Special Topics: Natural Language Processing

Introduction

Adam Meyers
New York University
2018
Outline

• Grades, Exams, Policies, etc.
• Text Books and Suggested Reading
• Discussion of Background Knowledge
• Defining the Field
• CL Applications
• Types of Text Analysis used in CL
• A Practice Manual Annotation Task
• Summary and Syllabus
• Homework No. 1
Basic Info

• Website: http://cs.nyu.edu/courses/spring18/CSCI-UA.0480-009/
• Class Room: 60 Fifth Avenue C10
• Schedule: Tuesday and Thursday 8:00AM—9:50AM
  – No Classes:
    • Tuesday March 13 and Thursday March 15 (Spring Break)
  – Important Dates:
    • Midterm: Tuesday March 20
    • Final Project Proposal: Tuesday March 27
    • Final Project First Draft: Tuesday April 17
    • Final Project Presentations: Thursday April 27 and Tuesday May 1
    • Final Project Final Version: May 15, 2018 (the day of the final exam)
• My office: 60 Fifth Avenue, Rm 301
• Office Hours: Mon: 1:30-3PM or Thurs: 10:30-12AM or by appointment
• My Email: meyers@cs.nyu.edu
• My Phone Number: 212-998-3482
Changes from Last Semester

• Elimination of Final Exam and 1 (out of 7) Homework Assignments
• Greater Emphasis on Final Project
• Different Order of Topics
• This is an Experiment
  – If aspects of these policies do not seem to be working, I may make some adjustments mid-semester, but I will avoid making large changes (until next term).
Grade $= \frac{1}{3} \text{ HW} + \frac{1}{3} \text{ Midterm} + \frac{1}{3} \text{ Final Project}$

- **Homework** – 6 Assignments to Submit through NYUClasses
  - 1 Annotation Assignment, 4 Programming Assignments and Final Project Proposal
  - Based on Assigned Readings, Lectures and Your Own Research
- **Midterm**
  - Covers the Lectures, Reading Material and HW for the first $\frac{1}{2}$ Semester
- **Final Project**
  - Sample Topics Available by 12th class (March 1)
  - Final Project Proposal Due by 17th class (March 27): Counts as 1 homework
  - First Draft of Final Project Due 23rd class (April 17)
  - Student Presentations Class 26 and 27: 3 min + 1 min for questions
  - Final Written Version due on May 15, 2018 (the day of the final exam)
  - Group Projects Encouraged (participant roles should be spelled out)
Policies

- **Late Homework**
  - **Natural Consequence**: You could fall behind, leading to lower marks on exams and the final project.
    - The midterm is partially based on what you learn by doing the homework
    - Doing homework assignments after the due date may take away time you could be spending on the final project.
    - Late homework is graded late at the grader's discretion
  - **Additional Consequences and Constraints**:
    - 1 point is taken off for lateness (on a 1-10 scale).
    - Please contact me before submitting homework more than 3 weeks late.
    - There is no guarantee that homework submitted after May 1 will be graded at all.

- **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)
  - You may discuss HW with anyone, but your work should be your own.
    - You should be prepared to solve it on your own after you submit your answer
    - If it is a creative assignment, 2 students should not have the same answers
    - There are special cases, e.g., annotation experiments, where we test to see if people get same answer independently
  - Midterm – I will help you during the midterm by explaining instructions or fixing errors in the phrasing of questions, but nobody else should help you.
  - Final Project
    - Testeach = 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)
Types of Final Projects

• Programming project including:
  – Write a program
  – A set of experiments evaluating the program
  – A write-up describing the program and the evaluation, as well as, discussion relating it to previous work described in articles from NLP Conferences and Workshops

• Resource project
  – Create a Resource for NLP (corpus annotation, a lexicon etc.)
  – A set of experiments evaluating the annotation
  – A write-up describing the annotation and evaluation, as well as, discussion relating it to previous work described in articles from NLP Conferences and Workshops

• Survey paper
  – Read lots of papers about a particular subarea of NLP
  – Write a paper summarize that subarea in light of your perspective on that topic, i.e., specialize in some interesting way.
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: I will provide at least one practice test. Previous midterms are available as well. However, there may cover different (but overlapping) material.

