Tecnológico de Monterrey

NLP 2

Inteligencia artificial avanzada para la ciencia de datos II Modulo 5 NLP 2

Corpus and tokenizers

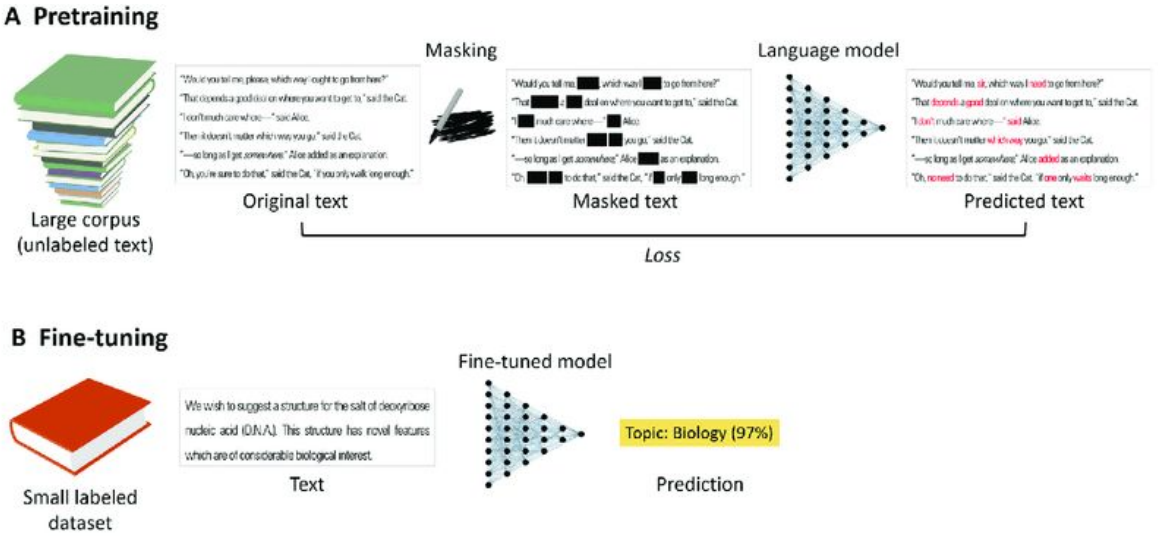
Corpus meaning

A corpus is a large and structured set of machine-readable texts that have been produced in a natural way, corpus might contain text originated also from videos, audio transcripts or some other representations of language.

Corpus units are the basic unit of storage and classification (text, articles, books,etc).

There might be also **metadata** related to each unit (title, author, pages, genres, etc).

Why do we need a corpus?



In order to train generative or discriminative language models we need **empirical evidence as examples**, this text ideally should be created , annotated and suitable for humans communication.

Corpus help us to train **general** pretrained models for context and attention tasks and also for specific **fine tuning** tasks.

Corpora examples

NLTK it's a python library that might help us to manage and use corpus for NLP.

Brown: 500 texts, 1M words in 15 genres. POS-tagged. **SemCor** subset (234K words) labelled with WordNet word senses.

WSJ: 6 years of *Wall Street Journal*; subsequently used to create Penn Treebank, PropBank, and more! Translated into Czech for the **Prague Czech-English Dependency Treebank**.

ECI: European Corpus Initiative, multilingual.

BNC: 100M words; balanced selection of written and spoken genres.

Redwoods: Treebank aligned to wide-coverage grammar; several genres.

Gigaword: 1B words of news text.

AMI: Multimedia (video, audio, synchronised transcripts).

Google Books N-grams: 5M books, 500B words (361B English).

Flickr 8K: images with NL captions

English Visual Genome: Images, bounding boxes ⇒ NL descriptions

Markup languages

Most of the biggest sources of information are available from the **world wide web**.

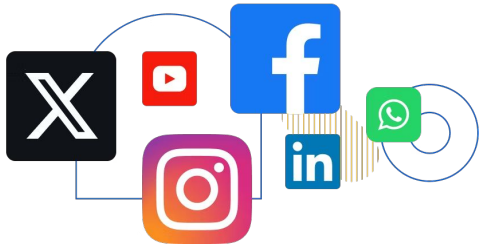
It is convenient that most of the time information is already stored and represented in a markup language.

Why corpus creation is challenging?

Ideally we need a source of language balanced, this means our corpus should be:

1. Representative (includes a variety of texts that represents real world scenario languages.

2. Well sampled (more is better, but usually corpus might contain a selection of text suitable for our purposes).



Unbalance reality

Frequency is not always an ally

Consider a corpus big enough as wikipedia datasource.

