

KDnuggets

[Subscribe to KDnuggets](#)[Submit a blog](#)

- [Blog](#)
- [Opinions](#)
- [Tutorials](#)
- [Top stories](#)
- [Courses](#)
- [Datasets](#)
- [Education: Online](#)
- [Certificates](#)
- [Events / Meetings](#)
- [Jobs](#)
- [Software](#)
- [Webinars](#)

[KDnuggets Home](#) » [News](#) » [2017](#) » [Feb](#) » [Tutorials, Overviews](#) » Natural Language Processing Key Terms, Explained ([17:n07](#))

Natural Language Processing Key Terms, Explained

[<= Previous post](#)[Next post =>](#)

http likes 421

Like 156

Share 156

Tweet

Share

Share

237

Tags: [Explained](#), [Information Retrieval](#), [Key Terms](#), [Natural Language Processing](#), [NLP](#), [Semantic Analysis](#), [Sentiment Analysis](#)

Natural language processing (NLP) concerns itself with the interaction between natural human languages and computing devices. NLP is a major aspect of computational linguistics, and also falls within the realms of computer science and artificial intelligence.

2. Tokenization

Tokenization is, generally, an early step in the NLP process, a step which splits longer strings of text into smaller pieces, or **tokens**. Larger chunks of text can be tokenized into sentences, sentences can be tokenized into words, etc. Further processing is generally performed after a piece of text has been appropriately tokenized.

3. Normalization

Before further processing, text needs to be normalized. Normalization generally refers to a series of related tasks meant to put all text on a level playing field: converting all text to the same case (upper or lower), removing punctuation, expanding contractions, converting numbers to their word equivalents, and so on. Normalization puts all words on equal footing, and allows processing to proceed uniformly.

4. Stemming

Stemming is the process of eliminating affixes (suffixes, prefixes, infixes, circumfixes) from a word in order to obtain a word stem.

running → run

5. Lemmatization

Lemmatization is related to stemming, differing in that lemmatization is able to capture canonical forms based on a word's [lemma](#).

For example, stemming the word "better" would fail to return its citation form (another word for lemma); however, lemmatization would result in the following:

better → good

It should be easy to see why the implementation of a stemmer would be the less difficult feat of the two.

6. Corpus

In linguistics and NLP, corpus (literally Latin for *body*) refers to a collection of texts. Such collections may be formed of a single language of texts, or can span multiple languages -- there are numerous reasons for which multilingual corpora (the plural of corpus) may be

useful. Corpora may also consist of themed texts (historical, Biblical, etc.). Corpora are generally solely used for statistical linguistic analysis and hypothesis testing.

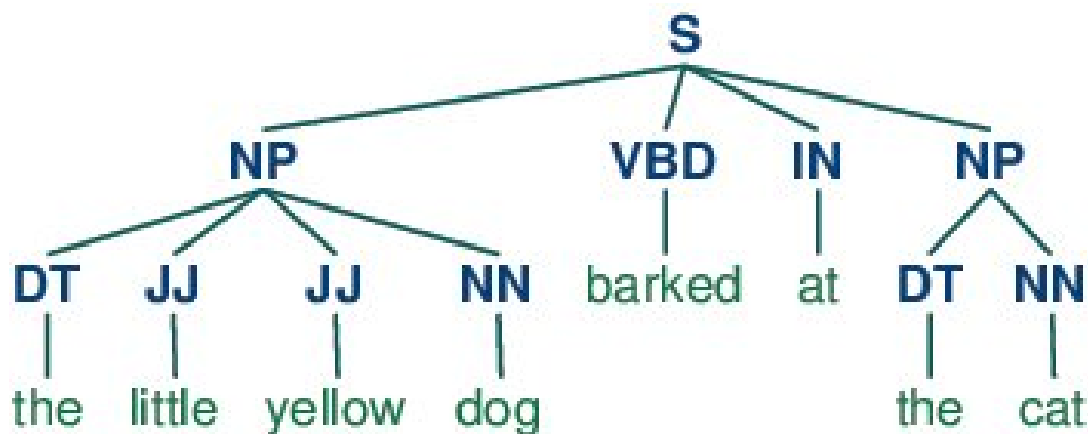
7. Stop Words

Stop words are those words which are filtered out before further processing of text, since these words contribute little to overall meaning, given that they are generally the most common words in a language. For instance, "the," "and," and "a," while all required words in a particular passage, don't generally contribute greatly to one's understanding of content. As a simple example, the following [panagram](#) is just as legible if the stop words are removed:

~~The~~ quick brown fox jumps over ~~the~~ lazy dog.

8. Parts-of-speech (POS) Tagging

POS tagging consists of assigning a category tag to the tokenized parts of a sentence. The most popular POS tagging would be identifying words as nouns, verbs, adjectives, etc.



