Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Plan TL
Plan de Impulso de las
Tecnologías del Lenguaje

Text Mining & NLP - quick methodology overview

Antonio Miranda-Escalada
Text Mining Research Engineer
antonio.miranda@bsc.es

Antonio Miranda

antoniomiresc@gmail.com

Spain

- BSc **Biomedical Engineering** - University Carlos III
- MSc Big Data Analytics - University Carlos III
- Bioinformatics Intern - CNIC
- Biometrics research assistant - University Carlos III
- Deloitte Analytics department
- NLP Research - Barcelona Supercomputing Center

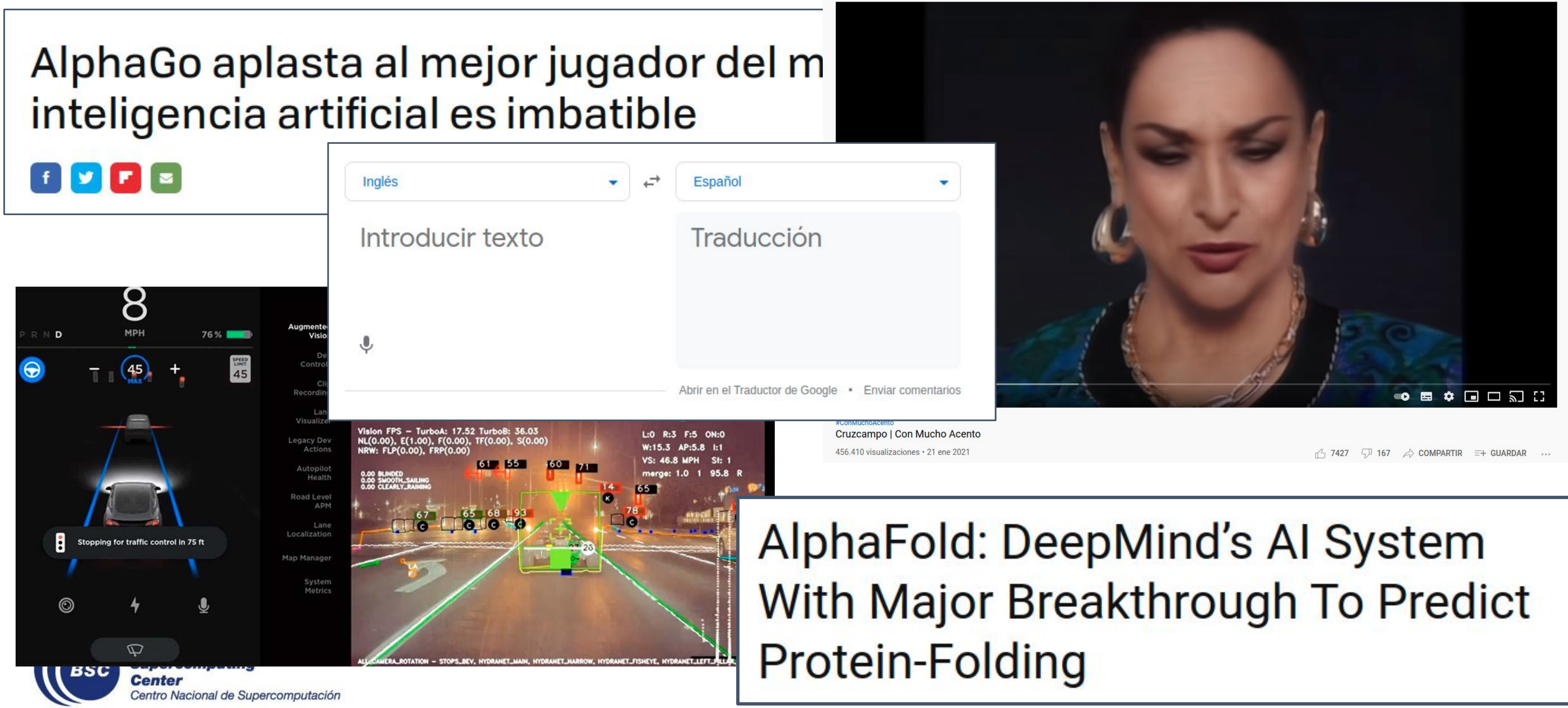
Abroad

- Exchange program at University of California, Irvine
- Short stay at Biomedical Research Foundation Academy Of Athens

Machine Learning

BSC Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Machine Learning breakthroughs



AlphaGo aplasta al mejor jugador del m
inteligencia artificial es imbatible

Inglés ⇄ Español

Introducir texto

Traducción

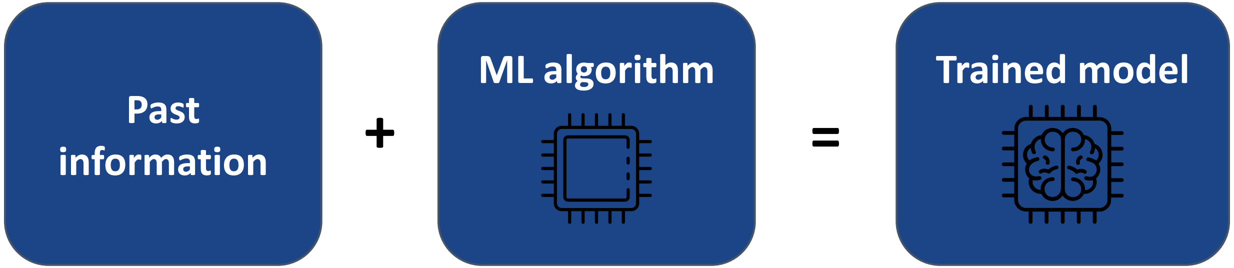
Abrir en el Traductor de Google • Enviar comentarios

#ConMuchoAcento
Cruzcampo | Con Mucho Acento
456.410 visualizaciones • 21 ene 2021

AlphaFold: DeepMind's AI System With Major Breakthrough To Predict Protein-Folding

Introduction to Machine Learning

- Wikipedia definition: “computer algorithms that improve automatically through experience and by the use of data”
- Machine **creates its own rules or behaviors** on how to respond to an information (this is called “training” the machine learning model).
- Machine creates these responses **based on previously provided information**

