# NLP Glossary

Unit 1

* **Natural Language Processing:** a subfield of linguistics, computer science, information engineering and artificial intelligence concerned with the interactions between computers and human language
* **Natural Language Understanding:** produce a useful representation of some inputted natural language
* **Natural Language Generation:** transform structured data into natural language
* **Named Entity recognition:** locating entities in unstructured text
* **Document Classification**: categorize documents into one or more classes or topics
* **Sentiment Analysis:** understanding sentiment (views/opinions/emotions) from text
* **Query Segmentation:** partition the queries into semantic units
* **Query Scoping:** map query segments into entity types
* **Query expansion:** broaden a query by adding additional phrases/tokens
* **Query Relaxation:** make the query less restrictive by removing tokens
* **Dialog Systems:** a system that communications with humans using natural language
* **Machine translation:** translate a source text from one language to another
* **Lexical Diversity:** the range of different words used in a text, with a greater range indicating a higher diversity

Unit 2

* Lexical Analysis:
  + Tokenization:
    - Token: word – or sequence of characters
    - Tokenization: convert a string of characters into a sequence of tokens
    - Separator: tabs or spaces
  + Stemming and Lemmatization:
    - Inflections: syntactic differences between words
    - Stemming: reducing a word to its root form
    - Lemmatization: uses lexical knowledge such as WordNet to obtain word base forms
  + Syntactic Analysis
    - Part of Speech tagging: can help determine important keywords in a document and assist in finding usages of a word
    - Parsing and Grammar: identifying verb/noun phrases and their relationships to each other
    - Semantic Analysis: process of understanding natural language – meanings of words and their relations to other words
  + Semantic Search
    - Taxonomy: hierarchical organization of entities (parent-child relationships)
    - Ontology: arbitrary complex relationships between entities
  + Text Corpus: a collection of text sources

Unit 3

* Chunking: classifying and labeling POS tags at major levels
* Deep Parsing: extending parsing to the most granular level
* Word Embeddings
  + Context-free embeddings: combine all the senses of a word into one
  + Contextual embeddings: Generate different embeddings for a word depending on the context/position of a word in a sentence
* Wide Component: a linear model that captures how the co-occurrence of a query-item feature pair correlates with the target label
* Deep Component: user needed – generalize matching items to queries that are close to each other in the embedding space
* Feature Engineering: the process of using domain knowledge to extract “hand crafted” features from raw data via data mining techniques
* Feature Learning> set of techniques that allow a system to automatically discover representations needed for feature detection or classification from raw data
* Top Down: build NLP framework around a set of high-level concepts and categories into which everything fit
* Bottom Up: build NLP by building up a huge list of stemmed keywords form raw text, then roll them up into phrases, tag clouds, hypernym trees and the like
* Leaf level node: semantic tag for an entity that uniquely identifies what it is

Unit 4

NONE

Unit 5

* Stop Words: extremely common “function words that are of little value in helping select documents matching a user’s need
* Inflections: syntactic differences between words (Toy vs Toys)
* Algorithmic stemmer: uses a program to decide whether two words are related
* Dictionary-based stemmer: relies on pre-created dictionaries of related terms to store term relationships
* Automatic Tagging
  + Default tagger: assigns the same tag to each token; good for establishing a baseline
  + Regular Expression Tagger: assigns tokens based on matching patterns
  + Lookup Tagger: find the “x” most frequent words and store their most likely tag
  + Backoff Strategy: words that are not in the lookup list will be assigned “None”
  + Unigram Tagging: uses statistical algorithm to assign the most likely tag for a token
  + N-gram tagger: generalization of the tagger whose context is the current word together with the POS tags of the n-1 preceding tokens

