Special Topics: Natural Language Processing

Introduction

Adam Meyers
New York University
2016
Outline

• Grades, Exams, Policies, etc.
• Text Books and Suggested Reading
• A Survey of the Students
• Defining the Field
• CL Applications
• Types of Text Analysis used in CL
• A Practice Manual Annotation Task
• Summary and Syllabus
• Homework No. 1
Grades, Homework, Exams, Final Projects

• Grade Breakdown:
  – ¼ Homework + ¼ Midterm Exam + ¼ Final Exam + ¼ Final Project

• Homework
  – 7 or 8 Homework Assignments to be submitted through NYUClasses
  – 11 reading assignments (helpful for both homework and tests)

• Final Project
  – Sample Topics Available by October 12
  – Final Project Proposal Due November 16
    • counts as 1 homework
  – Student Presentations Dec 7, 12 & 13: 3 min + 1 min for questions
  – Final Written Version Due Dec 14
  – Group Projects Encouraged (participant roles should be spelled out)
Succeeding in This Class

• Experiment and Ask Questions
  – Work out examples from readings on paper
  – Try out and modify NLTK programs
    • break them, read the error messages, fix your bugs, repeat
• Homework: If you have trouble, try to state clearly what you do not understand, so the grader can answer questions and/or send me your questions by email and I can go over them in class.
  – You can get credit for getting right answers or clearly stating problems and identifying source of your confusion – you could uncover a valid criticism of the assignment and/or a more general problem in the field
• Midterm and Final: I will provide practice tests
• Final project: Do in stages: (1) Proposal; (2) Baseline System; (3) Final System
  – Start working on your final project at least 1 month before the end of class.
Policies

• **Late Homework**
  – **Natural Consequence**: You could fall behind, leading to lower marks on exams and the final project.
    • The midterm and final exam are partially based on what you learn by doing the homework
  – **Late HW is graded late** (at the grader's discretion, see natural consequences)
    • 1 point is taken off for lateness (on a 1-10 scale)
    • If you have a large backlog of HW and you hand it all in at the end of the term, there can be no guarantees that it will be graded at all. The graders are also students and have their own finals to worry about.

• **Missing Homework**: I include the top N-1 homeworks in grade, i.e., Average ((N-1 * A) + F) = A

• **Intellectual Integrity** (context dependent):
  – [http://www.cs.nyu.edu/webapps/content/academic/undergrad/academic_integrity](http://www.cs.nyu.edu/webapps/content/academic/undergrad/academic_integrity)
  – Usually, you may discuss HW with anyone, but your work should be your own.
    • If it is a problem, you should be prepared to solve it on your own after you submit your answer
    • If it is creative, 2 students should not have the same answers
    • Special Cases, e.g., experiments, where we test to see if people get same answer independently
  – Midterm/Final Exam – no help, other than explaining instructions or fixing errors in the phrasing of questions
  – Final Project
    • research = your own (but you can get “normal” advice)
    • other people can help with experiments, e.g., annotation
    • multi-person projects and other collaboration is encouraged (as will be discussed)
Basic Info: CSCI-UA.0480-006 Fall 2016

- **Website:** [http://cs.nyu.edu/courses/fall16/CSCI-UA.0480-006/](http://cs.nyu.edu/courses/fall16/CSCI-UA.0480-006/)
- **Class Room:** WWH 101
- **Schedule:** Monday and Wednesday 11:00AM—12:15AM
  - No Classes:
    - October 10 (Fall Recess)
    - November 23 (Thanksgiving Recess)
  - Exams:
    - Midterm: Wednesday, October 26, 2016
    - Final: Monday, December 19, 2016, 10:00AM—11:50AM
- **My office:** 719 Broadway, Rm 702
- **Office Hours:** Monday: 1:30-3PM or Thursday: 10:30-12PM or by appointment
- **My Email:** meyers@cs.nyu.edu
- **My Phone Number:** 212-998-3482

Computational Linguistics
Lecture 1
2016
Text Books

• SPEECH and LANGUAGE PROCESSING 2\textsuperscript{nd} Edition
  – By Daniel Jurafsky and James H. Martin
  – \url{http://www.cs.colorado.edu/~martin/slp.html}
  – Overview of the Field, explanations of techniques, algorithms, etc.