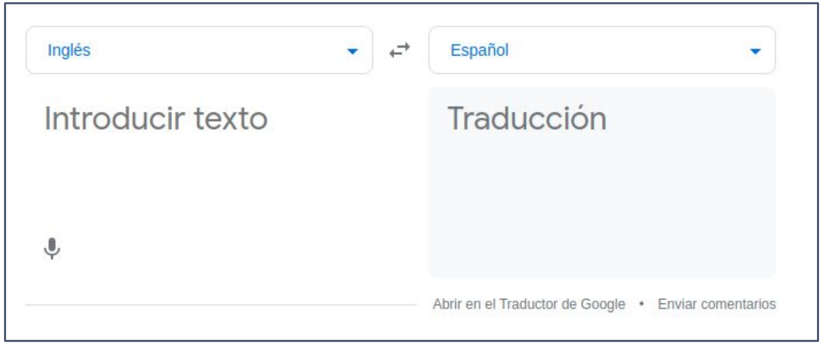


Past information + ML algorithm = Trained model

Introduction to Machine Learning

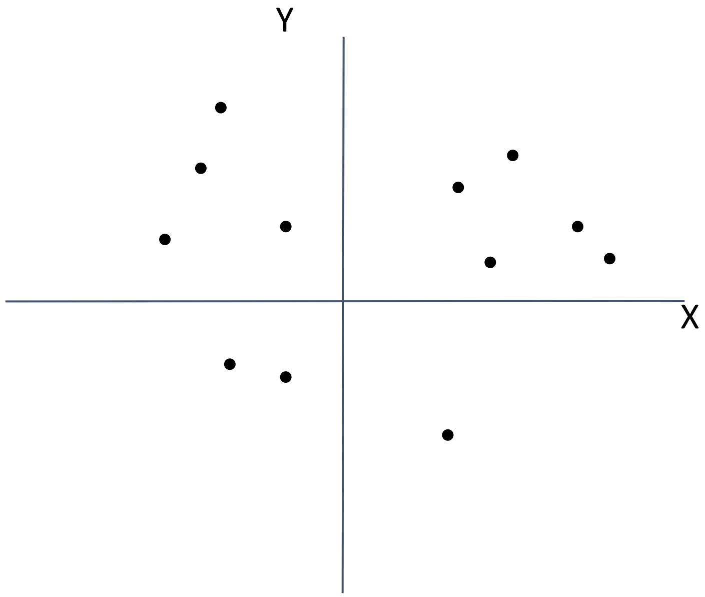
- Wikipedia definition: “computer algorithms that improve automatically through experience and by the use of data”
- Machine **creates its own rules or behaviors** on how to respond to an information (this is called “training” the machine learning model).
- Machine creates these responses **based on previously provided information**

Sentences in English and in Spanish + ML algorithm =



Introduction to Machine Learning

Example: I have some points and I know that, usually, points belong to two groups (**Black** and **Red)**. I want to design a system that classifies new points into **Black** and **Red** groups.



Introduction to Machine Learning

Example: I want to classify new points into **Black** and **Red** groups.

Traditional approach - expert knowledge:

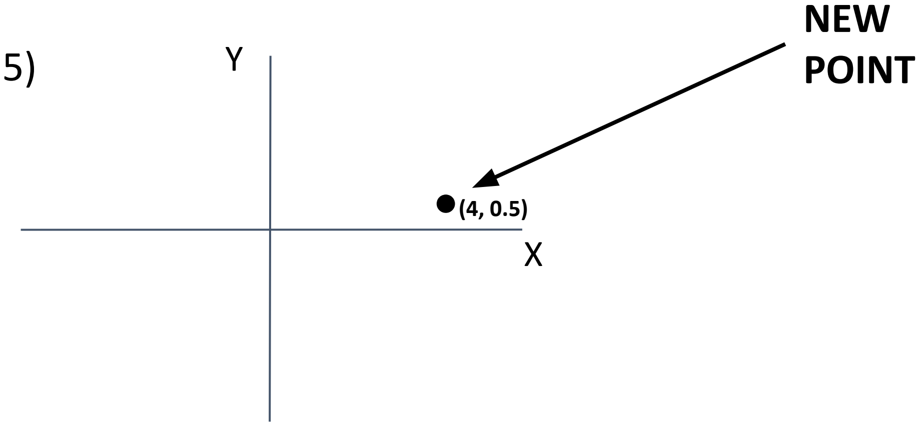
1. Since I have years of experience working with Black and Red groups, I know that points with X>3 usually belong to the Red group. Then, I write the following rule "If X>3, the new point belongs to the Red group" → ***handcrafted rules based on experience***

Introduction to Machine Learning

Example: I want to classify new points into **Black** and Red groups.

Traditional approach:

1. Since I have years of experience working with Black and Red groups, I know that points with X>3 usually belong to the Red group. Then, I write the following rule "If X>3, the new point belongs to the Red group" → *handcrafted rules based on experience*
2. When I have a new point, just **follow my handcrafted rule** to place it in the Black or Red group
   a. Example: new point (4, 0.5)

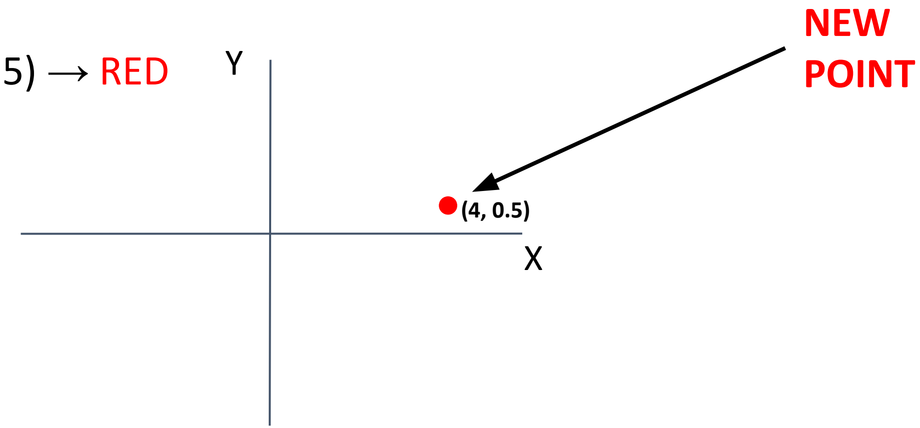


Introduction to Machine Learning

Example: I want to classify new points into **Black** and Red groups.

Traditional approach:

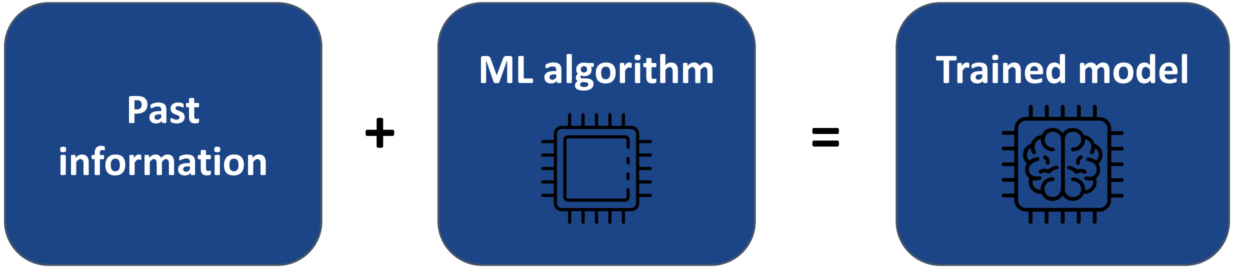
1. Since I have years of experience working with Black and Red groups, I know that points with X>3 usually belong to the Red group. Then, I write the following rule "If X>3, the new point belongs to the Red group" → *handcrafted rules based on experience*
2. When I have a new point, just follow my handcrafted rule to place it in the Black or Red group
   a. Example: new point (4, 0.5) → RED



Introduction to Machine Learning

Example: I want to classify new points into **Black** and Red groups.

Machine learning approach - algorithm learns from past data:



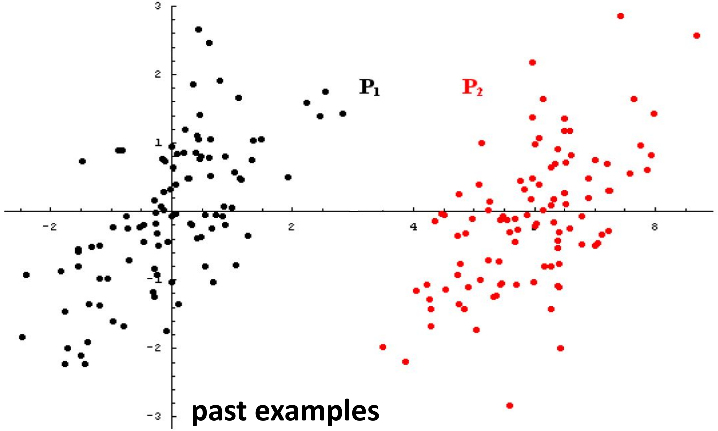
Past information + ML algorithm = Trained model

Introduction to Machine Learning

Example: I want to classify new points into **Black** and Red groups.

