Data Science

Session 6 - Working with text


hadrien.salem@centralelille.fr


introduction-to-data-science

Introduction

What did we do last time?

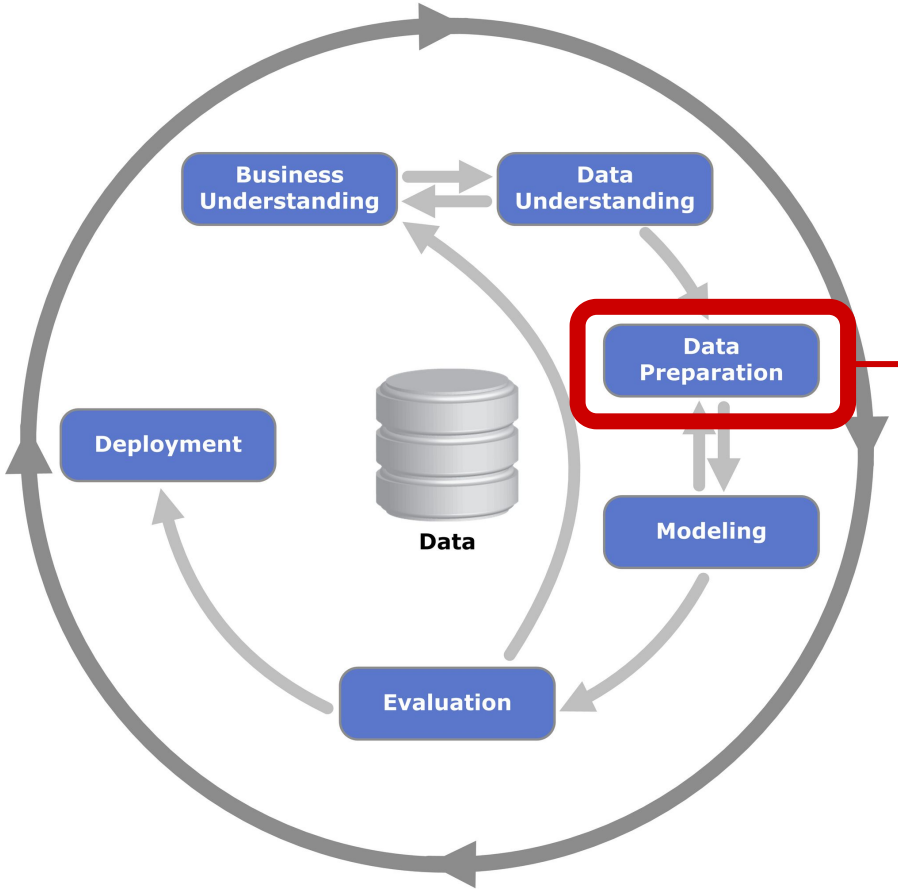
The CRISP-DM method

Cross-**I**ndustry **S**tandard **P**rocess for **D**ata **M**ining

- ➔ Published in 1999
- ➔ Common in the industry
- ➔ Still relevant today

The CRISP-DM method to carry out data-driven projects

(Image source: Wikipedia)



Today's class

The CRISP-DM method to carry out data-driven projects *(Image source: Wikipedia)*