9. Statistical Language Modeling

Statistical Language Modeling is the process of building a statistical language model which is meant to provide an estimate of a natural language. For a sequence of input words, the model would assign a probability to the entire sequence, which contributes to the estimated likelihood of various possible sequences. This can be especially useful for NLP applications which generate text.

10. Bag of Words

Bag of words is a particular representation model used to simplify the contents of a selection of text. The bag of words model omits grammar and word order, but is interested in the number of occurrences of words within the text. The ultimate representation of the

text selection is that of a bag of words (**bag** referring to the set theory concept of [multisets](#), which differ from simple sets).

Actual storage mechanisms for the bag of words representation can vary, but the following is a simple example using a dictionary for intuitiveness. Sample text:

"Well, well, well," said John.

"There, there," said James. "There, there."

The resulting bag of words representation as a dictionary:

```
{  
  'well': 3,  
  'said': 2,  
  'john': 1,  
  'there': 4,  
  'james': 1  
}
```

11. n-grams

n-grams is another representation model for simplifying text selection contents. As opposed to the orderless representation of bag of words, n-grams modeling is interested in preserving contiguous sequences of N items from the text selection.

An example of trigram (3-gram) model of the second sentence of the above example ("There, there," said James. "There, there.") appears as a list representation below:

```
[  
  "there there said",  
  "there said james",  
  "said james there",  
  "james there there",  
]
```

12. Regular Expressions

Regular expressions, often abbreviated *regexp* or *regex*, are a tried and true method of concisely describing patterns of text. A regular expression is represented as a special text string itself, and is meant for developing search patterns on selections of text. Regular expressions can be thought of as an expanded set of rules beyond the wildcard characters of

? and *. Though often cited as frustrating to learn, regular expressions are incredibly powerful text searching tools.

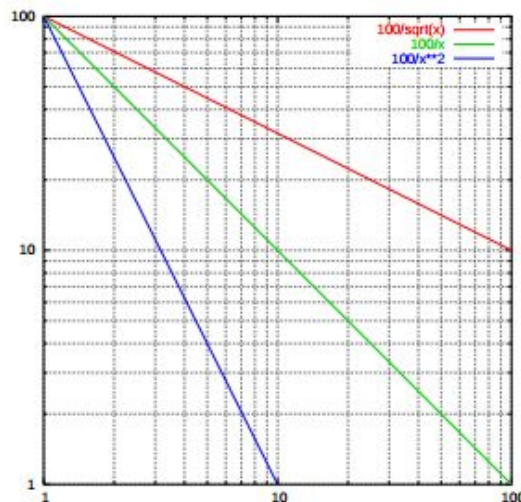
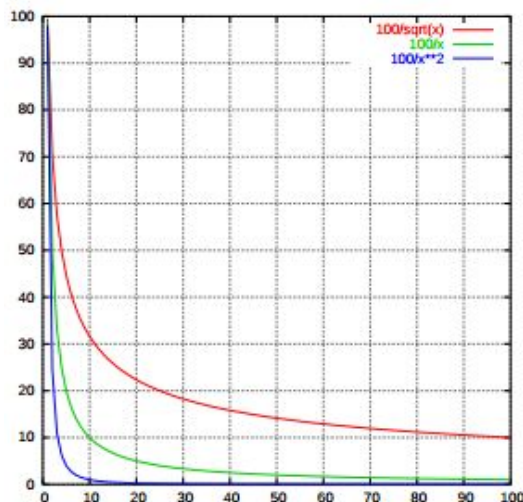
13. Zipf's Law

Zipf's Law is used to describe the relationship between word frequencies in document collections. If a document collection's words are ordered by frequency, and y is used to describe the number of times that the x th word appears, Zipf's observation is concisely captured as $y = cx^{-1/2}$ (item frequency is inversely proportional to item rank). More generally, [Wikipedia says](#):

Zipf's law states that given some corpus of natural language utterances, the frequency of any word is inversely proportional to its rank in the frequency table. Thus the most frequent word will occur approximately twice as often as the second most frequent word, three times as often as the third most frequent word, etc.

$$y = cx^{-k} \quad (k=1/2, 1, 2)$$

$$\log_{10} y = -k * \log_{10} x + \log_{10} c$$



14. Similarity Measures

There are numerous similarity measures which can be applied to NLP. What are we measuring the similarity of? Generally, strings.

- **Levenshtein** - the number of characters that must be deleted, inserted, or substituted in order to make a pair of strings equal
- **Jaccard** - the measure of overlap between 2 sets; in the case of NLP, generally, documents are sets of words
- **Smith Waterman** - similar to Levenshtein, but with costs assigned to substitution, insertion, and deletion

15. Syntactic Analysis

Also referred to as **parsing**, syntactic analysis is the task of analyzing strings as symbols, and ensuring their conformance to a established set of grammatical rules. This step must, out of necessity, come before any further analysis which attempts to extract insight from text -- semantic, sentiment, etc. -- treating it as something beyond symbols.

