UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

3546 – Deep Learning

Module 6 - Natural Language Processing

Learning Outcomes for this Module

- Develop some familiarity with key concepts in NLP
- Understand NLP pipelines, word and document representation, and topic modeling
- Have a look at some of the features of the Natural Language Toolkit (NLTK), spaCy, gensim, coreNLP and keras

Topics for this Module

- **1.1** What is Natural Language Processing?
- **1.2** What Makes NLP Challenging?
- **1.3** NLP Pipelines
- **1.4** Information Categorization and Retrieval
- **1.5** Encoding Meaning
- **1.6** Resources and Wrap-up

Module 6 – Section 1

What is Natural Language Processing?

Natural Language Processing

- Natural Language Processing (NLP) is the study of computational treatment of natural (human) language
- Combines linguistics with AI
- Like other areas of AI, NLP evolved from *symbolic* to *statistical* to *neural*
  - 1950's: Symbolic programming languages and translation
  - 1960's: ELIZA
  - 1970's: Ontologies
  - 1980's: Racter
  - 1990's: ML using corpora
  - 2000's: Web corpora and larger models
  - 2010's: Neural
  - 2020's: Spoken word to…?

Interacting with Machines via Language

- *Formal* logic literally means logic enforced by language constructs (e.g. Lambda Calculus, First Order Logic)
- **The Sapir-Whorf Hypothesis**: A principle suggesting that the structure of a language affects its speakers' worldview or cognition, and thus people's perceptions are relative to their spoken language (Wikipedia: https://en.wikipedia.org/wiki/Linguistic_relativity)
- **LogLan**: *Logical Language* designed to test the Sapir-Whorf hypothesis

NLP Applications

- **Search**: Web, documents, autocomplete
- **Editing**: Spelling, grammar, style
- **Dialog**: Chatbots, assistants
- **Writing**: Index, concordance, table of contents
- **Email**: Spam filter, classification, prioritization
- **Text mining**: Summarization, knowledge extraction, medical diagnosis
- **Law**: Legal inference, precedent search, subpoena classification
- **News**: Event detection, fact checking, headline composition, short article generation

NLP Applications (cont'd)

- **Attribution**: Plagiarism detection, literary forensics, style coaching
- **Sentiment analysis**: Community morale monitoring, product review triage, customer care
- **Behavior prediction**: Finance, election forecasting, marketing
- **Mood evaluation**: Depression detection, guidance for call centre interactions
- **Creative writing**: Movie scripts, poetry, song lyrics
- **Automatic question answering**: Self-serve help
- **Picture and video generation**: Create a scene as described
- **Coding assistants**: GitHub Copilot, TabNine

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 6 – Section 2

What Makes NLP Challenging?

Ambiguity in Natural Language

- Stolen painting found by tree
- Local high school dropouts cut in half
- Doctor: “No heart, cognitive issues”
- Students cook & serve grandparents
- 1 million get shot to save on loans
- San Jose cops kill man with knife
- Queen Mother tries to help abuse girl
- Big rig carrying fruit crashes on 210 Freeway, creates jam
- Princess Diana dresses to be auctioned
- How to prepare pets for Thanksgiving