Course outline

Data science course

Session 1: Understanding data

Session 2: Collaborative development

Session 3: Preparing data - Managing missing data

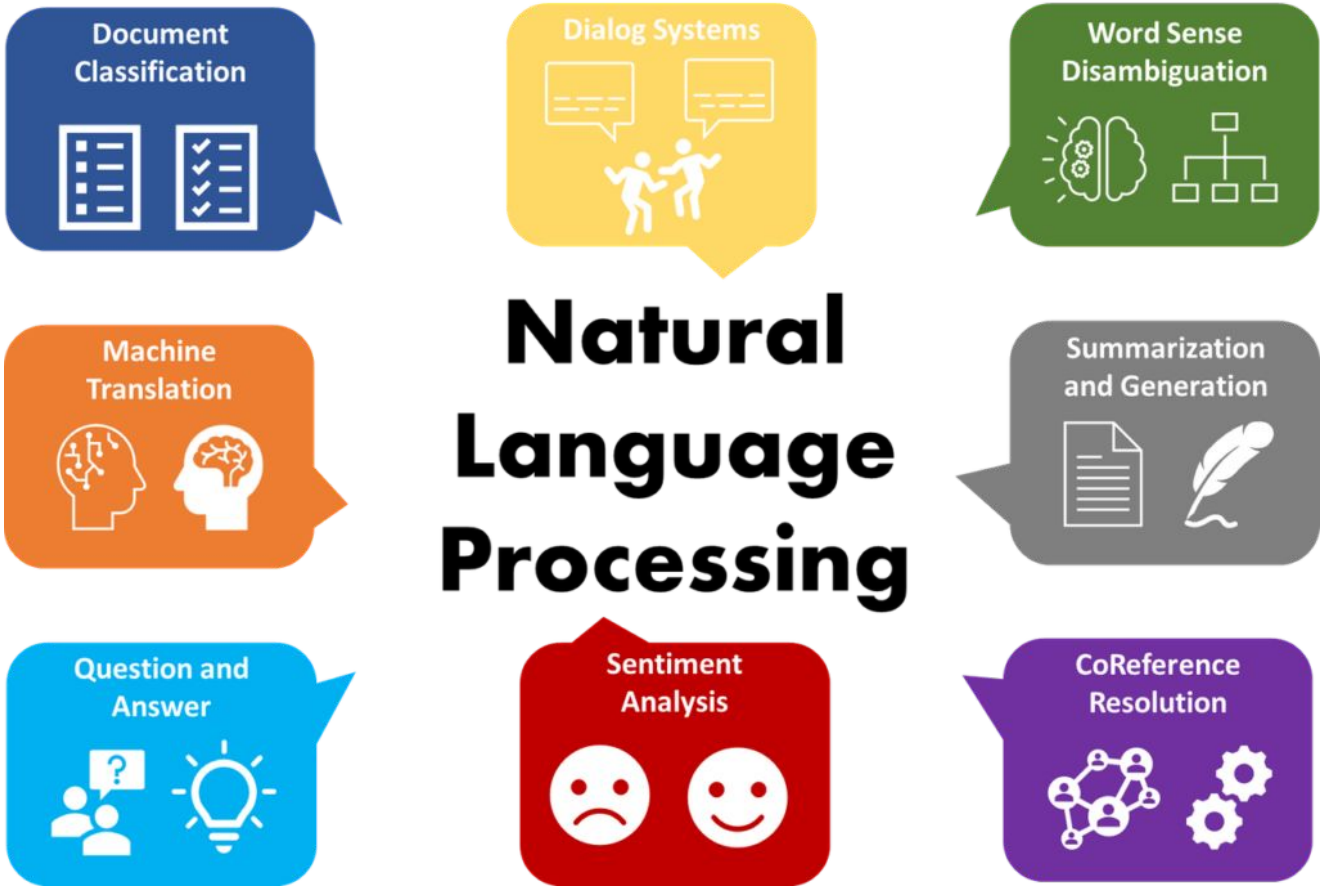
Session 4: Preparing data - Dimensionality reduction

Session 5: Imbalanced data and deidentification

Session 6: Working with text

Machine learning course

What is NLP?



Natural Language Processing encompasses several tasks

Image source

NLP has many applications in healthcare

In healthcare in particular, a lot of unstructured textual data is generated every day.

The automated processing of this data is a major challenge today.

EHR Analysis

Clinical coding

Outbreak detection

Healthcare chatbots

The difficulties of working with text

What challenges does one face when dealing with textual data ?

What challenges does one face when dealing with textual data ?

- Language, dialects and slang
- Contextual words and phrases, homonyms
- Synonyms
- Irony and sarcasm
- Ambiguity
- Misspellings
- Domain-specific language
- Long-range dependencies

Pre-processing is **absolutely crucial** in NLP

Text is extremely noisy by nature.

Processing the text helps algorithms learn from the meaningful parts of it.

How to prepare textual data for ML

Text contains many elements that can typically be cleaned up

- Lower casing
- Removal of punctuations
- Removal of frequent words
- Removal of rare words
- Removal / conversion of emojis
- Removal of URLs
- Spelling correction

I love Centrale Lille ❤️

I	love	Centrale	Lille	❤️

Tokenization is the process of splitting the whole text entity into smaller entities

Stemming vs Lemmatization

TURING

Stemming and Lemmatization

Both are techniques that reduce words to their root form.

They help reducing the dimensionality of the vocabulary.

Some information may be lost in the process.

