I256:
Applied Natural Language Processing

Marti Hearst
Week 4

Today

- Acquiring Vocabulary
  - Memorization vs Rules
  - Example: Morphology
- Lemmatization
- Stemming

Acquiring Vocabulary

- Children learn words quickly
  - Around age two they learn ~ 1 word every 2 hours.
    - (Or 9 words/day)
  - Often need one exposure to associate word/meaning
    - Can make mistakes, e.g., overgeneralization

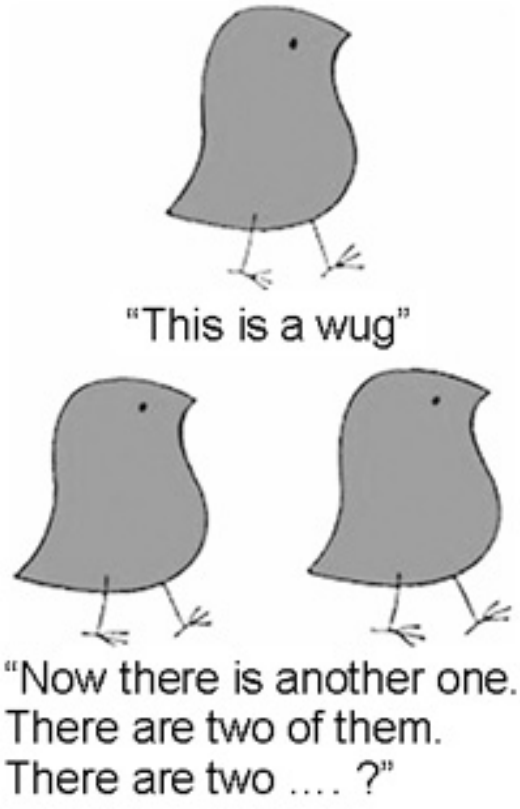
      "I goed to the store."
  - Exactly how they do this is still under study
- Adult vocabulary
  - Typical adult: about 60,000 words
  - Literate adults: about twice that.

Acquiring Vocabulary

- Dogs can do word association too!
  - Rico, a border collie in Germany
  - Knew the names of each of 100 toys
  - Could retrieve items called out to him with over 90% accuracy.
  - Could also learn and remember the names of unfamiliar toys after just one encounter, putting him on a par with a three-year-old child.

http://www.nature.com/news/2004/040607/pf/040607-8_pf.html

But is it all memorization?



Children age 4-7 insistently fill in the answer.

Similarly for rick, bing, or gling for a man doing the same thing yesterday.

But is it all memorization?

establish

establishment

the church of England as the official state church.