What are the most frequent words used in English language?

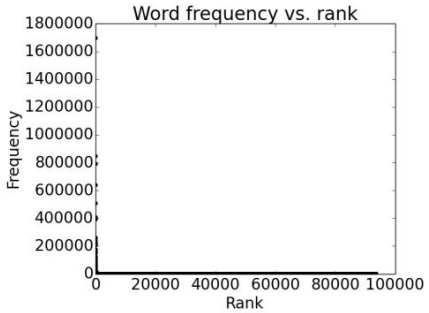
Most frequently words are useless without context (not so frequent words).

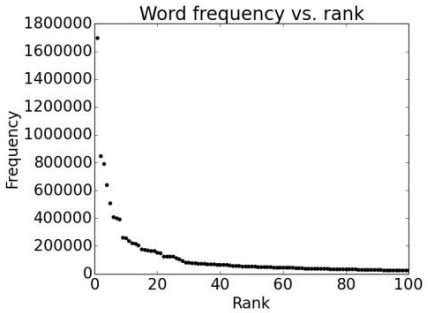
any word

Frequency	Type
1,698,599	the
849,256	of
793,731	to
640,257	and
508,560	in
407,638	that
400,467	is
394,778	a
263,040	I

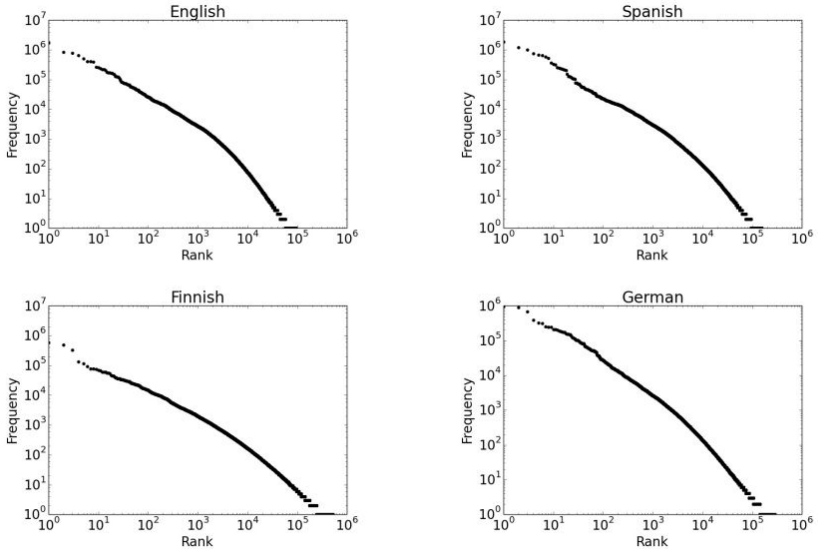
nouns

Frequency	Type
124,598	European
104,325	Mr
92,195	Commission
66,781	President
62,867	Parliament
57,804	Union
53,683	report
53,547	Council
45,842	States