Other Issues

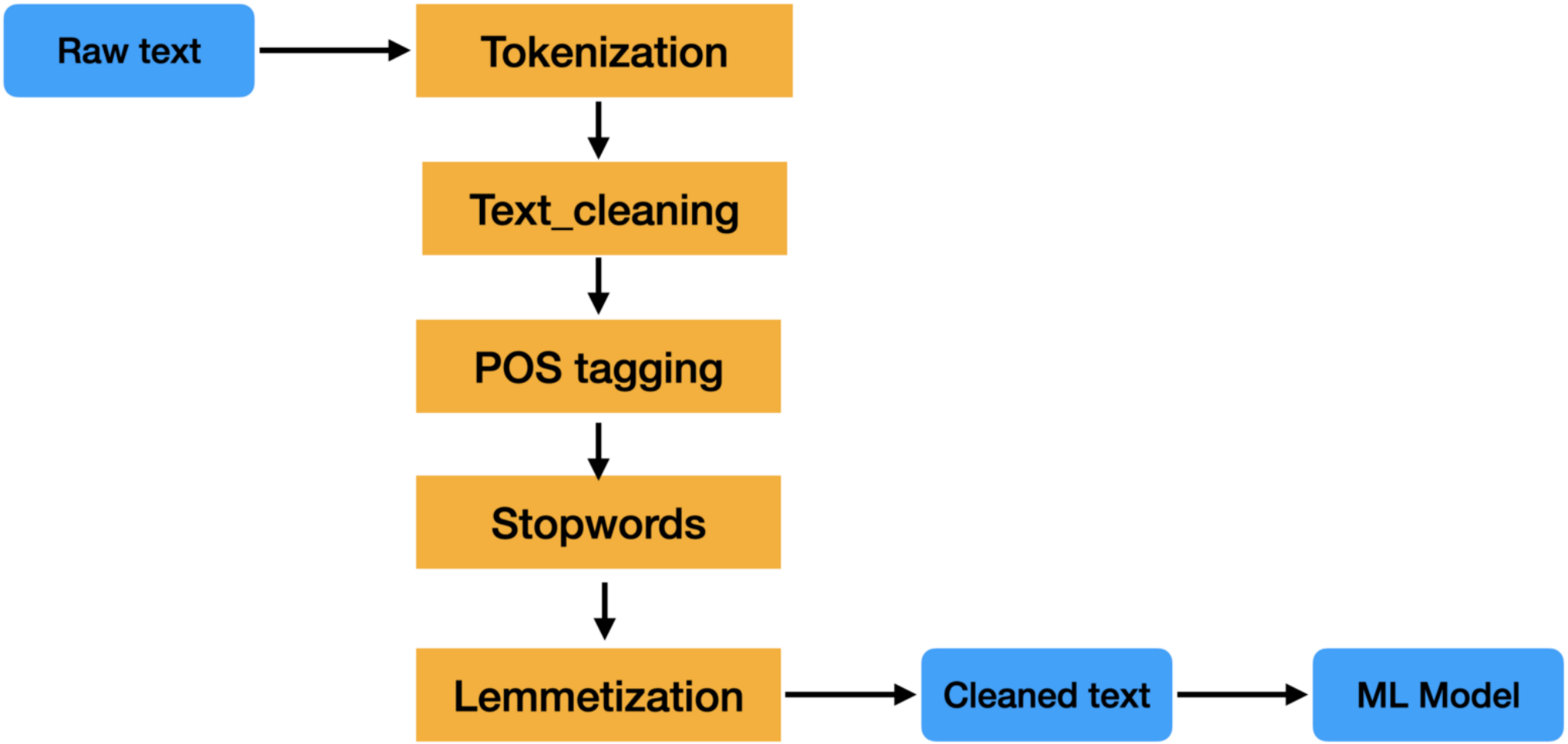
- **Synonyms**: A synonym is a word or phrase that means exactly or nearly the same as another lexeme in the same language
- **Homonyms**: Words which sound alike or are spelled alike but have different meanings
- **Misspellings**: Can be as a non-word or a different word
- **Sarcasm**: The use of irony to mock or convey contempt
- **Allegory**: A story, poem, or picture that can be interpreted to reveal a hidden meaning, typically a moral or political one
- **Dialects**: A particular form of a language which is peculiar to a specific region or social group
- **Language evolution**: Words, expressions and spelling change over time

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 6 – Section 3

NLP Pipelines

NLP Pipelines



Source: https://miro.medium.com/max/1838/1*CbzCcP3XFtYVJmWowZLugQ.png

Tokenization

- We could attempt to work with text as strings of characters: https://www.tensorflow.org/text/tutorials/text_generation
- Words (or pictograms) approximately represent a unit of meaning

Issues in Tokenization

- **Special cases**: Names (particularly multi-word), initials, hyphenated words, abbreviations, special forms (dates, phone numbers, URLs, etc.)
- **Punctuation**: Should ! be treated as a word?
- **Contractions**: How many words is "isn't"?
- **Named Entities**: e.g. Richard Feynman, CN Tower
- **Rare words**: Large vocabularies take a lot of CPU and/or memory
- **Very frequent words**: e.g. a, the

