Text mining theory and jargon

Nik Bear Brown

In this lesson we'll learn the theory and jargon you'll need to explore text mining in R.

# Additional packages needed

None. This lesson is just theory.

# Data

None. This lesson is just theory.

# Text Mining

[Text mining](https://en.wikipedia.org/wiki/Text_mining), also called text analytics, is the process of deriving high-quality information from text. Given the vast amount of text on the Internet, text mining is one of the most important research areas in machine learning. Text mining includes:

* [Information retrieval](https://en.wikipedia.org/wiki/Information_retrieval) - the process of obtaining structured data from free text.
* [Entity recognition](https://en.wikipedia.org/wiki/Named-entity_recognition) - identification of nouns (e.g. known people, places and things) in text.
* Fact extraction - identification of associations among entities and other information in text.
* [Sentiment analysis](https://en.wikipedia.org/wiki/Sentiment_analysis) is the assocaition of opinion and a topic within text.
* [Topic modeling](https://en.wikipedia.org/wiki/Topic_model) - identification of "topics" in text.
* Automated Tagging - a process to associate free-form keywords (not necessarily in the text) to index and search text.

  
*Text Mining*

# Text corpora and dictionaries

In linguistics, a corpus (plural corpora) or text corpus is a large and structured set of texts. They help annotate words.

* [corpus.byu.edu](http://corpus.byu.edu/) includes:
  + Hansard Corpus (British Parliament), 1.6 billion words
  + Global Web-Based English (GloWbE), 1.9 billion words
  + Corpus of Contemporary American English (COCA), 450 million words
  + TIME Magazine Corpus, 100 million words
  + Corpus of American Soap Operas, 100 million words
  + British National Corpus (BYU-BNC), 100 million words
  + Strathy Corpus (Canada), 100 million words
* The [British National Corpus](http://www.natcorp.ox.ac.uk/), extracts from 4124 modern British English texts of all kinds, both spoken and written; over 100 million words.
* The [Brown University Corpus](http://www.hit.uib.no/icame/brown/bcm.html): Approximately 1,000,000 words of American written English dating from 1960. The genre categories are parallel to those of the LOB corpus.
* The [LOB Corpus](http://clu.uni.no/icame/manuals/) (The Lancaster-Oslo/Bergen Corpus) is ) is a million-word collection of British English texts which was compiled in the 1970s.
* [The Kolhapur Corpus](http://www.essex.ac.uk/linguistics/external/clmt/w3c/corpus_ling/content/corpora/list/private/kolhapur.html): Approximately 1,000,000 words of Indian written English dating from 1978.
* The (Cambridge International Corpus)[<http://uk.cambridge.org/elt/corpus>] is a multi-billion word corpus of English language (containing both text corpus and spoken corpus)
* [The Longman-Lancaster Corpus](http://www.pearsonlongman.com/dictionaries/corpus/lancaster.html): Approximately 14.5 million words of written English from various geographical locations in the English-speaking world and of various dates and text types.\* [WordNet](http://wordnet.princeton.edu/) is a lexical database of English nuns, verbs, adjectives and adverbs which are grouped into sets of cognitive synonyms (synsets). The WordNet synsets are further characterized by hyperonymy, hyponymy or ISA relationships. We downloaded the WordNet database files and parsed them. Permission to use, copy, modify and distribute WordNet for any purpose and without fee or royalty is hereby granted, WordNet provided by WordNet as long as proper attribution is given to WordNet and any derivative products don’t use the WordNet trademark.
* [PubMed/Medline](http://www.ncbi.nlm.nih.gov/pubmed) comprises more than 25 million citations for biomedical literature. PubMed XML Data Retrieved from [<http://www.nlm.nih.gov/databases/journal.html>](http://www.nlm.nih.gov/databases/journal.html). You need to regiester with the National Library of Medicine to download the XML files.
* [arXiv](http://arxiv.org/) is an archive with over 100000 articles in physics, 10000 in mathematics, and 1000 in computer science. arXiv Bulk Data Access Retrieved from [<http://arxiv.org/help/bulk_data>](http://arxiv.org/help/bulk_data)
* AG's news corpus is AG's corpus of news articles. Retrieved from [<http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html>](http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html)
* Last.fm music tags can be retrieved from [<http://www.last.fm/charts/toptags>](http://www.last.fm/charts/toptags)
* Spambase can be retrieved from [<http://archive.ics.uci.edu/ml/datasets/Spambase>](http://archive.ics.uci.edu/ml/datasets/Spambase)
* Wikipedia. The Wikipedia Data Dump can be retrieved from [<http://en.wikipedia.org/wiki/Wikipedia:Database_download>](http://en.wikipedia.org/wiki/Wikipedia:Database_download)

