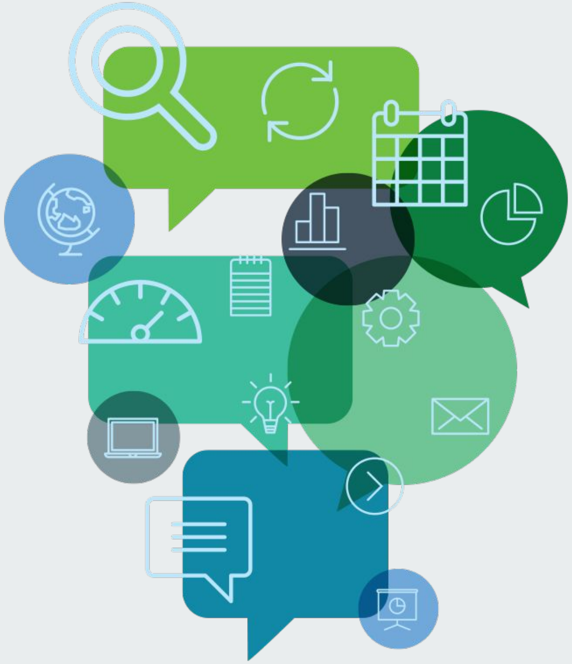
WOMEN WHO CODE DATA SCIENCE

Deep Dive 1

Feature Engineering -
POS Tagging, Chunking, Entity Parsing, Phrase Detection, N-Grams
Implement with NLTK, Spacy



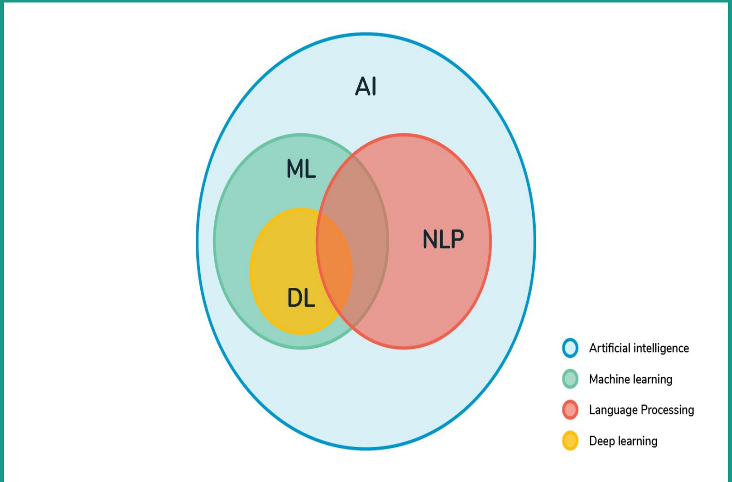
Source: https://www.sas.com/en_us/insights/analytics/what-is-natural-language-processing-nlp.html

WOMEN WHO CODE DATA SCIENCE

Agenda

1. Recap Week 1
2. **Feature Engineering** - Syntactic vs Semantic
3. Part of Speech (POS) Tagging
4. Shallow Parsing or Chunking
5. **Feature Engineering** - Entity Parsing
6. Named Entity Recognition
7. N-Grams
8. Implement with Google Colab
9. Questions?
10. Wrap-up and Next Steps

Recap - Week 1



What?

Where?

Recap - Week 1

WOMEN WHO CODE DATA SCIENCE

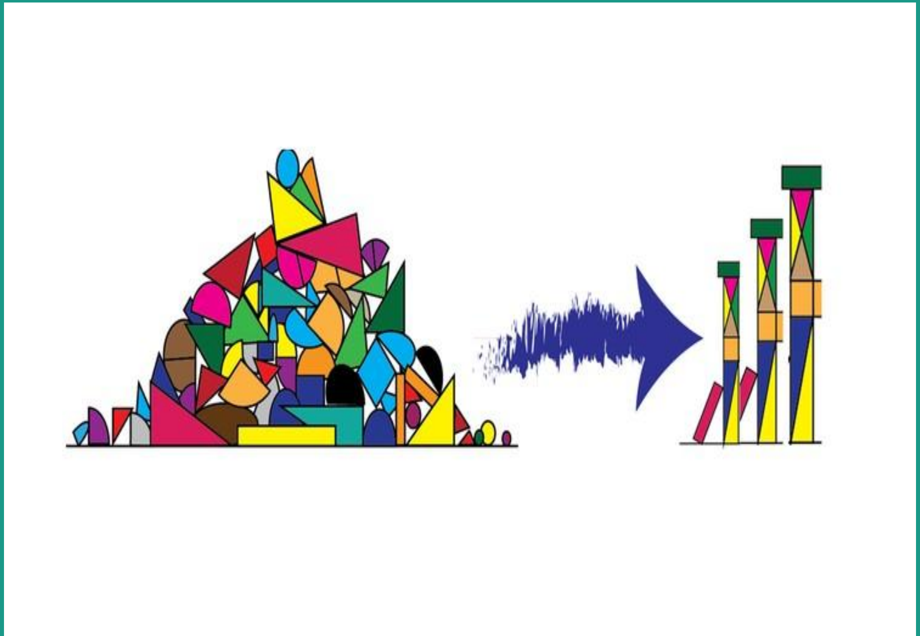
Challenges?

Must do
Pre-processing

WOMEN WHO CODE® DATA SCIENCE

1

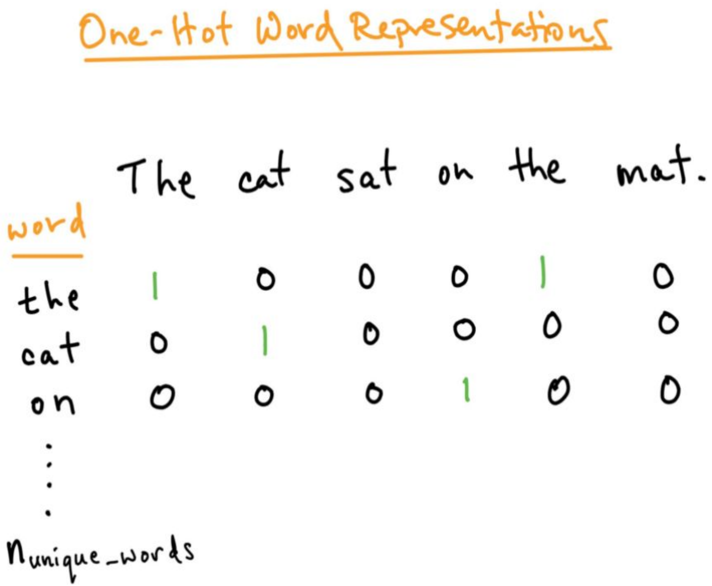
What is **Feature Engineering?**



Source: https://www.udemy.com/course/feature-engineering-for-machine-learning/

Feature Engineering Overview

- ML algorithms cannot work on the raw text directly
- Algorithms can only process numeric representation of an actual text
- FE techniques used to convert text into a matrix (or vector)
- Popular methods of feature extraction are : Bag-of-Words, TF-IDF



Source: https://www.kdnuggets.com/2019/10/introduction-natural-language-processing.html

What are features?