Issues in Tokenization (cont'd)

- **Issues specific to other languages**:
  - German and Welsh: Compounds words such as schadenfreude, Kraftfahrzeug-Haftpflichtversicherung and Llanfairpwllgwyngyllgogerychwyrndrob-wllllantysiliogogogoch https://youtu.be/fHxO0UdpoxM
  - Chinese: average of about 2 symbols/word
  - Japanese: kanji, hirigana, katakana, romanji

Stop Words

- Common words that convey little meaning
- Removed for some applications because vocabularies' size are usually limited
- Examples: the, a, is, at, which
- Removing them generally increases performance at tasks such as document comparison
- Removing them can cause problems with *named entity resolution* (e.g. "The Who") and translation
- There is no single agreed-on *stop words* list

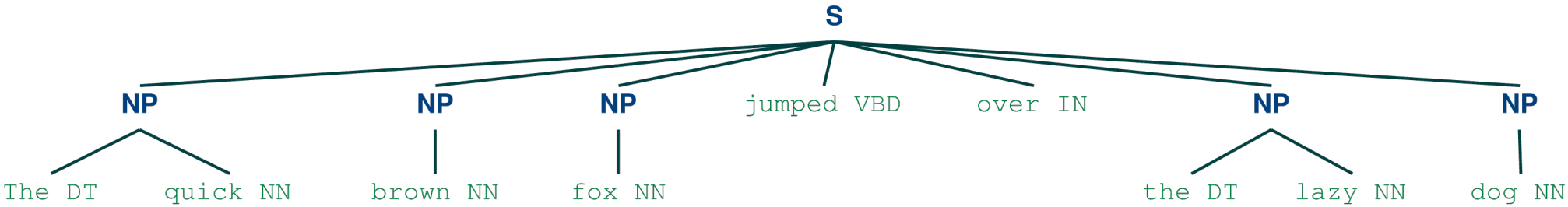
Sources of Text for Training

- A corpus (plural corpora) is a large and structured set of texts
- Examples:
  - Brown Corpus
  - Wikipedia
  - Common Crawl https://commoncrawl.org/
  - Gutenberg Corpus
  - The CMU Pronouncing Dictionary http://www.speech.cs.cmu.edu/cgi-bin/cmudict