Unit 6

* Graph: a powerful and versatile data structure that easily allows you to represent the relationship between different types of data
  + Nodes: the vertices where the data is stored
  + Connections: the edges which connect the nodes
* Synonym/antonym: a word or phrase that means exactly or nearly the same as another word or phrase in the same language
* Hyponym/hypernym: hyponymy shows the relationship between a generic term (hypernym) and a specific instance of it (hyponym)
* Holonym/meronym: the relationship between a term denoting the whole and a term denoting a part of, or a member of, the whole
* WordNet: example (in limited form) of a semantic network, which means it can tell you relations
* Taxonomy: Hierarchical organization of entities
* Ontology: arbitrary complex relationships between entities
* Structured Sources:
  + Relational databases, data feeds, catalogs, directories
* Unstructured sources:
  + Web pages, news, forums, emails, social media, speech, images, video
* Domain discovery: identifying relevant sites, datasets, and pages
* Crawling: building models of relevant content, identifying new content, overcoming anti-crawling measures

UNIT 7: POS TAGGING

* POS tagging: can help determine important keywords in a document and assist in finding specific usages of a word
* Structure hierarchy:
  + Sentence -> Clause -> Phrase -> Word
  + Word: smallest language unit, has four major categories (noun, verb, adjective, adverb)
  + Phrase: usually a group of two or more words, has 5 major categories (NP, VP, ADJP, ADVP, and prepositional phrase. Shallow parsing is used to extract phrases
* Dependency Grammars
  + Grammar: defines the syntax and structure of language
  + Dependency Grammar (word-based grammar): a word has a relation or depends on another word based on the position of the words in the sentence
  + Directed acyclic graph (DAG): every node in the tree has at most one incoming edge, except the root node
* Constituency Grammars
  + Syntax and rules that govern the hierarchy and order of the various constituents in the sentences
* Combined tagger: create a chain of taggers, and each tagger would fall back on a backoff tagger if it cannot tag the input tokens
* Brill tagger (supervised learning): analyzes corpus/training data to find the set of tagging rules that minimize tagging errors and best define the corpus
* Stochastic taggers
  + Stochastic: models that incorporate frequency and/or probability
  + Frequency-base: assign tag that occurs most frequently with the word in the training data
  + Probability-based: calculate the probability of occurrences of a given sequence of tags
  + Hidden Markov Models: combines frequency and probability-based approaches
* Hidden Markov Models (HMMs)
  + Emission probabilities: probabilities of making certain observations given a particular state
  + Transition probabilities: the probability of transition to another state given a particular state
* Viterbi Algorithm
  + Find the most likely sequence of tags given a set of observations
  + Dynamic Programming: a general technique to solve certain complex search problems by memorization
    - Recursively decompose the large each problem into smaller subproblems that can be solved efficiently
    - Store the solution of each subproblem

UNIT 8: Shallow Parsing

* Phrase: usually groups of two or more words (without both subject and verb)
  + 5 major categories: NP, VP, ADJP, ADVP, and Prepositional Phrases
* POS Tagging: Shallowest – POS tags only
* Shallow Parsing: grouping POS tags into “phrases”, to obtain semantically meaningful phrases and relations
* Chinking: syntax elements not part of chunks
* Full Parsing: complete internal syntax
* Regular Expression Parser: patterns for large-scale text corpus. Can be laborious to construct
* Greedy sequence classifier: analyze training data one sentence at a time, predicts a tag for the first element and store its features and predictions in history
* Dependency grammar: a word has a relation or depends on another word based on the position of the words in the sentence
* Directed acyclic graph (DAG): every node in the tree has at most one incoming edge except root node
* Constituency Grammars: syntax and rules that govern the hierarchy and ordering of the various constituents in the sentences
* Constituent Parsing: key-phrase extraction, document summaries, product/user review highlights
* Dependency Parsing: question-answering, chatbots, conversational AI

UNIT 10: Semantic Analysis

* Word Similarity:
  + Approaches
    - Statistical: How closely associated are two words in a corpus
    - Structure: How close are two words within a semantic graph
* PMI: pointwise mutual information measure. Positive means words are related/associated
* Vector Semantics:
  + Can be used to judge word similarity as well as text
  + Represents a distribution of other features found in same context
  + Two words are similar in meaning if their context vectors are similar
  + Subject to “garbage-in, garbage-out”
* Document Similarity:
  + Examples:
    - Jaccard distance: measures shared terms
    - Cosine Similarity
    - Hellinger Distance: probability distributions as measures of doc similarity
* Semantic Similarity
  + Similarity != Synonymy