## Unstructured text data

* [Common Crawl](http://commoncrawl.org/) is a openly accessible web crawl data that is freely available. [164] As of April 2013 the crawl has 6 billion pages and associated metadata. The crawl data is stored on Amazon’s Public Data Sets, allowing it to be directly accessed for map-reduce processing in EC2. Common Crawl Retrieved from [<http://commoncrawl.org/>](http://commoncrawl.org/)
* Twitter data.
  + Twitter Search API Retrieved from [<https://dev.twitter.com/docs/api/1/get/search>](https://dev.twitter.com/docs/api/1/get/search)
  + Twitter Streaming APIs. Retrieved from [<https://dev.twitter.com/docs/streaming-apis>](https://dev.twitter.com/docs/streaming-apis)
  + Twitter“Fire hose” real-time stream. See [<https://dev.twitter.com/streaming/firehose>](https://dev.twitter.com/streaming/firehose)
* Instagram
  + API - Instagram [<https://instagram.com/developer/>](https://instagram.com/developer/)

## Writing a web crawler

A Web crawler (also known as a Web spider or Web robot or bot) is a pscript which browses the World Wide Web and extracts web pages and links.

urls=<list of urls>  
while (urls)  
{  
 \* request url  
 \* request document  
 \* store text for later processing  
 \* parse document for links  
}

*Useful web crawling libraries in R*

* library(regex) - Regular Expressions as used in R
* library(httr) - Tools for Working with URLs and HTTP
* library(XML) - XML Parser and Generator
* library(RCurl) - RCurl: General Network (HTTP/FTP/...) Client Interface for R
* library(jsonlite) - JSON Parser and Generator for R
* library(stringr) - Simple, Consistent Wrappers for Common String Operations

# N-grams

An [n-gram](https://en.wikipedia.org/wiki/N-gram) is a contiguous sequence of n items from a given sequence of text or speech. An n-gram of size 1 is referred to as a "unigram"; size 2 is a "bigram" (or, less commonly, a "digram"); size 3 is a "trigram". Larger sizes are sometimes referred to by the value of n, e.g., "four-gram", "five-gram", and so on.

for example, bugaboo, nerd, student, data and new are one-grams. Data science, New York are two-grams. Note that New York means something different the the one grams new, and York seperately.

Note that if we take a sentence, say

"You have brains in your head. You have feet in your shoes. You can steer yourself in any direction you choose. You're on your own, and you know what you know. And you are the guy who'll decide where to go."

* Dr. Seuss

Most of the two-grams aren't meaningful, that is, "You have," "have brains," "brains in" don't make sense out of context.

# Tagging and Hashtags

Tagging is a process in which end users use free-form keywords to manually index content in an organic and distributed manner. The popularity of tagging has led some to claim that it is the primary classification scheme of the Internet. A tag can be thought of as an informative keyword. A user is very unlikely to tag an article with a word like “this” because it conveys very little information. Rather, they’ll often tag with a subject or sentiment.