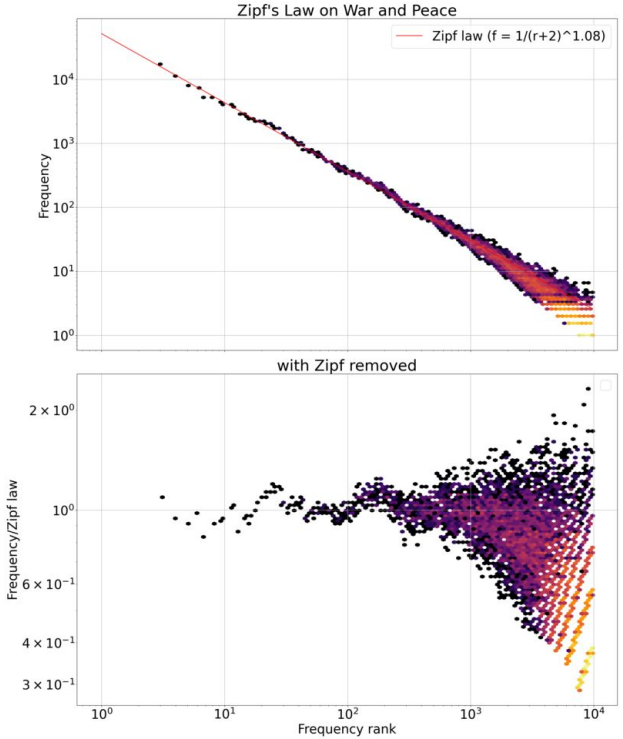
Zipf’s law



$$f \times r \approx k$$

- F = frequency
- R = position in a rank
- K = constant of the word

Zipf’s law normalization



$$f \times r \approx k$$

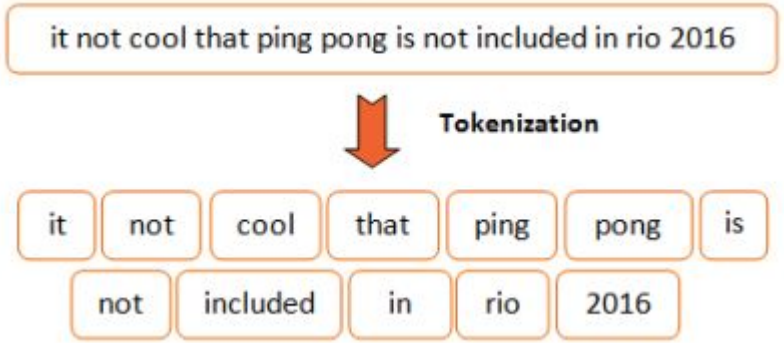
- F = frequency
- R = position in a rank
- K = constant of the word

Tokenizers

The first step to process corpus is the **extraction** and **validation** of words as vocabulary.

This is extremely useful in statistics models, every word in our corpus, is an **event** and observation.

Tokenizers takes an input (stream) and divides it into **words.**

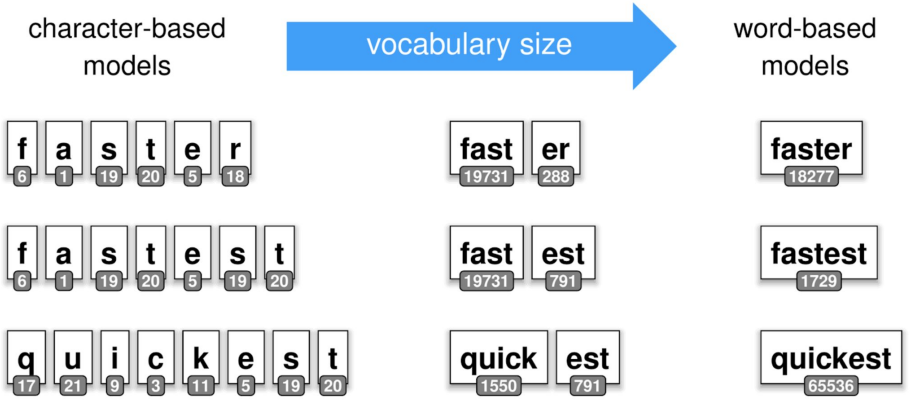


Sub tokenizers

Tokenizers might be more complex to reduce vocabulary (contractions).

This is useful in order to center a main word from derivatives.

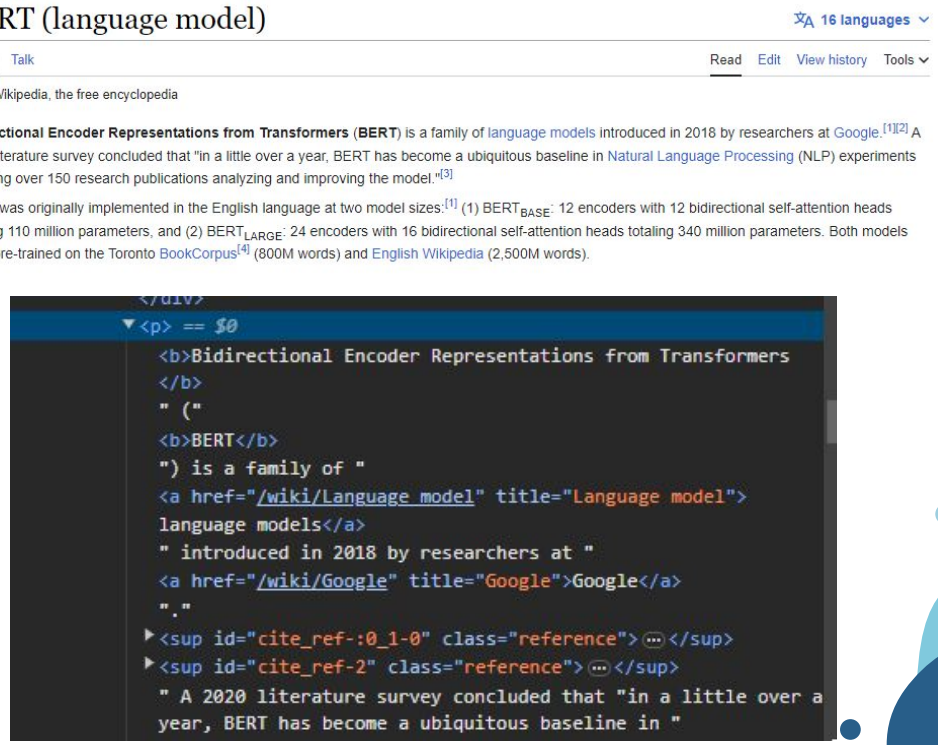
For example the word **cat** is close related to **cats** or **cat's**.