Machine learning approach - algorithm learns from past data:

1. Collect **past examples** of Black and Red points & feed them to the machine (this is called “**training**” the algorithm)



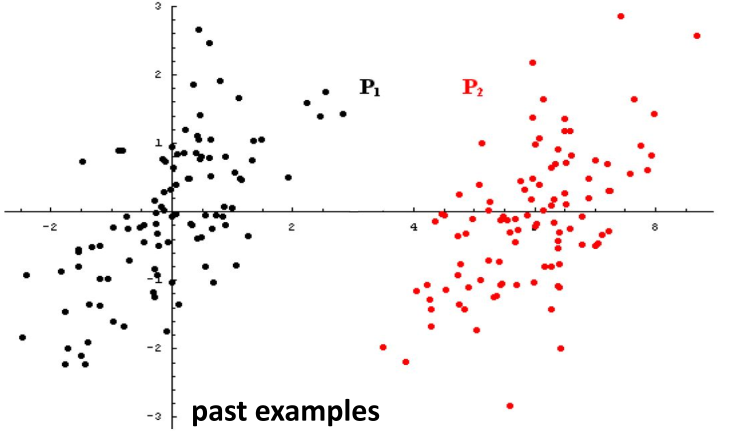
past examples

Introduction to Machine Learning

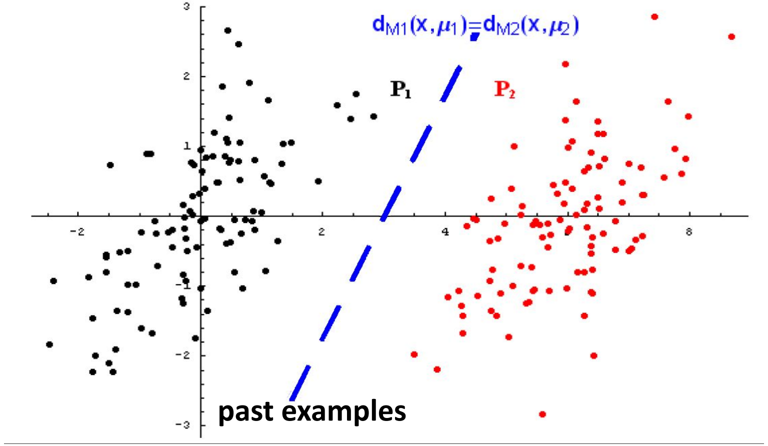
Example: I want to classify new points into **Black** and Red groups.

Machine learning approach:

1. Collect **past examples** of Black and Red points & feed them to the machine.
2. **It creates its own rule** → *self-created rule based on previous data* → that is the difference!



past examples



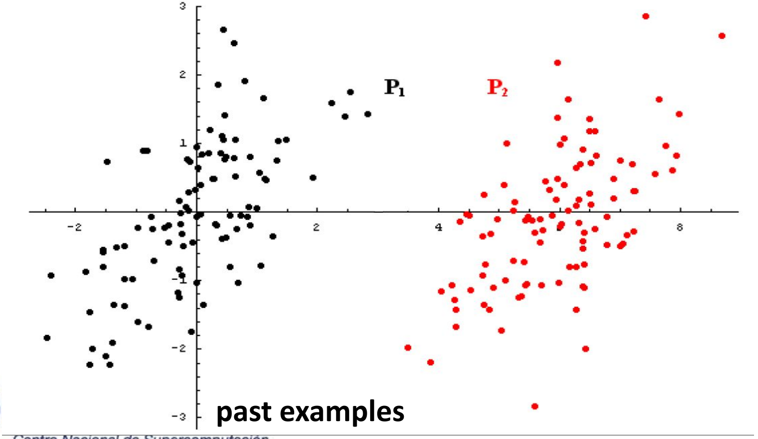
past examples

Introduction to Machine Learning

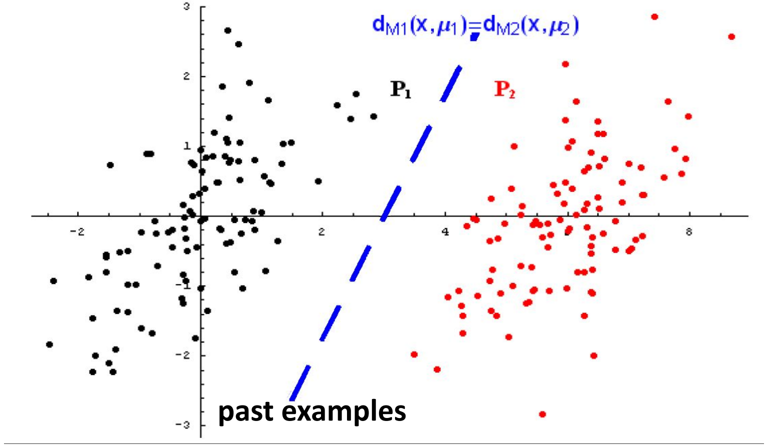
Example: I want to classify new points into **Black** and Red groups.

Machine learning approach:

1. Collect **past examples** of Black and Red points & feed them to the machine.
2. It creates its own rule
3. When I have a **new point**, just follow the rule *created by the machine* to place it in the Black or Red group



past examples



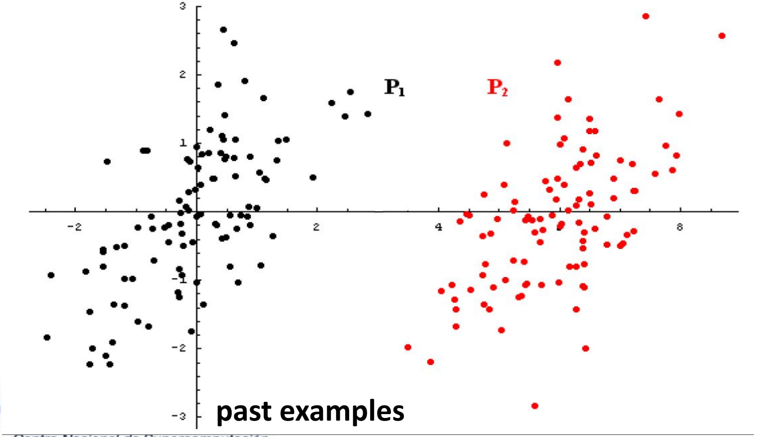
past examples

Introduction to Machine Learning

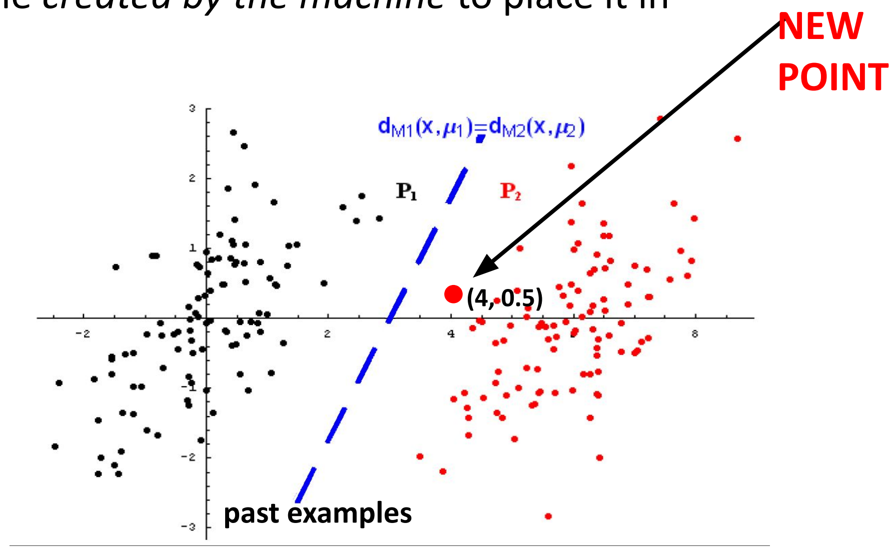
Example: I want to classify new points into **Black** and Red groups.