disestablishment

to deprive (a church) of official government support

antidisestablishment

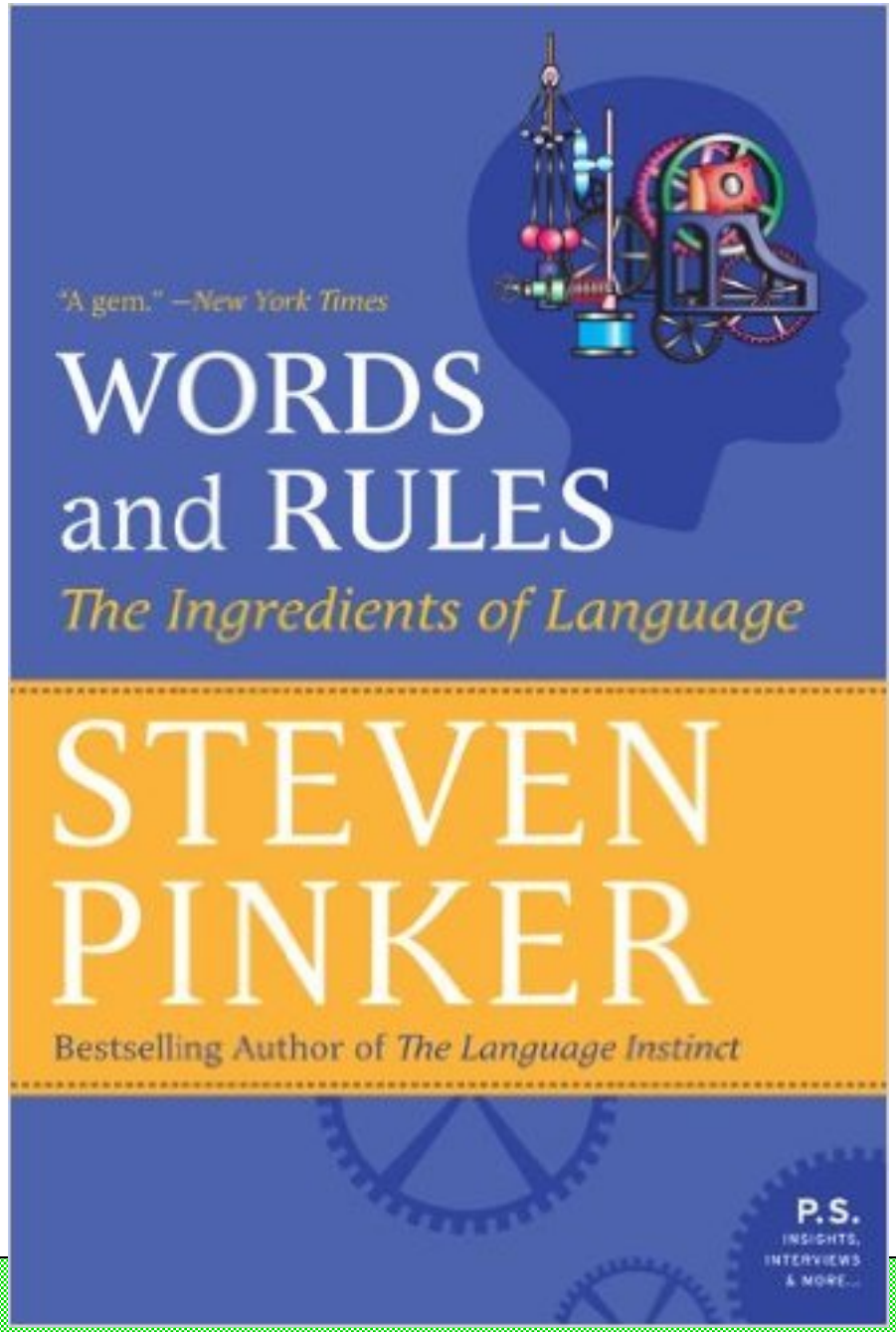