More work for tokenizers

Consider the following text, When we use unclean - real data might look different.

Tokenizers needs to adapt from the origin, format and is just the first step of several transformation steps.



BERT (language model)

16 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

Bidirectional Encoder Representations from Transformers (**BERT**) is a family of language models introduced in 2018 by researchers at Google.[1][2] A 2020 literature survey concluded that "in a little over a year, BERT has become a ubiquitous baseline in Natural Language Processing (NLP) experiments counting over 150 research publications analyzing and improving the model."[3]

BERT was originally implemented in the English language at two model sizes:[1] (1) $BERT_{BASE}$: 12 encoders with 12 bidirectional self-attention heads totaling 110 million parameters, and (2) $BERT_{LARGE}$: 24 encoders with 16 bidirectional self-attention heads totaling 340 million parameters. Both models were pre-trained on the Toronto BookCorpus[4] (800M words) and English Wikipedia (2,500M words).

```
</div>
▼<p> == $0
  <b>Bidirectional Encoder Representations from Transformers
  </b>
  " ("
  <b>BERT</b>
  ") is a family of "
  <a href="/wiki/Language_model" title="Language model">
  language models</a>
  " introduced in 2018 by researchers at "
  <a href="/wiki/Google" title="Google">Google</a>
  "."
 ▶<sup id="cite_ref-:0_1-0" class="reference">…</sup>
 ▶<sup id="cite_ref-2" class="reference">…</sup>
  " A 2020 literature survey concluded that "in a little over a
  year, BERT has become a ubiquitous baseline in "
```

Probability view

Using a word bases tokenizer, we can assume that every word is independent.

This means that the probability of a sentence is given as follows.

x = “I am your father”

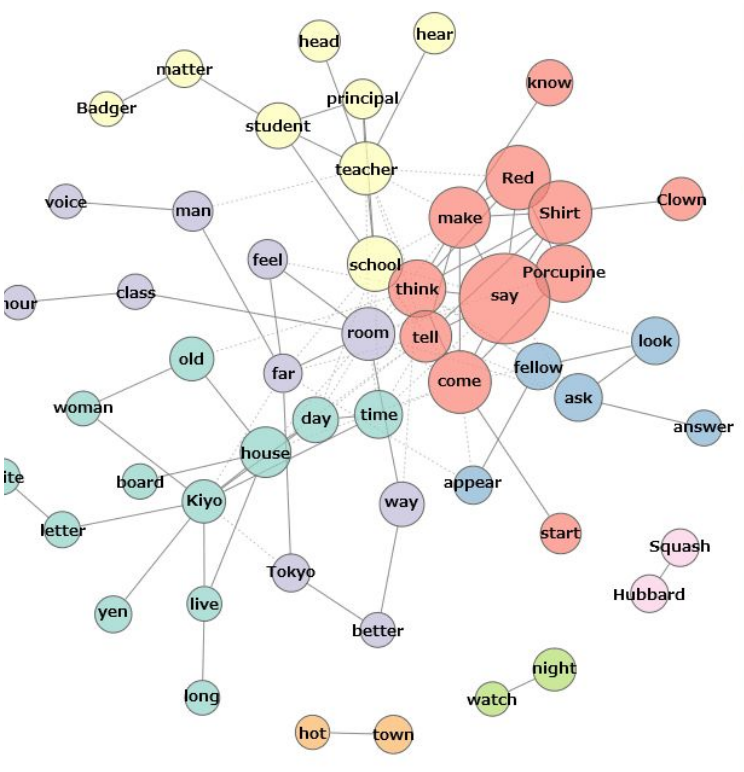
P(x) = p(I)*p(am)*p(your)*p(father)

Concordance and collocation

This analysis helps to understand the words (tokens) that usually surround a word.

