Tokenization

School of Information Studies
Syracuse University

Text Representation/ Vectorization

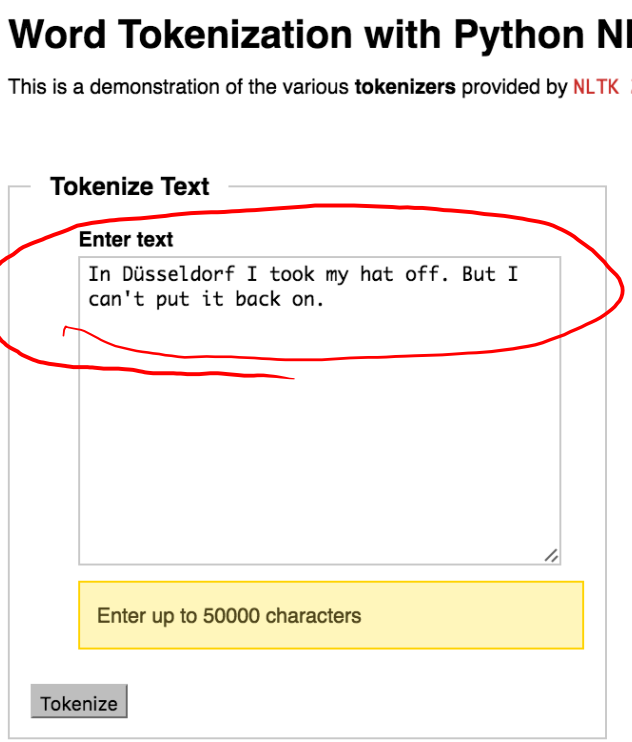
Computers can do only **one** thing, that is, **counting**!

First step toward text mining: convert text to numbers

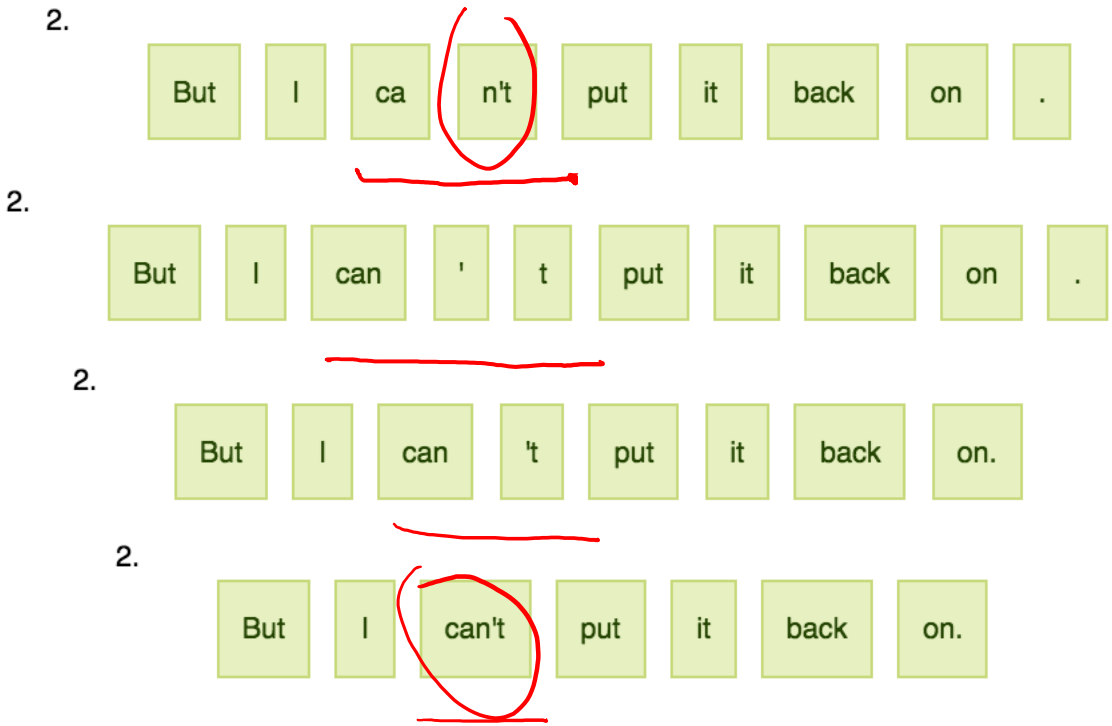
- What to count?
- How to count?