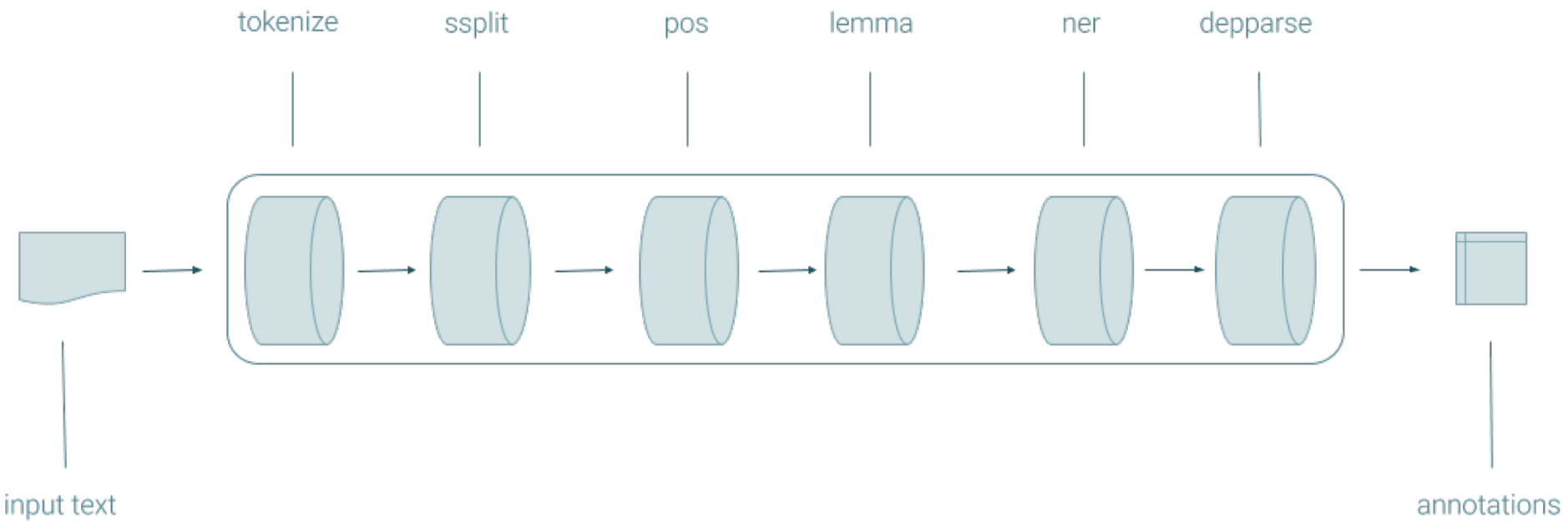
Normalization Techniques

- **Case folding**: Force all lower case
  - Can make named entity resolution more difficult
  - Becoming less common as a result
- **Stemming**: Crude chopping of affixes e.g. remove -s or -ing at end
  - Can cut vocabulary size in half (or more if aggressive)
  - Many algorithms: Porter's is most common English stemmer and has a lot of knowledge of English hardcoded in it
  - Useful for search where we are looking for similar, not exact matches (will improve recall but reduce precision)
- **Lemmatization**: Extraction of the base form e.g. "are", "am", "is" replaced with "be"
  - Better for most applications than stemmers which might take "better" and convert it to "bet"
- **Hashtag and emoji expansions**: e.g. #lol to laugh out loud

Parts Of Speech (POS) Taggers

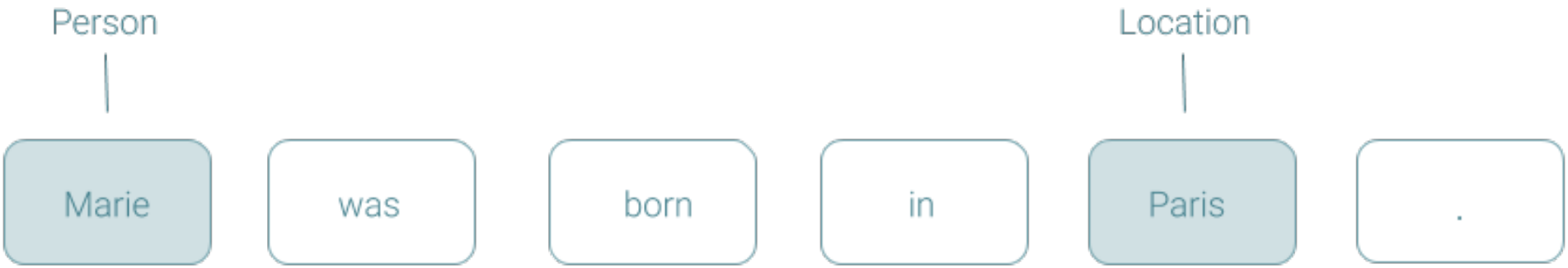


Annotators



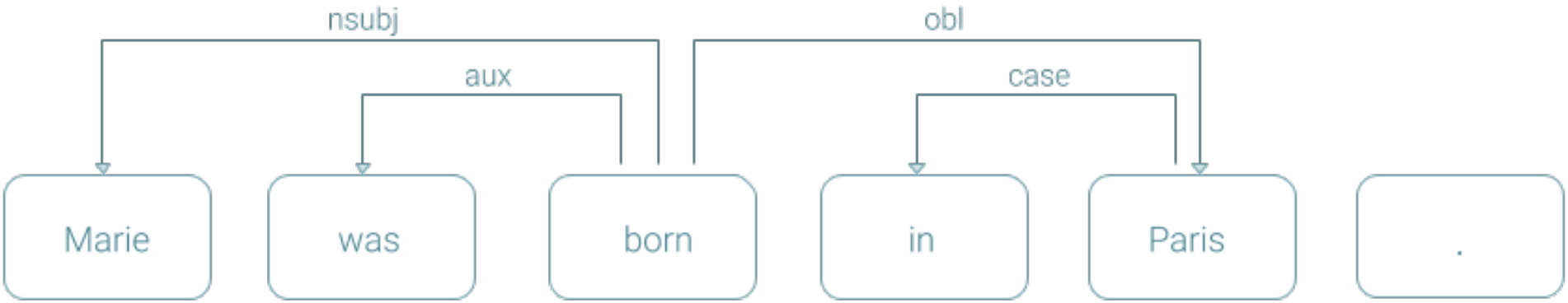
Source: https://stanfordnlp.github.io/CoreNLP/assets/images/pipeline.png

