Week 5: Text Mining

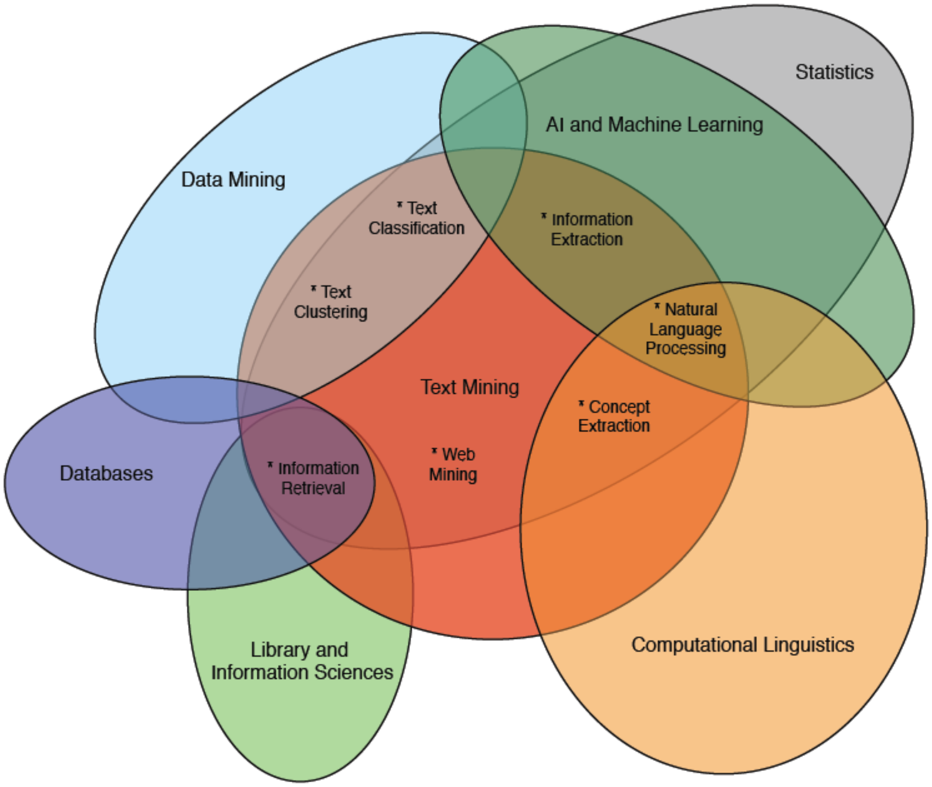
Text mining evolves rapidly as the results of information in the form of unstructured data such as social media, online discussion, reviews, emails, surveys, web pages, web blogs, digital books, corporate documents, technical documents, etc. are growing enormously. It is approximated that 80% of the new data uploaded to the Internet is in unstructured formats (Coronel/Morris: Database Systems). This information is complicated to work with but it contains insight and valuable information. Leveraging text could lead to better decision and prediction.

Text mining [Manning et.al. 2005] "is the process of compiling, organizing, and analyzing large document collections to support the delivery of target types of information to analysts and decision makers and to discover relationships between related facts that span wide domains of inquiry"

Text mining Applications:

- Sentiment analysis (e.g. twitter, Facebook)
- Spam detection, Email and news filtering
- Biomedical text analysis (e.g. association analysis in text collections to conclude medical cause and effect)
- Automated calendar updating using data from e-mail extraction
- Classifying news or web pages to appropriate categories
- Clustering similar documents
- Suggestions for auto-completed words
- Marketing research
- Analyzing open-ended survey responses for opinions and feeling
- Automatic processing of messages, emails, etc. Text such as messages and emails that are filtered or classified as junk mail are automatically discarded based on the certain words or terms appeared in the messages. In addition, legitimate messages can be automatically forwarded to the appropriate departments. In addition, inappropriate words in the messages are screened and returned to the sender to remove those offending contents.
- Analyzing warranty or insurance claims, diagnostic interviews.
  The information in the textual form can be exploited such as pinpointing common complaints or problems on certain automobiles, or patient symptoms that are useful for the medical diagnosis.
- Investigating competitors by crawling their web sites
  Contents of web pages including the links are crawled and processed. The activities of competitors could be analyzed for their capabilities.

Text mining borrows techniques and combines methodologies from many fields such as information retrieval (library and information sciences), natural language processing (computational linguistics), data mining, machine learning, statistics, and databases. The following figure shows the relationship between text mining and other fields.



