CS290D – Advanced Data Mining

Instructor: Xifeng Yan
Computer Science
University of California at Santa Barbara

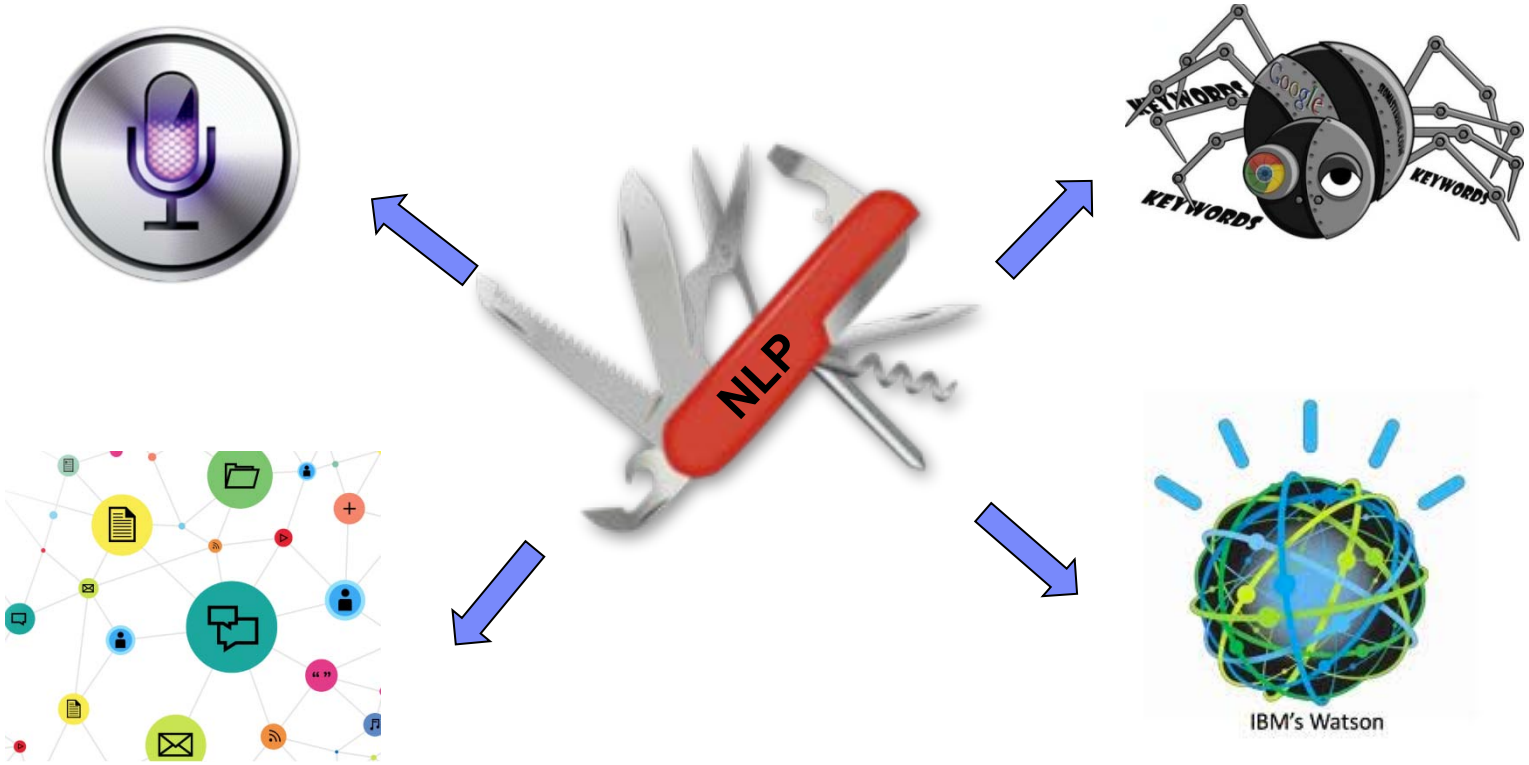
NLP Basics

Instructor: Yang Li
Computer Science
University of California at Santa Barbara

Slides adjusted from: https://www.fer.unizg.hr/_download/repository/TAR-02-NLP.pdf

Motivation: NLP as a tool!

- Most tasks addressed by NLP are useful for Text Analysis and Text Mining



It's All Practical!