Named Entity Recognition



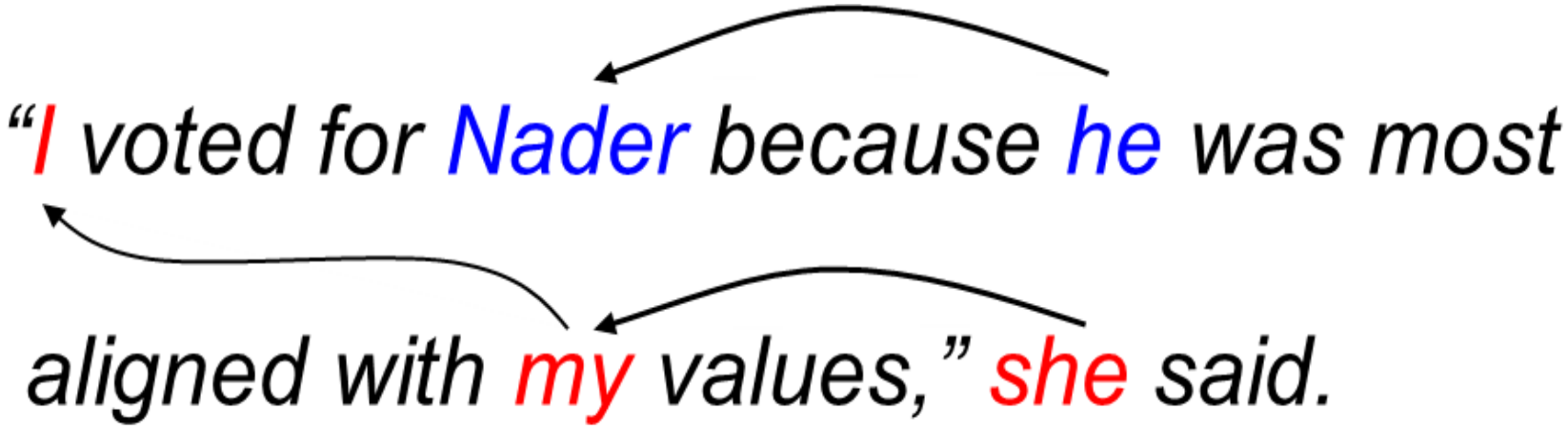
Source: https://stanfordnlp.github.io/CoreNLP/assets/images/ner.png

Constituent and Dependency Parsing



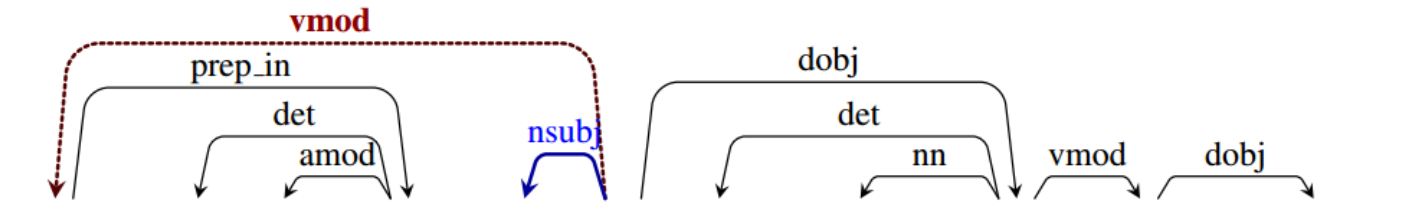
Source: https://stanfordnlp.github.io/CoreNLP/assets/images/depparse.png