Machine learning approach:

1. Collect **past examples** of Black and Red points & feed them to the machine.
2. It creates its own rule
3. When I have a **new point**, just follow the rule *created by the machine* to place it in the Black or Red group



past examples



past examples

Machine Learning applications in biomedical (and non-biomedical) NLP

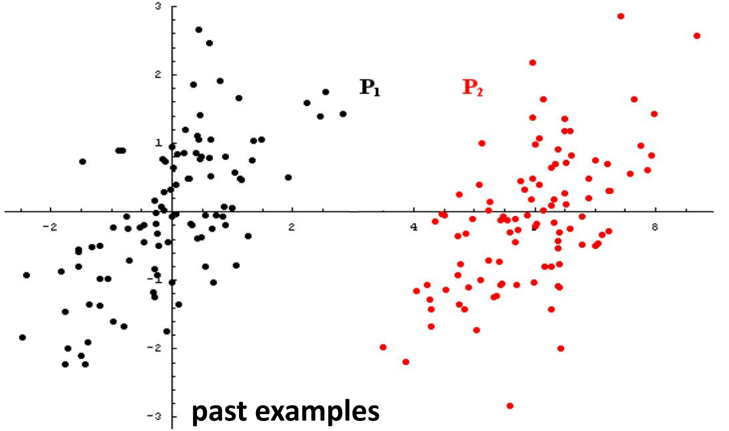
BSC
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Recap from Intro

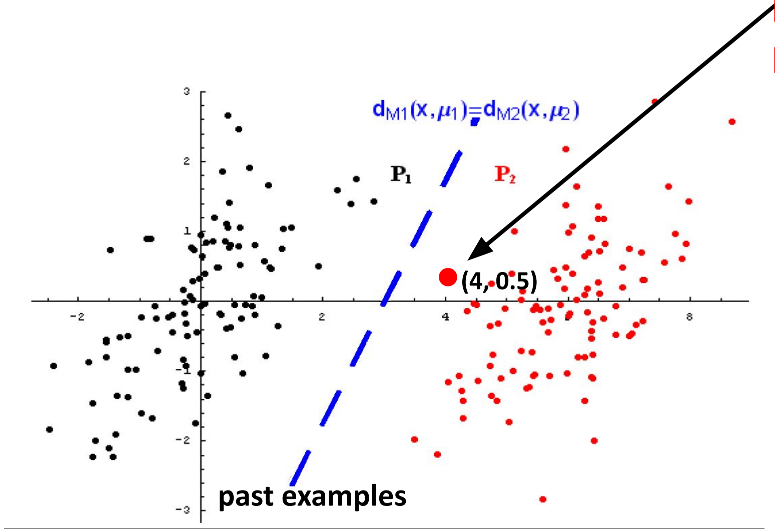
- Problem: I do not know the color or a point. I want to predict the color or a point (**Black** or **Red**)

- We gave to the algorithm: past **Black** and **Red** points

- The algorithm created rules to: decide which coordinates of a new point determine its colour



past examples

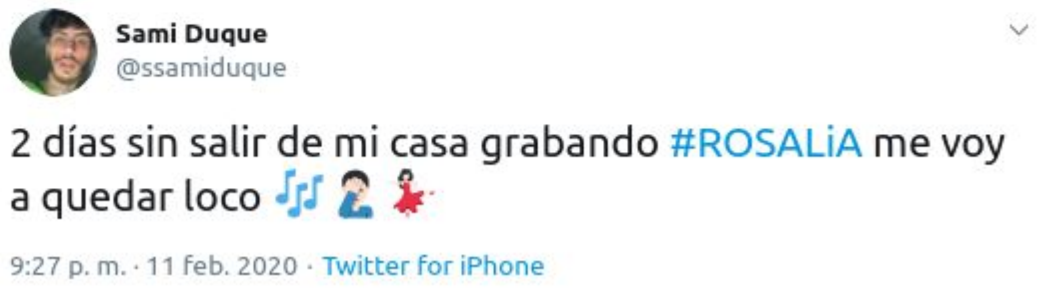


past examples

NEW POINT

DeepMoji

- **Problem**: I want to know which emoji I should add at the end of a tweet.
- **We gave to the algorithm:** past tweets with the emojis that a real user has written
- **The algorithm created rules to:** decide which emoji fits in the end of a new tweet