- After today's lecture, you'll:
  - Gain familiarity with the basic NLP tasks
  - Know about the tools and resources available
  - See how it all works in practice

NLP Components

- Sentence segmentation
- Tokenization
- POS tagging
- Stemming/Lemmatization
- Word-level (lexical) semantics
- Parsing (syntax)
- Coreference resolution

Sentence segmentation

- Sentence segmentation: finding boundaries of sentences
  - Often done heuristically, using regular expressions
  - Best performance with supervised machine learning models
    - predict for each full stop whether it denotes the end of the sentence

Natural language processing (NLP) is a field of computer science, artificial intelligence, and computational linguistics concerned with the interactions between computers and human (natural) languages. As such, NLP is related to the area of human–computer interaction. Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation.

Tokenization

- Tokenization: breaking a text up into tokens – words and other meaningful elements
  - Tokens are words, punctuation marks, and special characters
  - Rule-based (i.e., heuristic) methods achieve satisfactory performance

Natural language processing -lrb- NLP –rrb- is a field of computer science , artificial intelligence , and computational linguistics concerned with the interactions between computers and human –lrb- natural –rrb- languages . As such , NLP is related to the area of human–computer interaction . Many challenges in NLP involve natural language understanding , that is , enabling computers to derive meaning from human or natural language input , and others involve natural language generation .

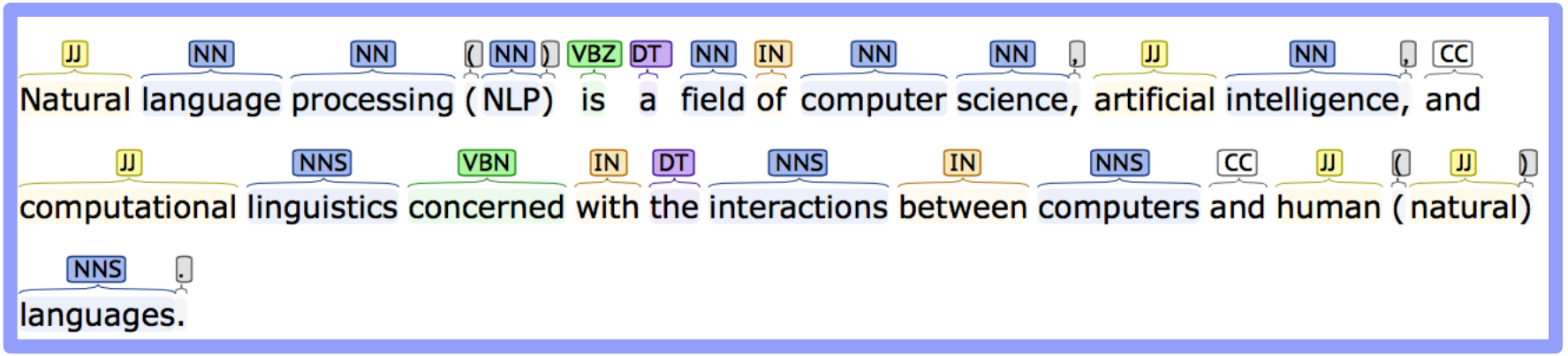
Parts-of-speech (POS)

- POS: the role that a word or phrase plays in a sentence
  - Explains not what the word is, but how it is used

Verbs assert something about the subject of the sentence and express actions, events, or states of being
Nouns are words that we used to name a person, an animal, a place, a thing, or an abstract idea
Adjectives modify nouns and pronouns by describing, identifying, or quantifying them.
Pronouns replace nouns or another pronouns and are essentially used to make sentences less cumbersome and less repetitive
Adverbs modify a verb, an adjective, another adverb, a phrase, or a clause. An adverb indicates manner, time, place, cause, . . .
Prepositions, conjunctions, . . .

POS Tagging

- POS tagging is the process of marking up a word in a text as corresponding to a particular part of speech
  - Assign tags from a finite predefined tag set (e.g. Penn Treebank)
  - State-of-the-art POS taggers are supervised machine learning models



  - Popular POS taggers:
  - Stanford POS tagger
  - CCG University of Illinois tagger

Stemming

- Stemming: reduction of word-forms to stems
  - adjustments → adjust
  - defensible → defens
  - revivals → reviv
  - Very useful in IR-related applications
- Typically by suffix stripping plus some extra steps/checks
  - simple and efficient.
  - prone to overstemming and understemming errors
    - e.g. university → universe
- Most popular English stemmer: Porter Stemmer