Boston Housing Dataset

1. average number of rooms
2. per capita crime rate by town
3. proportion of non-retail business acres per town
4. index of accessibility to radial highways

Understanding Syntax & Structure

dog the over he
lazy jumping is the fox
and is quick brown

- Syntax and structure are co-dependent
- A set of specific rules, conventions, and principles govern the way words are combined
- English language constituents include: words, phrases, clauses, and sentences
- Unordered words don’t convey much information
- Syntactic analysis (syntax) and semantic analysis (semantic) - primary techniques to understand natural language

Source: https://www.kdnuggets.com/2018/08/understanding-language-syntax-and-structure-practitioners-guide-nlp-3.html

Syntactic Vs. Semantic Analysis

- Syntax is the grammatical structure of the text, semantics is the meaning conveyed
- Sentence that is syntactically correct, may not always be semantically correct
- Syntactic analysis basically assigns a semantic structure to text
- Semantic analysis is the process of understanding the meaning and interpretation of words, signs and sentence structure
- Ex, “cats flow supremely” is grammatically valid (subject — verb — adverb) but it doesn't make any sense

Source: https://builtin.com/data-science/introduction-nlp; getty images

2

Feature Engineering
For text

Source: https://towardsdatascience.com/feature-engineering-what-powers-machine-learning-93ab191bcc2d

WOMEN WHO CODE DATA SCIENCE

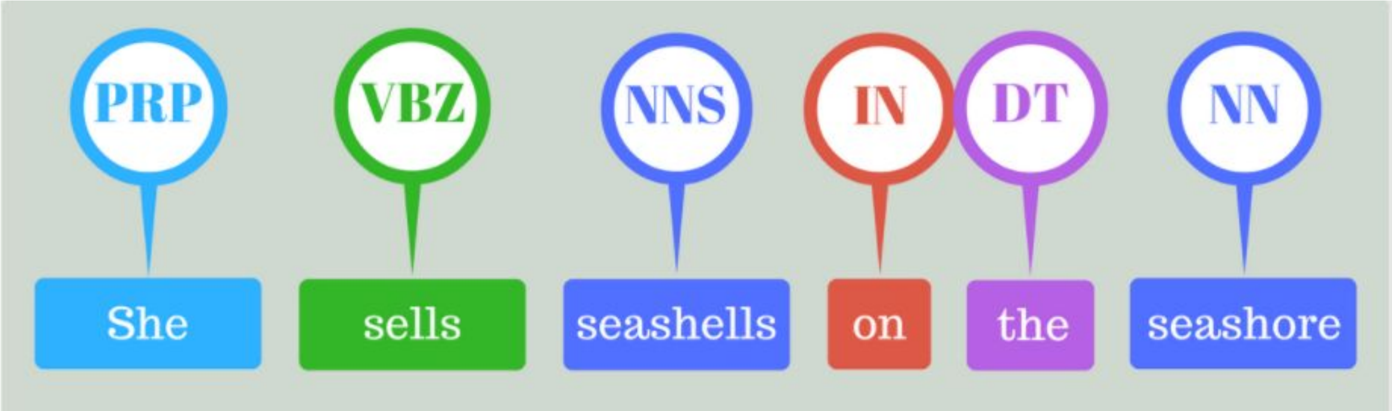
WOMEN WHO CODE DATA SCIENCE

Techniques to Understand Text

Part of Speech (POS) Tagging

Image Source:
https://medium.com/greyatom/learning-pos-tagging-chunking-in-nlp-85f7f811a8cb

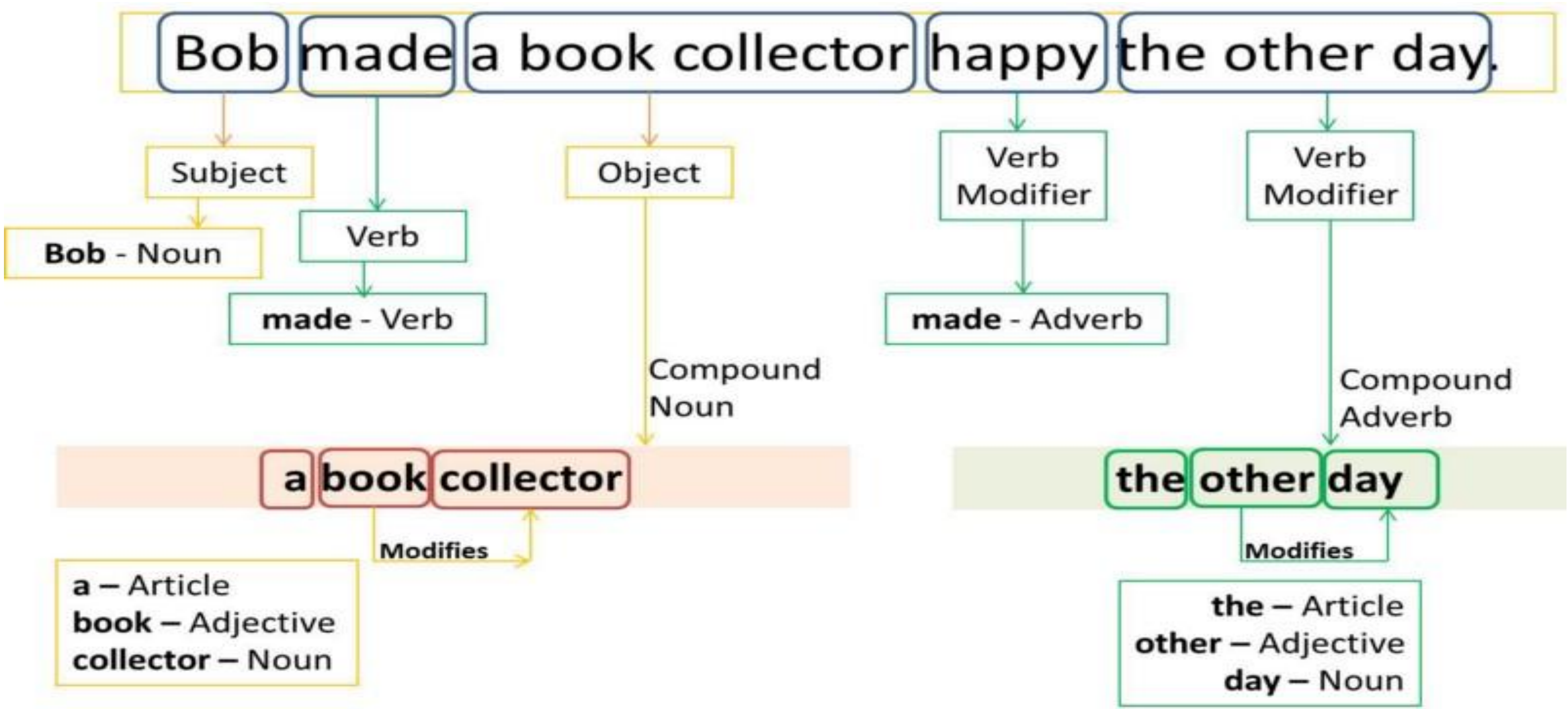
Part-of-Speech Tagging



Also called grammatical tagging or word-category disambiguation, is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition and its context

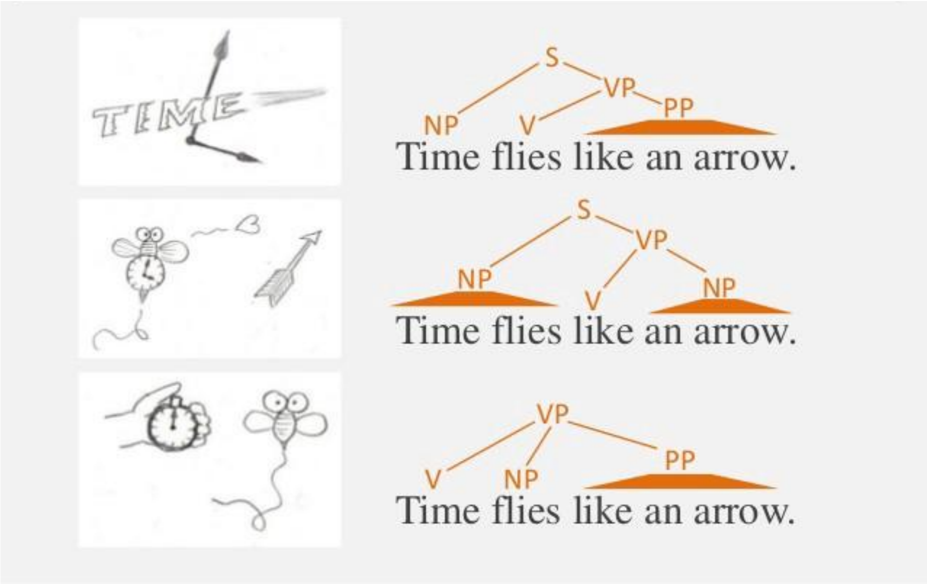
Source: https://nlpforhackers.io/wp-content/uploads/2016/08/Intro-POS-Tagging.png

Part-of-Speech Tagging



Source: https://www.freecodecamp.org/news/an-introduction-to-part-of-speech-tagging-and-the-hidden-markov-model-953d45338f24/

Why POS tagging is needed?