antidisestablishmentarian

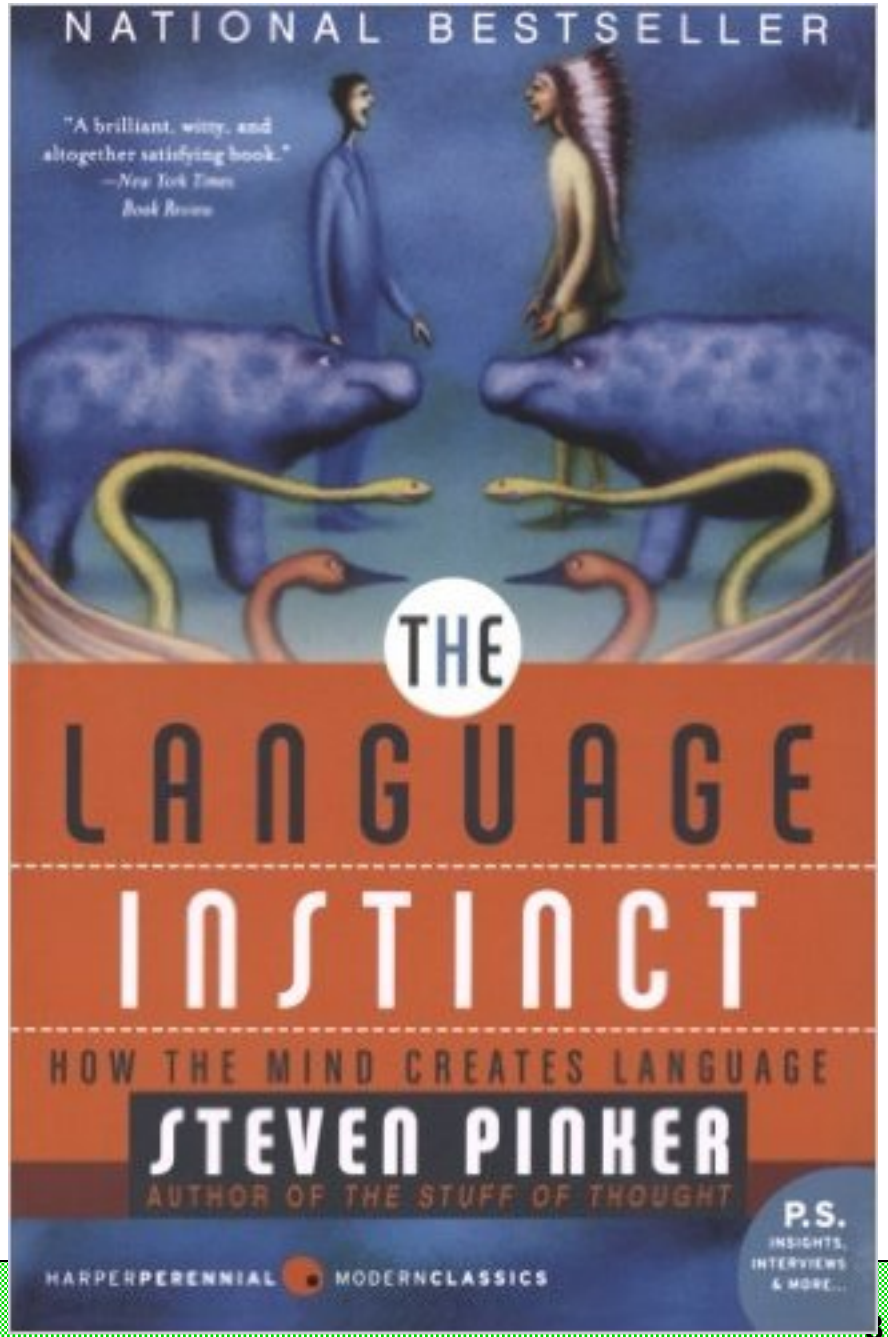
antidisestablishmentarianism