Lemmatization

- Lemmatization: transformation of a word-form into a linguistically valid base form, called the lemma
  - nouns → singular nominative form
  - verbs → infinitive form
  - adjectives → singular, nominative, masculine, indefinitive form
- Usually relies on:
  - a dictionary that maps word-forms to lemmas
  - a ML model, trained on a large number of word-lemma pairs
- Example of a machine learning-based lemmatizer:
  - CST lemmatizer: http://cst.dk/online/lemmatiser/uk/

Stemming vs. Lemmatization

- Stem is not necessarily a valid word, while lemma is always a linguistically valid word
  - revivals → reviv (stemming)
  - revivals → revival (lemmatization)
- Stemming is usually POS-independent, while lemmatization takes the POS into consideration.
  - lemmatization: dove (verb) → dive; dove (noun) → dove
  - stemming: dove → dove
- Stemmers are simpler, smaller and faster than lemmatizers

Stemming vs. Lemmatization

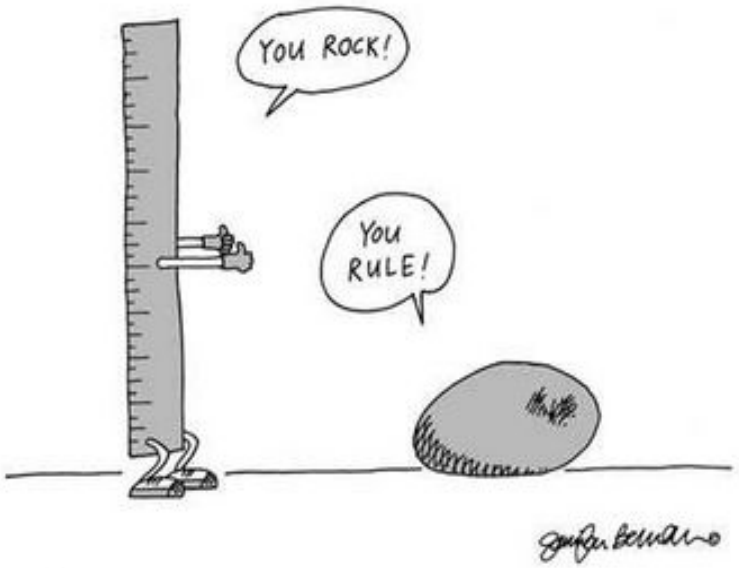
- For the following tasks, which operation is better?

1.Information Retrieval (Search Engine)

2. Machine Translation

Word-level (Lexical) Semantics

- Lexical semantics is concerned with the meaning of words
  - What the words mean and how they relate to each other
  - Verb and event semantics (semantic roles)
  - Distributional semantics



WordNet

- Manually constructed lexical database
  - contains Nouns, Verbs, Adjectives and Adverbs
  - https://wordnet.princeton.edu/

- Words are organized into synsets – sets of words with the same sense/meaning
  - e.g. synsets of the word “tire”:

tire.n.01 – hoop that covers a wheel

tire.v.01 – lose interest or become bored with something or somebody

tire.v.02 – exhaust or get tired through overuse or great strain

run_down.v.06 – deplete

bore.v.01 – cause to be bored

WordNet Synset

- For each synset WordNet provides:
  - a list of words that can be used in that sense
  - a gloss – a short description of the sense
  - semantic relations to other synsets (hyponymy, meronymy, . . .)

synset: tire.n.01
words: tire, tyre
gloss: hoop that covers a wheel

- Over 100k synsets for English, still quite limited

WordNet Relations

- Nouns:

  Hyperonymy/hyponymy – IS-A relation (chair – furniture)

  Meronymy – a part whole relation (finger – hand)

  Antonymy – opposite meaning (wet – dry)

  Similarity – similar (but not identical) meaning (warm – hot)

- Verbs:

  Troponymy – increasingly specific manner of an event (communicate – talk - whisper, move – jog - run)

  Entailment – one word entails the other (succeed – try, buy – pay)