Problems with tagging are well-known. Users often present idiosyncrasies, inaccuracies, inconsistencies, and other irregularities when tagging. Specifically, four areas are critical to tagging. The first three areas are straightforward enough:

1. tag misspelling;
2. tag heterogeneity, (that is, different tags denoting the same content, such as “Ziagen” and “abacavir sulfate,” which both refer to the same drug);
3. tag polysemy (i.e. identical tags that denote different meanings, such as, Apple may refer to fruit or a company. and;
4. semantic annotation of tags (i.e. abacavir sulfate is a drug).

An important area of research in text mining is call “semantic enrichment” is a particularly difficult problem. Lexical resources are often used to annotate terms. For example, lexical databases such as WordNet is often used as a source of tag annotation.

*#hashtags*

The social tagging in is done by placing a hash mark in front of a word or phrase, such as #BCSM, #Lyphoma, #BrainTumorThursday, #BreastCancer, #Infertility, #Diabetes, #lymphoedema, #RareDiseaseDay, #RareDisease, #ADHD, #Anorexia, #MultipleSclerosis. On social media sites (such as Twitter) a word or phrase preceded by a hash or pound sign (#) and used to identify messages on a specific topic. Hastags are essentially tags that in within text and annotated. As such they are easy to extract from text.

*Stop words*

In text mining, [stop words](https://en.wikipedia.org/wiki/Stop_words) are words which are filtered out becuase they'll interfer with text analysis. Stop words usually refer to the most common words in a language, such as *the*, *is*, *at*, *which*, and on.

# Regular expressions

Regular expressions (or RE or regex or regexp or rational expression) is a sequence of characters that define a search pattern, mainly for use in pattern matching with strings, or string matching.

Cerain characters have special meaning in regular expressions.

[] - A pair of brackets is used to indicate a set of characters.  
'' - Either escapes special character or signals a special sequence.  
? - The question mark indicates there is zero or one of the preceding  
'\*' - The asterisk indicates there is zero or more of the preceding element . + The plus sign indicates there is one or more of the preceding element.  
'^' (Caret.) Matches the start of the string.  
'$' Matches the end of the string.  
{m} (Braces) Specifies that exactly m copies of the previous RE should be matched.

For example, the regexp matches a seuqnece of at least one upper or lower case letters. The regexp [ ]+$ matches an e-mail pattern with starting and ending white spaces. There are many excellent books that describe regular expressions in detail.

# Term-Document matrices

A [term-document matrix](https://en.wikipedia.org/wiki/Document-term_matrix) is a matrix where the rows correspond to documents in the collection and columns correspond to terms. Creating term-document matrices are created using the [R text mining package tm](https://cran.r-project.org/web/packages/tm/index.html).

D1 = "I love R"  
D2 = "I love ice-cream"

then the document-term matrix would be:

which shows which documents contain which terms and how many times they appear.

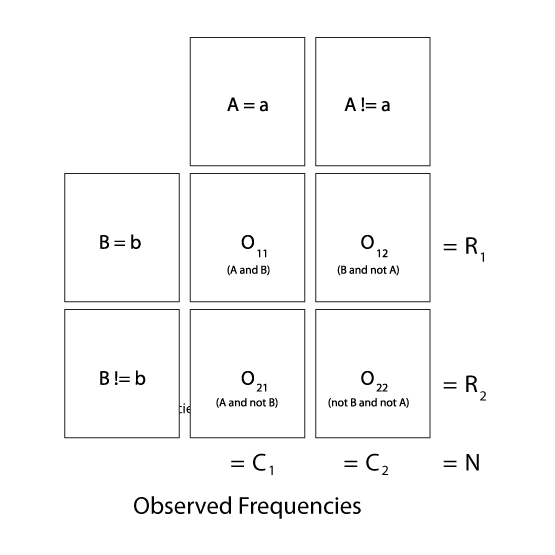
# Frequency signatures

For processing larger amounts of text, tag/word counts can be ineffecient using term-document matrices as these are typically very sparse matrices. Especially when one has a large dictionary of tags/words.