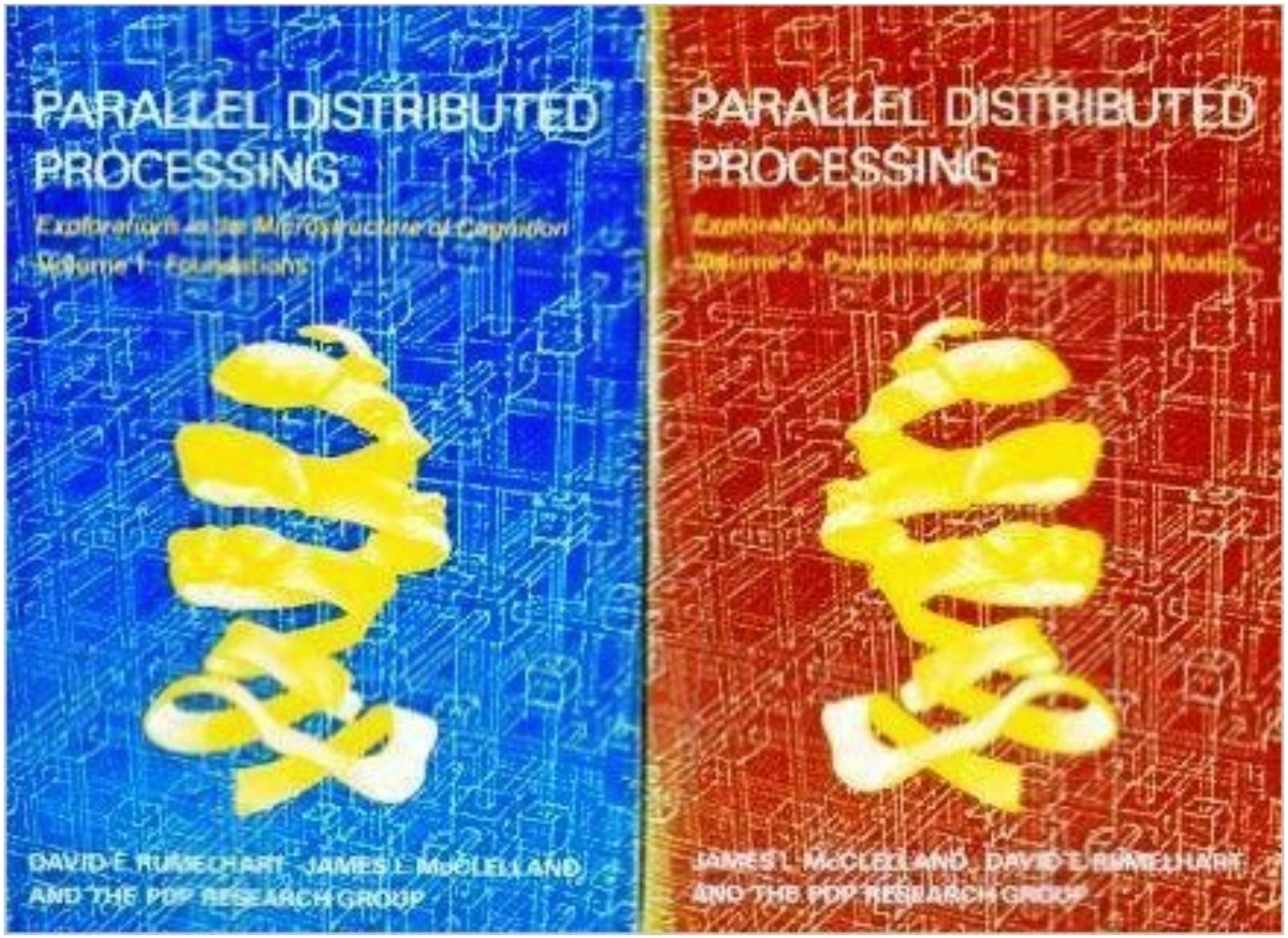
is a political philosophy that is opposed to the withdrawal of state recognition of the church.