What to Count? Tokens!

A tokenizer has a set of rules about grouping characters into tokens.



Tokenization Rules Can Vary



N-Gram: Multi-Word Tokens

Bag-of-Word representation (BoW) ignores the context of words

Multi-word tokens (n-grams) can capture local context of words; e.g., “digital library”

Common n-grams:

- Uni-grams: tokens of individual words
- Bi-grams: tokens of two consecutive words
- Tri-grams: tokens of three consecutive words

Tokenization Is Not Easy

Tokenizing URLs

- Choosespain.com

Tokenization Is Not Easy

Tokenize text strings with no whitespace Chinese (New Year couplets):

养猪大如山老鼠头头死

Raise|pigs|big|as|mountain|rats|all|die

养|猪|大|如|山|老鼠|头头|死

Raise|pigs|big|as|mountain rats, all|die

养|猪|大|如|山老鼠| 头头|死

Tokenization Is Not Easy

Lowercase vs. uppercase

Words with inflected forms

- "dishwasher" vs. "dishwashers"

Words with multiple senses

- "There is a money **bank** near the river **bank**."

WordNet

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: shoot [Search WordNet]

Display Options: (Select option to change) [Change]

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations

Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) **shoot** (a new branch)
- S: (n) **shoot** (the act of shooting at targets) *"they hold a shoot every weekend during the summer"*

Verb

- S: (v) **shoot**, hit, pip (hit with a missile from a weapon)
- S: (v) **shoot**, pip (kill by firing a missile)
- S: (v) blast, **shoot** (fire a shot) *"the gunman blasted away"*
- S: (v) film, **shoot**, take (make a film or photograph of something) *"take a scene"; "shoot a movie"*
- S: (v) **shoot** (send forth suddenly, intensely, swiftly) *"shoot a glance"*
- S: (v) dart, dash, scoot, scud, flash, **shoot** (run or move very quickly or hastily) *"She dashed into the yard"*
- S: (v) tear, **shoot**, shoot down, charge, buck (move quickly and violently) *"The car tore down the street"; "He came charging into my office"*
- S: (v) **shoot** (throw or propel in a specific direction or towards a specific objective)

Word Sense Disambiguation (WSD)

WSD techniques use word context to decide the word sense.

Could introduce more errors to next steps.

So far does not help search engines significantly.

Not widely used in text mining.

Vectorization (How to Count)

School of Information Studies
Syracuse University

How to Count Tokens?

Convert documents into word vectors

Bag of Words (BoW)

- Boolean
- Term frequency
- Normalized term frequency
- Tf*idf

Vectorization

Step 1: Create a dictionary of unique words.

1. "vector"
2. "number"
3. "text"
4. …

	"vector"	"number"	"text"	…
Doc1	1	0	0	
Doc2	1	1	1	
doc3	1	0	1	

Step 2: Represent every document as a word vector: each word is an attribute/feature.

Boolean Vectors

Word presence or absence

	"vector"	"number"	"text"	…
Doc1	1	0	0	
Doc2	1	1	1	
Doc3	1	0	1	

Frequency Vectors

Word frequency: the number of word occurrences

	"vector"	"number"	"text"	…
Doc1	5	0	0	
Doc2	1	3	6	
Doc3	2	0	8	

Normalized Frequency Vectors

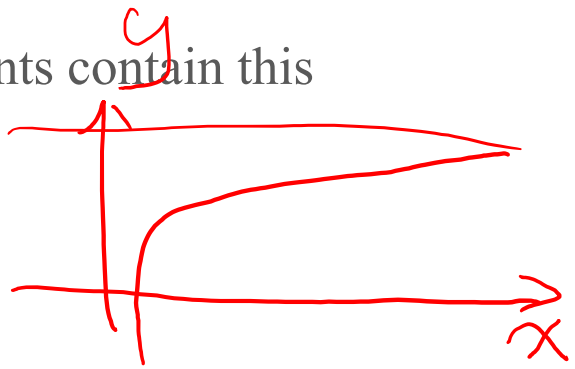
Normalized word frequency: word frequency normalized by the document length