For larger data we used a “frequency signature” approach to convert a [bag-of-words](https://en.wikipedia.org/wiki/Bag-of-words_model) output to a format that we can use to calculate tag co-occurrence associations and mutual information. Frequency signatures are described in detail in Stefan Evert’s PhD dissertation “The Statistics of Word Cooccurrences Word Pairs and Collocations.”

To calculate tag co-occurrence associations and mutual information for two tags, A and B, we need four items of data. The co-occurrence count of A and B, the count of A but not B, the count of B but not A, and the total number of tags in a corpus. This co-occurrence frequency data for a word pair (A,B) are usually organized in a contingency table show below. The contingency table stores the observed frequencies . The table below (adapted from Evert’s dissertation) shows an observed contingency table.

Contingency Tables



*Contingency table : is co-occurrence count of A and B, is the count of A but not B, is the count of B but not A, and is the count of not B and not A.*

However, while the co-occurrence count of A and B, and the total number of tags in a corpus are efficiently and easily counted the count of A but not B, the count of B but not A are tricky and computationally expensive. The insight and advantage of frequency signatures is that they calculate the count of A but not B, the count of B but not A by just counting A and B and the co-occurrence count of A and B. That is, the count of A but not B is equal to count of A minus the co-occurrence count of A and B. Likewise, the count of B but not A is equal to count of B minus the co-occurrence count of A and B.

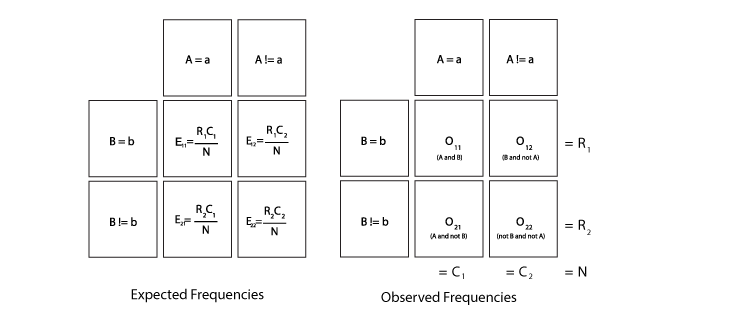
The frequency signature of a tag pair (A, B) is usually written as . Where is the co-occurrence count of A and B, is the count of A but not B, is the count of B but not A, and N is the total counts. Notice that the observed frequencies can be directly calculated from the frequency signature by the equations below:



Generating all of the data tag co-occurrence association and mutual information calculations using this approach can be generated using a single pass of the data and two associative arrays; one of the tag counts and another for the tag co-occurrence counts.

Calculating Associations and Mutual Information from Frequency Signatures

Evert shows the many association and mutual information statistics can be calculated from the observed frequencies if we can generate the expected frequencies . The table below (adapted from Evert’s dissertation) shows the expected versus observed contingency tables.

  
*Frequency Signatures*

The sum of all four observed frequencies (called the sample size N) is equal to the total number of pair tokens extracted from the corpus. R1 and R2 are the row totals of the observed contingency table, while C1 and C2 are the corresponding column totals. The expected frequencies can be directly calculated from observed frequencies by the equations below:



Evert went on to show that several association measures can be easily calculated once one has the expected and observed contingency tables. For example, the pointwise mutual information (MI) is calculated by below.

The Likelihood measures that can be calculated using the expected and observed contingency tables are:

* Multinomial-likelihood
* Binomial-likelihood
* Poisson-likelihood
* Poisson-Stirling approximation
* Hypergeometric-likelihood

The exact hypothesis tests that can be calculated using the expected and observed contingency tables are:

* binomial test
* Poisson test
* Fisher's exact test

The asymptotic hypothesis tests that can be calculated using the expected and observed contingency tables are:

* z-score
* Yates' continuity correction
* t-score (which compares O11 and E11 as random variates) \* Pearson's chi-squared test
* Dunning's log-likelihood (a likelihood ratio test)

The measures from information theory that can be calculated using the expected and observed contingency tables are:

* MI (mutual information, mu-value)
* logarithmic odds-ratio logarithmic relative-risk
* Liddell's difference of proportions
* MS (minimum sensitivity)
* gmean (geometric mean) coefficient
* Dice coefficient (aka. "mutual expectation")
* Jaccard coefficient
* MIconf (a confidence-interval estimate for the mu-value)
* MI (pointwise mutual information)
* local-MI (contribution to average MI of all co-occurrences)
* average-MI (average MI between indicator variables)

Stefan Evert also developed a R library called [UCS toolkit](http://www.collocations.de/software.html) for the statistical analysis of co-occurrence data with association measures and their evaluation in a collocation extraction task.