• Natural Language Processing with Python
  – By Steven Bird, Ewan Klein, and Edward Loper
  – \url{http://www.nltk.org/book} (look at the rest of the website also)
  – Book is available on line (or you can purchase it)
    • Online version may be more up-to-date then the paper version
  – Now available for both Python 2 and Python 3
    • Paper Version = Python 2
    • Electronic Version (with Python 3) being revised by authors
  – Downloadable open source programs to try out various computational linguistics tools and inspect their code
More Stuff to Read/Download, etc.

- Look at projects currently going on at NYU:
  - The Proteus website: http://nlp.cs.nyu.edu/
  - My website: http://nlp.cs.nyu.edu/people/meyers.html
    - Termolator – an open source terminology extraction tool
    - GLARF: processing tool written in Common Lisp (for linux)
    - NomBank: annotation project
    - COMLEX, NOMLEX: lexicon projects

- Other useful links:
  - Last term's NLP Class:
    - http://cs.nyu.edu/courses/spring16/CSCI-UA.0480-011/
  - Association for Computational Linguistics: http://aclweb.org/
  - ACL repository of conference papers in NLP: http://aclweb.org/anthology/
Some Pointers for Installing NLTK on your own Machine

• **Linux:** NLTK is easy to install in *linux*

• **Apple:** OK, but not as smooth as *linux*. To get all the bells and whistles, you may have to register as a developer

• **Windows:** There may be some limitations (last I checked), but most things relevant to this class will work.
  – I have not tested it, but Cygwin or AndLinux might be better for running NLTK

• **Python3 vs Python2:**
  – Python 2 – More popular, more libraries supported
  – Python 3 – Easier for processing non-ascii characters
    • You may need to use pip3 instead of pip when you install nltk
Computer Background Survey

• OS experience
  – UNIX experience?
    • Linux, Solaris, Using the Command line in Apple
    • Windows: Cygwin, Andlinux, …
  – How many people mostly use Windows?
    • Some UNIX platform is recommended
    • Many NLP resources work better/are easier to install/etc. in UNIX

• Programming Languages – which languages do people use?
  – Python
  – Java
  – C
  – Any variety of LISP: Common LISP, emacs LISP
  – Shell scripts

• UNIX utilities and script languages
  – grep
  – shell scripts, sed, awk
Linguistics Background Survey

• Syntax:
  – Descriptive Linguistics, e.g., comprehensive grammar of English
  – Chomskyan Linguistics?
  – Non-Chomskyan Frameworks
    • LFG, HPSG, Categorial Grammar, Dependency Grammar, Systemic Grammar, Other

• Phonetics, Phonology
  – Acoustics, Articulatory, Phonetics, Phonology, Intonation

• Discourse, Pragmatics
• Psycho-Linguistics
• Lexicography
• Historical
• Any Other Area
Role of Linguistic Theory in Computational Linguistics

- Framework = Language for Expressing Theory
- Theory = Set of Statements in Framework
- Different Theories/Frameworks are typically designed with different interests/biases/etc.
  - Chomskian Linguistics: Meta Grammar for all languages, set of primitives,
- Computational Linguistics is Applied Field of Study
  - Theories/Frameworks are important to the extent that they help make a successful application
  - Descriptive Adequacy is more important than Explanatory Adequacy
  - The authors of the answer key determine the framework
  - Some systems handle multiple theories/frameworks
- Frameworks that are popular in CL: Statistics-based Analysis (various), Dependency Grammar, Penn Treebank (based on 1980s Chomskian Linguistics), PropBank/Nombank (~ Relational Grammar), Frame Semantics (based on FrameNet), ...
- Only Broad Coverage Grammars are suitable, e.g., old theories with descriptive track records
- Proviso: there is a small niche within CL, in which researchers implement new theories
Defining Computational Linguistics

• AKA, Natural Language Processing (NLP), Language Engineering, ...

• **Domain**: The set of problems involving the interpretation and generation of human language text and speech

• **Properties**
  – As with applied science: the proof is in the pudding
  – Sometimes at odds with theoretical linguistics
    • Need not model human abilities and human methods
    • Need not correspond to published linguistic theories
    • But sometimes draws on one or both
  – Broad and changing domain influenced by available funding
CL Applications: Slide 1

• **Machine translation**
  – Methods are not at all based on how humans translate
  – Effective for gisting text, generating 1st draft translations, but not for high-level translation
  – Works better for “controlled languages” – technical manuals (Microsoft, Caterpillar, etc.)

