those facts get in the way of making progress
Course Material

- We’ll be intermingling discussions of:
  - Linguistic topics
    - E.g. Morphology, syntax, discourse structure
  - Formal systems
    - E.g. Regular languages, context-free grammars
  - Applications
    - E.g. Machine translation, information extraction

Topics: Linguistics

- Word-level processing
- Syntactic processing
- Lexical and compositional semantics
- Discourse processing
- Dialogue structure
Topics: Techniques

- Finite-state methods
- Context-free methods
- Augmented grammars
  - Unification
  - Lambda calculus
- First order logic
- Probability models
  - Supervised machine learning methods

Quotes

- It must be recognized that the notion “probability of a sentence” is an entirely useless one, under any known interpretation of this term.
  - Noam Chomsky, 1969
- Whenever I fire a linguist our system performance improves.
  - Frederick Jelinek, 1988
**Topics: Applications**

- **Small**
  - Spelling correction
  - Hyphenation
- **Medium**
  - Word-sense disambiguation
  - Named entity recognition
  - Information retrieval
- **Large**
  - Question answering
  - Conversational agents
  - Machine translation

- **Stand-alone**
- **Enabling applications**
- **Funding/Business plans**

**Categories of Knowledge**

- **Phonology**
- **Morphology**
- **Syntax**
- **Semantics**
- **Pragmatics**
- **Discourse**

Each kind of knowledge has associated with it an encapsulated set of processes that make use of it.

Interfaces are defined that allow the various levels to communicate.

This usually leads to a pipeline architecture.
Ambiguity

• Computational linguists are obsessed with ambiguity
• Ambiguity is a fundamental problem of computational linguistics
• Resolving ambiguity is a crucial goal

Ambiguity

• Find at least 2 meanings of these headlines:
  - Drunk Gets Nine Months In Violin Case
  - Farmer Bill Dies In House
  - Iraqi Head Seeks Arms
  - Enraged Cow Injures Farmer With Ax
  - Stud Tires Out
  - Eye Drops Off Shelf
  - Teacher Strikes Idle Kids
  - Squad Helps Dog Bite Victim
Ambiguity is Pervasive

• Phonetics!
  - I mate or duck
  - I’m eight or duck
  - Eye maid; her duck
  - Aye mate, her duck
  - I maid her duck
  - I’m aid her duck
  - I mate her duck
  - I’m ate her duck
  - I’m ate or duck
  - I mate or duck

Dealing with Ambiguity

• Four possible approaches:
  1. Tightly coupled interaction among processing levels; knowledge from other levels can help decide among choices at ambiguous levels.
  2. Pipeline processing that ignores ambiguity as it occurs and hopes that other levels can eliminate incorrect structures.
Dealing with Ambiguity

3. Probabilistic approaches based on making the most likely choices
4. Don’t do anything, maybe it won’t matter

Models and Algorithms

- By models we mean the formalisms that are used to capture the various kinds of linguistic knowledge we need.
- Algorithms are then used to manipulate the knowledge representations needed to tackle the task at hand.
Models

- State machines
- Rule-based approaches
- Logical formalisms
- Probabilistic models

Algorithms

- Many of the algorithms that we’ll study will turn out to be transducers; algorithms that take one kind of structure as input and output another.
- Unfortunately, ambiguity makes this process difficult. This leads us to employ algorithms that are designed to handle ambiguity of various kinds.
Paradigms

• In particular..
  • State-space search
    ▪ To manage the problem of making choices during processing when we lack the information needed to make the right choice
  • Dynamic programming
    ▪ To avoid having to redo work during the course of a state-space search
      • CKY, Earley, Minimum Edit Distance, Viterbi, Baum-Welch
  • Classifiers
    ▪ Machine learning based classifiers that are trained to make decisions based on features extracted from the local context

State Space Search

• States represent pairings of partially processed inputs with partially constructed representations.
• Goals are inputs paired with completed representations that satisfy some criteria.
• As with most interesting problems the spaces are normally too large to exhaustively explore.
  • We need heuristics to guide the search
  • Criteria to trim the space
Dynamic Programming

- Don’t do the same work over and over.
- Avoid this by building and making use of solutions to sub-problems that must be invariant across all parts of the space.