Outline

• Text Corpora in NLP
• Corpus Selection
• Corpus Annotation:
  – Purpose
  – Representation Issues
  – Linguistic Methods
  – Measuring Quality
• Homework No. 3
Characters, Encodings, Etc.

- A Text Corpus is a set of texts
- Corpora can be derived in different ways
  - Text that was originally electronic (published, letters, etc.)
    - Does it include “non-standard” characters?
  - Transcripts of spoken language
    - No punctuation
    - Possible representation of pauses
    - Possibly including pauses and false starts
  - Optical Character Recognition (with errors)
- Encodings (mappings between bits and characters)
  - Old Standards (English): ASCII (less than 1 byte), ISO-8859 (2 bytes)
  - New standards UTF-8 (back-compat w/ASCII) and UTF-16
    - More characters/alphabets – 4 or 16 bytes
  - Other encodings: GB (e.g., Chinese), EUC (e.g., Japanese)
Types of Texts

• “Genre” divides text into types along several dimensions
  – **Register?** (socio-ling division by social setting) : Fiction, News, Magazine, Scholarly Article, Legal Documents, Correspondence, Email, Discussion Groups, Twitter, Text Messages, Phone Calls, Instructions, Oral Narratives, Webpages
  – **Topic**: Sports, Games, Art, Natural Science, Social Science, Business, Fiction, Literary Criticism, …

• Spoken Language Transcripts have special properties (written language is the default)
  – Differences in Basic Units
    • Pauses/intonation, but no punctuation/capitalization
      – If transcribed at all, encoding is not as standard
  – Additional lexical items, syntactic phenomena
    • Disfluencies: false starts, stutters, ..
    • “uh”, “um”, “like”, ....
Choosing a Corpus for a Project

• Specialize in a single type of corpus
  – Idealization of language phenomenon
    • If noted, this is normal for academic studies
  – Particular corpus is appropriate for your project
    • A telephone Question Answer system → corpus of phone conversations

• A “Diverse” Corpus
  – For development of versatile system
  – To focus on common features of different genres
  – Keep corpora separate & focus on adaptability of system

• Your own Corpus or an Existing Standardized Corpus
  – Own corpus requires preparation, but will be suitable for your needs
    • Removing unwanted fields (tables), formatting codes, …
  – Standard/Shared Corpus: Next Slide
Standardized/Shared Corpora

- Why have shared or standardized Corpora?
  - Opportunities for Comparison and Collaboration
  - Use Other's Expertise/Avoid Duplicate Effort

- Brown Corpus (Kucera and Frances 1967)
  - 1 million words, sort of open source now
  - “balanced” (“diverse” is easier to define)
  - prose fiction, poetry, news, general interest, government documents, biography, ...

- Work using corpora flourished starting in the 1990s
  - Mostly government sponsored, mostly newspaper corpora
    - Wall Street Journal Corpus, incl Penn Treebank (1 million words)
      - Licensed by Linguistic Data Consortium
    - Depends on what was widely available
      - Hansard Corpus – Canadian French/English Parliamentary Proceedings

- Return to “Diverse” corpora
  - British National Corpus (BNC) – 100 million words, 1994
  - American National Corpus, incl Open American National Corpus (OANC) 2004 & ongoing
    - 21 million words (and growing) including (15 million words in OANC)
Statistical Info Derivable from Corpora (without Annotation)

- **Frequency:**
  - words: *eat, ate, cats, cat, Mary, because, ...*
  - base forms: *eat, cat, Mary, because, ...*
  - characters: *a, e, i, z, q, &, ., 5, 3, ?, @, ..*

- **Examples of Higher Level Statistics:**
  - Frequency of bi-grams: *ate the, the cat, house was, ...*
    - tri-grams, 4-grams, 5-grams, ... N-grams
  - **TF-IDF:** Term Freq × Inverse Document Freq
    - TF = Frequency of term in corpus
    - IDF = Num of Docs ÷ Num of Docs containing term
    - Examples: 100 documents, 100 instances of the word *cat*
      - If all in same document: $100 \times 100/1 = 10,000$
      - If each in a different document: $100 \times 100/100 = 100$
Statistical Info Derivable from Corpora (without Annotation)

• Frequency:
  – words: eat, ate, cats, cat, Mary, because, ...
  – base forms: eat, cat, Mary, because, ...
  – characters: a, e, i, z, q, &, ., 5, 3, ?, @, ..