• Final project:
  – Do in stages: (1) Proposal; (2) First Draft; (3) In-class Talks (4) Final Draft
  – Work on it steadily over the 2nd half of the semester. No homework during the 2nd half, i.e., no reason not to make progress on a weekly basis.
Purposes of NYUClasses

- Distributing resources subject to licencing conditions (and NYU has the license)
  - Details on Website and in Readme file in Resources Section of NYUClasses
  - Material without restrictive licenses is distributed through the class website
- Submitting homework assignments and final projects (proposals, first draft, final draft)
- Grading Details at the End of the Semester
  - I make the final grades “auditable”
  - At the end of the semester I advise you about how your grade was calculated
Text Books

• **SPEECH and LANGUAGE PROCESSING 2nd Edition**
  – By Daniel Jurafsky and James H. Martin
  – Overview of the Field, explanations of techniques, algorithms, etc.
  – Parts of the 3rd edition draft are available online:
    • [https://web.stanford.edu/~jurafsky/slp3/](https://web.stanford.edu/~jurafsky/slp3/)

• Natural Language Processing with Python
  – By Steven Bird, Ewan Klein, and Edward Loper
  – Book is available online (or you can purchase a paper version)
    • Online version may be more up-to-date than the paper version
  – Available for both Python 2 and Python 3 (2.7, 3.4, 3.5)
    • I have had trouble with Python 3.6 and 64K versions in the past – I do not know if these issues have been fixed yet.
    • 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
  - Web of Law – current research, including student researchers
    - Final projects using this data is encouraged

- Other useful links:
  - Last term's NLP Class:
    - http://cs.nyu.edu/courses/fall17/CSCI-UA.0480-006/
  - Association for Computational Linguistics: http://aclweb.org/
  - ACL repository of conference papers in NLP: http://aclweb.org/anthology/
Linguistic Data Consortium

• Their catalog: https://catalog.ldc.upenn.edu/

• They have data which may be useful for your final project and NYU has a license

• If you need LDC data, contact me or the nyulibrary (this is a relatively new thing)

• LDC has different types of corpora (text), some of it manually annotated.
  – Some data is already available elsewhere: freely downloadable or via the class NYUClasses page
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 recently. Unix recommended if you have trouble – details to follow.

- **Python3 vs Python2**:
  - Python 2 – Probably more popular, more libraries supported
  - Python 3 – Easier for processing non-ascii characters
  - Will Python 2 be phased out?
    - Allegedly, the core Python development team will end support in 2020
    - There seems to be substantial community support behind Python 2
Computer and Math Background

• UNIX
  – Many NLP resources work better/are easier to install/etc. in UNIX
  – Most common Unix platforms today: Apple (BSD) and Linux (preferred)
  – Unix Emulators in Windows: Cygwin, a VM running linux or the options described here:
    • https://www.howtogeek.com/170870/5-ways-to-run-linux-software-on-windows/
  – Any student in the class can have a CS linux account to log in to (e.g., by ssh) – just ask me by email.
  – NYU has servers for various purposes, including GPU-computing (e.g., deep learning)
    • https://cims.nyu.edu/webapps/content/systems/resources/computeservers
• UNIX utilities and script languages: grep, shell scripts, sed, awk, etc.
• Programming Languages
  – Some experience with Python helpful for NLTK
  – Some homework assignments use JAVA, but possible to use as black box
  – Any programming language OK for some assignments, e.g., Final Project
  – Common Lisp makes some NLP data structures easier to process, but few people use this now
• Experience writing large programs helpful
  – Programs that may take minutes or hours (not seconds) to run
• Math: Probability and Statistics are especially useful
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
    • Sometimes draws on linguistic theories and/or studies of human processing
  – 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/
  - Tools on NYU website include:
    - 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, part-of-speech tagging, chunking, parsing, predicate/argument structure, sense disambiguation, “semantic parsing”

• **Discourse:** anaphora, discourse argument structure, sentiment analysis

• **Other:** multi-lingual processing (including MT), summarization, IE, etc.
• **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
  – “Semantic Parsing”
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 bank, e.g.,
  – They took money out of the bank.
  – 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

They were eaten by a giant clam

John took a walk to the store
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.

ARG1

However, they did not possess the necessary skills.

ARG2
Semantic Parsing, e.g., GLARF

- Means different things to different researchers, but my version of semantic parsing is called GLARF:
  - http://nlp.cs.nyu.edu/meyers/GLARF.html
- One representation of the sentence that includes as much information as possible: lexical categories, predicate argument structure, discourse annotation, etc.
- Next slide is a representation of the sentence:
  - Afterwards, she decided to perform the operation.
  - When it occurs after the sentence: The doctor ran some tests
GLARF