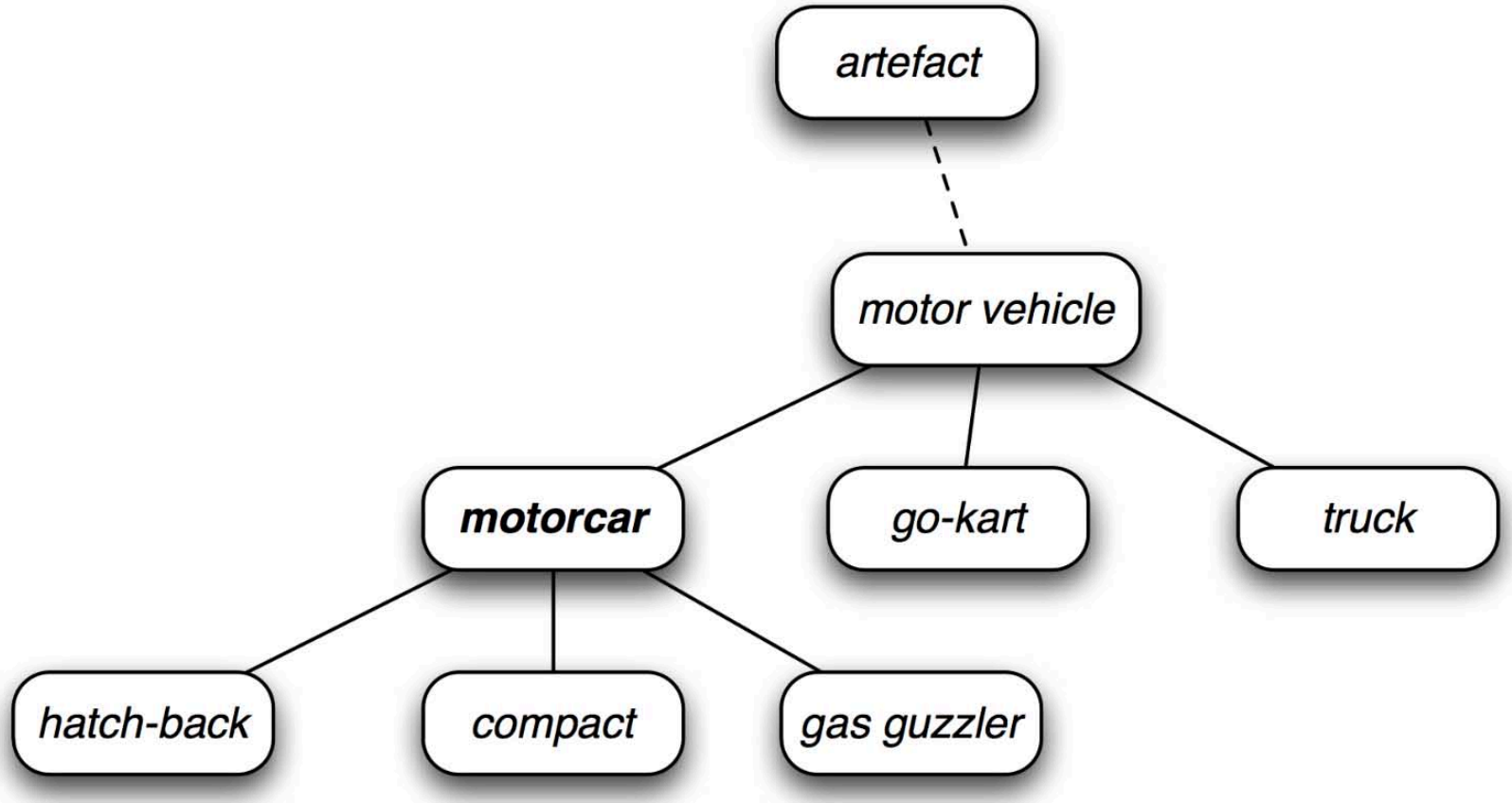
WordNet Relations

- Hypernyms of tire.n.01
  - hoop.n.02 – a rigid circular band of metal or wood or other material used for holding or fastening or hanging or pulling

- Hyponyms of tire.n.01
  - wagon_tire.n.01 – a metal hoop forming the tread of a wheel
  - pneumatic_tire.n.01 – a tire made of reinforced rubber and filled with compressed air; used on motor vehicles and bicycles etc
  - car_tire.n.01 – a tire consisting of a rubber ring around the rim of an automobile wheel

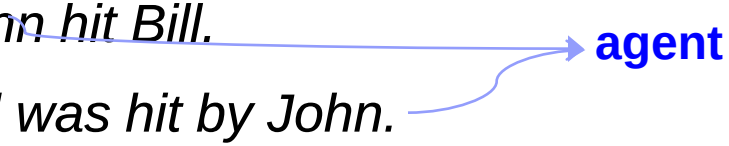
WordNet Hierarchy

- Example of the hierarchy from WordNet:



Semantic Roles

- The underlying relationship that a participant has with the main verb in a sentence
  - *John hit Bill.*
  - *Bill was hit by John.*

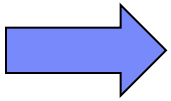


role	description
AGENT	the participant that initiates the action
PATIENT	the entity undergoing the effect of some action
THEME	the entity which is moved by an action
EXPERIENCER	the entity which is aware of the action or state described by predicate, but which is not in control
…	…

Semantic Roles Labeling

- The NLP task consisting of the detection of the semantic arguments associated with the verb of a sentence and the classification into their specific roles.