	"vector"	"number"	"text"	…
Doc1	0.51	0.02	0.01	
Doc2	0.12	0.15	0.35	
Doc3	0.02	0.13	0.43	

TF*IDF Vectors

Tf*idf weighting

- Tf: term (word) frequency
- Df: document frequency; i.e, how many documents contain this term; e.g., 2 out of 3 documents -> 2/3
- Idf: inversed-document frequency, 3/2 = 1.5
- Tfidf=tf*log(idf)



	"vector"	"number"	"text"
Doc1	1	0	0
Doc2	0.1	0.3	0.6
Doc3	0.2	0	0.8

	"vector"	"number"	"text"
Doc1	0	0	0
Doc2	0	0.3*log3	0.6*log1.5
Doc3	0	0	0.8*log1.5

History of TF*IDF

A concept borrowed from information retrieval

A “blind” weighting strategy for text classification

Importance of Normalization

School of Information Studies
Syracuse University

Importance of Normalization

Different examples/vectors might differ greatly in length.

Example: For text documents, the vector lengths of long documents are much greater than short documents.

An Example of Normalization in Information Retrieval

	a	against	but	camera	gallery	hit	husband	images	imagined
music.1	13	0	3	0	0	0	0	0	0
music.2	18	0	7	0	0	2	0	0	0
music.3	33	0	2	0	3	1	0	0	0
music.4	28	0	11	0	0	1	0	0	0
music.5	10	0	0	0	1	0	0	0	0
art.1	20	0	3	2	0	0	1	0	0
art.2	51	0	9	1	4	0	0	2	1
art.3	55	1	6	11	1	0	2	8	0
art.4	64	2	7	0	0	0	0	0	2
art.5	11	1	1	0	0	0	0	2	0

	instruments	melody	new	old	photographs	photography	songs	wife
music.1	3	1	0	0	0	0	0	0
music.2	0	0	1	1	0	0	0	0
music.3	0	0	2	1	0	0	3	0
music.4	0	0	2	0	0	0	0	1
music.5	0	1	2	1	0	0	1	0
art.1	0	0	1	0	0	1	0	1
art.2	0	0	3	3	1	4	0	1
art.3	1	0	5	2	0	3	0	2
art.4	0	0	1	0	0	0	0	2
art.5	0	0	0	0	1	1	0	0

Table 2: Bag-of-words vectors for five randomly selected stories classified as "music", and five classified as "art" (but not music), from the *Times* corpus. The table shows a selection of the 700 features.

http://www.stat.cmu.edu/~cshalizi/350/lectures/01/lecture-01.pdf

Longer Documents Tend to Be Far Away From Short Ones Based on Raw Euclidean Distance

	a	against	but	camera	gallery	hit	husband	images	imagined
music.1	13	0	3	0	0	0	0	0	0
music.2	18	0	7	0	0	2	0	0	0
music.3	33	0	2	0	3	1	0	0	0
music.4	28	0	11	0	0	1	0	0	0
music.5	10	0	0	0	1	0	0	0	0
art.1	20	0	3	2	0	0	1	0	0
art.2	51	0	9	1	4	0	0	2	1
art.3	55	1	6	11	1	0	2	8	0
art.4	64	2	7	0	0	0	0	0	2
art.5	11	1	1	0	0	0	0	2	0

	instruments	melody	new	old	photographs	photography	songs	wife
music.1	3	1	0	0	0	0	0	0
music.2	0	0	1	1	0	0	0	0
music.3	0	0	2	1	0	0	3	0
music.4	0	0	2	0	0	0	0	1
music.5	0	1	2	1	0	0	1	0
art.1	0	0	1	0	0	1	0	1
art.2	0	0	3	3	1	4	0	1
art.3	1	0	5	2	0	3	0	2
art.4	0	0	1	0	0	0	0	2
art.5	0	0	0	0	1	1	0	0