16. [Semantic Analysis](#)

Also known as **meaning generation**, semantic analysis is interested in determining the meaning of text selections (either character or word sequences). After an input selection of text is read and parsed (analyzed syntactically), the text selection can then be interpreted for meaning. Simply put, syntactic analysis is concerned with what words a text selection was made up of, while semantic analysis wants to know what the collection of words actually **means**. The topic of semantic analysis is both broad and deep, with a wide variety of tools and techniques at the researcher's disposal.

17. [Sentiment Analysis](#)

Sentiment analysis is the process of evaluating and determining the sentiment captured in a selection of text, with sentiment defined as feeling or emotion. This sentiment can be simply positive (happy), negative (sad or angry), or neutral, or can be some more precise measurement along a scale, with neutral in the middle, and positive and negative increasing in either direction.



18. [Information Retrieval](#)

Information retrieval is the process of accessing and retrieving the most appropriate information from text based on a particular query, using context-based indexing or

metadata. One of the most famous examples of information retrieval would be Google Search.

Related:

- [Machine Learning Key Terms, Explained](#)
- [Deep Learning Key Terms, Explained](#)
- [Deep Learning Research Review: Natural Language Processing](#)

[<= Previous post](#)

[Next post =>](#)

Top Stories Past 30 Days

Most Popular

1. [How to create stunning visualizations using python from scratch](#)
2. [We Don't Need Data Scientists, We Need Data Engineers](#)
3. [How to Get Your First Job in Data Science without Any Work Experience](#)
4. [Telling a Great Data Story: A Visualization Decision Tree](#)
5. [Data Science vs Business Intelligence, Explained](#)

Most Shared

1. [We Don't Need Data Scientists, We Need Data Engineers](#)
2. [How to Get Your First Job in Data Science without Any Work Experience](#)
3. [How to create stunning visualizations using python from scratch](#)
4. [All Machine Learning Algorithms You Should Know in 2021](#)
5. [Data Science vs Business Intelligence, Explained](#)

[Latest News](#)

- [5 Supporting Skills That Can Help You Get a Data Scienc...](#)
- [6 Web Scraping Tools That Make Collecting Data A Breeze](#)
- [The Difficulty of Graph Anonymisation](#)
- [How Reading Papers Helps You Be a More Effective Data S...](#)

- [Why Do Machine Learning Projects Fail?](#)
- [Pandas Profiling: One-Line Magical Code for EDA](#)

Top Stories Last Week

Most Popular

1. [We Don't Need Data Scientists, We Need Data Engineers](#)
2. [Telling a Great Data Story: A Visualization Decision Tree](#)
3. [Cartoon: Data Scientist vs Data Engineer](#)
4. [Data Science vs Business Intelligence, Explained](#)
5. [Approaching \(Almost\) Any Machine Learning Problem](#)



Most Shared

1. [We Don't Need Data Scientists, We Need Data Engineers](#)
2. [Evaluating Deep Learning Models: The Confusion Matrix, Accuracy, Precision, and Recall](#)
3. [Telling a Great Data Story: A Visualization Decision Tree](#)
4. [Approaching \(Almost\) Any Machine Learning Problem](#)
5. [Feature Store as a Foundation for Machine Learning](#)

More Recent Stories

- [Pandas Profiling: One-Line Magical Code for EDA](#)
- [KDnuggets 21:n08, Feb 24: Powerful Exploratory Data Analysis...](#)
- [Using NLP to improve your Resume](#)
- [10 Statistical Concepts You Should Know For Data Science Inter...](#)
- [Data Observability, Part II: How to Build Your Own Data Qualit...](#)
- [Top Stories, Feb 15-21: We Don't Need Data Scientists, We Ne...](#)
- [An overview of synthetic data types and generation methods](#)
- [Powerful Exploratory Data Analysis in just two lines of code](#)
- [Inside the Architecture Powering Data Quality Management at Uber](#)
- [Cartoon: Data Scientist vs Data Engineer](#)

- [People Skills for Analytical Thinkers](#)
- [Evaluating Deep Learning Models: The Confusion Matrix, Accuracy, Precision, and Recall \[Silver Blog\]](#)
- [Feature Store as a Foundation for Machine Learning](#)
- [Multidimensional multi-sensor time-series data analysis framework](#)
- [Approaching \(Almost\) Any Machine Learning Problem](#)
- [6 Data Science Certificates To Level Up Your Career](#)
- [Forecasting Stories 5: The story of the launch](#)
- [Distributed and Scalable Machine Learning \[Webinar\]](#)
- [GPT-2 vs GPT-3: The OpenAI Showdown](#)
- [10 resources for data science self-study](#)

[KDnuggets Home](#) » [News](#) » [2017](#) » [Feb](#) » [Tutorials, Overviews](#) » Natural Language Processing Key Terms, Explained ([17:n07](#))

© 2021 KDnuggets. | [About KDnuggets](#) | [Contact](#) | [Privacy_policy](#) | [Terms of Service](#)

[Subscribe to KDnuggets News](#)

X