Concordance gives us examples of where an specific word is used.

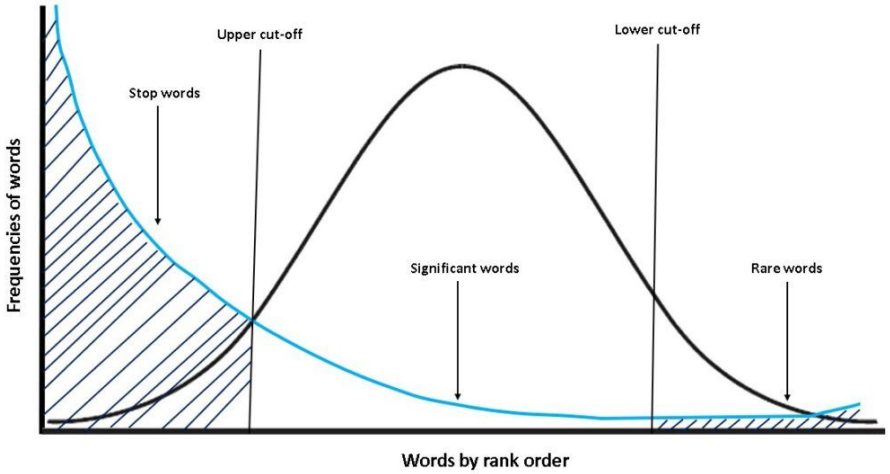
Collocation give us words that are found usually together to a word.



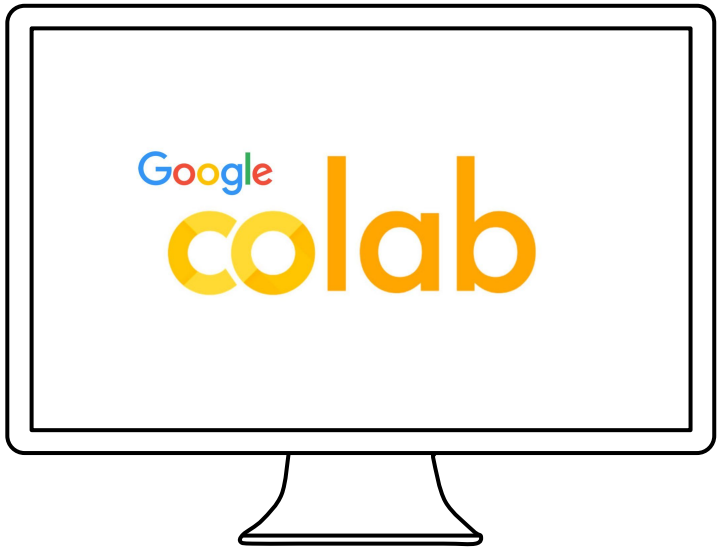
Stop words

As you might understand, there are several known tokens we would like to ignore for statistical analysis.

Removing this tokens is called a “**stop word”** filter.



Coding Time



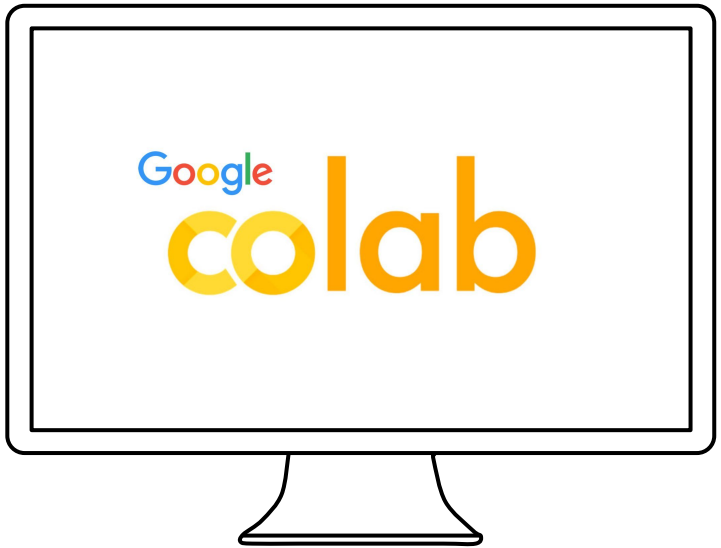
Activity 1

(For the final report)

Create your own corpus from any source (web scraping, files, databases, etc).

Remember must be text generated in the human way.

Clean your corpus and perform analysis of words.



Thanks

Do you have any questions?

emmanuel.paez@tec.mx
Slack #module-5-nlp-1