John hit Bill.
Bill was hit by John.



John <Agent> hit
Bill <Patient> hit

- State-of-the-art performance with supervised machine learning models

Semantic Role Labeling Tools

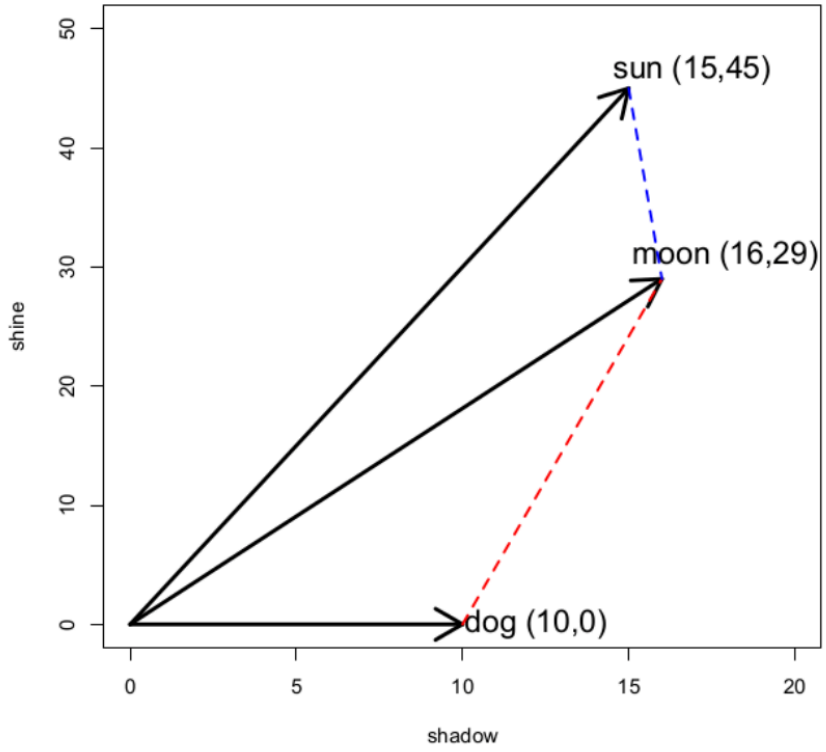
- Illinois Semantic Role Labeler
  - http://cogcomp.cs.illinois.edu/page/demo_view/srl
- Shalmaneser
  - http://www.coli.uni-saarland.de/projects/salsa/shal/
- SwiRL
  - https://github.com/kvh/SwiRL

Distributional Semantics

- Represent word meaning based on distributional hypothesis
  - correlation between words' context similarity and semantic similarity
- Words represented as vectors of context features obtained from corpus
- Semantic similarity predicted via vector similarity
- Useful for: finding semantically similar and related words
  - Similarity = pragmatic relation
  - Relatedness = syntagmatic relation

Distributional Semantics

	planet	night	full	shadow	shine	crescent
moon	10	22	43	16	29	12
sun	14	10	4	15	45	0
dog	0	4	2	10	0	0



Word Sense Disambiguation (WSD)

- Identifying which sense (meaning) of a word is used in a sentence, when the word has multiple meanings

- WSD addresses both polysemy and homonymy
  - Polysemy – a word has multiple, related meanings
    - e.g. ring (wedding ring vs. boxing ring)
  - Homonymy – two, unrelated words, have the same form
    - e.g., saw (past tense of see vs. a tool)

- The more fine-grained the senses, the more difficult the disambiguation task

WSD Approaches

- Dictionary-based methods:
  - Rely on lexical resources, most often WordNet
- Supervised WSD
  - One classifier for each polysemous word
  - Manual sense annotations in text, very time- and resource-consuming
- Unsupervised WSD
  - Clustering different contexts in which the word appears
  - Ideally, each cluster corresponds to one sense