• **Spoken Language**
  – dictation (IBM ViaVoice, Dragon Naturally Speaking)
  – Telephone-based customer support (phone mazes)

• **Information Retrieval**
  – Finding documents based on a query, e.g., Web Searches
CL Applications Slide 2

- Information Extraction
  - Dealtime, Google Products, Monster.com (job search)
  - Some open source tools:
    - https://opennlp.apache.org/
    - http://alias-i.com/lingpipe/
  - NYU
    - Some tools on website
      - http://nlp.cs.nyu.edu/projects/index.shtml#t-r-i
      - http://cs.nyu.edu/grishman/jet/jet.html
      - http://nlp.cs.nyu.edu/ice/
      - http://nlp.cs.nyu.edu/termolator/
    - Example from disease domain http://nlp.cs.nyu.edu/info-extr/biomedical-snapshot.jpg

- Question Answering
  - ask.com, Wolfram Alpha, MIT start: http://start.csail.mit.edu/

- Summarization: http://newsblaster.cs.columbia.edu/

- Spelling/Grammar Checking, etc. https://languagetool.org/
Types of Analysis

• Phonetics/Phonology: speech recognition and speech synthesis (not in this class)
  – We will focus on text analysis
  – Text does not represent some phonological features
  – Text has punctuation

• Syntactic/Semantic: sentence splitting, tokenization, pos tagging, chunking, parsing, predicate/argument structure, sense disambiguation

• Discourse: anaphora, discourse argument structure, sentiment analysis

• Other: multi-lingual processing (including MT), summarization, IE, etc.
Lowest Level Syntactic Processing (text)

- **Tokenization and Segmentation**
  - Given a sentence, determine the words or word-like units that it consists of:
    - *They announced in unison, “We don't agree with each other.”*
    - Tokenization: *They | announced | in | unison | , | “ | We | do | n't | agree | with | each | other | . | ”*
      - Controversial parts: *n't, each other*
      - NLTK command: `nltk.word_tokenize('this is a sentence')`

- **Part of Speech Tagging (modified PTB)**
  - Apply a set of part of speech tags to a set of tokens
    - *They/PRP announced/VBD in-IN unison/NN ,/PU “/PU We/PRP do/VBP n't/RB agree/VB with-IN each/DT other/JJ ./PU ”/PU*
  - NLTK command: `nltk.pos_tag(tokens)`
Low Level Syntactic Processing

- **Named Entity Tagging (with a little semantics)**
  - Mark boundaries of names of type PERSON, ORGANIZATION, FACILITY, GPE, LOCATION, …
  - `<ENAMEX TYPE="PERSON"> Adam Meyers</ENAMEX>` works for `<ENAMEX TYPE="ORGANIZATION">New York University</ENAMEX>`
  - `test_sentence = 'Adam Meyers works for New York University.'`
  - NLTK command: `nltk.chunk.ne_chunk(nltk.pos_tag(nltk.word_tokenize(test_sentence))`

- **Chunking -**
  - mark verb groups and/or noun groups, convenient approximations of syntactic units
  - `[NG The book] with [NG the blue cover] [VG will end up] on [NG the shelf].`
  - do not include “right modifiers”, like constituents derived in parsing (next slide)
  - NLTK:
    - `sentence = 'The book with the blue cover will end up on the shelf.'`
    - `chunks = r"""
      NG: {(<DT|JJ|NN>)*(<NN|NNS>)}
      VG: {<MD|VB|VBD|VBN|VBZ|VBP|VBG>*<VB|VBD|VBN|VBZ|VBP|VBG><RP>?>}
      """
    - `chunks_grammar = nltk.RegexpParser(chunks)`
    - `chunks_grammar.parse(nltk.pos_tag(nltk.word_tokenize(sentence)))`
Parsing: High Level Syntactic Processing

• (S (NP (DT the) (NN book))
  (PP (IN with))
  (NP (DT the)
    (JJ blue)
    (NN cover))))

  (VP (VBZ is)
    (PP (IN on)
      (NP (DT the) (NN shelf))))
Semantics – ish

• Semantics – A wide range of topics loosely referring to “meaning”