• Examples of Higher Level Statistics:
  – Frequency of bi-grams: ate the, the cat, house was, ...
    • tri-grams, 4-grams, 5-grams, … N-grams
  – TF-IDF: Term Freq \times \text{Inverse Document Freq}
    • TF = Frequency of term in corpus
    • IDF = \text{Num of Docs} \div \text{Num of Docs containing term}
    • Examples: 100 documents, 100 instances of the word cat
      – If all in same document: 100 \times 100/1 = 10,000
      – If each in a different document: 100 \times 100/100 = 100
Multi-lingual Corpora

• Parallel Corpora: bi-texts, tri-texts, etc.
  – 2 (or more corpora), such that corresponding segments are (literal) translations of each other
  – Useful for Machine Translation
  – Ex: Hansard Corpus

• Comparable Corpora
  – 2 (or more corpora) about similar/same topics, e.g., Wikipedia articles in multiple languages
Role of Manual Annotation in CL

• Together, Annotation and Specifications define a task
  – Can be used to “score” the output of any type of system

• For Supervised Machine Learning, Corpus is Divided
  – 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 used infrequently and system should not be tuned to get better results

• More annotated text often yield more effective patterns

• Different genres may have different properties
  – Systems can “train” separately on different genres
  – Systems can “train” on one diverse corpus
Annotation by Directly Marking Text

• Example: The Penn Treebank

• Input: This is a sentence.

• Output: (S (NP (DT This)))
  (VP (VBZ is))
  (NP (DT a))
  (NN sentence))
  (. .))

• Can be difficult to align original text with the annotation
  – Spaces, newlines, etc. not explicitly represented
  – Words --> tokens not always obvious
    • cannot --> can/MD not/RB
    • 'Tis → T-/PRP is/VBZ
    • fearlast → fear/NN last/JJ
  – token standardization, typos and other accidental changes
Encoding Annotation with a Markup Language

• Input: *This is a sentence.*
• Output: `<S><NP><DT>This</DT></NP> <VP><VBZ>is</VBZ> <NP><DT>a</DT> <NN>sentence</NN></NP><VP><.></S>`
  – (all on one line, preserving spaces)

• Markup language
  – Markup languages are designed to add information to text and typically distinguish beginning and ending tags `<X>` vs. `</X>`
  – Examples
    • HTML – language for website creation
    • XML, SGML – standards for more specific markup languages

• Programs often treat text and markup separately, e.g., turn markup into instructions (text color = red, bold, underline, italic, hyperlink, ...).
  – Example program: web browser treats html markup as instructions
• Annotation is usually designed so deleting the markup will remove all changes
  – sed 's/<[^>]*>// annotated_file > copy_of_original_file
  – diff original_file copy_of_original_file
• Markup relies on assumption that certain characters will not appear in the original text (< and >)
  – Suppose the corpus included the sentence: “I used an “<NP>” tag today”
  – To handle this special characters are often substituted, e.g., html uses the following codes for ampersands and greater than signs
    • &amp;
    • &gt;
  – See for example http://rabbit.eng.miami.edu/info/htmlchars.html
  – Same/similar codes are often used in non-html text for NLP purposes
  – This adds a layer of complexity if one wants to compare (e.g., align) the annotated version with the original text.
Offset Annotation

• Many newer annotation frameworks use annotation that “points” to the original file
  – There is a file of plain text containing the words, sentences, etc. being classified.
  – 1 or more annotation files “point” to positions in the original file by means of character offsets from the beginning of the file.

• For example, a tag of the form:
  – <S :start 0 end: 57> could mean that there is a sentence beginning at the start of the file and ending 57 characters after the start of the file.
  – As in many programming environments, positions in strings are before and after characters and begin with 0, e.g.,
    • the python slice: 'This string'[0:4] selects the substring between 0 and 4, assuming: 0 T h i s 1 2 3 4 5 6 7 8 9 10 11
Offset Annotation – Slide 2

• Overcomes the shortcomings of other methods
  – No special characters are needed
  – Relation to original text transparent
  – Multiple Annotations with the Same Scheme
    • Easy to Compare
  – Multiple Annotations with Different Schemes
    • Easier to compare, combine, etc.