WSD Tools

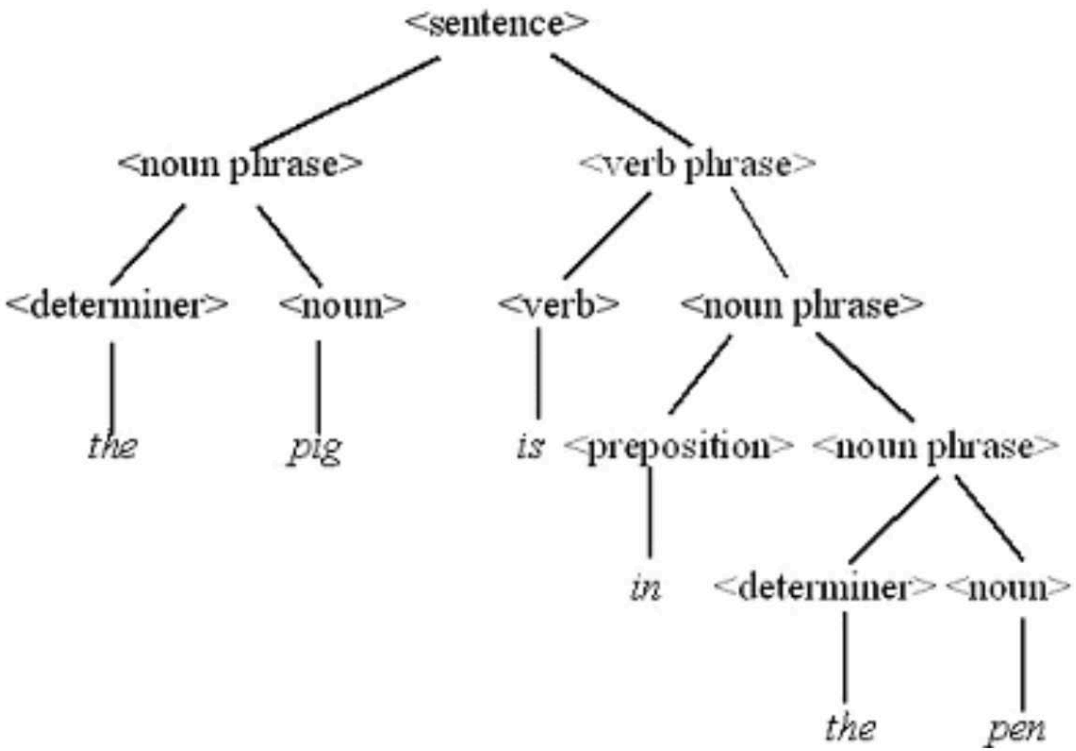
- University of North Texas WSD Tool:
  - lit.csci.unt.edu/index.php?P=research/downloads
  - Unsupervised graph-based WSD
- KYOTO UKB graph-based WSD
  - http://ixa2.si.ehu.es/ukb/
- pyWSD (Simple WSD algorithms in Python)
  - https://github.com/alvations/pywsd

Parsing

- Parsing is the task of analyzing the grammatical structure of natural language sentences
- Given a sequence of words, a parser forms units like subject, verb, object and determines the relations between them according to some grammar formalism
- Two types of parsers
  - Constituency parsers/phrase structure tree (PST) parsers –
    - based on constituency/PS grammars
  - Dependency parsers
    - based on dependency grammars