Table 2: Bag-of-words vectors for five randomly selected stories classified as "music", and five classified as "art" (but not music), from the *Times* corpus. The table shows a selection of the 700 features.

http://www.stat.cmu.edu/~cshalizi/350/lectures/01/lecture-01.pdf

Normalization by Document Length (L-1)

2.1 Normalization

Just looking at the Euclidean distances between document vectors doesn't work, at least if the documents are at all different in size. Instead, we need to **normalize** by document size, so that we can fairly compare short texts with long ones. There are (at least) two ways of doing this.

Document length normalization Divide the word counts by the total number of words in the document. In symbols,

$$\vec{x} \mapsto \frac{\vec{x}}{\sum_{i=1}^{p} x_i}$$

Notice that all the entries in the normalized vector are non-negative fractions, which sum to 1. The i^{th} component is thus the probability that if we pick a word out of the bag at random, it's the i^{th} entry in the lexicon.

Normalization by Euclidean Length (L-2)

Euclidean length normalization Divide the word counts by the Euclidean length of the document vector:

$$\vec{x} \mapsto \frac{\vec{x}}{\|\vec{x}\|}$$

For search, normalization by Euclidean length tends to work a bit better than normalization by word-count, apparently because the former de-emphasizes words which are rare in the document.

Cosine "distance" is actually a similarity measure, not a distance:

$$d_{\cos}\,\vec{x}, \vec{y} = \frac{\sum_i x_i y_i}{\|\vec{x}\|\,\|\vec{y}\|}$$

It's the cosine of the angle between the vectors $\vec{x}$ and $\vec{y}$.

Compare Results With/Without Normalization

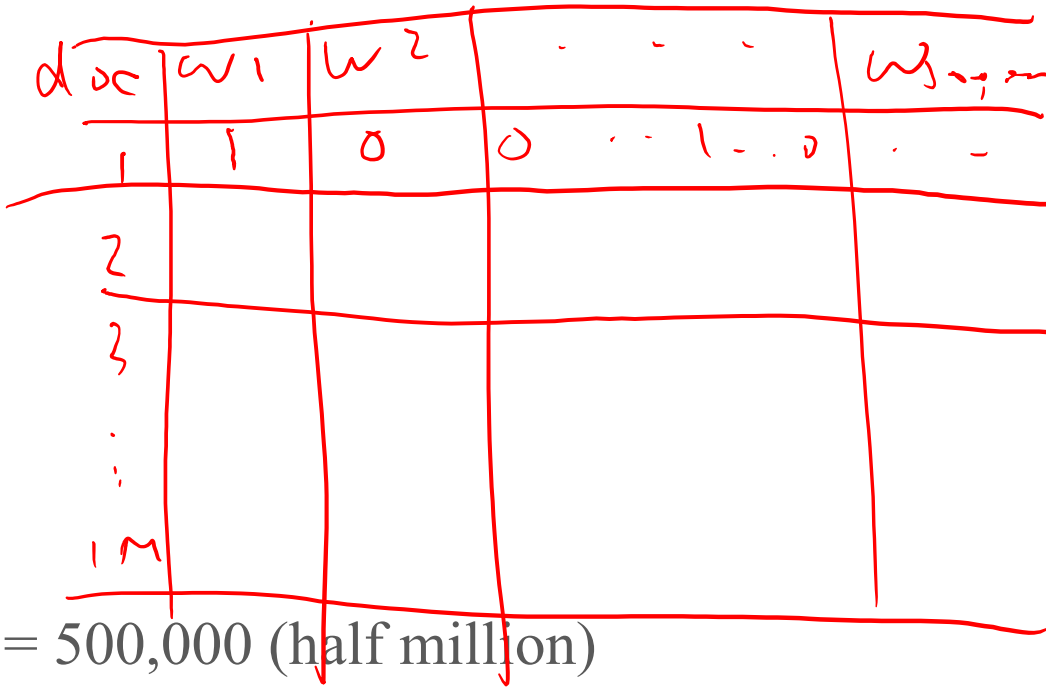
	Best match by similarity measure		
	Euclidean	Euclidean + word-count	Euclidean + length
music.1	art.5	art.4	art.4
music.2	art.1	music.4	music.4
music.3	music.4	music.4	art.3
music.4	music.2	art.1	art.3
music.5	art.5	music.3	music.3
art.1	music.1	art.4	art.3
art.2	music.4	art.4	art.4
art.3	art.4	art.4	art.4
art.4	art.3	art.3	art.3
art.5	music.1	art.3	art.3
error count	6	2	3

