Natural Language Processing

AI and Machine Learning

Hult International Business School

Michael de la Maza

Version 1.0

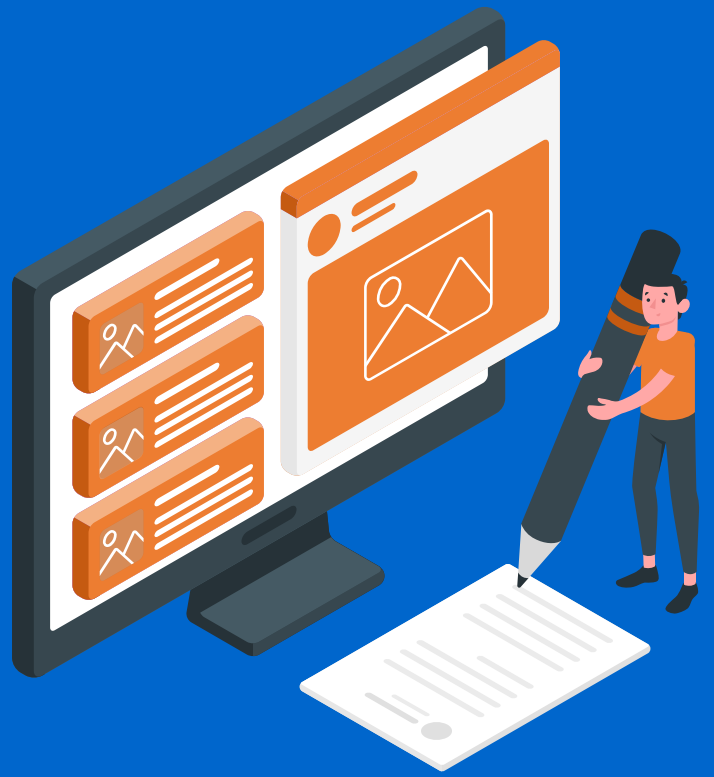
Natural language is unstructured data

- We have been working with structured data. Each instance has interpretable attributes / features / fields.
  - Example: Bike Rentals database has temp, humidity fields
- Consider a magazine article, an example of natural language
  - How would you represent this as structured data?
  - How would you 'feed' the article into a neural network?
  - Let's say that you want to classify magazine articles into two categories (sports and non-sports). How would you do that?

Examples of unstructured datasets

- Magazine articles
- Electronic medical records
- Insurance claims
- Tweets, Facebook posts
- Images (will examine in the next session)

Sometimes datasets will have both structured and unstructured data (e.g., medical records)



“Bag of Words” technique

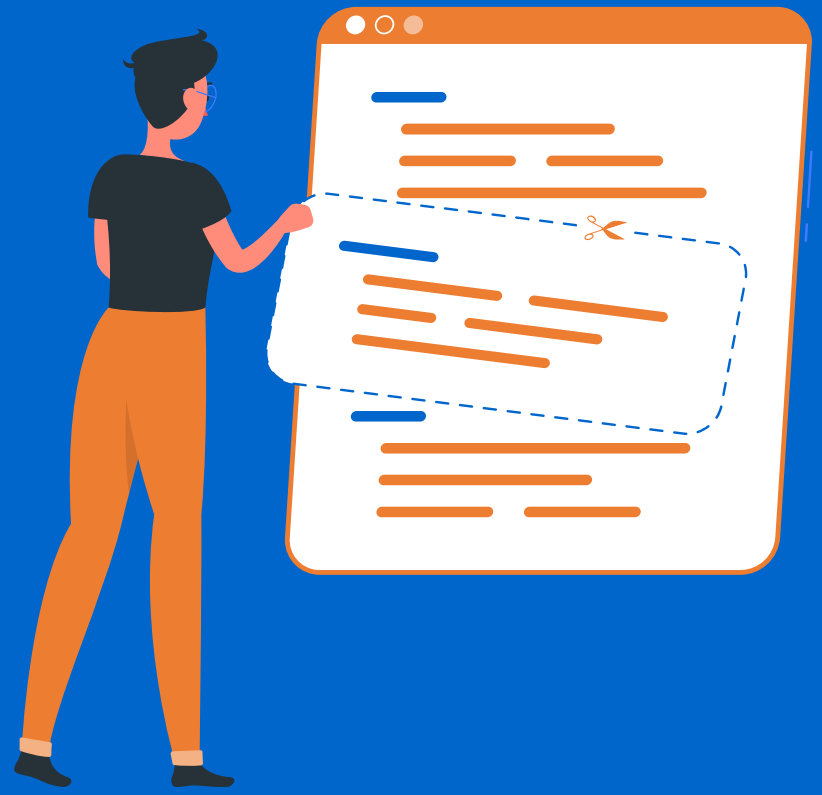
- A key challenge is to *represent* natural language in a manner that works for a neural network
- One simple technique is called ‘bag of words’
- Text is represented as a vector with the ith position corresponding to either:
  - Presence: 1 if the ith word exists in the text; 0 otherwise
  - Frequency: An integer corresponding to the number of times a word appears in the text
- Simple example:
  - The cat and the dog are playing
  - The cat is playing

AND	ARE	CAT	DOG	IS	THE	PLAYING
1	1	1	1	0	1/2	1
0	0	1	0	1	1	1

- Typically limit vocabulary to 10,000 words or less. Words that are not in the vocabulary are not represented. These words are extremely infrequent.
- Critically, this approach ignores the order of words. These two sentences have the identical ‘bag of words’ representation:
  - The dog bit the person
  - The person bit the dog
- Despite this (severe?) limitation, this technique works well in many situations!

Cleaning up the text: Stop-word removal and stemming

- Stop-word removal: Eliminate ‘insignificant’ words such as:
  - And | the | a | it | For
- Stemming: Replace words with their root form.
  - fishing => fish
  - playing => play
- Exercise: For the two sentences on the previous slide, perform stop-word removal and stemming



Other ways to preprocess text

- Frequency filters: Eliminate words that almost always or almost never appear.
  - Often domain specific
- Ignore case
- Remove punctuation

End result is a sequence of *tokens*

Let’s look at some code!