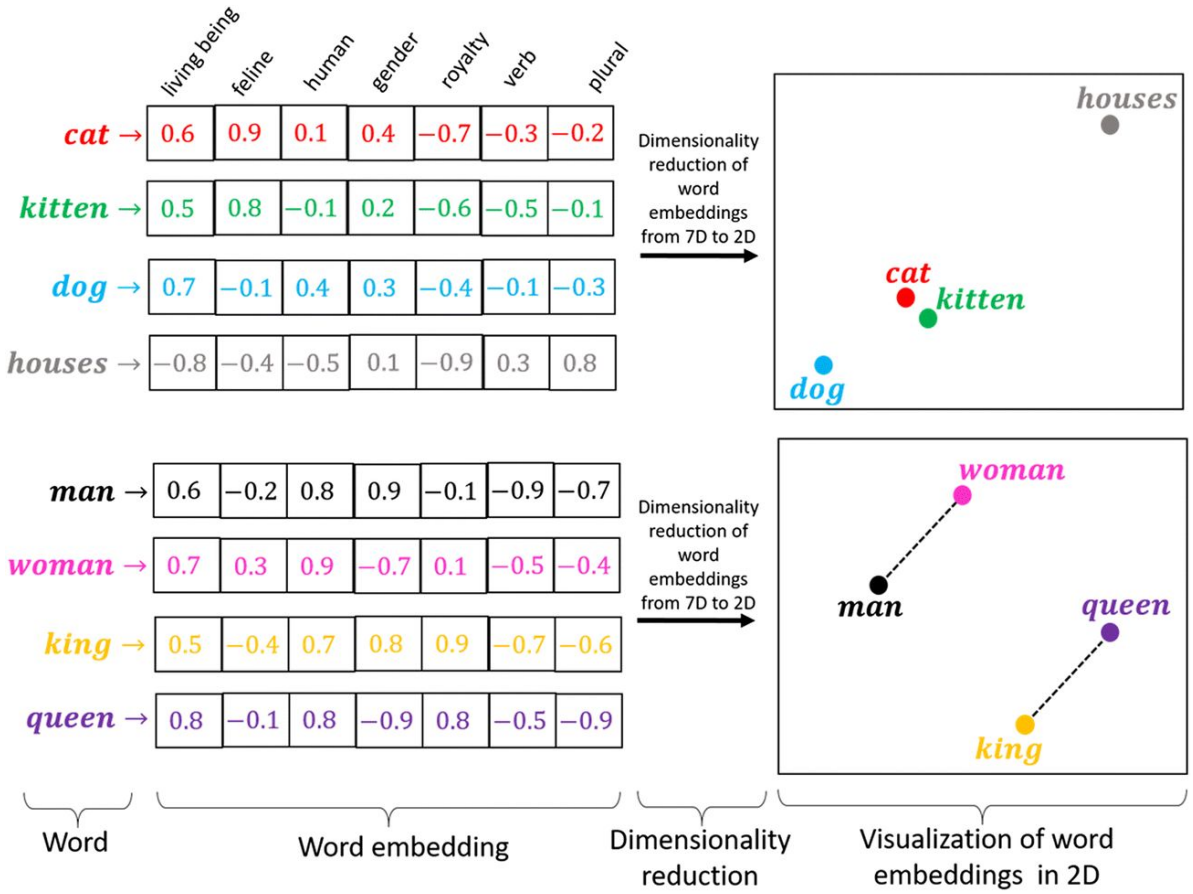
Stemming

- Consists in chopping off prefixes and suffixes
- Fast and simple
- Has limitations for complex word morphologies

Lemmatization

- Using the context to find the dictionary form
- More accurate than stemming
- Computationally expensive

The cleaned up text needs to be converted into vectors: this is called **embedding**



Embedding is the process of turning tokens into vectors

[Image source]

How do you convert tokens into vectors?