Table 3: Closest matches for the ten documents, as measured by the distances between bag-of-words vectors, and the total error count (number of documents whose nearest neighbor is in the other class).

Sparse Matrix

School of Information Studies
Syracuse University

Sparse Matrix



N = 1 million docs

Each doc = 1,000 words

Each word = 6 bytes

Total corpus size = 6 GB

Total distinct words (types) = 500,000 (half million)

A matrix of 500,000 X 1 million:

- Half-a-trillion cells (0s or 1s): $5*10^{11}$
- 99.8% values are zeros
  - The number of 1s is up to 1 million*1,000 = $1*10^{9}$

Compressed Text Representation

Look-up dictionary of "index:word" pairs

- 1: "a," 2: "great," 3: "is," 4: "this," 5: "movie," 6: "terrible," …

Compressed storage of sparse vector:

- Each word that occurs in a document will be recorded as an "index:value" pair. For example:
  - Document: "this is a great movie"
  - Boolean vector: "1:1, 2:1, 3:1, 4:1, 5:1"

Reducing Vocabulary Size

School of Information Studies
Syracuse University

Approaches for Reducing Vocabulary Size

Stemming

Case merging

Removing stop words

Word clustering

Stemming

Can be used in inflected languages like English

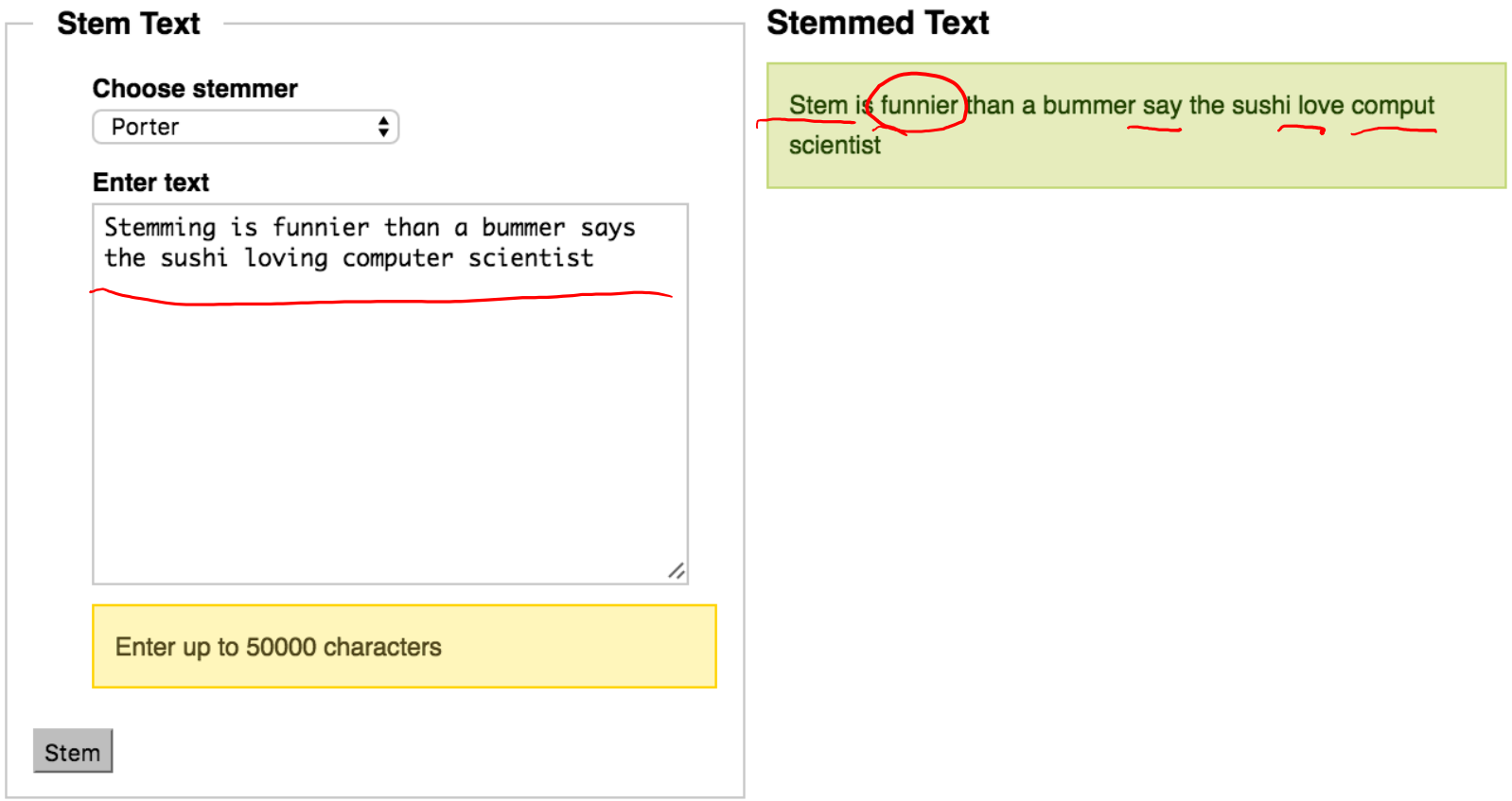
Stemmer: remove postfixes to find the root form