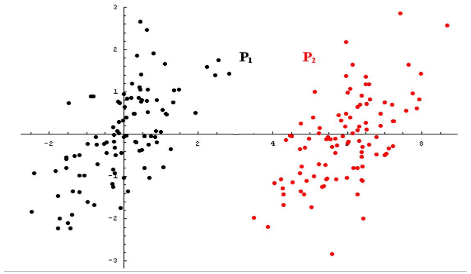


Sami Duque
@ssamiduque

2 días sin salir de mi casa grabando #ROSALiA me voy a quedar loco 🎶🤦‍♂️💃

9:27 p. m. · 11 feb. 2020 · Twitter for iPhone

past examples

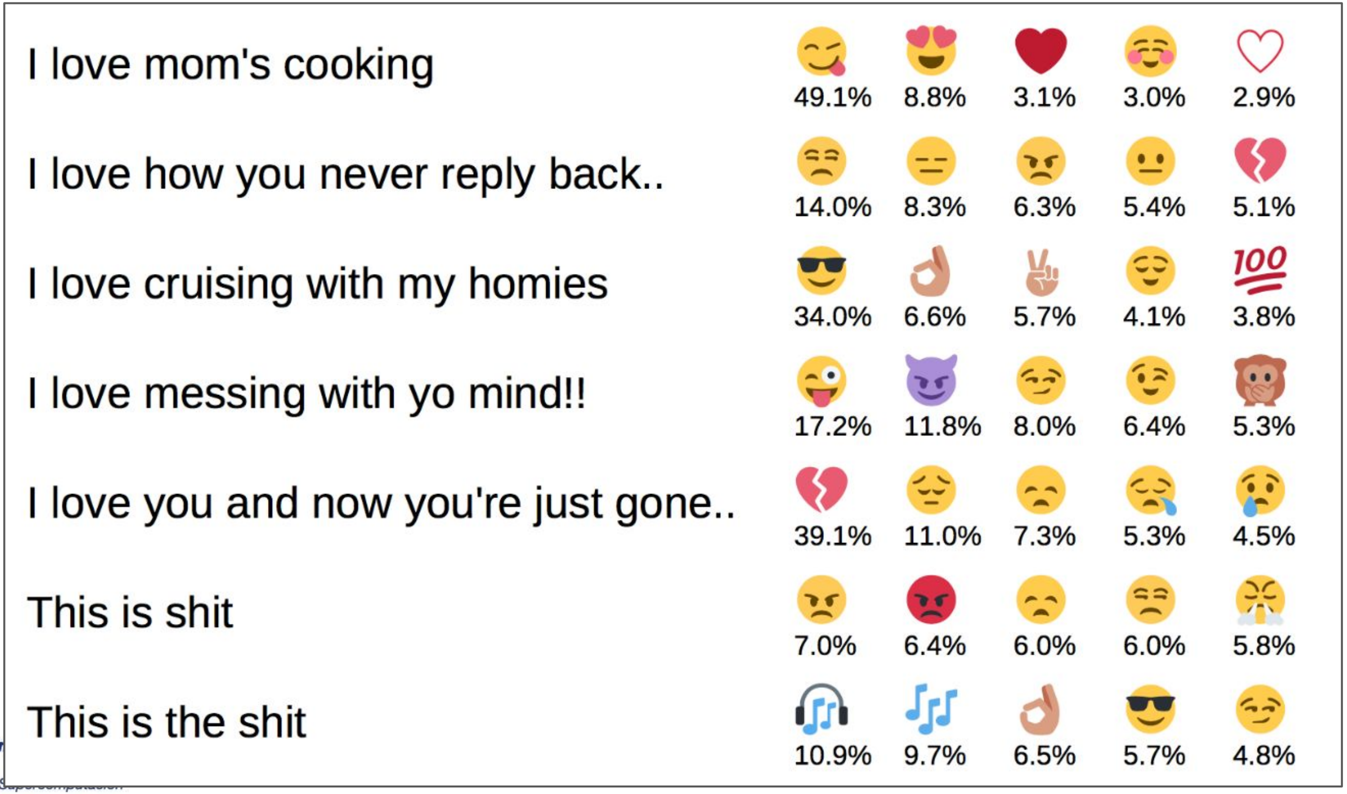


past examples

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

https://deepmoji.mit.edu/

DeepMoji



New tweets	Predicted emoji				
I love mom's cooking	😋 49.1%	😍 8.8%	❤️ 3.1%	☺️ 3.0%	♡ 2.9%
I love how you never reply back..	😒 14.0%	😑 8.3%	😠 6.3%	😐 5.4%	💔 5.1%
I love cruising with my homies	😎 34.0%	👌 6.6%	✌️ 5.7%	😌 4.1%	💯 3.8%
I love messing with yo mind!!	😜 17.2%	😈 11.8%	😏 8.0%	😉 6.4%	🙊 5.3%
I love you and now you're just gone..	💔 39.1%	😔 11.0%	😞 7.3%	😪 5.3%	😢 4.5%
This is shit	😠 7.0%	😡 6.4%	😞 6.0%	😒 6.0%	😤 5.8%
This is the shit	🎧 10.9%	🎶 9.7%	👌 6.5%	😎 5.7%	😏 4.8%

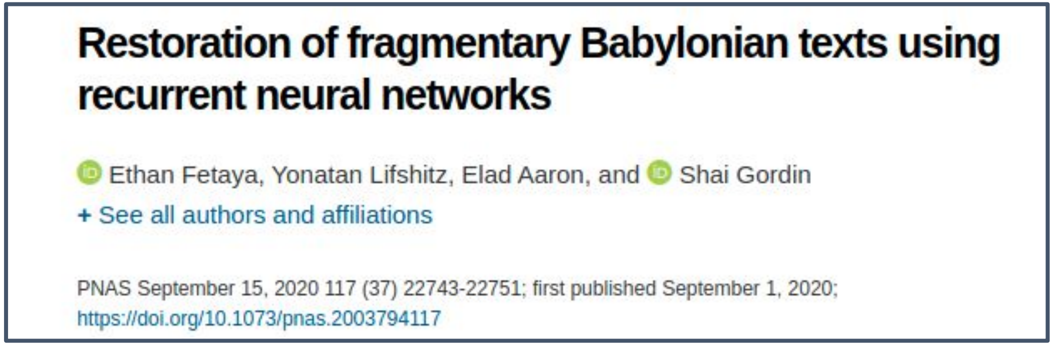
Restoration of fragmentary Babylonian texts

- **Problem**: In old cuneiform clay tablets there are gaps (because they are old). When there is a gap in the middle of a sentence, I want to predict which word fits there.

- **We gave to the algorithm:** past complete sentences

- **The algorithm created rules to:** decide which word fits in the gap of a new sentence

ción



Restoration of fragmentary Babylonian texts using recurrent neural networks

Ethan Fetaya, Yonatan Lifshitz, Elad Aaron, and Shai Gordin

+ See all authors and affiliations

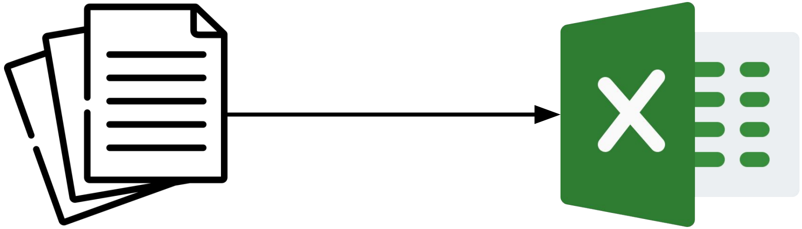
PNAS September 15, 2020 117 (37) 22743-22751; first published September 1, 2020; https://doi.org/10.1073/pnas.2003794117

https://www.pnas.org/content/117/37/22743

Our job at BSC: Finding symptoms in COVID-19 reports

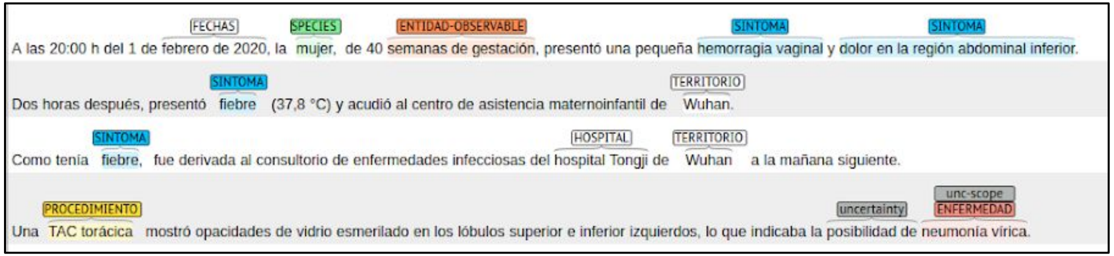
- **Problem**: I have 50K medical reports from COVID-19 patients from a major hospital in Barcelona. I want to get the symptoms and previous diseases of a patient. As well as the medications he/she received.

- **Motivation:** I want to learn which medications work better for patients with different symptoms and previous diseases. But I cannot read 50K medical reports. If I extract the symptoms, previous diseases and medications together with the outcome of the patient and put them in a table, I could extract correlations.



Our job at BSC: Finding symptoms in COVID-19 reports

- **We gave to the algorithm:** past sentences with the symptoms, diseases and medications outlined



A las 20:00 h del 1 de febrero de 2020, la mujer, de 40 semanas de gestación, presentó una pequeña hemorragia vaginal y dolor en la región abdominal inferior.

Dos horas después, presentó fiebre (37,8 °C) y acudió al centro de asistencia maternoinfantil de Wuhan.

Como tenía fiebre, fue derivada al consultorio de enfermedades infecciosas del hospital Tongji de Wuhan a la mañana siguiente.

Una TAC torácica mostró opacidades de vidrio esmerilado en los lóbulos superior e inferior izquierdos, lo que indicaba la posibilidad de neumonía vírica.

- **The algorithm created rules to:** decide which word is a symptom, which one a disease, etc.

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Basic NLP concepts

BSC
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Basic NLP concepts

- **Gold Standard:** A thing of superior quality which serves as a point of reference. It is used by the Machine Learning algorithm to create the rules. For example, in the points example, it would be the collection of past examples of Black and Red points.