Embedding method #1: Bag of Words

	she	loves	pizza	is	delicious	a	good	person	people	are	the	best
She loves pizza, pizza is delicious	1	1	2	1	1	0	0	0	0	0	0	0
She is a good person	1	0	0	1	0	1	1	1	0	0	0	0
good people are the best	0	0	0	0	0	0	1	0	1	1	1	1

Problem: Gives a lot of weight to common words like “the”

Image source

Embedding method #2: TF-IDF

$$\textbf{tf-idf}_{i,j} = \text{Term Frequency}_{i,j} \times \text{Inverse Document Frequency}_i$$

Where,

$$\textbf{Term Frequency}_{i,j} = \frac{\text{Term i frequency in document j}}{\text{Total no. of terms in document j}}$$

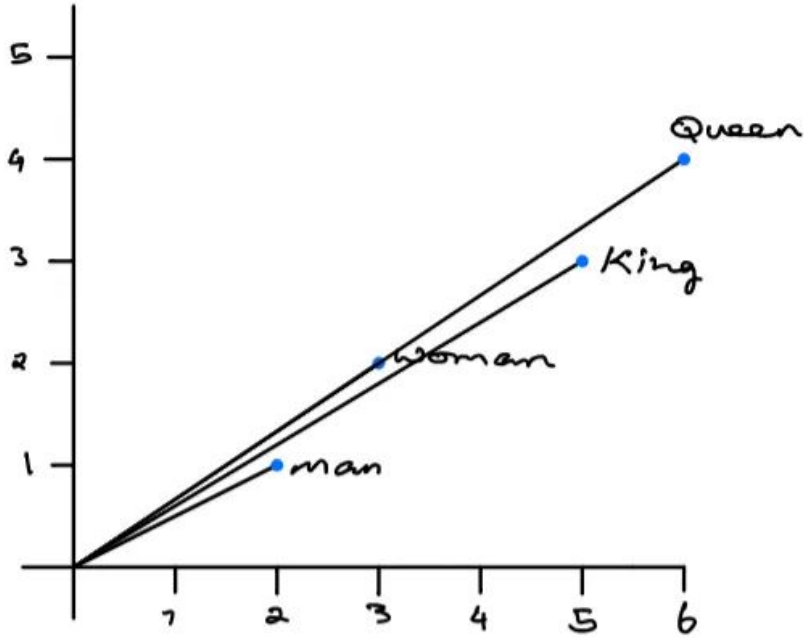
$$\textbf{Inverse Document Frequency}_i = \log\left(\frac{\text{Total documents}}{\text{No. of documents containing term i}}\right)$$

Similar to BoW

Reweighting to give less importance to common words

Image source

Embedding method #3: Word2Vec

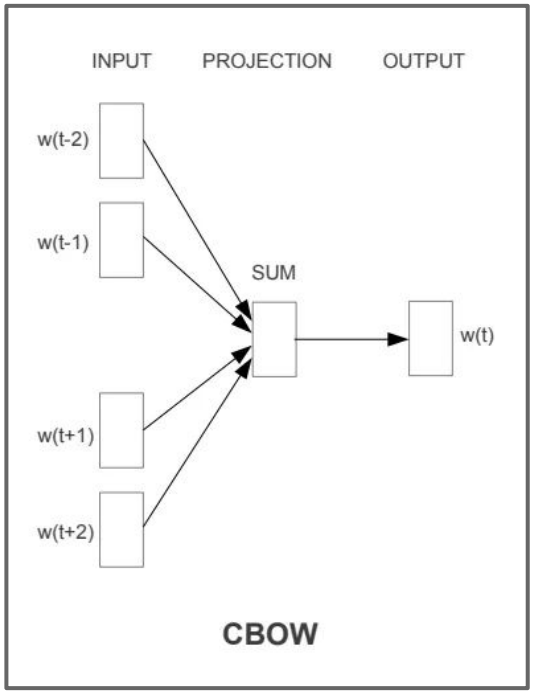


Queen = King - Man + Woman

Shallow neural network that creates **word embeddings** such that **words that are close in meaning are close in the embedding space**.