Source: Practical Text Mining [Miner et.al 2012]

Both text mining and information retrieval (IR) have the same focus on dealing with text. Thus, several algorithms and methods from text mining are borrowed from IR. Nevertheless, the ultimate objective of IR is to retrieve documents that best match the query. On the other hand, text mining's goal is to find insights or discover new knowledge in the text collections using statistical relations, lexical, and semantics of the text.

Text mining is difficult because language is ambiguous such as same words may have different meanings (e.g. root) but different words can mean the same thing, word-level ambiguity (e.g. design can be a noun or a verb), various spellings (e.g. colour, color), abbreviations, misspellings. In addition, extracting words (concepts) usually create very huge dimensions, possibly thousands new fields but each field is sparse (primarily with zeros as elements in the matrix).

Ambiguity is the nature characteristic of the free text. Understanding text semantics can facilitate the way particular words should be processed. Semantics refers to the study of meaning to understand human expressions through language.
The summary of semantics concepts:

- Antonymy – words that have opposite meaning
  Example: graded antonym (big vs. small, fast vs. slow, good vs. bad)
  Complementary pairs (male vs. female, present vs. absent)
  Relational pairs (buy vs. sell, pull vs. push)

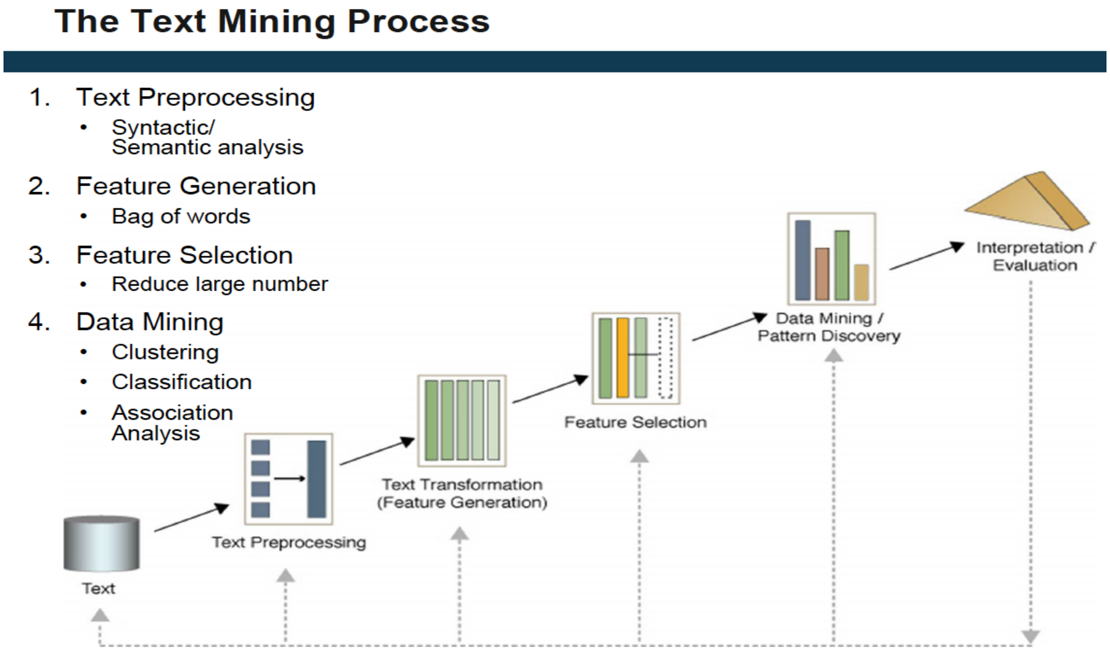
In English, there are many ways to form antonyms such as add prefix {un}, {non}, {in}, {mis}, {dis}, etc.

- Synonymy – different words but similar or same meaning
  Example: a) big = large
  b) small = little
- Polysemy – words that have multiple meanings
  Example crane: 1) a bird 2) a construction equipment 3) to strain out one's neck
- Hyponymy – concept hierarchy or subclass (subordinated). It is the relationship between general and specific term.
  Example beverage (general): tea, coffee, milk
  musical instrument (general): piano, violin, guitar, flute, etc.
  The hierarchical diagram (or taxonomy) is the relationship between general term and the specific instances.
- Idiom – a group of words, in which the meanings cannot be inferred from each individual word that made it.
  Example: give it a shot (try)
  a piece of cake (very easy)
  see eye to eye (agree)
  bee in her bonnet (she is upset)

According to Miner et.al [2012], seven practice areas of text mining contain:

- Search and information retrieval (IR) engage in storage and retrieval of text documents, including keyword search and search engines.
- Document classification based on grouping terms, snippets, paragraphs, or documents using data mining classification methods (with labeled examples)
- Document clustering related to categorized terms, snippets, paragraphs or documents by utilizing data mining clustering methods (no labeled examples).
- Web mining centers on text data on the Internet or interconnectedness of the web.
- Information extraction involves in extracting related or relevant facts from unstructured text. This includes making structured data from unstructured and semi-structured.
- Natural language processing (NLP) or computational linguistics is the low-level language processing.
- Concept extraction based on words and phrases grouping into semantically similar groups.