Rules vs Memorization

- Current thinking in psycholinguistics is that we use a combination of rules and memorization
  - However, this is controversial
- Mechanism:
  - If there is an applicable rule, apply it
    - Plural: add 's' to the end
  - However, if there is a memorized version, that takes precedence.
    - (Important for irregular words.)
    - Artists paint "still lifes"
      - Not "still lives"
    - Past tense of
      - think → thought
      - blink → blinked
    - This is a simplification; for more on this, see Pinker's "Words and Rules".


"A gem." —New York Times
WORDS and RULES
The Ingredients of Language
STEVEN PINKER
Bestselling Author of The Language Instinct
P.S.
INSIGHTS, INTERVIEWS & MORE...

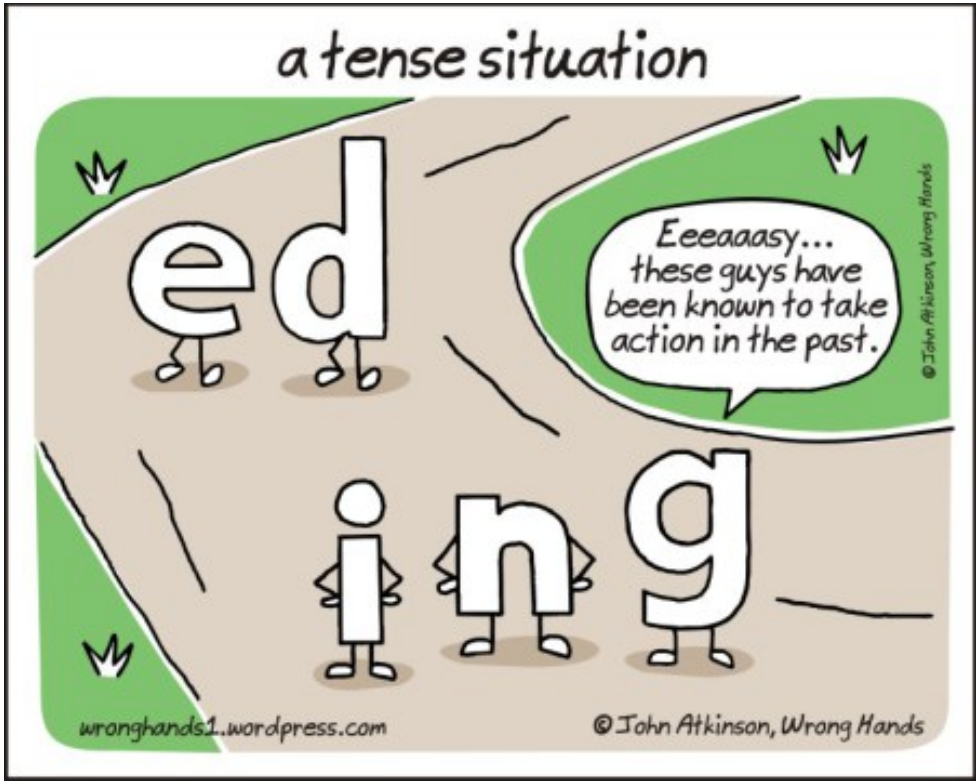

NATIONAL BESTSELLER
"A brilliant, witty, and altogether satisfying book."
—New York Times Book Review
THE LANGUAGE INSTINCT
HOW THE MIND CREATES LANGUAGE
STEVEN PINKER
AUTHOR OF THE STUFF OF THOUGHT
P.S.
INSIGHTS, INTERVIEWS & MORE...
HARPERPERENNIAL MODERNCLASSICS



Parallel Distributed Processing, Vol. 1&2: Foundations by Rumelhart, David E.; McClelland, James L, 1987

MORPHOLOGY

morphe (form) + -logy (study of)



MORPHOLOGY

A study and description of word formation (as inflection, derivation, and compounding), in language. (Merriam-Webster online)

Morphology

- The system of word-forming elements in language
- Morphemes:
  - Smallest grammatical unit in language
  - May or may not stand alone
- *Derivational* morphemes:
  - Change the meaning or part of speech (when combined with a root word)
- *Inflectional* morphemes:
  - Modify qualities of the word such as tense and number
  - The inflections can be captured to some extent by rules

The past, the present, and the future walked into a bar. It was tense.

Inflectional Morphology

- Rules for an algorithm:
  - according = accord + ing
  - composing = compose –e + ing
  - flows = flow + s
  - modified = modify –y + ied
  - partly = part + ly
  - overregularized = over + regular + ize + d

- But … there are many irregulars
  - s z iz (pronounciation)
  - hawks dogs horses
  - hits sheds chooses
  - Pat's Fred's George's

A Poem (Richard Lederer)

The verbs in English are a fright
How can we learn to read and write?
Today we speak, but first we spoke;
Some faucets leak, but never loke.
Today we write, but first we wrote;
We bite our tongues, but never bote.
Each day I teach, for years I taught,
And preachers preach, but never praught.
This tale I tell; this tale I told;
I smell the flowers, but never smold.
If knights still slay, as once they slew,
Then do we play, as once we plew? […]

JABBERWOCKY

Lewis Carroll

(from *Through the Looking-Glass and What Alice Found There*, 1872)

'Twas brillig, and the slithy toves
Did gyre and gimble in the wabe:
All mimsy were the borogoves,
And the mome raths outgrabe.

"Beware the Jabberwock, my son!
The jaws that bite, the claws that catch!
Beware the Jubjub bird, and shun
The frumious Bandersnatch!"

He took his vorpal sword in hand:
Long time the manxome foe he sought --
So rested he by the Tumtum tree,
And stood awhile in thought.

And, as in uffish thought he stood,
The Jabberwock, with eyes of flame,
Came whiffling through the tulgey wood,
And burbled as it came!

One, two! One, two! And through and through
The vorpal blade went snicker-snack!
He left it dead, and with its head
He went galumphing back.

"And, has thou slain the Jabberwock?
Come to my arms, my beamish boy!
O frabjous day! Callooh! Callay!'
He chortled in his joy.

`Twas brillig, and the slithy toves
Did gyre and gimble in the wabe;
All mimsy were the borogoves,
And the mome raths outgrabe.

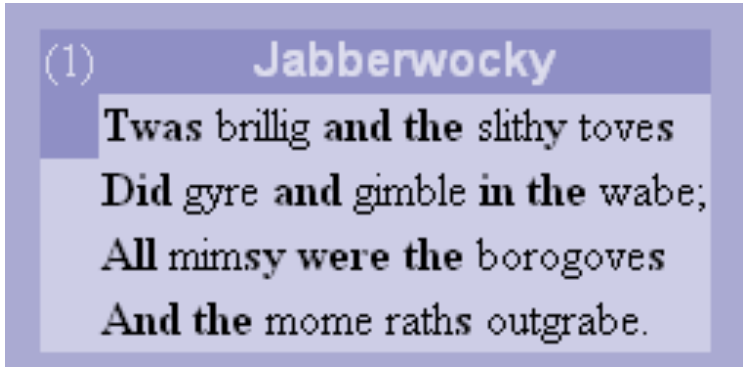
Jabberwocky Analysis

- This is nonsense … or is it?
- This is not English … but it's much more like English than it is like French or German or Chinese or …
- Why do we pretty much understand the words?