# tf–idf

[Tf–idf](https://en.wikipedia.org/wiki/Tf%E2%80%93idf) or term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document.

The tf-idf value increases proportionally to the number of times a word/tag appears in the document, but is offset by the frequency of the word/tag. It is a measure of jargon. If a word appears frequently in a document, it's important. Give the word a high score. But if a word appears in every documents, it's not specifc to a topic. Give the word a low score.

If we want to find jargon, that is, topic or subject specifc words/tags then this is a reasonable metric. Words like 'the', 'a', get low scores as the are in every document. While words/tags like 'Machine Learning', 'Twitter,' or 'Text Mining' get high scores since they are used a lot in specific contexts.

If with call the raw frequency of a term in a document, i.e. the number of times that term t occurs in document d and the maximum raw frequency of any term in the document, then the term frequency is:

The inverse document frequency is a measure of how much information the word provides, It is the logarithmically scaled fraction of the documents that contain the word, obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.

with

* : total number of documents in the corpus
* : number of documents where the term t appears (i.e., ). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to add a single "pseudocount."

Mathematically the base of the log function does not matter for these purposes.

Then tf–idf is calculated as

A high weight in tf–idf means a tag/word has high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents.

For how this is done in R see [The tf-idf-Statistic For Keyword Extraction](http://www.r-bloggers.com/the-tf-idf-statistic-for-keyword-extraction/)

# Word entropy and entropy rate

In information theory, entropy is also a measure of the uncertainty in a random variable. Like tf–idf, entropy quantifies the expected value of the information contained in a message (or the expected value of the information of the probability distribution). Typically this is expressed in the number of ‘bits’ or ‘nats’ that are required to encode a given message.

In this sense entropy can be used to estimate (like tf–idf) how much information is in a word or tag. Entropy can also be used to estimate the generating probability distribution for a text document or corpus. The entropy of many languages has been determined. English has 1.65 bits per word, French has 3.02 bits per word, German has 1.08 bits per word, and Spanish has 1.97 bits per word. Given the probability density function of word entropies and the average bits per word of a single tweet we could then assign probabilities that it is English, French, German, or Spanish.

## Entropy rate

The entropy rate (or mean entropy rate) describes the limiting entropy over an entire probability distribution. This can be thought of as the average entropy over a sufficiently long realization of a stochastic process, whereas the entropy is relevant to a single random variable at a given point in time.

In statistics, ergodicity describes a random process wherein the average time for one sufficiently long realization of events is the same as the ensemble average. That is, the ensemble’s statistical properties (such as its mean or entropy) can be deduced from a single, sufficiently long sample of the process. In other words, there are long-term invariant measures that describe the asymptotic properties of the underlying probability distribution, and they can be measured by following any single reprehensive portion if followed long enough. For example, if I look at two particles in an ergodic system at any time, those particles may have very different states; but if I follow those particles long enough, they become statistically indistinguishable from one another. This means that statistical properties of the entire system can be deduced from a single sample of the process if followed for a sufficiently long time.

Stationarity is the property of a random process which guarantees that the aggregate statistical properties of the probability density function, such as the mean value, its moments and variance, remain the same at every point in time. A stationary process, therefore, is one whose probability distribution is the same at all times. Its statistical properties cannot necessarily be deduced from a single sample of the process. There are stochastic processes that exhibit both stationarity and ergodicity called stationary ergodic processes. These are random processes that will not change their statistical properties with time; hence, the properties, including the disorder (entropy) of the system, can be deduced from a single, sufficiently long sample realization of the process. There are weaker forms of the stationary condition in which the first- and second-order moments (that is, the mean and variance) of a stochastic process are constant but other properties of the probability density function can vary. Likewise, there are stationary stochastic processes that are not themselves ergodic but are composed of a mixture of ergodic components.