• Difficult to read without programs (visualization tools, tools that write-out inline tag versions, etc.)
Annotation of Annotation

• Annotation Often Performed in Layers
  – One Project (or phase) Annotates Constituents
  – Another Project (or phase) Annotates Relationships Between Those Constituents

• Typical Cases:
  – Coreference:
    • Constituents X and Y are “mentions” of one Entity
  – Argument Structure
    • Predicate is in relation R with X as ARG1 and Y as ARG2

• 2 Layers of Annotation for: *John and Mary said that they were leaving.*
  – NP₁ = *[John and Mary]*, verb₁ = *said*, NP₂ = *[they]*, S₁ = *[that they were leaving]*
  – Coref(NP₁, NP₂), ARG0(verb₁, NP₁), ARG1(verb₁, S₁)

• Examples of Projects: ACE, Penn Treebank + PropBank, NomBank and PDTB
Annotation Entry Tools

• Help humans create computationally viable annotation
  – simulate inline annotation, while creating offset annotation

• Well-formedness
  – Only Legal Labels are permitted
  – Other Constraints can be hard-coded (e.g., distance)
  – Constraints can be automated
  – Warning statements can be included for “unusual” labelings

• Ease of Annotation
  – Specification Help Menus Can be Included
  – System can automatically propose next item
  – Common options can be automated, e.g., previous tags for particular strings can be proposed by system
The MAE annotation tool

- Original (Amber Stubbs at Brandeis):
  - http://code.google.com/p/mae-annotation/
- Alternative Version (modified at NYU by Giancarlo Lee):
  - http://nlp.cs.nyu.edu/meyers/IE_TECH_NYU.html

- java -jar mae.jar
- Write dtd file: specifications for annotation
- Load txt file and create xml file
- Process
  - Mae separates the document into 2 XML fields:
    - Copy of original text between: “<TEXT><![CDATA[“ and “”]]></TEXT>”
    - Annotation between <TAGS> and </TAGS>
- Annotation of entities is offset annotation
- Annotation of relations: refers to entity annotation
AttributionTask Example

• Let's do a little bit of sample “AttributionTask”
  – Load dtd file
  – Load file
  – A little bit of annotation (the description task makes more sense to me than the action task)

• Let's look at the output file in emacs (my preferred text editor)

• In this output, character positions begin at the end of [CDATA[
  – i.e., = 0

• Ctrl-U N – does following command N times
  – Ctrl-u N Ctrl-f – moves forward N spaces

• The relation (ATTRBIUTION) refer to the IDs of the entities: COMMUNICATOR and MESSAGE

• Each annotated tag has several feature=value pairs
  – Some are calculated by the program start/end
  – Others we added in explicitly (function/type/comment)
AttributionTask Example

- Let's do a little bit of sample “AttributionTask”
  - Load dtd file
  - Load file
  - A little bit of annotation (the description task makes more sense to me than the action task)

- Let's look at the output file in emacs (my preferred text editor)

- In this output, character positions begin at the end of [CDATA[
  - i.e., = 0

- Ctrl-U N – does following command N times
  - Ctrl-u N Ctrl-f – moves forward N spaces

- The relation (ATTRBIUTION) refer to the IDs of the entities: COMMUNICATOR and MESSAGE

- Each annotated tag has several feature=value pairs
  - Some are calculated by the program start/end
  - Others we added in explicitly (function/type/comment)
Now Let's Look at the Penn Treebank and NomBank

- Penn Treebank: wsj_0003.mrg
  - In emacs, Cntrl-Meta-B and Cntrl-Meta-N are useful for finding corresponding brackets particularly in lisp-mode

- NomBank (and PropBank): wsj_0003.nombank
  - Identifies nodes in Penn Treebank Trees
    - Token:length-of-path-from-first-leaf
  - File = wsj_0003
  - Tree = 10 (11th tree because count starts with 0)
  - predicate *amount*(s) = token 11 (staring with 0)
  - sense/roleset number 01 – see lexical entry
  - ARG1 = (NP-SBJ-1 (NN asbestos)) as connected to its empty category
  - Support Chain = *used* + *in* (tokens 7 and 9)
Designing Content Component of Annotation Task

• Goals:
  – Task must describe desired phenomena
  – Humans must be able to make distinctions consistently

• Write detailed specs and test them on data
  – Use multiple annotators
  – Do annotators agree N %
    • Easy task: N>90%
    • Medium Task: N>85%
    • Difficult Task: N>70%, ...
  – Annotator Agreement is Upper Bound for System Output Quality
    • Different levels of agreement may be required for different applications