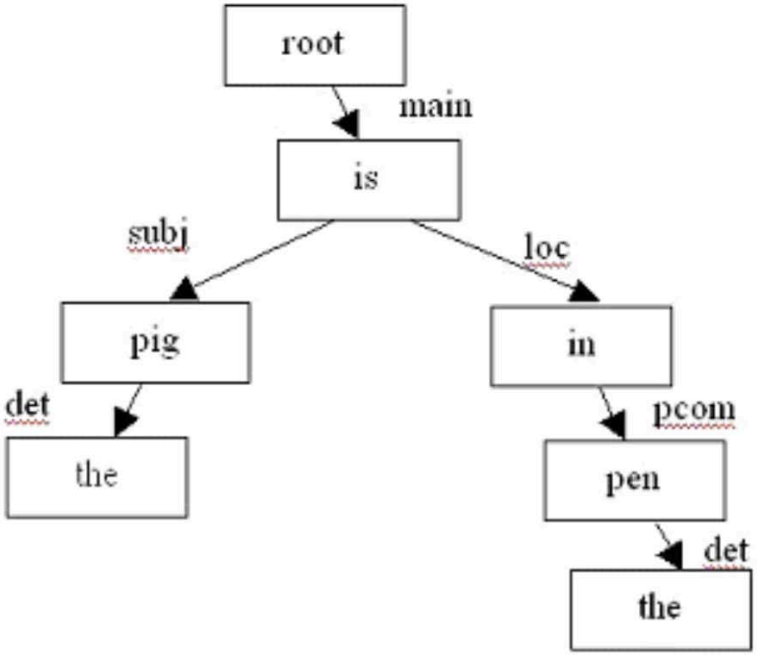
Constituency Parsing

- Constituency parser produces a tree that represents the syntactic structure of a sentence
  - Words appear only as leaves of the tree



Dependency Parsing

- Dependency parsing represents the structure of the sentence as the tree of syntactic dependencies between pairs of words
  - Each dependency relation has a governing word and a dependent word
  - Verb is the syntactic center of the clause, all other words directly or indirectly dependent on the verb



Popular Parsers

- Stanford parsers (both constituency and dependency)
  - http://nlp.stanford.edu/software/lex-parser.shtml
- Berkeley parser (constituency)
  - https://code.google.com/p/berkeleyparser/
- Collins's parser (constituency)
  - http://people.csail.mit.edu/mcollins/PARSER.tar.gz

Shallow Parsing

- Shallow parsing ("chunking", "light parsing") only identifies the constituents (noun phrases, verbs phrases, prepositional phrases, etc.) but does not specify their internal structure nor their role in the sentence

[**NP** Jack and Jill] [**VP** went] [**ADVP** up] [**NP** the hill] [**VP** to fetch] [**NP** a pail] [**PP** of] [**NP** water].

- Much faster than full parsing!

Shallow Parsers

- CCG University of Illinois shallow parser
  - http://cogcomp.cs.illinois.edu/page/run_demo/ ShallowParse
- Apache Open NLP shallow parser
  - http://opennlp.apache.org/index.html

Coreference Resolution

- Coreference resolution is the task of clustering together of expressions that refer to the same entity/concept.

Michelle LaVaughn Robinson Obama is an American lawyer and writer. She is the wife of the 44th and current President of the United States, Barack Obama, and the first African-American First Lady of the United States.

Coreference Resolution

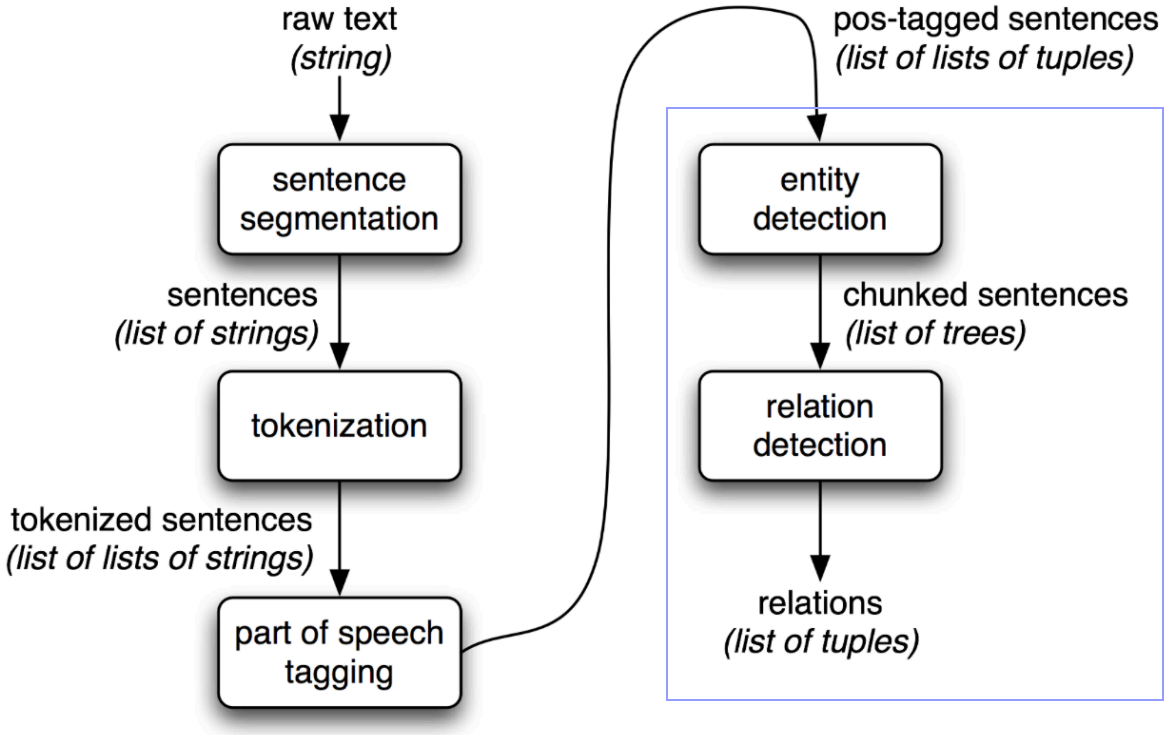
- Important for discourse semantics
- State-of-the-art methods:
  - Supervised machine learning (e.g. Berkeley)
  - Unsupervised multi-pass rule matching (e.g. Stanford)
- ~80% overall performance, poor on pronoun resolution