The text mining process consists of a) text preprocessing b) text transformation c) feature selection d) data mining/pattern recovery and e) interpretation and evaluation. Notice that the process is similar to CRISP-DM and can be illustrated as:



The Text Mining Process

1. Text Preprocessing
   - Syntactic/ Semantic analysis
2. Feature Generation
   - Bag of words
3. Feature Selection
   - Reduce large number
4. Data Mining
   - Clustering
   - Classification
   - Association Analysis

Text
Text Preprocessing
Text Transformation (Feature Generation)
Feature Selection
Data Mining / Pattern Discovery
Interpretation / Evaluation

Source: Paulheim et.al [2014]

1) Text Preprocessing

Generally, a document is treated as a bag of words or terms. The order of the words is ignored. In the vector space model, each document is represented as a vector. Each term is weighted based on its frequency in the documents.

For example, "Tom likes fruit. Tim likes fruit and vegetable." Based on these two texts and bag of words approach, the results are: Tom 1, likes 2, fruit 2, vegetable 1, Tim 1.

Text preprocessing activities include:

1.1) Tokenization, which refers to splitting text (or stream of characters) into single word or n-grams. In English, each word is separated by white space. Words may contain special characters such as comma (,), apostrophe ('), or dash (-). Some languages such as Chinese, Japanese, and Korean do not have white space.

Syntactic/linguistic text analysis may incorporate which can be simple or advance analysis. Simple syntactic analysis is based on text clean up (e.g. remove HTML tags, remove punctuation); normalize case (e.g. lower case), separate text into single word or n-grams. Advanced linguistic analysis emphasizes on word sense dis-ambiguity such as normalized synonyms (e.g. United States, USA, US), normalize pronouns (she -> Melissa) and part of speech (POS) tagging. In English, POS set includes noun, verb, adjective,

adverb, preposition, and conjunctions. The function of each term is determined. For example, I (noun), drink (verb), water (noun), quickly (adv.).
Most of English words have one POS. However, some common English words can be ambiguous. Thus, it is possible to have multiple tagging results.
Other things to consider:
- unfold abbreviations such as “2moro” to “tomorrow”
- slang words should be replaced with the standard English
- clean up transcription and typing error (e.g. ca n’t, stuyd)
- synonym normalization using catalogs such as WordNet (http://wordnet.princeton.edu). WordNet is a lexical database for the English language. It resembles a thesaurus. WordNet’s structure is a useful tool for natural language processing.

1.2) Stop words removal

Words such as “*a, an, the, to, in, on, or, as, and, with, be, by” etc*. are used very often but do not carry useful information. Therefore, these words (aka stop words) will be removed. Removing stop words results in reducing dimensionality. When irrelevant features are reduced, the efficiency improves.

1.3) Stemming

Stemming is a process to remove prefixes and suffixes to its stem or root form. Thus, different term variations can be represented with one single representation. The stemming process includes:

a) inflectional stemming such as removing plural (s), verb tenses (e.g. past tense (ed), present-progressive (ing)) and transform words.
For example, talks, talked, talking (inflected form) -> talk (base form)
goes, went, gone (inflected form) -> go (base form)
b) Stemming to root refers to reducing word into the most basic form.
For example: computer, computing, computerized -> compute
apply, applications, reapplied -> apply
Exception case: department and depart.

The most common stemmer is the Porter stemmer. The official website for the Porter stemmer algorithm is at: http://tartarus.org/~martin/PorterStemmer/.
For the JavaScript Porter Stemmer Online: http://9ol.es/porter_js_demo.html

The benefits of stemming include reducing index sizes, decreasing storage space, and increasing text mining effectiveness.

A large amount of effort is devoted to this preprocessing step. It is important to mention that no single algorithm works perfectly.

2) Feature generations (e.g. bag of words)

A document is considered as a bag of words (terms). Each word represents a feature. The order of words ***is not important*** so the order can be ignored.