– To solve word sense disambiguation

– Information Retrieval

– Text to Speech: object(N) vs. object(V)

E.g. Time (N) vs. Time(V)

– Machine Translation

Refer to Penn Treebank Project: https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

Source: https://nlpforhackers.io/wp-content/uploads/2016/08/Intro-POS-Tagging.png

Methods for POS tagging

Rule-Based POS tagging –
e.g. ENGTWOL [Voutilainen, 1995]

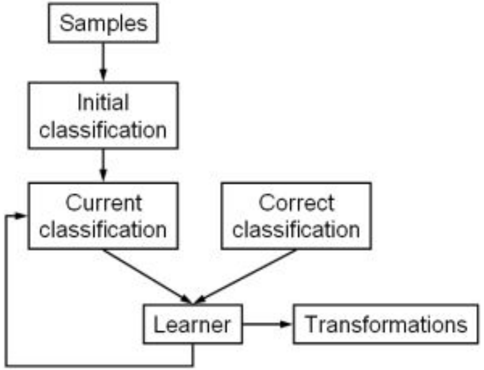
1. Use contextual information to assign tags to unknown
2. Disambiguation is done by analyzing the linguistic features of the word
3. Manual, time consuming, not scalable

Example of a rule:

If an ambiguous/unknown word X is preceded by a determiner and followed by a noun, tag it as an adjective.

Transformation-based tagging –
e.g.,Brill's tagger [Brill, 1995]

1. Transformation-based Error-driven Learning (TEL)
2. Tagger is based on transformations or rules, and learns by detecting errors.



Stochastic (Probabilistic) tagging – e.g., TNT [Brants, 2000]

1. Based on probability of certain tag occurring
2. Necessitates a training corpus
3. Brown Corpus - 1M words
4. Hidden Markov Model (HMM) - uses both tag sequence probabilities and word frequency measurements

Source: https://pdfs.semanticscholar.org/e5b1/c7446207b0beac658cd1631b4da06ed17c82.pdf; https://pdfs.semanticscholar.org/ae31/1cb0f1d821840d4309c9d33c0da0525f621d.pdf

POS Tagging Challenges

POS tags are not generic. Problem is **Ambiguity** in English language
A **single word** can have different tag in different sentences based on **different contexts**

Eg 1:
She saw a bear (Bear → Noun)
Your efforts will bear fruit (Bear → Verb)

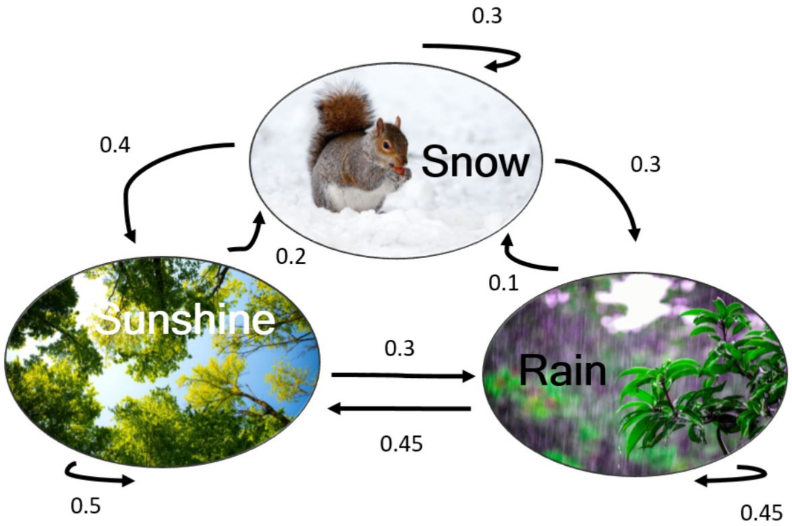
Eg 2:
The trash can is hard to find (Can → Noun)
I can do better (Can → Modal Verb)

Retrieving POS tags have 2 part components:

1. Individual words with statistical preferences for their POS
2. Context has an important effect on the POS for a word

Mathematical concept for POS tagging

Markov Models



Basic Formula:

$$\begin{bmatrix} \text{NEXT} \quad \text{STATE} \end{bmatrix} = \begin{bmatrix} \text{MATRIX OF} \\ \text{TRANSITION} \\ \text{PROBABIIITIES} \end{bmatrix} \begin{bmatrix} \text{CURRENT STATE} \end{bmatrix}$$

States: SUNSHINE, SNOW & RAIN

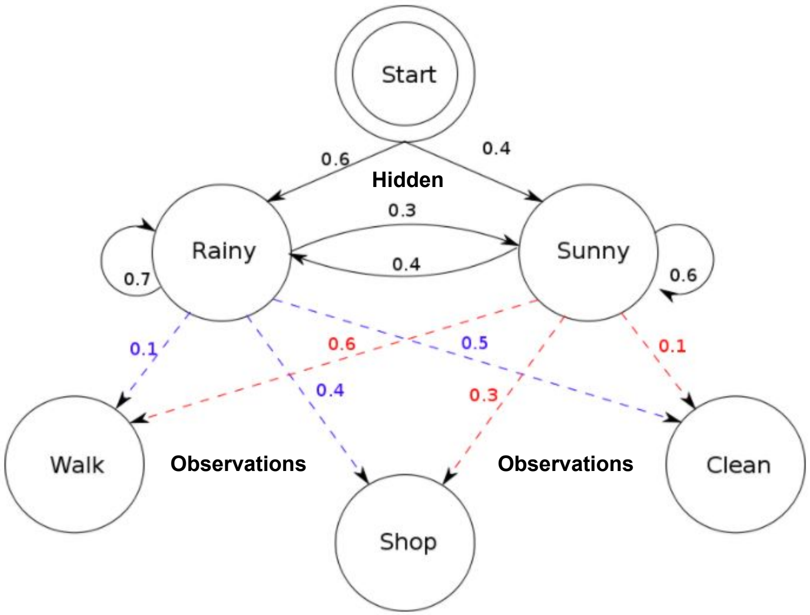
State transition probability: Decimal numbers (State1 $\rightarrow$State2)

E.g.: There is 0.2 probability of SNOW tomorrow if today it is SUNSHINE.

Source : https://towardsdatascience.com/introduction-to-hidden-markov-models-cd2c93e6b781

Mathematical concept for POS tagging

Hidden Markov Models



Hidden States: SUNNY, RAIN

Observation States: WALK, SHOP, SUNNY

Transition Prob: Prob of 1 hidden state to another

Emission Prob: Obs are emitted from Hidden States

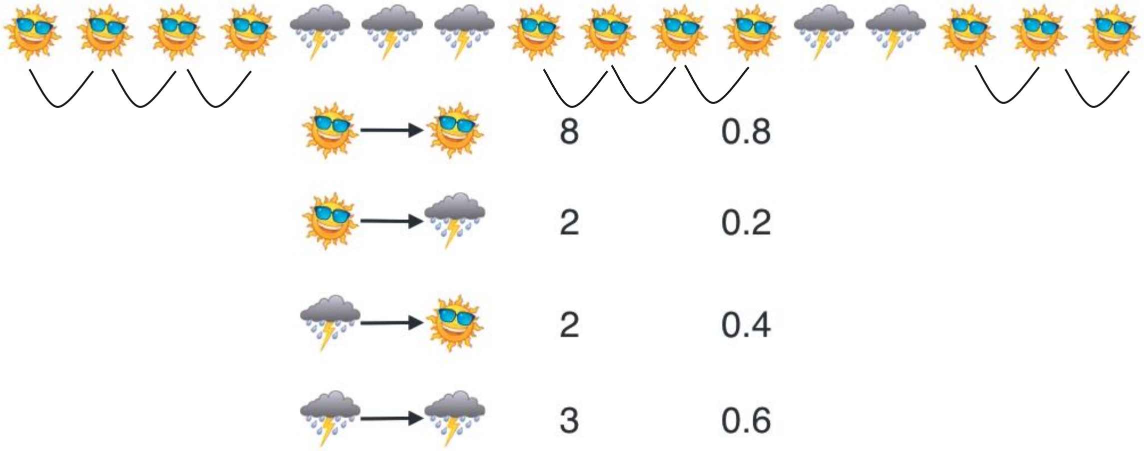
Logic:

Predict sequence of states not directly observable, given another sequence of states that are observable and hidden states have some **dependence** on the observable states

Source : https://towardsdatascience.com/introduction-to-hidden-markov-models-cd2c93e6b781

How are we getting the probability?