Coreference Resolution Tools

- Stanford Deterministic Coreference Resolution System
  - http://nlp.stanford.edu/software/dcoref.shtml
- Reconcile
  - https://www.cs.utah.edu/nlp/reconcile/
- Berkeley Coreference Resolution System
  - http://nlp.cs.berkeley.edu/projects/coref.shtml

NLP Pipeline

- The basic NLP components can be used for many applications
- An example for constructing knowledge graph:



to be covered in next class

Popular NLP software

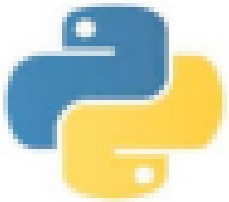
- UIMA
- NLTK
- GATE
- Stanford CoreNLP
- Other libraries

UIMA

- Java framework for developing NLP pipelines
  - http://uima.apache.org
- Provides Eclipse plugins
- Wrappers for a wide variety of C++ and Java-based component libraries
- Focus is on integrating various NLP tools and scaling up to huge amounts of data

NLTK

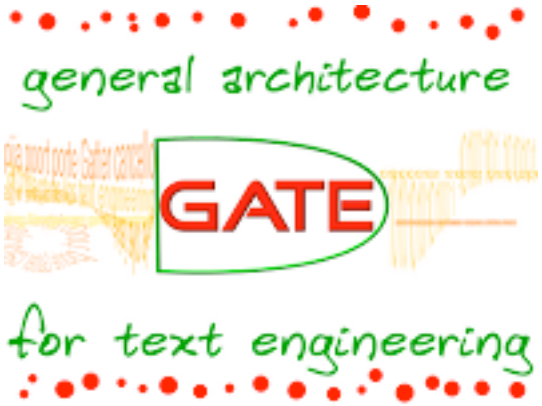
- Python library for NLP
  - http://www.nltk.org/



- Pros:
  - Good out-of-box models and algorithms for basic NLP tasks
  - Training new models with user defined features is very easy

- Cons:
  - Lacks good pretrained models for more advanced tasks
  - Can be slow compared to other tools

GATE



- A Java framework for NLP pipelines
  - http://gate.ace.uk
- Includes a number of rule-based NLP components
- Provides wrappers to a huge amount of NLP libraries (including UIMA, OpenNLP and Stanford parser)

Stanford CoreNLP

- Java NLP library, integrates all Stanford NLP tools
  - http://nlp.stanford.edu/software/corenlp.shtml
- Offers very good out-of-the-box models for advanced NLP tasks
- Trained models are also available for languages other than English

Other libraries

- OpenNLP
  - Java machine learning based text processing library
  - https://opennlp.apache.org/
- LingPipe – Java toolkit for processing text
  - http://alias-i.com/lingpipe/
- Mallet – Java MAchine Learning for LanguagE Toolkit
  - http://mallet.cs.umass.edu/
- Proxem Antelope – .NET-based NLP framework
  - https: //www.proxem.com/en/technology/antelope-framework/

What you should know from this class

- Gain familiarity with the basic NLP tasks
- Know about the tools and resources available
- See how it all works in practice

A good practice

- Build a simple knowledge graph extraction system using existing NLP tools and resources
  - Input: text corpus (e.g. a novel)
  - Output: a knowledge graph where nodes are noun phrases and edges are verbs connecting noun phrases

- Design a NLP pipeline for it and see how it magically works ☺

Questions?