TOKENIZATION AND Part-of-speech tagging
POS tagging: introduction

• Annotate word with part-of-speech information
• State of art: 95%+ for English
• Often 1 wd/sent error
• Syntagmatic approach: consider close tags
• Frequency (‘dumb’) approach: over 90%
• Various standardized tagsets
Tokenization

- Splitting up words from an input document
- How hard can that be? What is a word? Issues:
  - Whitespace
  - Spliced punctuation
  - Hyphenation
  - Clitics
  - Word separation (some languages)
  - Capitalization
- Normalization sometimes refers to this cleanup
- It’s easy to underestimate this task!
- Related: sentence boundary detection
Tokenize this!

file: FL977416_CP-1195236 04 05 06
file: FL203088_TN-833756 05 06 07
file: FL83567_TN-330011 19 20 21
file: FL83567_TN-330011 29 30 31
file: FL83567_TN-330011 25 26 27
file: FL1047444_CP-679926 17 18 19
file: FL1047444_CP-679926 55 56 57
file: FL1047444_CP-679926 82 83 84
file: FL65052_TN-1341174 054 055 056
file: FL65052_TN-1341174 151 152 153
file: FL65052_TN-1341174 064 065 066
file: FL1310736_CP-544963 15 16 17
file: FL1310736_CP-544963 18 19 20
file: FL1310736_CP-544963 21 22 23
file: FL1310736_CP-544963 30 31 32
file: FL1040493_CP-1152140 11 12 13
file: FL1040493_CP-1152140 15 16 17
file: FL1040493_CP-1152140 20 21 22
file: FL84174_TN-379660_07 050 051 052
file: FL84174_TN-379660_07 106 107 108
file: FL84174_TN-379660_07 075 076 077
file: FL84174_TN-379660_07 022 023 024
file: FL225982_TN-672458 125 126 127
file: FL225982_TN-672458 019 020 021
file: FL225982_TN-672458 111 112 113
file: FL225982_TN-672458 058 059 060
file: FL225982_TN-672458 062 063 064
file: FL225982_TN-672458 032 033 034
file: FL225982_TN-672458 073 074 075
file: FL1728583_CP-1124436 39 40 41
file: FL1728583_CP-1124436 36 37 38
file: FL1034992_CP-561723 032 033 034
file: FL1034992_CP-561723 063 064 065
Another common task

• Part-of-speech assignment (aka tagging)
• Label each word (etc.) with a tag describing its function
Why are POS helpful?

- Pronunciation
  - I will lead the group into the lead smelter.
- Predicting what words can be expected next
  - Personal pronoun (e.g., I, she) ____________
- Stemming (web searches)
  - -s means singular for verbs, plural for nouns
- Translation
  - (E) content +N ⇒ (F) contenu +N
  - (E) content +Adj ⇒ (F) content +Adj or satisfait +Adj
Annotating POS

- Most English sets have about 40-75 tags
- Much study has been carried out over the last 15 years or so
- Extremely important for many linguistic applications
  - You’ll use some of these during the semester
How hard is POS tagging?

• Easy: open classes (verb, noun, adjective, adverb)
• A little harder: Closed classes
  • conjunctions: and, or, but
  • pronouns: I, she, him
  • prepositions: with, on
  • determiners: the, a, an

• Hard:
  • provided, as in “I’ll go provided John does.”
  • there, as in “There aren’t any cookies.”
  • might, as in “I might go.” or “I might could go.”
  • no, as in “No, I won’t go.”
Tagsets

- Brown corpus tagset (87 tags)
- Penn Treebank tagset (45 tags)
- Claws7 tagset (146 tags)
Automating the process

- Rule-based vs. probabilistic approaches
- Steps:
  - Token lookup in dictionary
  - Not found: morphology, heuristics
  - Ambiguity: crucial problem (for English)
- Rule-based: use linguistic rules, clues
- Probabilistic: use probabilistic values
Bigram tagging

• “a new play”: \( P(\text{NN}|\text{JJ}) = 0.45, \ P(\text{VBP}|\text{JJ}) = 0.0005 \) (Brown Corpus)

• Training: FSA, probability measures
  • Bayes: tag-bigram and word-tag prob’s

• Testing: producing tag seq’s
  • DP algorithms can reduce exponentiality (e.g. Viterbi)
  • OOV (out-of-vocabulary) words:
    • Anything goes
    • Use morphological, orthographic clues
Trigram tagging

• Much more versatile, but
  • After comma, not useful
  • Data sparseness

• Linear interpolation (combine 1,2,3-grams): combine context lengths

• Variable memory: shift between lengths of N-grams
Transformation-based learning

• Premise: learn transformations to correct erroneous taggings
• Input: tagged corpus, dictionary
• Rule: triggering environment $\rightarrow$ rewrite
• Triggers
  • Previous tags, previous words
  • Previous word/tag combinations
  • Previous morphological operations
Other approaches

- Decision trees
- Neural networks
- k-nearest-neighbor approaches
- Analogical modeling
- Entropy measures
- Rule-based approaches (EngCG)
Applications

• Partial parsing, shallow parsing
• Information extraction: template-filling
• Information retrieval: indexing terms
• Question answering

• Full parsing can elim. need for tagging
• Fast, lightweight component
Factors

• Amount of training data available
• Tagset
• Dictionary/training/test corpus differences
• Unknown words
• Language being tagged
• Machine resources
QTAG

- Two sources of information
  - Dictionary of words, possible tags, freqs
  - Matrix of tag sequences, freqs
  - Can generate both from pre-tagged corpus
- Trigram window, dico lookup, guess if word not found
  - $P_w = P(\text{tag}|\text{token})$, $P_c = P(\text{tag}|t1,t2)$,
  - $P_{w,c} = P_w \times P_c$
Supervised approaches

• Connectionist
• (Naïve) Bayesian
• Information Theory
• AML
• Decision trees
Solutions to supervision

- Bootstrapping: hand-tag some training data
- Cluster via some method, hand-tag each class
- Leverage bilingual corpora, evidence from translation contexts
- Sparseness: smoothing
Using lexical resources

• Use a dictionary directly
  • Use bag-of-words from definitions, glosses
• Categorize words in microcontext
  • Thesaurus entries, subject codes
• Leverage bilingual dictionaries
  • Compare trx occurrences in src, tgt lang’s
• Leverage distributional properties
  • Conjunction, collocation, subcategorization
Open problems

• How to address context (bag-of-words, relational information, microcontext, distance measures)
• Sense division (psychological validity, lexicographer intuitions, standards)
• Granularity (metaphor, src/tgt mismatch)
• Evaluation (MUC, TREC, SENSEVAL)
Unsupervised learning

• Necessary when resources inadequate
• Sense tagging is unrealistic
• Sense discrimination is possible
• Initialize parameterized model, refine via EM algorithm
• Can process finer-grained results than exist in reference source (e.g. bank: corporate entity vs. physical building, suit: civil vs. criminal)