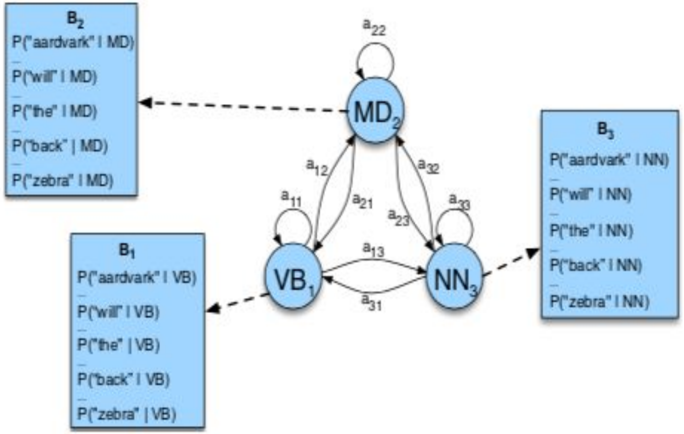
From past data observations:



Source : https://www.youtube.com/watch?v=kqSzLo9fenk

POS tagging and HMM - Related how?

Observable States: words in a sentence
Hidden States: POS Tags
Estimating POS tags: using HMM



Matrix A: Matrix contains the tag **transition** probabilities P(ti|ti−1)

Set of possible Tags: Calculate A[Verb][Noun]:

P (Noun|Verb)= Count(Noun & Verb)/Count(Verb)

Matrix B = **emission** probabilities, P(wi|ti)
Sequence of observation (words in the sentence):
Given a tag (Verb), it's associated with a given word (Running)

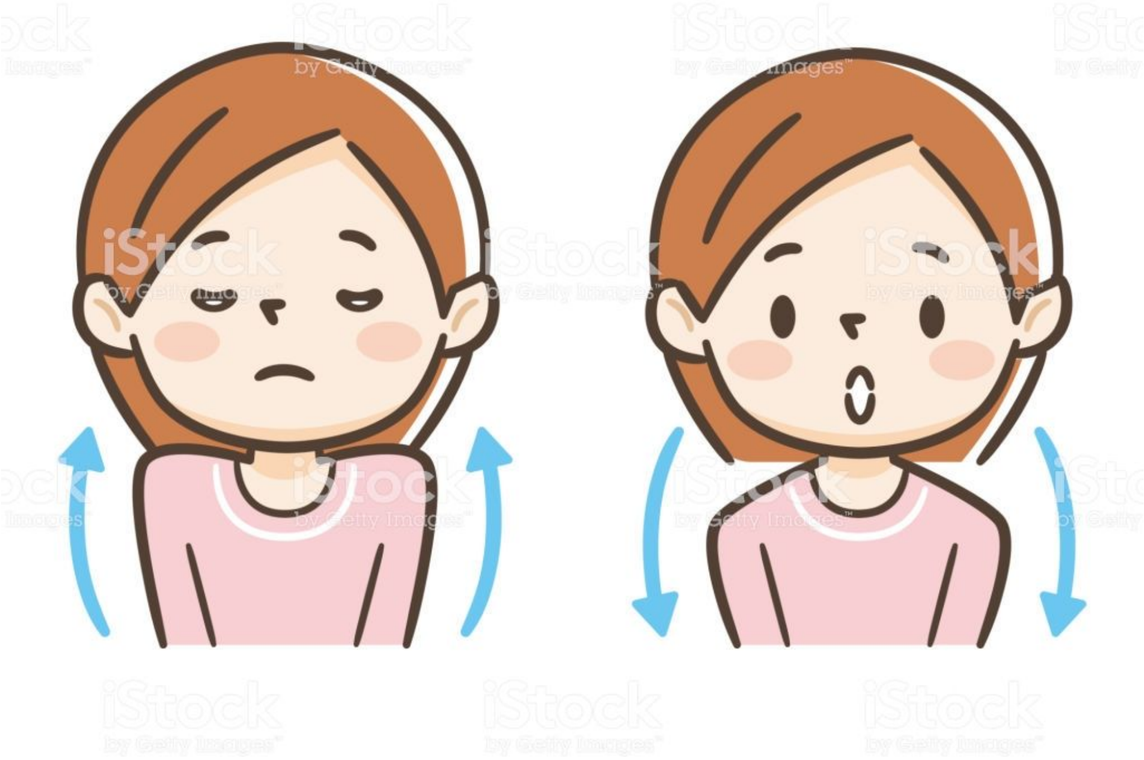
The emission probability B[Verb][Running]:

P(Running|Verb)=Count(Running & Verb)/Count(Verb)

Source : https://towardsdatascience.com/introduction-to-hidden-markov-models-cd2c93e6b781

WOMEN WHO CODE DATA SCIENCE

Let’s take few deep breaths now!



Techniques to Understand Text

Shallow Parsing or Chunking

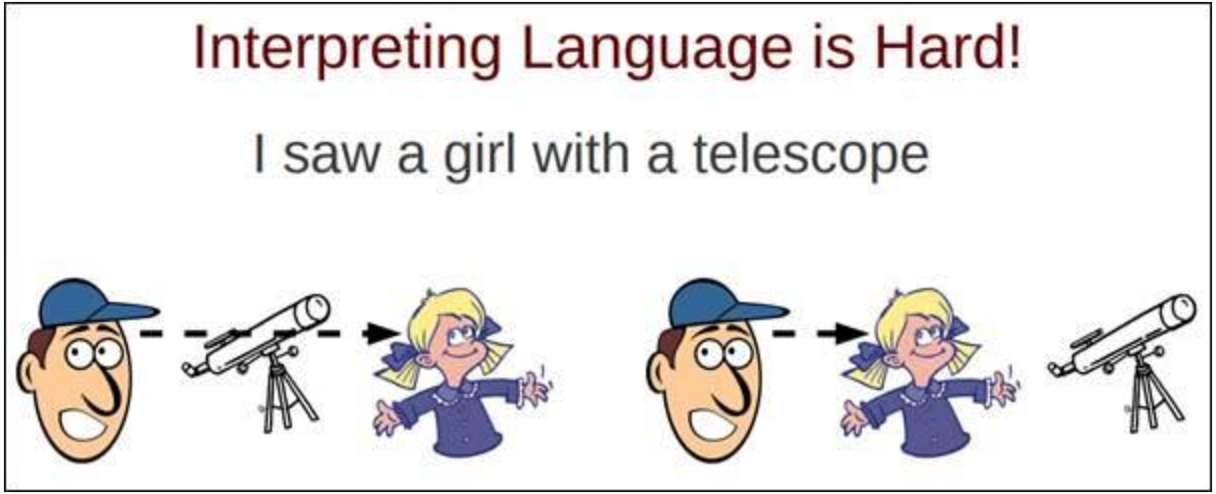
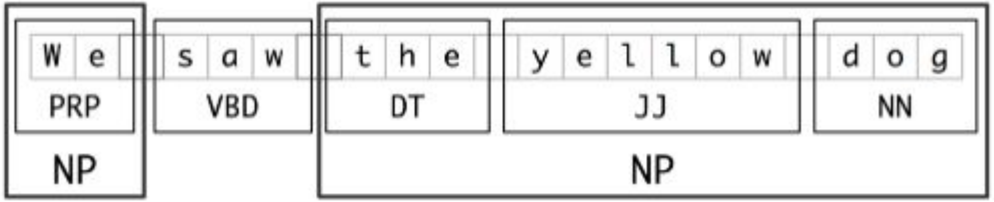


Image Source:
https://www.analyticsvidhya.com/blog/2017/12/introduction-computational-linguistics-dependency-trees/

What is Shallow Parsing?