### Rényi entropies

The [Rényi entropies](https://en.wikipedia.org/wiki/R%C3%A9nyi_entropy) generalize the Shannon entropy, the Hartley entropy, the min-entropy, and the collision entropy. As such, these entropies as an ensemble are often called the Rényi entropies (or the Rényi entropy, even though this usually refers to a class of entropies). The difference between these entropies is in the respective value for each of an order parameter called alpha: the values of alpha are greater than or equal to zero but cannot equal one. The Renyi entropy ordering is related to the underlying probability distributions and allows more probable events to be weighted more heavily. As alpha approaches zero, the Rényi entropy increasingly weighs all possible events more equally, regardless of their probabilities. A higher alpha (a) weighs more probable events more heavily. The base used to calculate entropies is usually base 2 or Euler's number base e. If the base of the logarithm is 2, then the uncertainty is measured in bits. If it is the natural logarithm, then the unit is nats.

### Rényi entropies

The Rényi entropy of order , where and , is defined as

Here, X is a discrete random variable with possible outcomes 1,2,...,n and corresponding probabilities and the logarithm is base 2.

#### Hartley entropy

The Hartley entropy (Gray, 1990) is the Rényi entropy with an alpha of zero.

the probabilities are nonzero, is the logarithm of the cardinality of X, sometimes called the Hartley entropy of X:

#### Shannon entropy

The Shannon entropy (Gray, 1990) is the Rényi entropy with an alpha of one. The Shannon entropy is a simple estimate of the expected value of the information contained in a message. It assumes independence and identically distributed random variables, which is a simplification when applied to word counts. In this sense it is analogous to naïve Bayes, in that it is very commonly used and thought to work well in spite of violating some assumptions upon which it is based.

The limiting value of is the Shannon entropy:

#### collision entropy

The collision entropy (Gray, 1990) is the Rényi entropy with an alpha of two and is sometimes just called "Rényi entropy," refers to the case ,

where and are independent and identically distributed.

#### min-entropy

The min-entropy (Gray, 1990) is the Rényi entropy as the limit of alpha approaches infinity. The name min-entropy stems from the fact that it is the smallest entropy measure in the Rényi family of entropies. In the limit as , the Rényi entropy :

Equivalently, the min-entropy is the largest real number b such that all events occur with probability at most .

#### Kullback-Leibler divergence

[Kullback-Leibler divergence](https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence) (Gray, 1990) is a non-symmetric measure of the difference between two probability distributions. The Kullback-Leibler measure goes by several names: relative entropy, discrimination information, Kullback-Leibler (KL) number, directed divergence, informational divergence, and cross entropy. Kullback-Leibler divergence is a measure of the difference between the observed entropy and its excepted entropy. We calculate the KL divergence by weighting one distribution (like an observed frequency distribution) by the log of probabilities of some other distribution D2. For discrete probability distributions P and Q, the Kullback–Leibler divergence of Q from P is defined to be

In words, it is the expectation of the logarithmic difference between the probabilities P and Q, where the expectation is taken using the probabilities P.

#### Mutual Information

[Mutual information](https://en.wikipedia.org/wiki/Mutual_information) (Gray, 1990) quantifies the mutual dependence of the two random variables. It is a measure of the “stickiness” between two items. It measures how much knowing one of these variables reduces uncertainty about the other. We can use mutual information to quantify the association between two tags. Mutual information (Equation 10) is given by:

the mutual information of two discrete random variables X and Y can be defined as:

where is the joint probability distribution function of and , and and are the marginal probability distribution functions of and respectively. In the case of continuous random variables, the summation is replaced by a definite double integral:

where is now the joint probability density function of and , and ) and are the marginal probability density functions of and respectively.