• Some Example Topics which may be part of Semantics (Next Few Slides)
  – Word Sense Disambiguation
  – Predicate Argument Structure
  – Anaphora
  – Discourse Argument Structure
WordNet Noun entry for bank

1. S: (n) bank (sloping land (especially the slope beside a body of water)) "they pulled the canoe up on the bank"; "he sat on the bank of the river and watched the currents"

2. S: (n) depository financial institution, bank, banking concern, banking company (a financial institution that accepts deposits and channels the money into lending activities) "he cashed a check at the bank"; "that bank holds the mortgage on my home"

3. S: (n) bank (a long ridge or pile) "a huge bank of earth"

4. S: (n) bank (an arrangement of similar objects in a row or in tiers) "he operated a bank of switches"

5. S: (n) bank (a supply or stock held in reserve for future use (especially in emergencies))

6. S: (n) bank (the funds held by a gambling house or the dealer in some gambling games) "he tried to break the bank at Monte Carlo"

7. S: (n) bank, cant, camber (a slope in the turn of a road or track; the outside is higher than the inside in order to reduce the effects of centrifugal force)

8. S: (n) savings bank, coin bank, money box, bank (a container (usually with a slot in the top) for keeping money at home) "the coin bank was empty"

9. S: (n) bank, bank building (a building in which the business of banking transacted) "the bank is on the corner of Nassau and Witherspoon"

10. S: (n) bank (a flight maneuver; aircraft tips laterally about its longitudinal axis (especially in turning)) "the plane went into a steep bank"
Word Sense Disambiguation

• For interesting characterizations of word senses (and relation between senses), use WordNet (online or download it)
  – wordnet.princeton.edu/

• Fewer than 10 obviously distinct senses of \textit{bank}, e.g.,
  – \textit{They took money out of the bank}.
  – \textit{The water flooded over the bank of the river}.

• Difficult sense disambiguation
  – Example: senses 2, 6 and 9 are arguably not distinct
  – Lexicographers are acutely aware of the merging vs. splitting problem of enumerating senses
  – CL systems usually collapse some WordNet distinctions
Predicate/Argument Structure

- For thousands of years, linguists have employed systems to characterize predictable paraphrases, e.g., Pāṇini, a Sanskrit linguist from the 4th Century BC
- In 21st Century CL, semantic role labeling is popular

![Diagram showing predicate/argument structure with examples: They were eaten by a giant clam and John took a walk to the store.](image-url)
Anaphora

• Coreference
  – Though **Big Blue** won the contract, this official is suspicious of **IBM**.
  – **Mary** could not believe what **she** heard.

• Other Varieties
  – John ate a **sandwich** and Mary ate **one** also.
  – The **amusement park** is very dangerous. The **gate** has sharp edges. The **rides** have not been inspected for years.
  – This **book** is valuable, but the **other book** is not.
Discourse Argument Structure

- Adverbs, Subordinate/Coordinate Conjunctions, among other words link clauses

They wanted to steal the diamonds.

However, they did not possess the necessary skills.

ARG1

ARG2
Role of Manual Annotation

• Used to create, test and fine-tune task definitions/guidelines.
  – For a task to be well-defined, several annotators must agree on classification most of the time.
  – If humans cannot agree, it is unlikely that a computer can do the task at all.
  – Popular, but imperfect measurement of agreement:
    • \[ \text{Kappa} = \frac{\text{Percent (Actual Agreement)} - \text{Prob (Chance Agreement)}}{1 - \text{Prob (Chance Agreement)}} \]

• Used to create answer keys to score system output
  – One set of measures are: recall, precision and f-score

  \[ \text{Recall} = \frac{|\text{Correct}|}{|\text{Answer Key}|} \quad \text{Precision} = \frac{|\text{Correct}|}{|\text{System Output}|} \quad \text{F – Score} = \frac{1}{2} \left( \frac{1}{\text{Precision}} + \frac{1}{\text{Recall}} \right) \]
Manual Annotation in Supervised Statistical ML

• Divide the corpus into sub-corpora
  – A training corpus is used to acquire statistical patterns
  – A test corpus is used to measure system performance
  – A development corpus is similar to a test corpus
    • Systems are “tuned” to get better results on the dev corpus
    • Test corpora are only used infrequently to insure accuracy/fairness
      – The system should not be tuned to get better results

• More annotated text often yield better results

• Different genres may have different properties
  – Systems can “train” separately on different genres
  – Systems can “train” on one diverse corpus
Summary

• Computational Linguistics is an applied discipline with an increasingly large inventory of applications.