- Shallow parsing or chunking is a process dividing a text into syntactically related group
- Divide the whole text into non-overlapping contiguous subsets of tokens
- Segments and labels multi-token sequence
- Crucial for information extraction from text to create sub-components such as Locations, Person Names

Source: http://www.nltk.org/book/ch07.html

POS-tagger vs Chunking?

E.g.: Sentence = "the little yellow dog barked at the cat"

POS tagged:

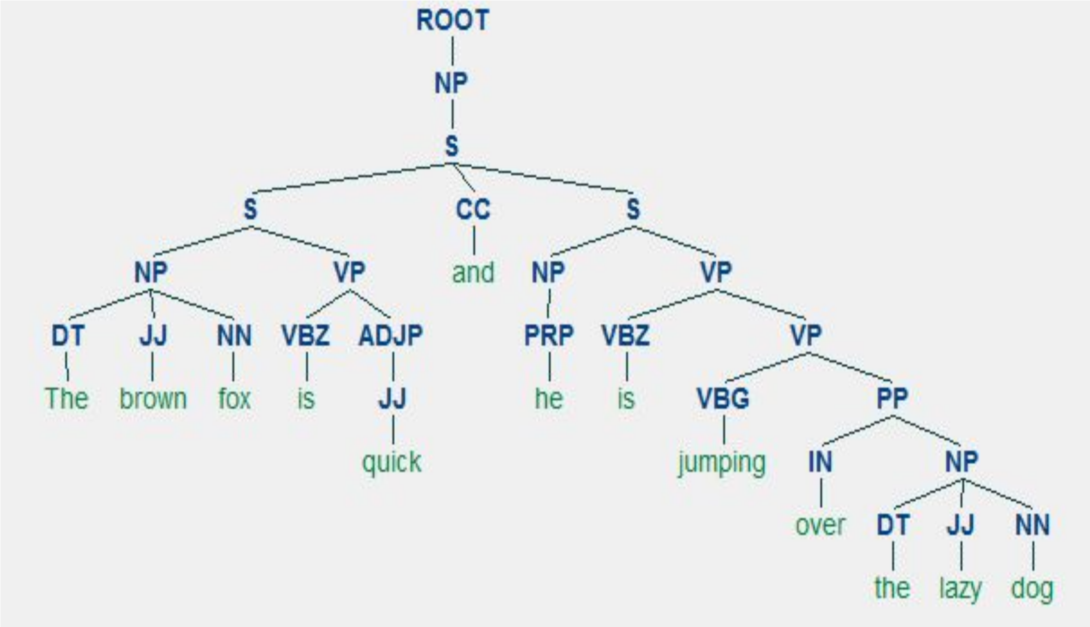
[('the', 'DT'), ('little', 'JJ'),
('yellow', 'JJ'), ('dog', 'NN'),
('barked', 'VBD'), ('at', 'IN'),
('the', 'DT'), ('cat', 'NN')]

Chunked Tree:

Source: https://medium.com/greyatom/learning-pos-tagging-chunking-in-nlp-85f7f811a8cb

Other methods of Parsing

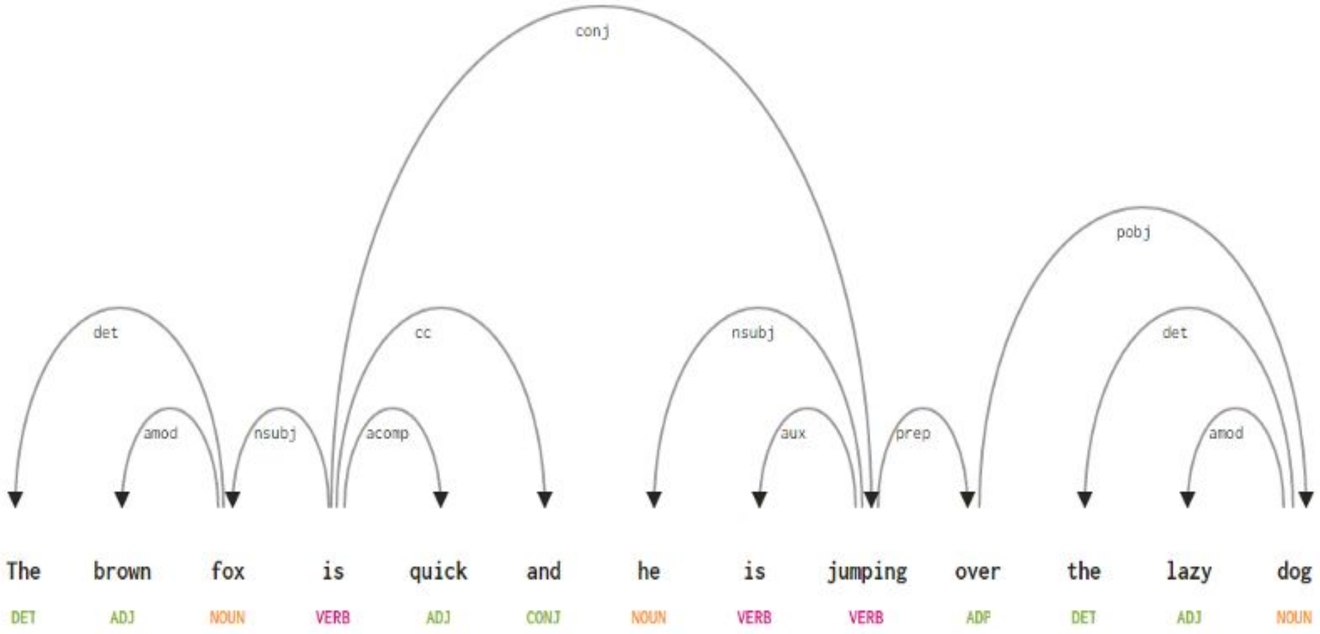
Constituency Parsing



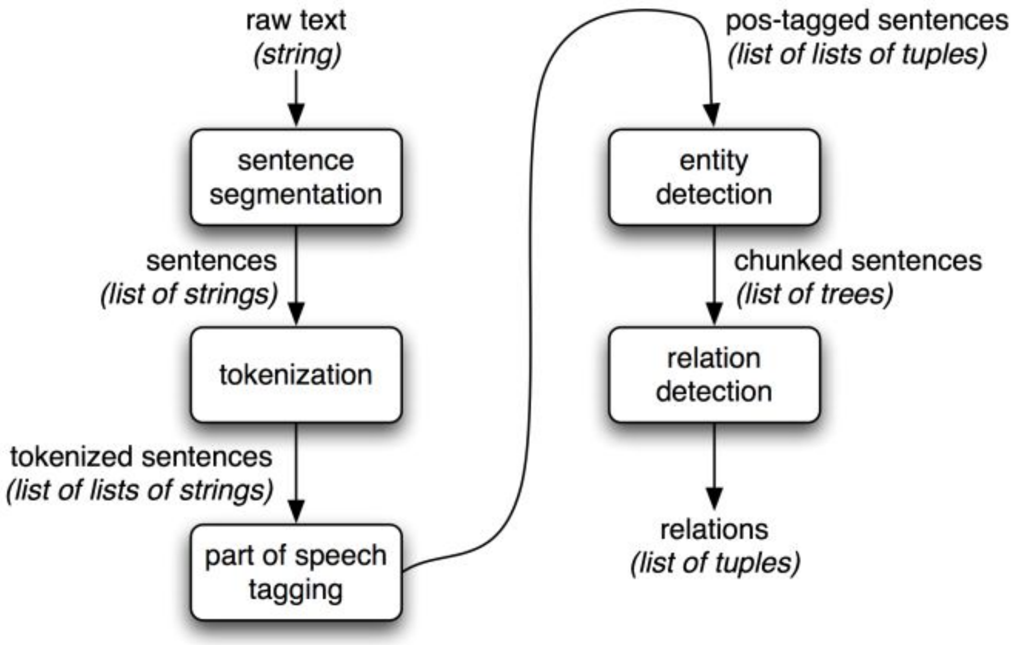
Source: https://web.stanford.edu/~jurafsky/slp3/13.pdf