- "Applied" and "application" -> "appli"

Lemmatizer: transform the root to a real word

- "Applied" and "application" -> apply

NLTK Stemming Demo



Stemming Issues

How far should it go?

- “denormalization” -> denormalize -> denormal -> normal -> norm?

How accurate can it be?

- “bore”/ “He wanted to bore a hole” / “He bore the students on his heart”

How Useful Is Stemming?

No consistent conclusion

Information retrieval

- Search “dishwasher” to know how it works
- Search “dishwashers” to shop around

Text categorization

- Future tense vs. past tense in company performance report
  - “Will do” vs. “have done”

Convert Uppercase to Lowercase?

Emily Dickinson's poem

- "Joy" vs. "joy"
- "Love" vs. "love"

Uppercase

But pompous
Joy
Betrays us, as
his first
Betrothal
Betrays a
Boy.

The Treason
of an Accent
Might vilify
the Joy -
To breathe -
corrode the
rapture
Of Sanctity
to be

Boundlessness -
Expanse cannot
be lost -
Not Joy, but
a Decree
Is Deity -
His Scene,
Infinity -

Lowercase

Could she have guessed that it would be - Could but a Crier of the joy Have climbed the distant hill! -	I want to send you joy, I have half a mind to put up one of these dear little Robin's, and . . .	I cant believe you are coming - but when I think of it, and tell myself it's so, a wondrous joy comes over me, and my old fashioned life . . .

Remove Stop Words

Observation: words that occur in most documents are not useful of distinguishing documents

Stop words are usually function words that bear no specific meaning, compared to content words

Example of the Start of a Stop Word List

a
about
across
after
afterwards
again
against
all
alone
along
already
also
always
amongst
an
and
another
any
anyhow
anyone
anything
are
around
as
be
because
becomes
becoming
been
before
beforehand
behind
being
below
besides
between
beyond
but
can