• A wide variety of levels of analysis are used to implement these applications.
  – Many, but not all of these levels are derived from or inspired by theoretical linguistics

• One popular paradigm for producing an analysis automatically involves manually annotating text
Syllabus: Subset of these Topics

- Introduction (today)
- Formal Languages and Transducers
- Corpus Annotation
- English Syntax and Parsing
- POS Tagging and Hidden Markov Models
- Named Entities and Machine Learning
- Lexical Semantics and Semantic Role Labeling
- Information Extraction: Entities, Relations, Events, Time
- Anaphora and Coreference Resolution
- Feature Structures and Representing Multiple Phenomena
- Machine Translation
Specifications for Annotation
Task in the Homework

- Adjectives occur in two main positions
  - Attributive
    - Adjectives precede nouns that they modify
    - Ex: *the big sandwich*
  - Predicative
    - A noun phrase is linked to an adjective by predication
    - Ex: *The sandwich is big*
    - Ex: *I made the sandwich big*

- Adjectives can have three morphological forms
  - Normal: *big*
  - Comparative: *bigger*
  - Superlative: *biggest*

- Adjectives should not be confused with nouns
  - Ex: *The truck salesman*
    - *truck* is not an adjective
    - Nouns, not adjectives can occur in the plural, e.g., *trucks*
    - Nouns, not adjectives are modified by determiners like *the* or *a*, e.g., *a truck, the truck*
Adjective Specifications: Slide 2

• Difficult cases
  – A word can have distinct meanings as an adjective and as a noun
    • They are studying for the **final**. (Noun)
    • This was their **final** attempt. (Adjective)
  – Adjectives may be used as nouns with an adjective + one meaning
    • They exploit the **poor**.
      – **poor** means something like poor people
      – **poor** is used much more frequently as an adjective
      – Compare with predicative position
        » They are **poor**
        » If it was a noun, poor would be plural (**poors**)
  – Frequency is an issue, e.g., assume color words are adjectives because their noun-like uses are rare
    • I really love this **red** (noun) vs That clown nose is **red** (adj) or The **red** nose

• Determiners are not adjectives (they occur before adjectives)
  – **such several one most more many less few enough both all your those**
  – **this these their the that some our no neither my its his her every either**
  – **each any another an a**

• Cardinal Numbers ARE NOT adjectives: **one, two, three, ...** (they are determiners)

• Ordinal Numbers ARE adjectives: **first, second, third, ...**
Adjective Specifications Slide 3

• Adjectives can be modified by other words:
  – light red, very hungry, quite upset

• In attributive position, they occur after the determiners and before any nouns, e.g.,
  – the hairy mountain gorilla
    • the = determiner
    • hairy = adjective
    • mountain = noun
    • gorilla = noun
Demo of Mae Annotation Tool

- nMAE – Giancarlo Lee's (NYU) modification of Amber Stubb's (Simmons College) Mae annotation tool
- Put all files in the same directory:
  - nMAE.jar
  - adjective.dtd
  - FullMorph.txt
  - state_of_the_union.txt
  - alm6888_state_of_the_union.xml
    - sample with Instructor annotation of first 2 paragraphs
- `java -jar nMAE.jar` (or double click on nMAE.jar)
- File → load → dtd → adjective.dtd (dtd file defines the task – write a different dtd file for a different task)
- File → load → state_of_the_union.txt
- To mark an adjective:
  - Drag left mouse over adjective and click, only one choice
  - Change attributes in list of adjectives (left click and select on slot)
  - Click yes on done
    - go to repeated cases by double clicking on new row created
    - Change the new row if necessary
    - Or delete the new row by right click and select
- Periodically save:
  - File → Save File as XML – choose a name that includes your ID number, e.g., alm6888_state_of_the_union.xml
Homework and Readings

  - Readings
  - Installation Instructions
  - Files to download
  - Annotation Specifications (same as previous slides)
  - Optional Independent Work
  - Due September 14 and will be discussed September 19