Other methods of Parsing

Dependency Parsing

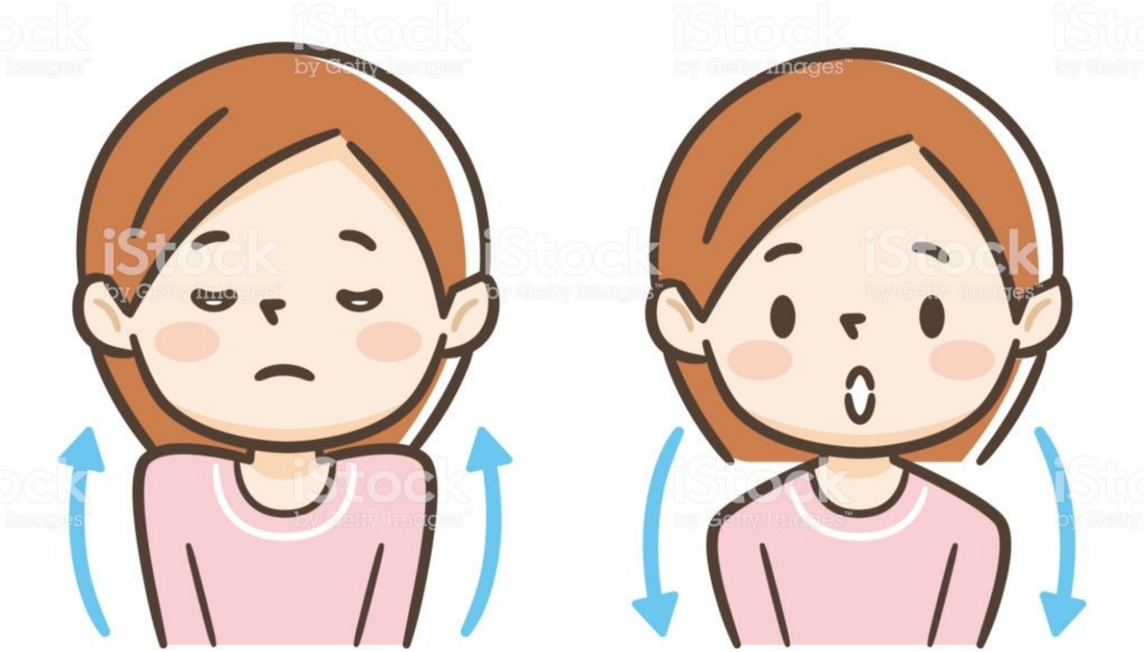


Understanding Entity Parsing



Source: https://nickcdryan.com/2017/02/13/shallow-parsing-for-entity-recognition-with-nltk-and-machine-learning/

Let’s take few deep breaths now!



WOMEN WHO CODE DATA SCIENCE

Techniques to Understand Text

Named Entity Recognition (NER)

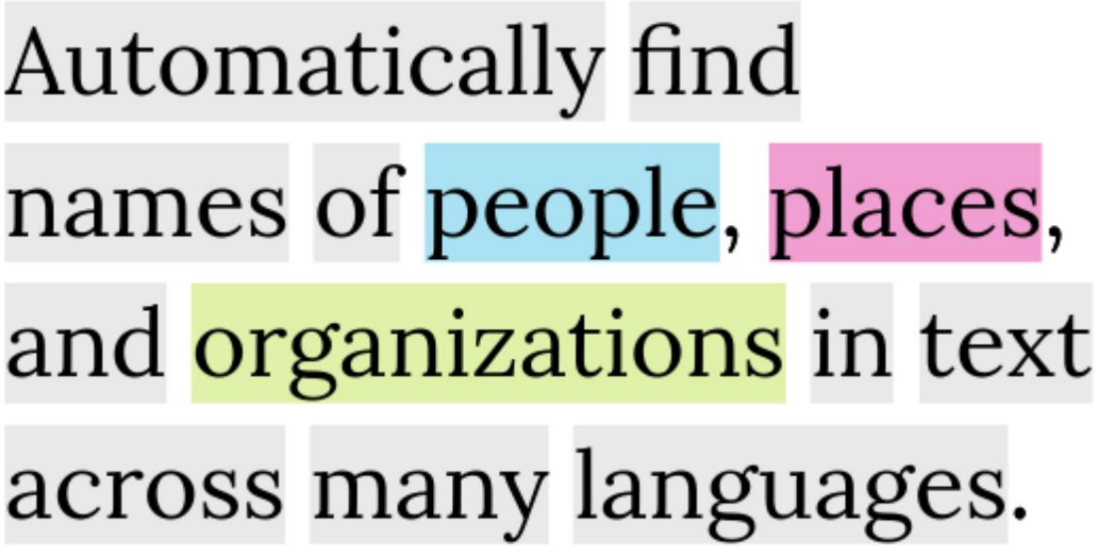


Image Source: https://www.splunk.com/en_us/blog/machine-learning/named-entity-recognition-and-extraction.html

What is NER?

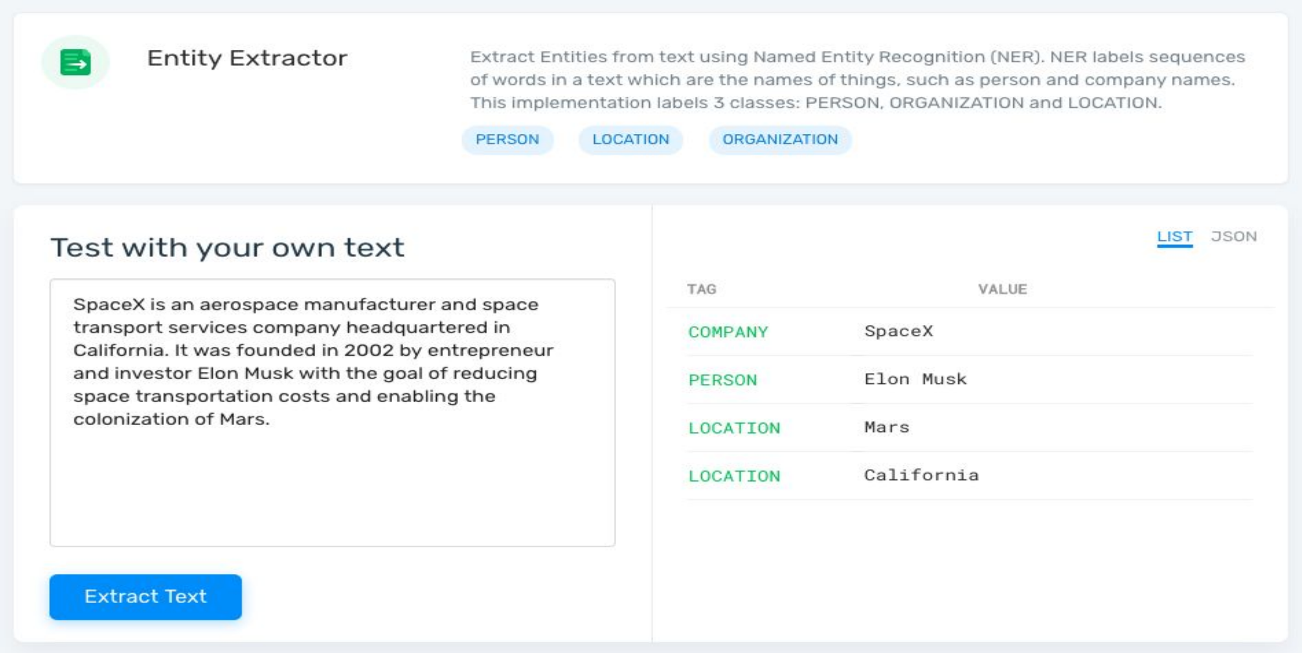
- Terms that represent specific entities that are more informative and have a unique context
- Represent real-world objects like people, places, organizations, and so on, which are often denoted by proper names
- Used in information extraction to identify and segment the named entities in predefined classes