Little Words Can Make Big Difference

School of Information Studies
Syracuse University

Little Words Can Make Big Difference

Function words are useful for certain text mining tasks

- Genre classification
- Authorship attribution
- Gender detection

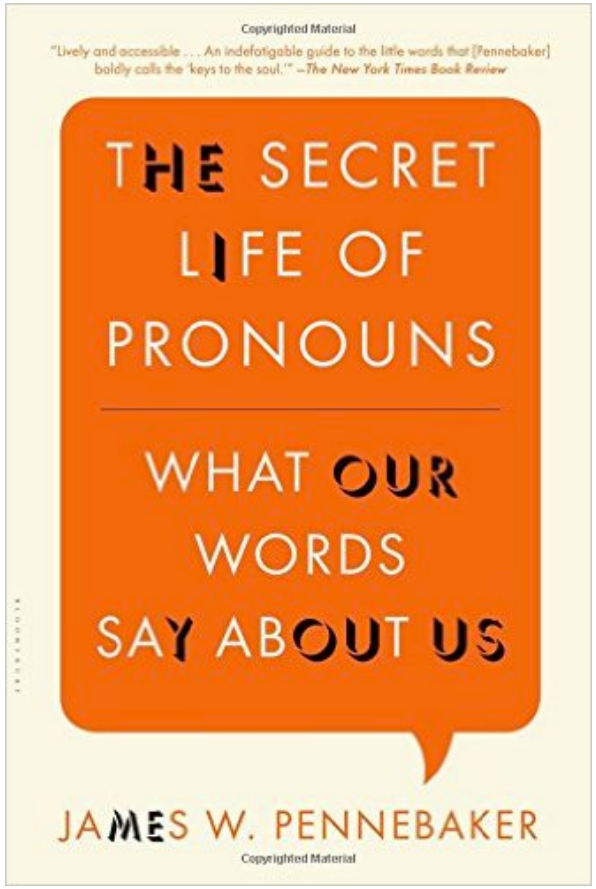
Genre Classification

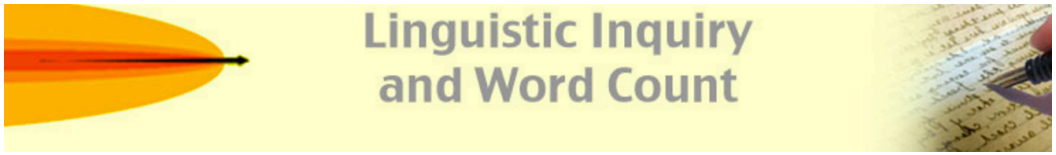
Personal Homepage Identification (Riloff, 1995)

- Top features “I” and “my”

Riloff, E. (1995, July). Little words can make a big difference for text classification. In Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval (pp. 130–136). ACM.

Personal Pronouns





Testing LIWC Online

We understand completely. You are a student, a poor faculty member, or a researcher who wants to analyze a few cases without having to buy the LIWC program for almost $100. We've been there, and, because we know your plight, this page is for you. This is a no-frills page whereby you can enter text (by typing it directly or copying it from some other place and pasting it here) and get the basic LIWC output. All you have to do is enter the text file you want to analyze, press SUBMIT, and voila, we will give you feedback on some of the LIWC dimensions. That's the kind of people we are.

OK, we admit it. We aren't completely altruistic. We would like to keep a copy of your text files to add to our growing archive of 50,000+ files. To help us with our data, could you enter the age and gender of the author of the text (if you know it). If you don't know them or don't want to enter them, then choose 'No details' from the 'Gender of text author' selector.

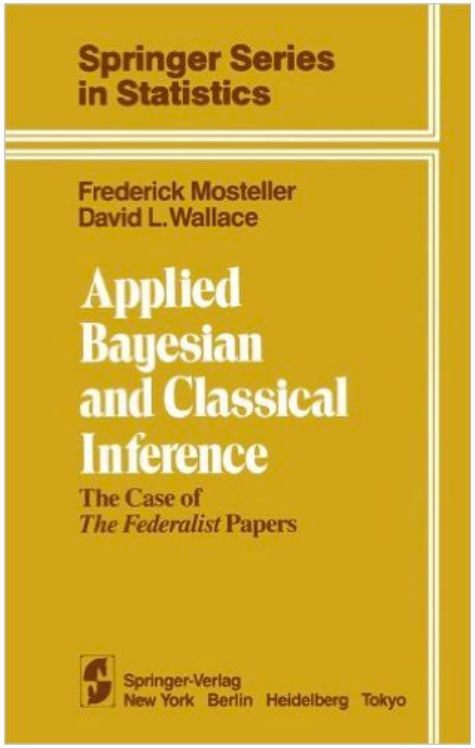
Gender of author: ? *Age of author:* ?