Jabberwocky Analysis

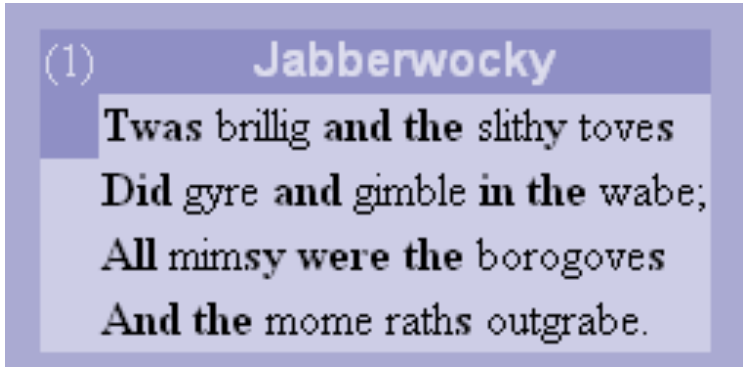
- Why do we pretty much understand the words?
- We recognize combinations of morphemes.
  - **Chortled** - Laugh in a breathy, gleeful way; (Definition from Oxford American Dictionary) A combination of "chuckle" and "snort."
  - **Galumphing** - Moving in a clumsy, ponderous, or noisy manner. Perhaps a blend of "gallop" and "triumph." (Definition from Oxford American Dictionary)

Exercise: Define Jabberwocky Terms; Justify Your Answers



- ***toves***:
- **gimble:**
- ***wabe***:

Exercise: Define Jabberwocky Terms; Justify Your Answers



(1) **Jabberwocky**

Twas brillig **and the** slithy toves
Did gyre **and** gimble **in the** wabe;
All mimsy **were the** borogoves
And the mome raths outgrabe.

- ***toves***: probably can perform an action (because they **did *gyre*** and ***gimble***)
- ***wabe***: is probably a place. (they **did** … **in the *wabe***)

Stemming

- The removal of the inflectional ending from words (strip off any affixes)
  - *Laughing, laugh, laughs, laughed→ laugh*
  - Problems
    - Can conflate semantically different words
      - *Gallery* and *gall* may both be stemmed to *gall*
  - A further step is to make sure that the resulting form is a known word in a dictionary, a task known as *lemmatization*
- Stemming is a well-defined process in the details.

Regex for Stemming

```
>>> re.findall(r'^(.*)(ing|ly|ed|ious|ies|ive|es|s|ment)$', 'processing')
[('process', 'ing')]
```

```
>>> re.findall(r'^(.*)(ing|ly|ed|ious|ies|ive|es|s|ment)$', 'processes')
[('processe', 's')]
```

- Note: the star operator is "greedy" and the .* part of the expression tries to consume as much of the input as possible. If we use the "non-greedy" version of the star operator, written *?, we get what we want:

```
>>> re.findall(r'^(.*?)(ing|ly|ed|ious|ies|ive|es|s|ment)$', 'processes')
[('process', 'es')]
```

Regex for Stemming

- The book defines a function to perform stemming, and applies it to a text:

```
>>> def stem(word):
...     regexp = r'^(.*?)(ing|ly|ed|ious|ies|ive|es|s|ment)?$'
...     stem, suffix = re.findall(regexp, word)[0]
...     return stem
...
>>> raw = """DENNIS: Listen, strange women lying in ponds distributing swords
... is no basis for a system of government.  Supreme executive power derives from
... a mandate from the masses, not from some farcical aquatic ceremony."""
>>> tokens = nltk.word_tokenize(raw)
>>> [stem(t) for t in tokens]
['DENNIS', ':', 'Listen', ',', 'strange', 'women', 'ly', 'in', 'pond',
'distribut', 'sword', 'i', 'no', 'basi', 'for', 'a', 'system', 'of', 'govern',
'.', 'Supreme', 'execut', 'power', 'deriv', 'from', 'a', 'mandate', 'from',
'the', 'mass', ',', 'not', 'from', 'some', 'farcical', 'aquatic', 'ceremony', '.']
```