UNIT 11: Document Clustering

* Organization of similar documents based on characteristics
* Types (several):
  + Centroid-based -> each cluster has a central representative member
    - Centroid: point within the cluster that is considered the most representative of that group
  + Hierarchical -> smaller clusters are members of larger clusters

UNIT 12: Automated Classifying of Documents

* Types of Text Classification:
  + Usually longer texts, but able to classify short texts
  + Content-based
    - Start with two or more classes of existing content
    - Most are binary classifications
  + Descriptor-based
    - User-inputted description of content desired
    - Binary or Multiclassification
    - Can use taxonomy
* Subject Based Classification
  + Two approaches:
    - Multinomial Naïve Bayes
      * Prior probability of class: global distribution of individuals
      * Prior probably of predictor: global distribution of individuals into the predictor
      * Posterior probability of predictor: the probability of having the predictor attributes
      * Posterior probability of class: the probability of falling into the target class
    - SVM-based classifiers
      * Can train a plurality of SVMs and organize them into a system

UNIT 13: Semantic Analysis (Topic Modeling)

* Canonical: match a preestablished list of topics for our domain
  + Goal is to determine a subset of canonical topics that are materially treated in a given corpus, showing which topics are contextually related in that corpus
  + Approaches:
    - Constrain the organic topic model to the canonical list
    - Use an IR approach, leveraging the canonical topic list to build queries.
    - Extension vs Intension
      * Extension: concepts are extensionally related when they extend to some of the same referents in the world
      * Concepts are intentionally related when their meanings overlap
* Organic: discover the “natural” topics of a corpus
  + Idea is to let topics bubble up from out of the “lake” of unstructured docs
  + Main focus in DS community
* Entity-centric: topics are strongly related to sets of Nes that may change over time
  + Goal: model whatever topics are strongly related to a set of named entities in a domain
  + Topics are worth little unless tied to NEs.
  + NEs are worth little unless tied to topics
* Organic Topic Modeling:
  + LSA: Latent Semantic Analysis
    - Tries to find groups of words associated with the largest variances between documents
    - Starts with a large term-document matrix
    - Built for systemically making these discoveries for us
  + LDA: Latent Dirichlet Allocation
    - Constructs topics as groups of words that have high co-occurrences among different documents in the corpus
    - Parameters
      * Alpha: high alpha means the document likely to contain a mixture of topics
      * Bega: high beta means each topic is likely to contain a mixture of many words
    - Produces more usable results than NMF
  + NMF: Non-negative matrix factorization
    - Version of LDA in which parameters have been tweaked to enforce a sparce number of topics
    - Inherent sparseness means not the best solution for finding lots of topics in long documents
    - Cheaper computationally than LDA

UNIT 14: General Sentiment Scoring

* Goal: detection of positive or negative sentiment
* Scoring
  + Usually scored from -1 to 1
  + 0 usually represents neutral, non-detection, or balanced
* Approaches:
  + Supervised ML approach
    - Pros:
      * Quick to implement
      * Don’t need to develop a coded vocab
    - Cons:
      * Very opaque
      * Only as granular as the training data annotations
  + Unsupervised lexical KB approach
    - Pros:
      * Does not require training data
      * Eminently explainable
    - Cons:
      * Needs a coded vocabulary
      * Can be cumbersome to maintain in face of new tropes
* Ways to improve general scoring
  + Determining referents and/or topics to which sentiment attaches
    - Run a chunker and send chucks into sentiment analyzer
    - Run deeper parser
    - Follow dependency paths until object is found
  + Classifying sentiment into more categories (other than Pos and Neg)
    - Shaver’s hierarchy of emotions
  + Picking up on non-sentiment vocab differences that align with sentiment around topic