Coreference Resolution



Source: https://nlp.stanford.edu/projects/corefexample.png

Stanford CoreNLP Open Information Extraction

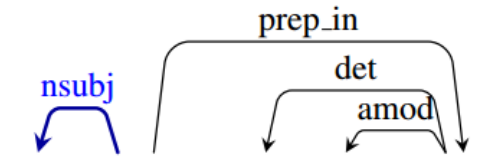


(input)

she took the midnight train going anywhere
Born in a small town, she took the midnight train
Born in a town, she took the midnight train

she took the midnight train
she took midnight train
. . .

(she; took; midnight train)



(extracted clause)

she Born in small town
she Born in a town
she Born in town

(she; born in; small town)
(she; born in; town)

Source: https://nlp.stanford.edu/software/openie.html

Python-friendly NLP Tools

- **NLTK**: The Natural Language Toolkit
- **SpaCy**: An alternative to NLTK
- **Spark NLP**: NLP tools for Apache Spark
- **Gensim**: Primarily topic modelling
- **TextBlob**: Extensions to NLTK
- **KerasNLP**: NLP tools for Keras
- **PyTorch NLP**: NLP tools for PyTorch
- **Hugging Face**: NLP tools and language models
- **MonkeyLearn**: Cloud-based simple NLP
- **Amazon Comprehend**: Amazon's NLP API
- **Google Cloud NLP API**: Google's NLP API

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 6 – Section 4

Information Categorization and Retrieval

Representing Documents

- Bag of words
- Bag of n-grams
- TF-IDF vectors

Bag of Words and One Hot Encoding

- Ignore word order
- Fix the vocabulary size
- One Hot encoding
- Enables arithmetic with vectors: union, intersection, addition, subtraction

Bag of N-Grams

- **n-gram**: A continuous sequence of n items from a given sample of text or speech
- Useful for translation, spelling correction, speech recognition, question answering
- We can add 2, 3, etc.-grams
- A vocabulary of about 20,000 words is sufficient to track 95% of words in a corpus of tweets, blog posts and news articles

TF-IDF Vectors

- Term Frequency, Inverse Document Frequency
- A measure of how important a word is to a document in a collection
- Let's say we want to divide up a corpus (collection) of documents into similar clusters
- How should we decide how similar two documents are?
  - How many words they have in common
  - How specialized those words are
- To capture these two aspects we need two measures for each word in our vocabulary:
  - Term Frequency: How frequently the word occurs in each document
  - Document Frequency: How often the word occurs in our corpus
- Document Frequency is usually measured on a log scale