Basic NLP concepts

- **Gold Standard:** A thing of superior quality which serves as a point of reference. It is used by the Machine Learning algorithm to create the rules. For example, in the points example, it would be the collection of past examples of Black and Red points.
- **Corpus:** Collection of text documents

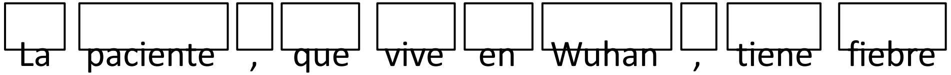
Basic NLP concepts

- **Gold Standard:** A thing of superior quality which serves as a point of reference. It is used by the Machine Learning algorithm to create the rules. For example, in the points example, it would be the collection of past examples of Black and Red points.
- **Corpus:** Collection of text documents
- **Token:** sequence of characters in some particular document that are grouped together as a useful semantic unit for processing

1 Sentence:

La paciente, que vive en Wuhan, tiene fiebre

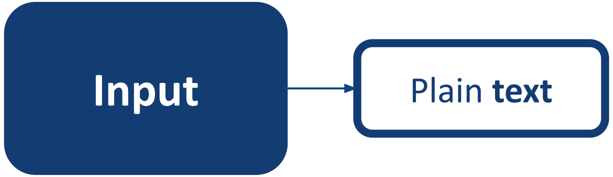
10 Tokens:

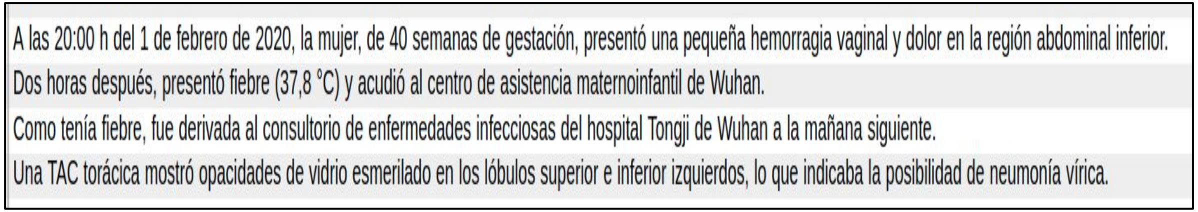


La | paciente | , | que | vive | en | Wuhan | , | tiene | fiebre

Basic NLP concepts

- **Named Entity Recognition (NER)**: detection and classification of relevant parts of text (entities)





A las 20:00 h del 1 de febrero de 2020, la mujer, de 40 semanas de gestación, presentó una pequeña hemorragia vaginal y dolor en la región abdominal inferior.

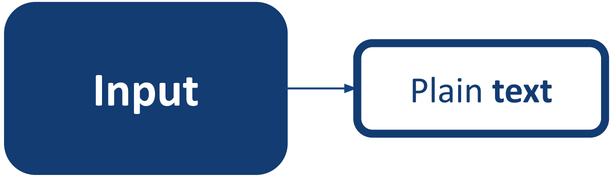
Dos horas después, presentó fiebre (37,8 °C) y acudió al centro de asistencia maternoinfantil de Wuhan.

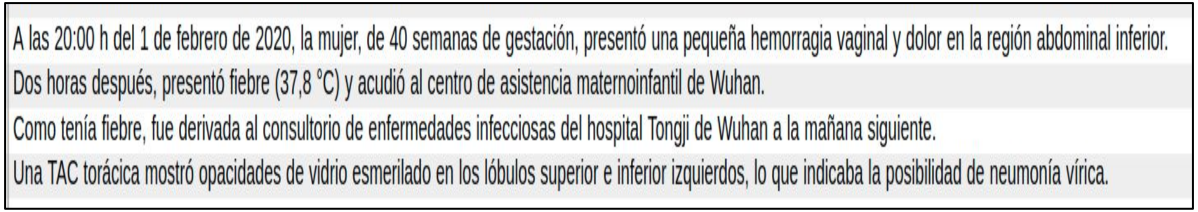
Como tenía fiebre, fue derivada al consultorio de enfermedades infecciosas del hospital Tongji de Wuhan a la mañana siguiente.

Una TAC torácica mostró opacidades de vidrio esmerilado en los lóbulos superior e inferior izquierdos, lo que indicaba la posibilidad de neumonía vírica.

Basic NLP concepts

- **Named Entity Recognition (NER)**: detection and classification of relevant parts of text (entities)



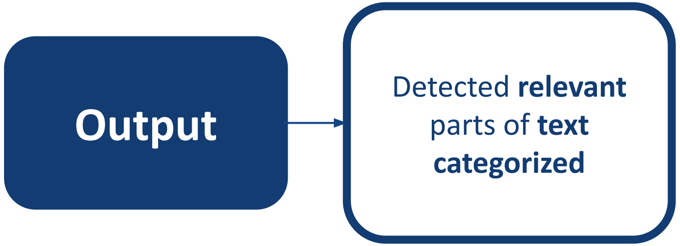


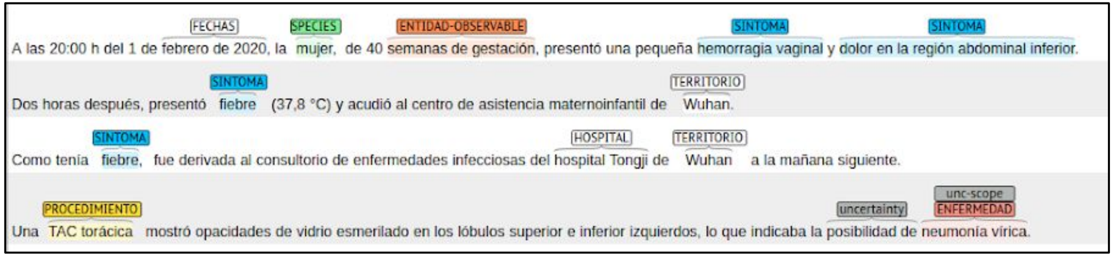
A las 20:00 h del 1 de febrero de 2020, la mujer, de 40 semanas de gestación, presentó una pequeña hemorragia vaginal y dolor en la región abdominal inferior.

Dos horas después, presentó fiebre (37,8 °C) y acudió al centro de asistencia maternoinfantil de Wuhan.

Como tenía fiebre, fue derivada al consultorio de enfermedades infecciosas del hospital Tongji de Wuhan a la mañana siguiente.

Una TAC torácica mostró opacidades de vidrio esmerilado en los lóbulos superior e inferior izquierdos, lo que indicaba la posibilidad de neumonía vírica.





A las 20:00 h del 1 de febrero de 2020, la mujer, de 40 semanas de gestación, presentó una pequeña hemorragia vaginal y dolor en la región abdominal inferior.

Dos horas después, presentó fiebre (37,8 °C) y acudió al centro de asistencia maternoinfantil de Wuhan.