Type or paste the text you want analysed into the box below and then hit the submit button.

Submit text for online LIWC analysis

LIWC

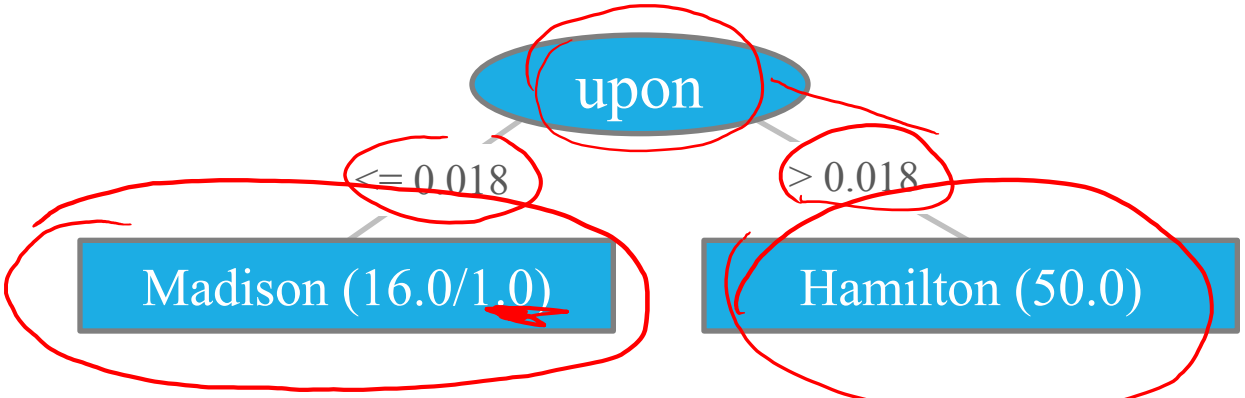
Function Words for Authorship Attribution



The Federalist Papers

This web-friendly presentation of the original text of the Federalist Papers (also known as The Federalist) was obtained from the e-text archives of Project Gutenberg. For more information, see About the Federalist Papers.

No.	Title	Author	Publication	Date
1	General Introduction	Hamilton	For the Independent Journal	--
2	Concerning Dangers from Foreign Force and Influence	Jay	For the Independent Journal	--
3	The Same Subject Continued: Concerning Dangers from Foreign Force and Influence	Jay	For the Independent Journal	--
4	The Same Subject Continued: Concerning Dangers from Foreign Force and Influence	Jay	For the Independent Journal	--
5	The Same Subject Continued: Concerning Dangers from Foreign Force and Influence	Jay	For the Independent Journal	--
6	Concerning Dangers from Dissensions Between the States	Hamilton	For the Independent Journal	--



School of Information Studies
Syracuse University

Gender Difference in Speech and Writing

TABLE 1 (*Continued*)

LIWC Dimension	*Examples*	*Female*		*Male*		
		M	*SD*	*M*	*SD*	*Effect Size (d)*
Pronouns		14.24	4.06	12.69	4.63	0.36
First-person singular	I, me, my	7.15	4.66	6.37	4.66	0.17
First-person plural	we, us, our	1.17	2.15	1.07	2.12	*ns*
Second person	you, you're	0.59	1.05	0.65	1.15	–0.06
Third person	she, their, them	3.41	3.45	2.74	3.01	0.20

Newman, M. L., Groom, C. J., Handelman, L. D., & Pennebaker, J. W. (2008). Gender differences in language use: An analysis of 14,000 text samples. Discourse Processes, 45(3), 211–236.