# Zipf's law

[Zipf's law](https://en.wikipedia.org/wiki/Zipf%27s_law) is an empirical law formulated using mathematical statistics, refers to the fact that many types of data studied in the physical and social sciences can be approximated with a Zipfian distribution, one of a family of related discrete power law probability distributions. It states that while only a few words are used very often, many or most are used rarely,

where is the frequency of a word ranked nth and the exponent a is almost 1. This means that the second item occurs approximately 1/2 as often as the first, and the third item 1/3 as often as the first, and so on. The is a so-called "power law" distributon.

The law is very common when looking a the distributions of words in all lagnuages. In fact, it is named after the American linguist [George Kingsley Zipf](https://en.wikipedia.org/wiki/George_Kingsley_Zipf) when he observed this law in 1935 in his academic studies of word frequency.

# Text-mining pipeline

Common Natural language processing tasks, such as tokenization, stemming, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, word disamiguation, summarization, and coreference resolution are often done on every new text document before other types of analysis. As such it is common to write a "pipeline" (i.e. a sequence of scripts) that automatically perform these tasks.

*Tokenization*

[Tokenization](http://nlp.stanford.edu/IR-book/html/htmledition/tokenization-1.html) is the task of chopping it up into pieces, called tokens. This is often done by throwing away certain characters, such as punctuation.

*Stemming*

[Stemming](https://en.wikipedia.org/wiki/Stemming) reduces words to their word stem, base or root form. For example, "argue", "argued", "argues", "arguing", and "argus" reduce to the stem "argu." The can allow for "argue", "argued", "argues" to be counted as a single word.

*Sentence segmentation*

[sentence segmentation](http://www.monlp.com/2012/03/13/segmenting-words-and-sentences/) is the process plitting text into words and sentences. This is often done by looking for certain punctuation (full stop, question mark, exclamation mark, etc.).

*Part-of-speech tagging*

Given some text [part-of-speech tagging](https://en.wikipedia.org/wiki/Part-of-speech_tagging) determines the [part of speech](https://en.wikipedia.org/wiki/Part_of_speech) for each word. (e.g. a noun, verb, adjective. etc.)

*Named entity extraction*

[Named entity recognition](https://en.wikipedia.org/wiki/Named-entity_recognition) determines which items in the text map to proper nouns, such as people or places, or things.

*Chunking*

Chunking is also called shallow parsing. It is the identification short meaningful n-grams (like noun phrases, named entities, collocations, etc.). In this processing of a tweet, it could refer to sperating the hashtags, the links and the tweet.

*Parsing*

[Parsing](https://en.wikipedia.org/wiki/Parsing) or syntactic analysis generates [<https://en.wikipedia.org/wiki/Parsing>] the parse tree (grammatical analysis) for each sentence or fragment.

*Word-sense disambiguation*

Many words have more than one meaning. [Word-sense disambiguation](<https://en.wikipedia.org/wiki/Word-sense_disambiguation>] selectx the meaning which makes the most sense in context.

*Coreference resolution*

Given a sentence or larger chunk of text, [coreference resolution](https://en.wikipedia.org/wiki/Coreference) determines which words ("mentions") refer to the same objects ("entities")

*Automatic summarization*

Automatic summarization produce a readable summary of a chunk of text. The summary need not be grammatical, for example, a [tag cloud](https://en.wikipedia.org/wiki/Tag_cloud) could be thought of as a visual summarization of some text.

Which elements are part of a Text-mining pipeline depend on the application.

# Small Talk and the Social Web

One particularly interesting aspect of the social web is the nature of text made available for public consumption. From the time of the Gutenberg printing press until the advent of Web 2.0, nearly all text presented in public was written by professionals. Whether it was a book, a business or government record, a sermon, a news or opinion article, a scientific paper, or an advertisement, it was written by a professional with the intent to communicate information and/or ideas. Not until the social web and Twitter did musings about what a noncelebrity ate for breakfast or whether someone likes naps was widely available for public consumption.