Source: https://towardsdatascience.com/named-entity-recognition-with-nltk-and-spacy-8c4a7d88e7da

WOMEN WHO CODE DATA SCIENCE

Steps of NER

1. Detect a Named Entity
2. Extract the Entity
3. Categorize the Entity



Source: https://app.monkeylearn.com/main/extractors/ex_isnnZRbS/

Methods for NER extraction

Lexicon approach

1. relies on a knowledge base called ontology
2. contains all terms related to a particular topic, grouped in different categories
3. system looks for matches with named entities

Cons:
1. doesn't work to extract new words not in lexicon

Example:
lexicon of cities, states, and countries to recognize locations in data.

Rule-based systems

1. series of grammatical rules hand-crafted by computational linguists
2. can get results of high precision but low recall

Cons:
1. Defining rules takes time
2. Domain specific

Example:
Build a model to extract "legal terms", you need to manually tag tokens of legal methods, cases, process

Machine learning-based systems

1. build an entity extractor
2. feed the model with a large volume of annotated training data

1. Cons:

Need tagged and clean training data

Example:
Build a model to extract "legal terms", you need to manually tag tokens of legal methods, cases, process

Source: https://monkeylearn.com/blog/named-entity-recognition/

Industry Use-Case of NER?

Problem

Reduce time for processing customer queries and tickets

Goal

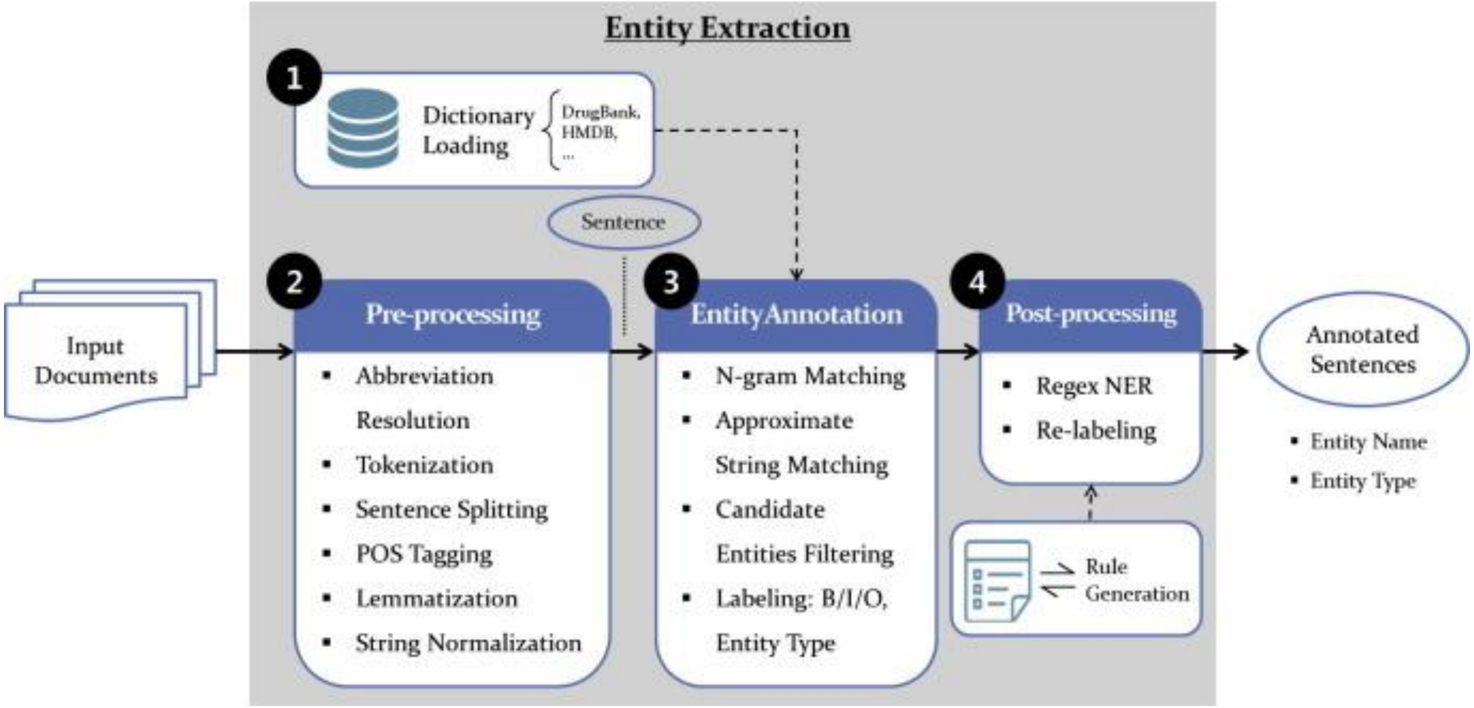
Increase customer satisfaction by reducing queue time and solving problems faster

Constraints

Limited number of agents to respond to tickets, minimize burden

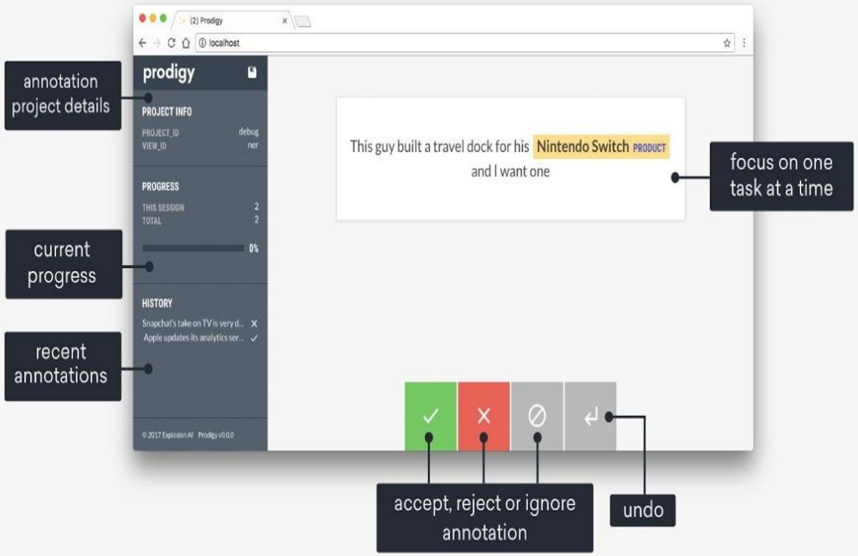
Source: http://www.nltk.org/book/ch07.html