(S (ADV (ADVP (HEAD (ADVX (HEAD (RB Afterwards 0))))
  (P-ARG1 (S (EC-TYPE PB) (INDEX 0+0))
  (P-ARG2 (S (EC-TYPE PB) (INDEX 0))
  (RELATION-TYPE AFTER)))
  (INDEX 1) (POINTER 0:1))))
(PUNCTUATION (, , 1))
(SBJ (NP (HEAD (PRP she 2)) (INDEX 2) (POINTER 2:1))))
(PR (VP (HEAD (VG (HEAD (VBD decided 3))
  (P-ARG0 (NP (EC-TYPE PB) (INDEX 2)))
  (P-ARG1 (S (EC-TYPE PB) (INDEX 5))))
  (P-ARGM-TMP (ADVP (EC-TYPE PB) (INDEX 1)))
  (SEM-TENSE PAST)))
(COMP (S (L-SBJ (NP (EC-TYPE INF) (INDEX 2)))
  (PRD (VP (HEAD (VG (AUX (TO to 4))
    (HEAD (VB perform 5))
    (P-ARG0 (NP (EC-TYPE PB) (INDEX 2)))
    (P-ARG1 (NP (EC-TYPE PB) (INDEX 4)))
    (INDEX 3))))
  (OBJ (NP (Q-POS (DT the 6))
    (HEAD (NX (HEAD (NN operation 7)))
    (P-SUPPORT (VG (EC-TYPE PB) (INDEX 3)))
    (P-ARG0 (NP (EC-TYPE PB) (INDEX 2)))))
  (INDEX 4) (POINTER 6:1))
  (PB-POINTER 4:1))
  (POINTER 4:2) (INDEX 5))
  (POINTER 3:1))
  (PUNCTUATION (.. 8) (POINTER 0:2) (TREE-NUM 1) (INDEX 0))

Computational Linguistics
Lecture 1
2018
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.

  - $\text{Recall} = \frac{|\text{Correct}|}{|\text{Answer Key}|}$
  - $\text{Precision} = \frac{|\text{Correct}|}{|\text{System Output}|}$
  - $\text{F – Score} = \frac{1}{2} \left( \frac{1}{\text{Precision}} + \frac{1}{\text{Recall}} \right)$
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
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
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
Specifications for Homework Annotation Task

- 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, sometimes with arguments of the adjective
    - Ex: The sandwich is big
    - Ex: I made the sandwich big
    - Ex: Philosophy may seem difficult to understand

- Adjectives can have three morphological forms
  - Normal: big
  - Comparative: bigger (Do not mark multi-word non -er cases as comparative, e.g., “more angry”)
  - Superlative: biggest (Similarly, do not mark cases like “most significant” as superlative)
  - This is a morphological classification of a word which pertains to suffixes and prefixes
  - It is not a semantic classification and it is not a classification of word sequences (more than 1 word)

- 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

- 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(s)* meaning
  - *They exploit the poor.*
    - *poor* means something like poor ones or 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*

- Idiomatic Constructions involving verbs and adjectives – if the word occurs elsewhere as an adjective and there is no other obvious part of speech, it is probably a predicative adjective
  - *Mary fell ill. John went ballistic.*

- Idioms and near idioms with literal interpretations that favor adjective marking – mark as adjectives
  - *red herring, cold shoulder, …*

- Present participles (*flowing, flying, …*) and past participles (*understood, fixed, …*) of verbs
  - Mark as adjective in attributive position (*flowing hair, fixed position, …*)
  - Only mark as adjective in predicative position if an adjective meaning is clear, e.g., understanding means two different things in these examples – only the first meaning is adjectival. Other clues are modifiers (*very vs easily*)
    - *John was very understanding*
    - *John was understanding the lecture easily.*
Adjective Specifications Slide 3

• Determiners (see list) 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, …

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

• In attributive position, they occur after any determiners and before any
nouns, e.g.,
  – the hairy mountain gorilla

• the = determiner
• hairy = adjective
• mountain = noun
• gorilla = noun
Demo of Mae Annotation Tool

• MAE – Amber Stubb's (Simmons College) Mae annotation tool
  - https://github.com/amber-stubbs/mae-annotation

• Put all files in the same directory:
  - mae_v0.9.6.jar
  - adjective.dtd
  - state_of_the_union.txt
  - AM_state_of_the_union.xml
    - sample with Instructor annotation of first 2 paragraphs

• java -jar mae_v0.9.6.jar (or double click)

• 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
  - You can also load your saved xml file

• 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 and Self-Study
  – Chapter 1 in Jurafsky and Martin
  – Install NLTK, Read Chapter 1 and follow examples
  – Optional: Read through the full Penn Treebank Part of Speech tagset description:
    https://catalog.ldc.upenn.edu/docs/LDC99T42/tagguid1.pdf

• Written Assignment:
  – http://cs.nyu.edu/courses/spring18/CSCI-UA.0480-009/homework1.html