Usually, a document is represented by a vector. There are different methods for creating a vector such as:

a) Boolean model – A document is represented by a set of keywords. Each keyword attribute is Boolean, which signifies whether the term appears in the document (e.g. 1/0 for presence/absence). Suppose there are *t* terms (T), and *n* documents (D). The matrix form is:

$$\left[\begin{array}{cccccc} & T_1 & T_2 & T_3 & \dots & T_t \\ D_1 & 1 & 0 & 1 & \dots & 0 \\ D_2 & 1 & 0 & 1 & \dots & 1 \\ D_3 & 0 & 1 & 0 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ D_n & 1 & 1 & \dots & \dots & 1 \end{array}\right]$$

b) Vector space model – A collection of n documents is represented by a term-document matrix. Suppose there are *t* terms (T), and *n* documents (D).

i) Term document count matrices
Each document is a count vector.

$$\left[\begin{array}{cccccc} & T_1 & T_2 & T_3 & \dots & T_t \\ D_1 & 45 & 0 & 91 & \dots & 34 \\ D_2 & 1 & 18 & 1 & \dots & 1 \\ D_3 & 30 & 7 & 6 & \dots & 9 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ D_n & 7 & 44 & \dots & \dots & 28 \end{array}\right]$$

ii) tf-idf weighting

$$\left[\begin{array}{cccccc} & T_1 & T_2 & T_3 & \dots & T_t \\ D_1 & w_{11} & w_{21} & w_{31} & \dots & w_{t1} \\ D_2 & w_{12} & w_{22} & w_{32} & \dots & w_{t2} \\ D_3 & w_{13} & w_{23} & w_{33} & \dots & w_{t3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ D_n & w_{1n} & w_{2n} & \dots & \dots & w_{tn} \end{array}\right]$$

Each entry in the matrix $(w_{11} \dots w_{tn})$ corresponds to the term weighting in the document. TF-IDF (term-frequency – inverse document frequency) is a conventional weighting schema. The concept is a term is considered important if that term appears frequently in a single document but less frequent in a text collection (corpus). There are many variations of weighting schemes.

$$TF - IDF = TF * IDF$$

TF measures how often a term appears in a document. Since documents can vary in length, it is usually divided by the document length (number of terms in that document)

$$TF(t) = \frac{Number\ of\ times\ a\ term\ (t) appears\ in\ a\ document}{Total\ number\ of\ terms\ in\ the\ document}$$

$IDF(t)$ measures the importance of the term. Some terms such as “a, an, the, is…” can appear often but not very important. Thus, these terms should be weighted down. On the other hand, the infrequent term or rare term should be weighted up.

$$IDF\ (t) = log_2\ \frac{total\ number\ of\ document\ in\ the\ collection(corpus)}{Number\ of\ documents\ with\ term\ (t)}$$

$$IDF\ (t) = log_2\ \frac{N}{DF(t)}$$

N is the total number of documents in the collection (corpus)
DF is the number of documents containing term (t)

For example:
A document contains 200 words, and a term (t) appears 40 times. So,

$TF = \frac{40}{200} = \frac{1}{5}$

Also, assume that a collection consists of 100000 documents, and a term t appears in 2000 documents.

$IDF\ (t) = log_2\ \frac{100000}{2000} = 5.64$

$TF*IDF = \frac{1}{5} * 5.64 = 1.128$

N-gram is a contiguous sequence of items, where items can be phonemes, syllables, letters, words, or base pair depending on the applications. Here n indicates the size. 1-gram is referred to as a unigram, 2-gram is known as bigram or digram, 3-gram is trigram, etc.

For instance, split this string “iPhone” into:
1-gram (unigram): e.g., “i”, “P”, “h”, “o”, “n”, “e”
2-grams (digrams or bigrams): e.g. “iP”, “Ph”, “ho”, etc.
3-grams (trigrams): e.g. “iPh”,”Pho”,”hon”, etc.

The probability of a sequence (e.g. letters, words) allows us to predict the next sequence. More information about computing probability of n-grams: https://class.coursera.org/nlp/lecture/14

3) Feature selection involves reducing the number of features since not all features are useful, but misleading and redundant. Selecting a subset of features to represent a document can improve text presentation and improve efficiency of the algorithms.
High dimensionality is a tough task for learning algorithms. Features can be reduced through methods such as principle component analysis (PCA), and singular value decomposition (SVD). The concept is reducing the dimensionality of the input matrix (number of input documents by number of extracted words) to a lower dimension, where each consecutive dimension represents the largest degree of variability (differences between words and documents) possible.