Image source

Embedding method #3: Word2Vec

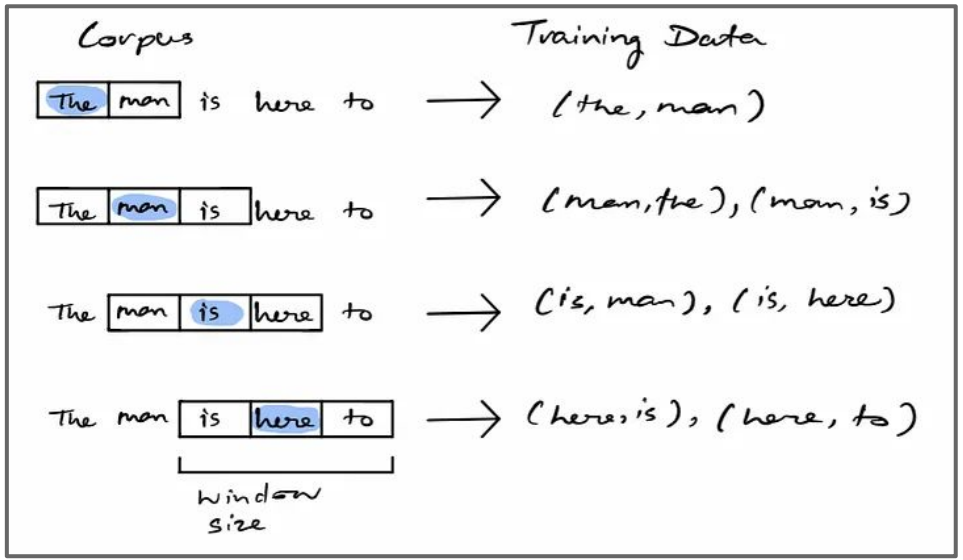


Continuous Bag of Words

You can create the embedding either by predicting the current word…

Image source

Embedding method #3: Word2Vec

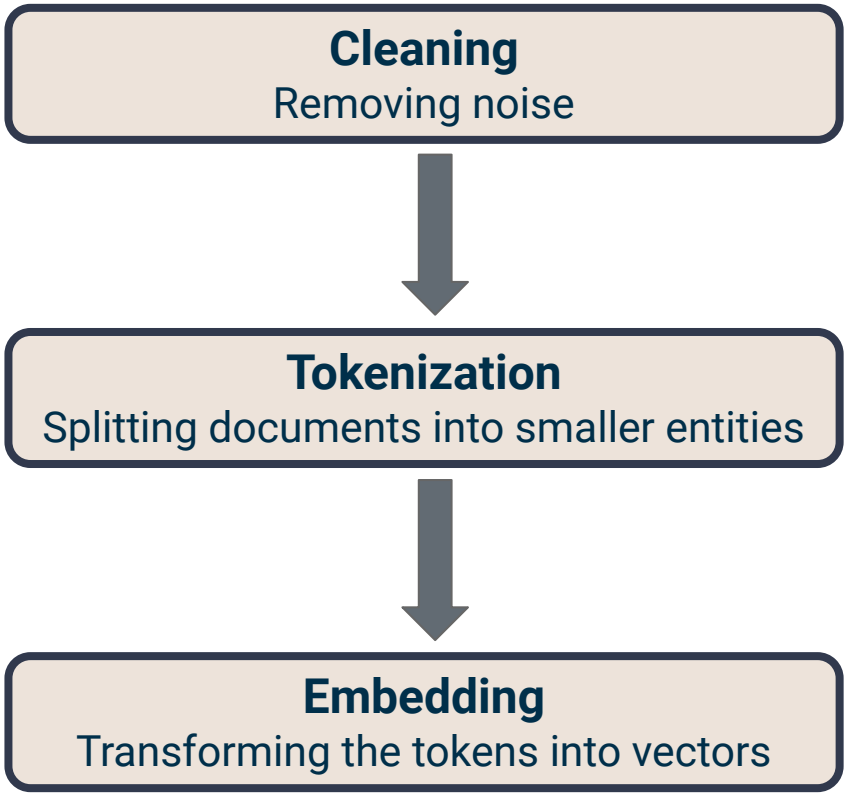


… Or by predicting the words before and after the target word.

Continuous skip-gram

[Image source]

Preparing data for NLP - Summary



Practical work

Get the latest version of the notebook from GitHub

Don’t forget to upload your work!

Debrief

Debrief

What did we learn today?

What could we have done better?

What are we doing next time?

Closing words on this first course

Course outline

Data science course

Session 1: Understanding data

Session 2: Collaborative development

Session 3: Preparing data - Managing missing data

Session 4: Preparing data - Dimensionality reduction

Session 5: Imbalanced data and deidentification

Session 6: Working with text

Machine learning course

What we saw so far

1. Understanding data

- Asking the right questions
- How to visualize data

2. Collaborative development

- How to use Git and GitHub
- The basics of collaborative development

3. Managing missing data

- Sources of missing data
- Removing or imputing missing values

4. Dimensionality reduction

- Feature selection and feature extraction
- Principal Component Analysis

5. Imbalanced data and deidentification

- Undersampling and oversampling
- Deidentifying data

6. Working with text

- Introduction to NLP
- Cleaning, tokenizing and embedding

What we will see next