• If results are insufficient, revise Specs and test new Specs again
  – Repeat Until Results are good enough for your purpose
Measuring Annotation Quality

- Popular, but imperfect measurement of agreement:

  \[
  \text{Kappa} = \frac{\text{Percent (Actual Agreement)} - \text{Prob(Chance Agreement)}}{1 - \text{Prob(Chance Agreement)}}
  \]

  - Kappa works provided it is possible to estimate “chance agreement”

- For POS tagging each token gets exactly one tag. So estimates can be based on:
  - tags assigned to previous instances of token
  - tags assigned to tokens in general

- Multiply annotated data can be adjudicated and then each annotator can be scored against the corrected annotation. These same scores are often used for system evaluations:

  \[
  \text{Recall} = \frac{|\text{Correct}|}{|\text{Answer Key}|} \quad \text{Precision} = \frac{|\text{Correct}|}{|\text{System Output}|} \quad F - \text{Score} = \frac{1}{2} \times \left( \frac{1}{\text{Precision}} + \frac{1}{\text{Recall}} \right)
  \]
Annotation Tasks Vary in Difficulty

- Penn Treebank Part of Speech Tagging
  - Approximately 97% accuracy/agreement
  - Annotation = Fast process
- Penn Treebank Bracketing Annotation
  - Mid 90s? (a guess)
  - Now mostly by one experienced annotator (Ann Bies)
- PropBank – Approximately 93%
  - About 1 instance per minute
- NomBank – Approximately 85%
  - About 1 instance per 2 minutes
- Temporal Relations – (big variation, approx 75%)
- Sentiment Annotation (about 75%)
Who Should Annotate?

• Most Common for Difficult Annotation
  – Linguistics Academics: PostDocs and Students
  – Penn Treebank: Ann Bies
  – Other Experts: Classics students
  – Researchers (small projects)
  – Domain Experts (biology, physics, etc.)

• Crowd Sourcing
  – For easier annotation tasks
  – Some research breaking down hard tasks into sequences of easy ones
Crowd Sourcing

• Unknown annotators contribute via a web browser
• Tasks formulated so non-experts can do OK
  – break down decisions into multiple choice questions
  – use qualification tests
  – do more annotation and filter through consensus
• Amazon Turk: currently the most common conduit
  – Inexpensive (including Amazon's commission)
• Some People have set up their own sites, e.g.:
  – https://anawiki.essex.ac.uk/phrasedetectives/
• Limitation: difficult to formulate sophisticated tasks for crowd sourcing
• If trend continues, may end up changing the role of linguists in annotation projects (e.g., linguists may design tasks rather than annotate)
URLs for Corpora w/English Bias

• Organizations that distribute corpora (and other resources) for fees
  – Linguistic Data Consortium: https://www.ldc.upenn.edu/
• The British National Corpus: http://www.natcorp.ox.ac.uk/
• American National Corpus (including OANC):
  – http://www.americannationalcorpus.org/
• The Brown Corpus (also through NLTK)
  – http://www.hit.uib.no/icame/brown/bcm.html
  – https://archive.org/details/BrownCorpus
• PubMed Corpus of Scientific Abstracts: http://www.americancorpus.org/
• Links to more links: http://www.americancorpus.org/
Annotation Project URLs w/ English Bias

- Examples of Shared Tasks with Associated Corpora & Annotation
  - Automatic Content Extraction: Coreference, Named Entities, Relations, Events, English, Arabic, Chinese, Spanish (little bit) – organized by US government
    - https://www.ldc.upenn.edu/collaborations/past-projects/ace
  - CONLL (yearly since 1997, diverse, internationally organized)
    - http://ifarm.nl/signll/conll/
    - I was on the committee for the 2008 & 2009 tasks
  - BIONLP (yearly IE task for biological texts)
- Penn Treebank: http://www.cis.upenn.edu/~treebank/
- PropBank: http://verbs.colorado.edu/~mpalmer/projects/ace.html
- NomBank: http://nlp.cs.nyu.edu/meyers/NomBank.html
- Penn Discourse Treebank: http://www.seas.upenn.edu/~pdtb/
- TimeML (incl TimeBank): http://www.timeml.org/site/index.html
- Pittsburgh Opinion Annotation: http://mpqa.cs.pitt.edu/
Readings and Homework

• TBA