According to Callan, J.A. (Text data mining 2004, CMU), there are two approaches for feature selections.

a) select features before using them in the methods (e.g. classifier). Features are independent of the methods.
b) select features based on how well they work in the method (e.g. classifier) Here, the classifier is part of the feature selection process and it is an iterated process.

4) Data mining
Traditional data mining techniques are employed.

a) Clustering – find the document similarity in a given set of documents using similarity measurements such as Jaccard coefficient and cosine similarity. The clustering algorithm such as K-means is trained with no-labeled documents.

Similarity Measure: the examples come from [Paulheim, et.al 2014]*:*

i) Jaccard Coefficient

$$dist(x_i, x_j) = \frac{M_{11}}{M_{01} + M_{10} + M_{11}}$$

In simple words, the Jaccard distance is calculated from the number of matching terms divided by number of attributes without 00 matches
Example:

- Sample documents set:

D1 = “Saturn is the gas planet with rings”
D2 = “Jupiter is the largest gas planet”
D3 = “Saturn is the Roman god of sowing”

- Documents as vectors

Vector structure:
(Saturn, is, the, gas, planet, with, rings, Jupiter, largest, Roman, god, sowing)

D1: 111111100000
D2: 011110011000
D3: 111000000111

- sim (d1,d2) = 0.44
- sim (d1,d3) = 0.30
- sim (d2,d3) = 0.20

ii) Cosine similarity

$$\cos(x, y) = \frac{x.y}{||x||||y||}$$

The numbers used here are randomly chosen for illustration purposes.
D1 = 3 2 0 5 0 0 0 2 0 0
D2 = 1 0 0 0 0 0 0 1 0 2
D1*D2 = 3*1 + 2*0 + 0*0 +5*0 + 0*0 +0*0 +0*0 + 2*1 + 0*0 + 0*2 = 5
||D1|| = $(3*3 + 2*2 + 0*0 + 5*5 + 0*0 + 0*0 + 0*0 + 2*2 + 0*0 + 0*0)^{0.5}$ = 6.481
||D2|| = $(1*1+ 0*0+ 0*0 + 0*0 + 0*0 + 0*0 + 0*0 + 1*1 + 0*0 + 2*2)^{0.5}$=2.449
cos (D1,D2) = 0.315

It is also possible to combine cosine with weighting scheme such as TF-IDF.

b) Classification – assign unseen documents to the pre-determined classes as accurate as possible. Text classification methods include Naïve Bayes, support vector machine, KNN, decision trees.
Sentiment analysis is an example of a text mining classification task. Here, the sentimental targets could be positive, negative, or neutral. The training data set is used for training the model. In supervised classification tasks, training data could be in the form of pairs of features and labels such as <text, sentiment>. The test set is used to evaluate the performance of the model. Usually, the train set and test set should not overlap.

5) Interpretation and Evaluation
The process can be terminated if the result is applicable for the problem at hand. Iterate the process as necessary if results are not convincing.

Here are the lists of the free, open source and commercial text mining/text analytics commercial software: http://www.kdnuggets.com/software/text.html

Reference:

Hearst, Marti (Oct 17, 2003). *What is Text Mining?* SIMS, UC Berkeley. Access on Dec, 2014. Retrieved from: http://people.ischool.berkeley.edu/~hearst/text-mining.html

Manning, C. Raghavan, P. (2005) *Web Search and Mining CS276B*. Stanford University. Retrieved from: http://web.stanford.edu/class/cs276b/handouts/lecture10.pdf

Miner, G., Delen, D, Elder, J. Fast, A, Hill, T, and Nisbet, A.R. (2012) *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. Elsevier Inc.

Paulheim, H. Bryl, V. Meusel, R. and Lehmberg, O.(2014) *Data Mining* | *Text Mining*. University of Mannheim. Retrieved from: http://dws.informatik.uni-mannheim.de/fileadmin/lehrstuehle/ki/Lehre/DataMining1/HWS2014/DM06-TextMining-HWS2014-V2.pdf

Stavrianou, A. Andritsos, P. and Nicoloyannis (2007). Overview and Semantic Issues of Text Mining. SIGMOD Record, Sep 2007 (Vol. 36, No. 3) Retrieved from: http://www.cs.toronto.edu/~periklis/pubs/sigrec07.pdf

Text Mining (Big Data, Unstructured Data) (2014). Statsoft.com. Retrieved from: http://www.statsoft.com/Textbook/Text-Mining/button/3

Zhu, X. (2010*) Basic Text Process*. CS769 Advanced Natural Language Processing. Retrieved from: http://pages.cs.wisc.edu/~jerryzhu/cs769/text_preprocessing.pdf