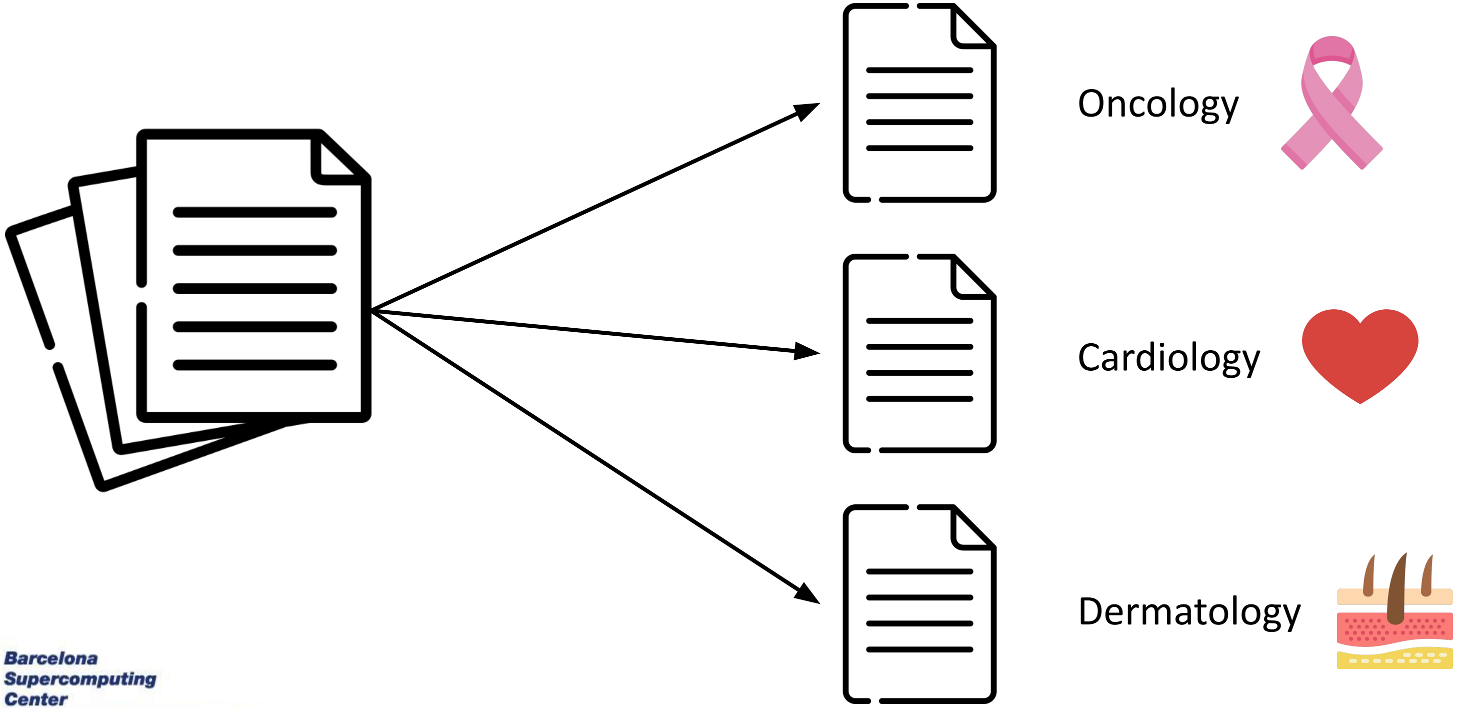
Como tenía fiebre, fue derivada al consultorio de enfermedades infecciosas del hospital Tongji de Wuhan a la mañana siguiente.

Una TAC torácica mostró opacidades de vidrio esmerilado en los lóbulos superior e inferior izquierdos, lo que indicaba la posibilidad de neumonía vírica.

COVID-19 clinical case report with entities detected by NER system

Basic NLP concepts

- **Text Classification:** categorizing text into organized groups (for example, into medical areas)



Example: Text Classification with Named Entity Recognition

Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Expectations

This is a quick introduction:

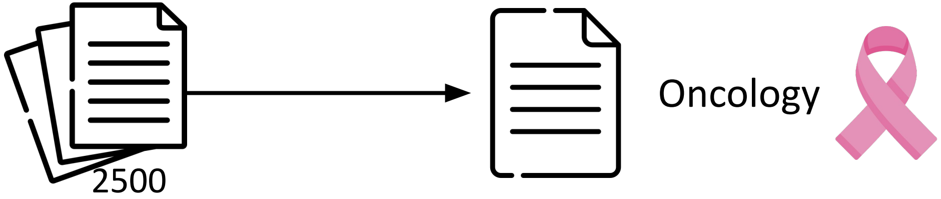
- I expect you to understand the example
- I expect you to understand the motivation behind every step we do to solve the use case
- I **do not** expect you to understand every line of code. In NLP we mostly use Python because it is the language that allows us to do all of the NLP steps.

Problem

I have **2500 documents** about the condition and evolution of patients suffering from diverse problems: COVID-19, oncology problems, cardiology problems, urology and many other different medical areas.

Goal: I am an oncology doctor and I want to study just the ones about oncology.

Obviously, I do not want to read the 2500 documents before starting to study them...



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Problem

I have **2500 documents** about the condition and evolution of patients suffering from diverse problems: COVID-19, oncology problems, cardiology problems, urology and many other different medical areas.

Goal: I am an oncology doctor and I want to study just the ones about oncology.

Obviously, I do not want to read the 2500 documents before starting to study them...

I know:

- There are **some** documents that talk about **oncology**.
- All documents are from **Spanish-speaking countries**

I do not know:

- I do not know **how many** of them are **about oncology.**
- I do not know what are the **other medical areas** that I need to discard.
- I do not know which **types of cancer** my clinical reports talk about

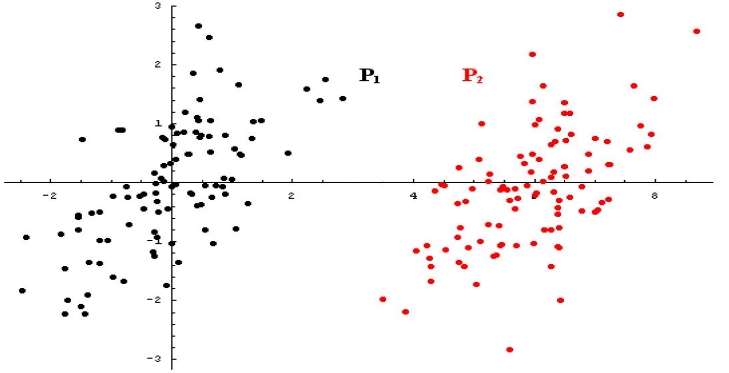
Approach 1: Text Classification

Approach 1: Text classification

Approach 1: Text Classification

Approach 1: Text classification

Similar to the Black and Red points.



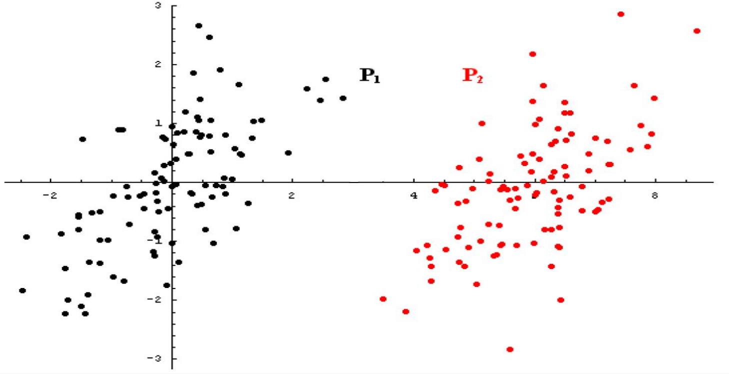
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Approach 1: Text Classification

Approach 1: Text classification

Similar to the Black and Red points.

If I had past examples of oncology, cardiology, covid, etc documents, I could input them to a classification algorithm and it would create its own classification rules. Then, I could classify my new 2500 documents.

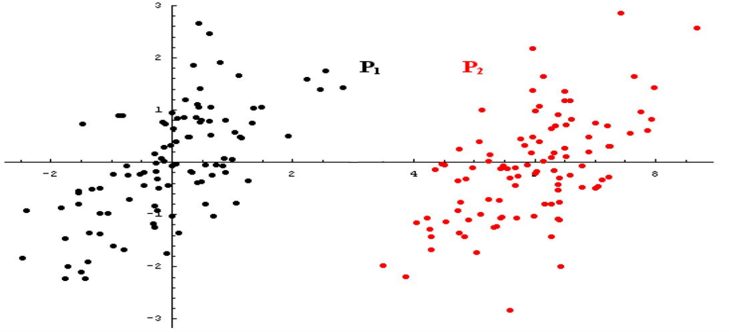


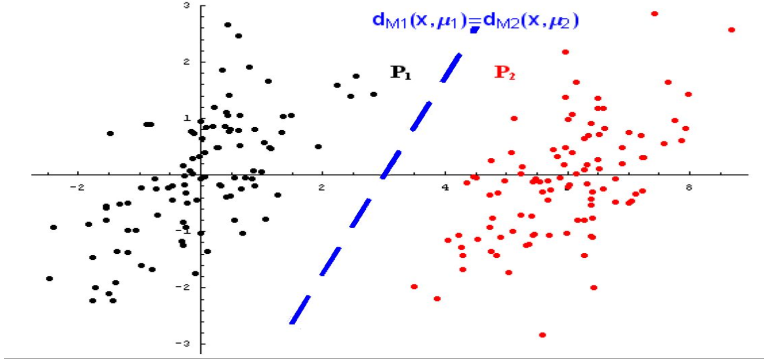
Approach 1: Text Classification

Approach 1: Text classification

Similar to the Black and Red points.