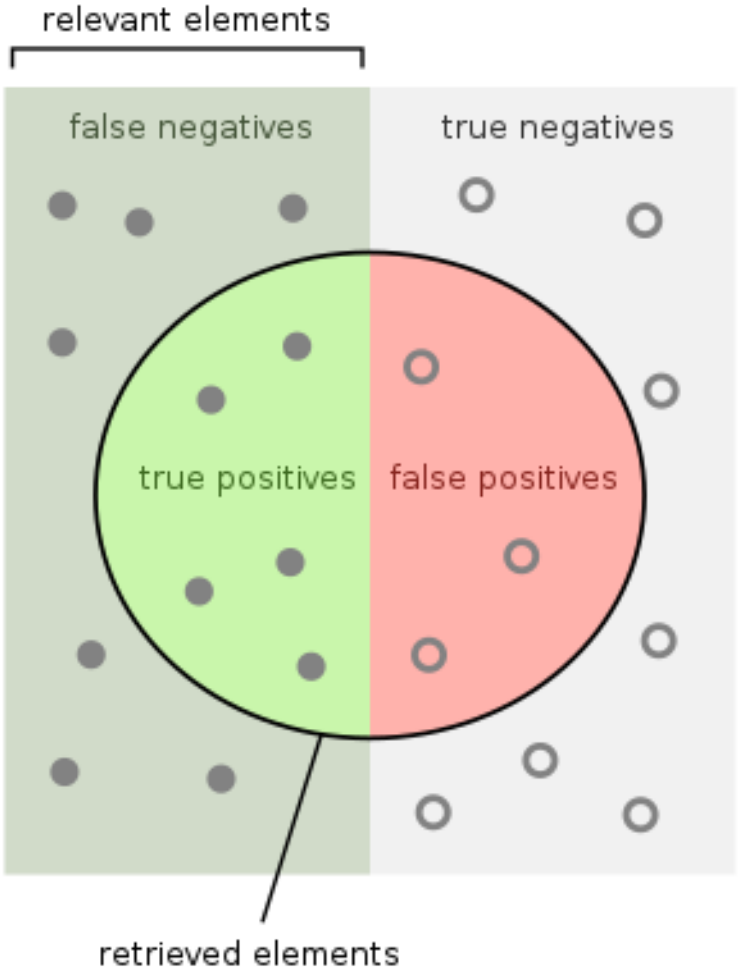
Cosine Similarity

- We can measure the similarity of vectors (such as TF-IDF vectors) using Cosine Similarity
- The more similar the vectors are, the smaller the angle there should be between them
- The cosine similarity of two vectors, x and y, is easily calculated using dot product:

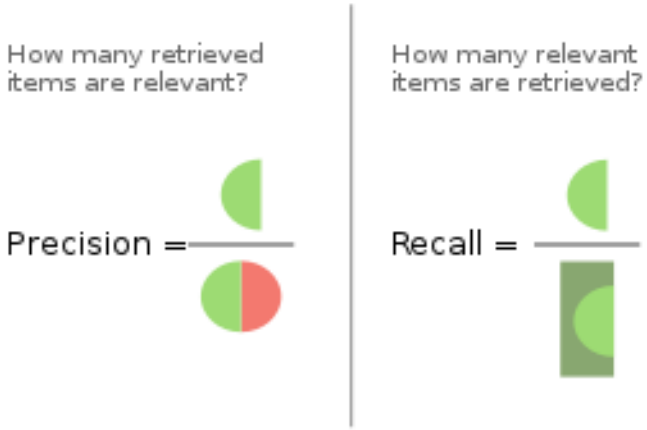
$$\cos(a, b) = \frac{a \cdot b}{\|a\|\|b\|}$$

- Cosine similarity is a number that runs between 0 (nothing in common) to 1 (identical) for TF-IDF vectors