Musings about one’s foot fungus or a statement like “ Hahahahahaha!!!! You should have come to NKLA!!! So many beautiful pitties! And pittie lovers....” are what we call small talk. Small talk is light, intimate banter, often understandable only by the authors’ close friends. A lot of the communication on the social web is small talk, even though it was very rare in public writing prior to the social web.

This free very unstructured nature of language on the social web should be taken into account when mining these data. It can add a considerable amount of noise and magnifies the importance of good semantic and lexcial resources.

# Assingment

Consider the text from Dr. Seuss below:

You have brains in your head. You have feet in your shoes. You can steer yourself in any direction you choose. You're on your own, and you know what you know. And you are the guy who'll decide where to go.

* Dr. Seuss

Perform the following tasks (either by hand or in R):

\* Create a term by document matrix for the Dr. Seuss quote. Assume each sentence is a new documment.   
\* Calculate the td-idf for three terms in the text. Assume each sentence is a new documment.   
\* Write a regular expression to segment the Dr. Seuss quote in to seperate sentences.   
\* Write a regular expression to tokenize the Dr. Seuss quote.   
\* Create a frequency signature for the Dr. Seuss quote. Assume each sentence is a new documment.

# Resources

* AG's news corpus (2013). AG's corpus of news articles. Can be retrieved from <http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html>
* arXiv (2013) arXiv Bulk Data Access Can be retrieved from <http://arxiv.org/help/bulk_data>
* Common Crawl Can be retrieved from <http://commoncrawl.org/>
* Evert, Stefan 2004. The Statistics of Word Cooccurrences: Word Pairs and Collocations. PhD dissertation, University of Stuttgart. (www.collocations.de)
* Fellbaum, Christiane . 1998. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
* Gray, J., Chambers, L., & Bounegru, L. 2012. The Data Journalism Handbook: O'Reilly Media, Inc.
* Last.fm (2013). Last.fm music tags. Can be retrieved from <http://www.last.fm/charts/toptags>
* Manning, Christopher D. ; Raghavan, Prabhakar; Schuetze, Hinrich. 2008. Introduction to Information Retrieval The MIT Press
* Manning, Christopher D. ; Schuetze, Hinrich. 1999. Foundations of Statistical Natural Language Processing The MIT Press
* Miller, George A. 1995. WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.
* pROC. Can be retrieved from <http://web.expasy.org/pROC/>
* PubMed/Medline (2013) PubMed XML Data Can be retrieved from <http://www.nlm.nih.gov/databases/journal.html>
* R Package ‘lda’. Can be retrieved from <http://cran.r-project.org/web/packages/lda/lda.pdf>
* R Package ‘topicmodel’. Can be retrieved from <http://cran.r-project.org/web/packages/topicmodels/vignettes/topicmodels.pdf>
* R text mining package tm. Can be retrieved from <http://cran.r-project.org/web/packages/tm/vignettes/tm.pdf>
* Salton G; McGill MJ 1986. Introduction to modern information retrieval. McGraw-Hill. ISBN 0-07-054484-0.
* Smith, G. 2008. Tagging: People-powered Metadata for the Social Web.
* Stefan Evert. 2013. UCS toolkit Can be retrieved from <http://www.collocations.de/software.html>
* Twitter (2013). REST API v1.1 Resources. Can be retrieved from <https://dev.twitter.com/docs/api/1.1>
* Twitter Search API Can be retrieved from <https://dev.twitter.com/docs/api/1/get/search>
* Twitter Streaming APIs. Can be retrieved from <https://dev.twitter.com/docs/streaming-apis>
* UC Irvine Machine Learning Repository (2013). Spambase. Can be retrieved from <http://archive.ics.uci.edu/ml/datasets/Spambase>
* Wikipedia. 2013. Data Dump Can be retrieved from <http://en.wikipedia.org/wiki/Wikipedia:Database_download>