Training a NER model



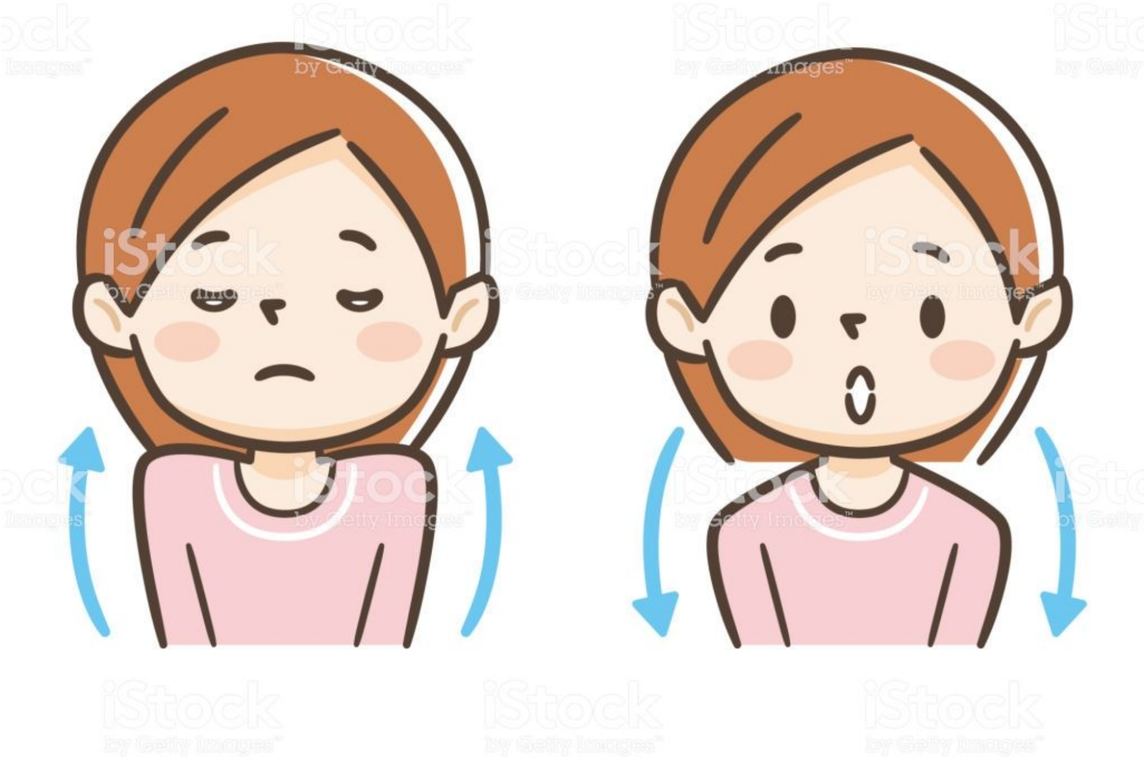
Source: https://www.sciencedirect.com/science/article/pii/S1532046415001756

Annotation for NER



Source: https://prodi.gy/docs

Let’s take few deep breaths now!

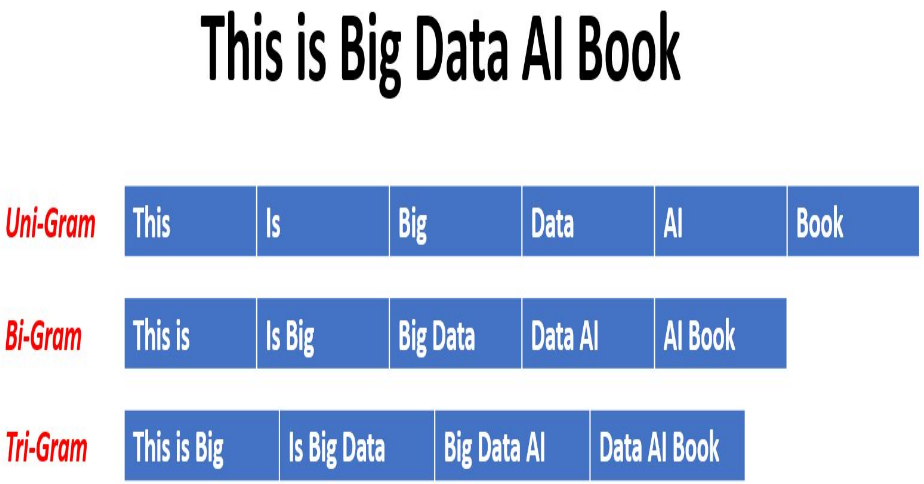


WOMEN WHO CODE DATA SCIENCE

Techniques to Understand Text

N-Grams

What are n-grams?



- N-gram model is a type of Language Model (LM), which is about finding the probability distribution over word sequences
- N-gram means a sequence of N words
- N-grams cut out the noise from the data in your analyses
- Identify themes quickly
- NLP applications including speech recognition, machine translation and predictive text input

Source: https://devopedia.org/images/article/219/7356.1569499094.png

Mathematical concept of n-grams

Probability of a word w, given some history, h = P(w|h)

Eq: P(the|today the sky is so clear that)

Here,
w =the
h = today the sky is so clear that

Approach 1: relative frequency count

Step1: Take a large corpus
Step2: count the number of times **today the sky is so clear that** appears
Step3: count the number of times it is followed by **the**

Eq:
P(the|today the sky is so clear that) =
C(today the sky is so clear that the)/
C(today the sky is so clear that)

Basically, we need to answer:
Out of the times you saw the history h, how many times did the word w follow it

Source: https://towardsdatascience.com/introduction-to-language-models-n-gram-e323081503d9

Mathematical concept of n-grams

Cons of Approach 1:

1. If we have a large corpus, Approach 1 needs to go over entire corpus
2. Not feasible for scaling as well as time performance
3. To decompose the probability function into smaller chunks leads to the usage of **chain rule**

Instead of computing probability using the entire corpus, chain rule using N-grams would approximate it by just a **few historical words**

Source: https://towardsdatascience.com/introduction-to-language-models-n-gram-e323081503d9

Mathematical concept of n-grams

Probability of a word w, given some history, h = P(w|h)

Eq: P(the|today the sky is so clear that)

Here,
w_n = the
h = today the sky is so clear that
w_{n-1} = that

Approach 2: Bigram Model

Process: Approximates the probability of a word given all the previous words by using only the conditional probability of one preceding word

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-1})$$

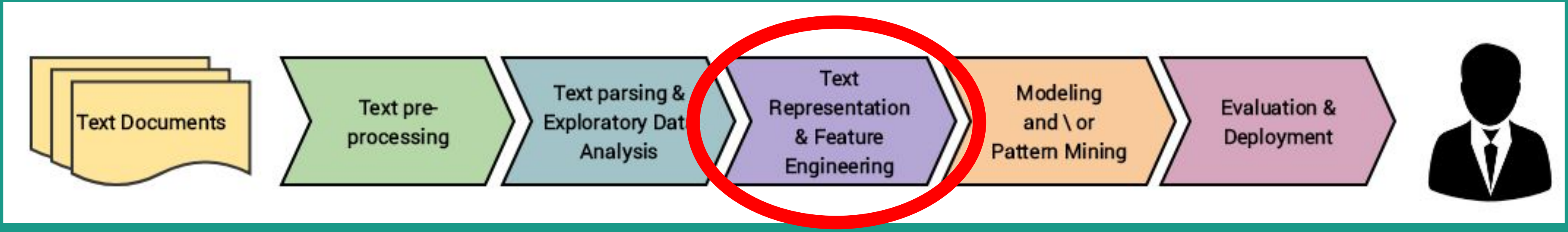
Assumption that the **probability of a word** depends only **on the previous word** = Markov assumption

Def: Markov models are the class of probabilistic models that assume that we can **predict the probability of some future unit** without looking too far in the past.

Source: https://towardsdatascience.com/introduction-to-language-models-n-gram-e323081503d9

4

NLP Workflow



Source: https://towardsdatascience.com/a-practitioners-guide-to-natural-language-processing-part-i-processing-understanding-text-9f4abfd13e72

5

Theory Wrap-up & Next Steps

WOMEN WHO CODE DATA SCIENCE

Recap

01 Syntax vs Semantics

02 Part of Speech (POS) Tagging

03 N-Grams

04 Named Entity Recognition

05 Shallow Parsing or Chunking

Google Colab Project

https://bit.ly/introtonlp-week2-notebook

WOMEN WHO CODE DATA SCIENCE

WOMEN WHO CODE DATA SCIENCE

Homework #1

Additional Resources

- (Youtube) Hidden Markov Model by Luis Serrano
- (FreeCodeCamp) Part of Speech Tag and Hidden Markov Model
- (AnalyticsVidhya) Dependency
- (TowardsDataScience) N-grams

WOMEN WHO CODE DATA SCIENCE

See you next week!

Questions?

Join us on Slack and post your questions to the #help-me channel