Precision and Recall



Source:
https://en.wikipedia.org/wiki/Precision_and_recall

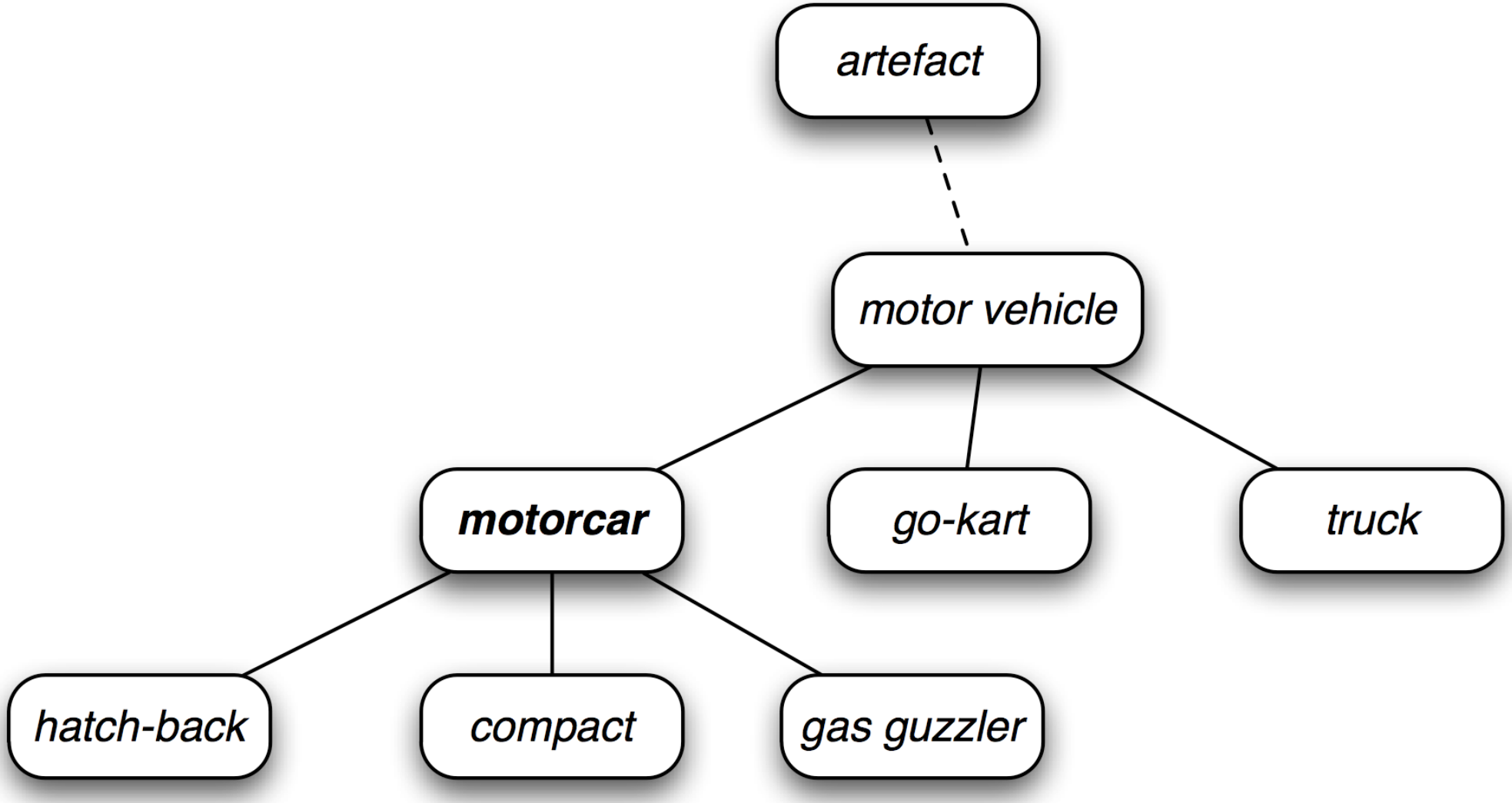


UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 6 – Section 5

Encoding Meaning

WordNet


artefact
motor vehicle
motorcar
go-kart
truck
hatch-back
compact
gas guzzler

The Meaning of Words

- **Wittgenstein (1949)**: “For a large class of cases–though not for all–in which we employ the word “meaning” it can be defined thus: *the meaning of a word is its use in the language*”.
- **Firth (1957**): ”You shall know a word by the company it keeps”.

Representing Meaning as Numerical Vectors

- WordToVec
- Glove
- TagLM
- ELMO
- GPT
- BERT

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Module 6 – Section 6

Resources and Wrap-up

Resources

- http://nlpprogress.com/english/dependency_parsing.html
- https://nlp.stanford.edu/IR-book/html/htmledition/tokenization-1.html
- https://wordnet.princeton.edu/
- https://www.nltk.org/
- https://github.com/mikhailklassen/Mining-the-Social-Web-3rd-Edition/tree/master/notebooks
- https://en.wikipedia.org/wiki/Regular_expression
- Lane, Howard & Hapke. Natural Language Processing in Action. Manning. 2019.
- Jurafsky & Martin. Speech and Language Processing, 3rd Ed. https://web.stanford.edu/~jurafsky/slp3/
- SpaCy: https://spacy.io/
- gensim: https://radimrehurek.com/gensim/
- Natural Language Processing with Python Steven Bird, Ewan Klein, and Edward Loper

Next Class

- Word Vectors
- Sequence-to-Sequence Models
- Attention and Transformers

Follow us on social

Join the conversation with us online:

 facebook.com/uoftscs

 @uoftscs

 linkedin.com/company/university-of-toronto-school-of-continuing-studies

 @uoftscs

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Any questions?

UNIVERSITY OF TORONTO
SCHOOL OF CONTINUING STUDIES

Thank You

Thank you for choosing the University of Toronto School of Continuing Studies