January 24, 2017

Natural Language Processing Glossary for Programmers

I created this Natural Language Processing (NLP) glossary as a resource for folks who are only just getting introduced to NLP. I don't claim that this is exhaustive. I want to follow the Pareto's principle here - this probably has about 20% of the definitions but will help you with 80% of the concepts you would actually use in NLP projects. Also, I would like to keep it a short and easy read.

Sentence segmentation

This is the process of splitting a document of text into individual sentences.

Tokenization

This is the process of splitting a document into individual words. Tokenization and sentence splitting usually go hand in hand. For example, Stanford's CoreNLP expects tokenization to be done before sentence segmentation.

Part of Speech tagging

Remember those old English grammar rules? I don't either. My first language isn't English, although English was the medium of instruction during my schooling and beyond. So I am fairly comfortable using English although I don't really understand its rules of grammar.

In any case, part of speech (POS) tagging inspects your text and decides if the individual words are nouns, adjectives, verbs, adverbs etc. There are a lot of POS tags (over 35 according to this list on StackOverflow).

Stemming

Stemming is the process of getting the "root" from a word. For example, the words organize, organized and organizing are all derived from organize, and when you do a search for the word organize, you would expect to also get the other forms of the word since they represent the same idea.

A very important thing to note (especially if you end up using stemming in your NLP projects) is that the stem of a word does not have to be and often isn't a dictionary word. E.g the stem of the word "saw" is just "s".

Lemmatization

Lemmatization is similar to stemming in that it tries to get the root of a word, except that it tries to regularize the word to end up with a dictionary word. E.g. the lemma of the word "saw" is either "see" or "saw" based on whether the token was a verb or a noun.

Named Entity Recognition

Named Entity Recognition or NER is simply the process of extracting nouns from your text. For example, using NER, you could automatically detect all the occurrences of a brand name in a person's Twitter feed.

Syntax parse tree (or Constituency parse tree)

A sentence can be parsed into a grammar tree. I think this concept is best explained with a picture.

Dependency parse tree

While the constituency parse tree is concerned with how words *combine* to form constituents, a dependency parse tree is a tree representation of the *relationships* between the words in a sentence.

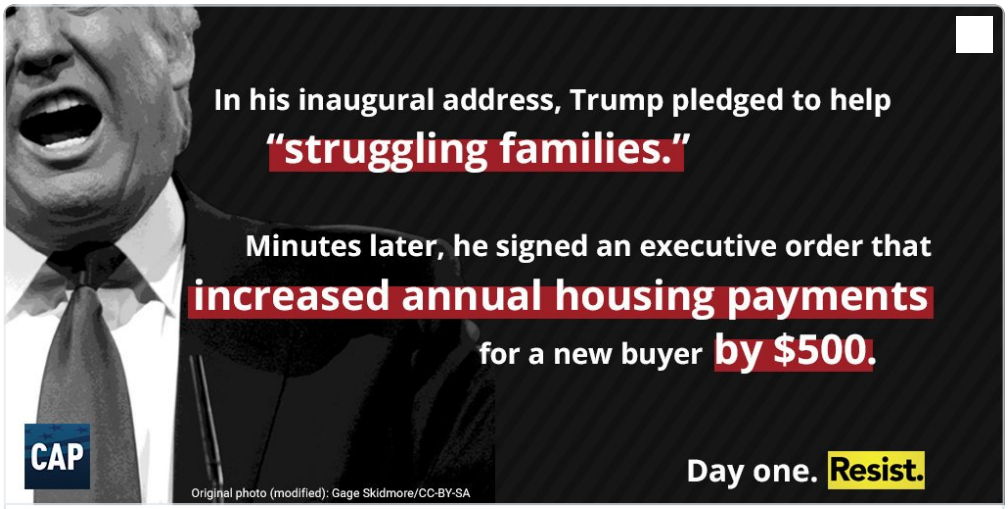
Coreference resolution

Coreference occurs when two or more expressions in a text refer to the same person or thing. For example, in the sentence "Bill said he would come", the word "he" refers to Bill. Coreference resolution is the ability to *resolve* the co-reference to find what it is referring to.

Polarity detection

This is a fancy term for deciding whether a piece of text conveys a positive or a negative sentiment. Imagine if you are writing a program for figuring out whether a tweet says something positive or negative about a brand.

For example, does this person think Trump is amazing?



Information Extraction

Information Extraction is the process of extracting facts (about the world) from text information. For example, if you saw the sentence "Nigeria is a country in Africa" you should be able to answer the question "India is a country in _____".

If the first word you thought of for the blank was "disarray" or "trouble"- I feel ya! But that is also an excellent example of why the progress in Natural Language Processing has been somewhat slow. There is just a lot of ways a given idea could be expressed in the English language.

While this is not an exhaustive glossary, it covers many key concepts in NLP which should give you a quick idea of the kinds of things you can do using Natural Language Processing and text analytics.

Feedback

Do you think there are items which definitely should be included in this glossary? Let me know your thoughts in the comments below.

Article by aravindmc / Natural Language Processing / 2 Comments