- The RE removed *s* from *ponds* but also *is* from *basis* and produced some non-words like *distribut* and *deriv*.
- However, these are acceptable in some applications

The Porter Stemmer

- No lexicon
- Rewrite rules
  - ATIONAL → ATE (e.g. *relational, relate*)
  - FUL → ε (e.g. *hopeful, hope*)
  - SSES→ SS (e.g. *caresses, caress*)
- Errors of Commission
  - *Organization→ organ*
  - *Policy → police*
- Errors of Omission
  - *Urgency* (not stemmed to *urgent*)
  - *European* (not stemmed to *Europe*)
- Detailed rules: http://people.ischool.berkeley.edu/~hearst/irbook/porter.html

LEMMA

In morphology and lexicography, a lemma is the canonical form, dictionary form, or citation form of a set of words. (English Wikipedia)

Lemmatization in NLTK

- Ensure the resulting form is a **known word in a dictionary**
- NLTK includes the WordNet lexical network
- The WordNet lemmatizer only removes affixes if the resulting word is in its dictionary
  - Does the transformation, compares the result to the WordNet dictionary
  - If the transformation produces a real word, then keep it, else use the original word.
- Uses an understanding of inflectional morphology
  - Use an exception list for irregulars
  - Handles collocations in a special way
    - Based on WordNet's morphy()
    - More details: http://wordnet.princeton.edu/man/morphy.7WN.html

Exercise:
Compare Stemming Algorithms

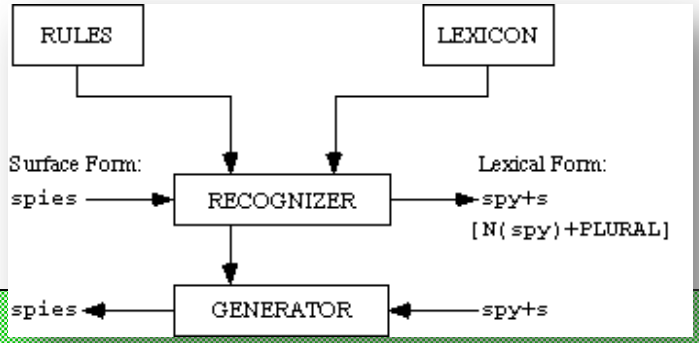
- See notebook

Is Stemming Useful?

- For word counts, can be useful
  - If the details matter, accuracy matters
  - Build vs building
- But … using bigrams / noun phrases with the simplest transformations (plurals to singular, normalizing verb tenses) is probably sufficient.
- For part-of-speech tagging, yes
- For classification algorithms, results are mixed
- For information retrieval, results are mixed

Morphologically-Based Tools

- Very sophisticated programs have been developed using a technique called Two-Level Phonology
  - Has been applied to numerous languages
- Best known: PCKimmo
  - After Kimmo Koskenniemi, based in part on work by Lauri Kartunnen in 1983
  - Uses:
    - A **rules file** which specifies the alphabet and the phonological (or spelling) rules,
    - A **lexicon file** which lists lexical items and encodes morphotactic constraints.
  - http://www.sil.org/pckimmo/
    - NLTK used to have it (in perl)



Morphologically-Based Tools

- Another version based on augmenting lemmas: morpha
  - Minnen, Carroll, Pearce. "Applied morphological processing of English." Natural Language Engineering 7.03 (2001).
  - http://sro.sussex.ac.uk/1213/1/minnen.pdf
- Based on Unix Flex and other filter tools, now written in Java
  - https://github.com/knowitall/morpha
- 1400 rules for morphological generalizations along with exceptions for specific words, e.g.:
  - {A}+{C}"ied" {return(lemma(3, "y", "ed"));}
    - {A}+ is a regex for alphabetic characters
    - {C} is a regex for a single consonant
    - "ied" is a literal
    - This converts "carried" to "carry + ed" by replacing the last 3 characters and inserting a "+"
  - Exceptions are then just listed as such:
    - "boogied" {return(lemma(1, "", "ed"));}

Next Week

- Part of Speech Tagger Algorithms
  - Regex Taggers
  - Unigram, Bigram, and Trigram Taggers with Backoff
- Training, Development, and Test Sets
- Confusion Matrices for Error Assessments
- More on Language Modeling