If I had past examples of oncology, cardiology, covid, etc documents, I could input them to a classification algorithm and it would create its own classification rules. Then, **I could classify my new 2500 documents**.





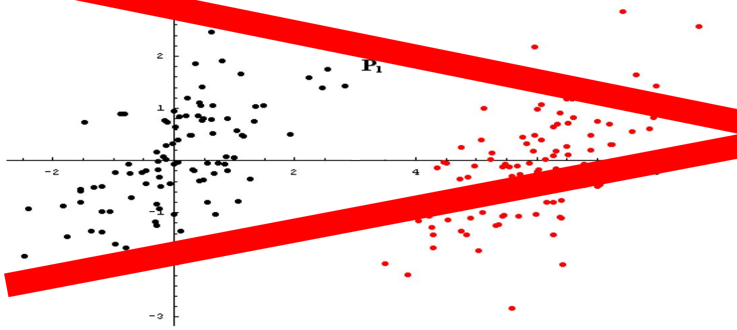
Approach 1: Text Classification

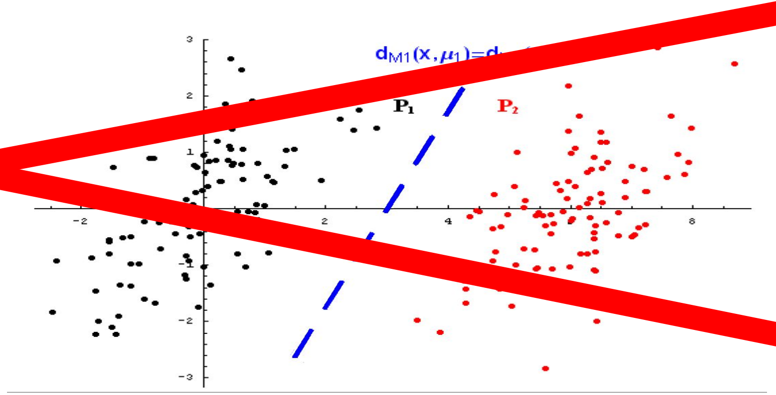
Approach 1: Text classification

Similar to the Black and Red points.

If I had past examples of oncology, cardiology, covid, etc documents, I could input them to a classification algorithm and it would create its own classification rules. Then, I could classify my new 2500 documents.

But this time, I do not know what are the other medical specialties. And **I do not have these past examples.**

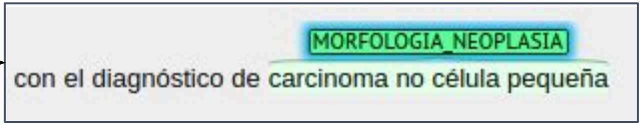




Approach 2: Named Entity Recognition

Approach 2: Named Entity Recognition

I will **locate** the clinical case reports that **mention entities related to oncology** (such as “metástasis” or “carcinoma lobulillar infiltrante”).



con el diagnóstico de carcinoma no célula pequeña

I will assume that if a **document mentions an oncology entity**, I will be **interested** in this document.

Approach 2: Named Entity Recognition

Traditional Named Entity Recognition:

1. Get a list of all oncology-related terms
2. Search for those terms in my documents

Approach 2: Named Entity Recognition

Traditional Named Entity Recognition:

1. Get a list of all oncology-related terms
2. Search for those terms in my documents

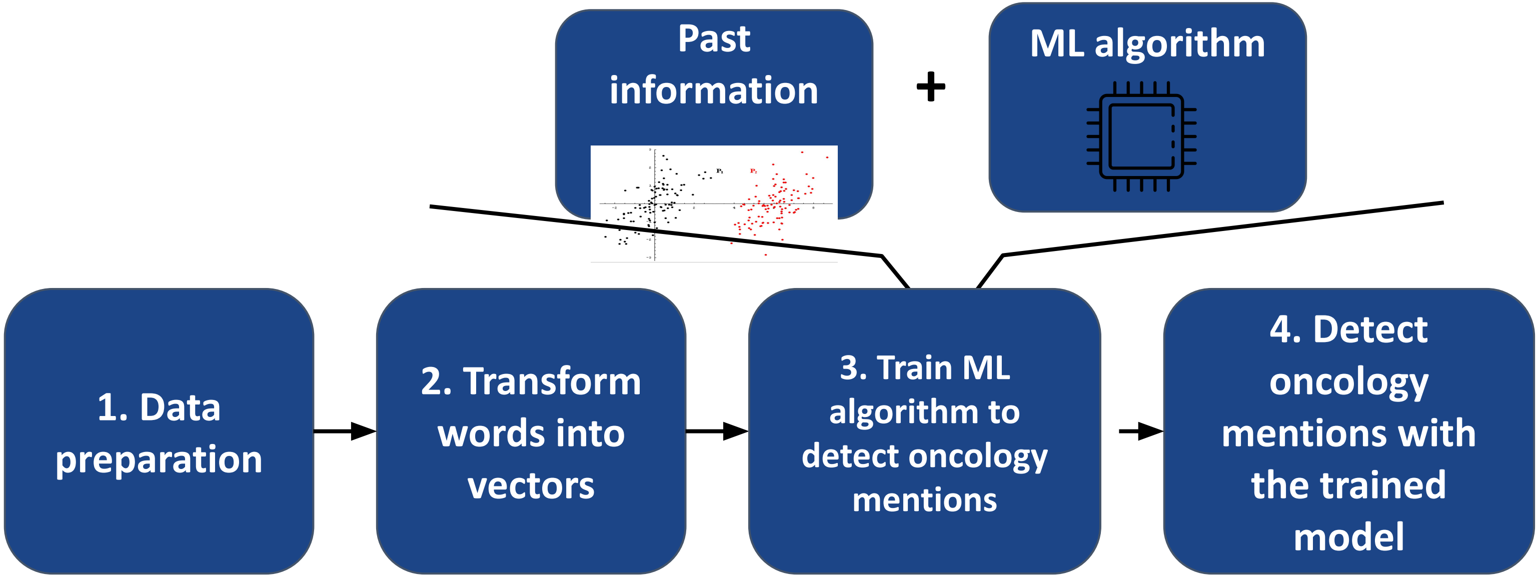
Problems:

- Do I have such **100% complete list**? Probably not…
- **Real-world vocabulary** is different from textbook vocabulary. Medical doctors (and nobody) do not write as in textbooks.
- Documents are written in Spanish-speaking countries.
  - **Other languages**: what happens if some documents are written in Catalan or Galician?
  - **Dialects**: what happens if there are Spanish variants between Spain and México?
- How do I account for typographical **errors**?
  - Instead of “carcinoma lobu**lill**ar infiltrante”, I could have “carcinoma lobu**llili**ar infiltrante”

Approach 2: Named Entity Recognition

Today's Named Entity Recognition:

Use Machine Learning (**Deep Learning**) to find the oncology mentions



Past information

+

ML algorithm

1. Data preparation

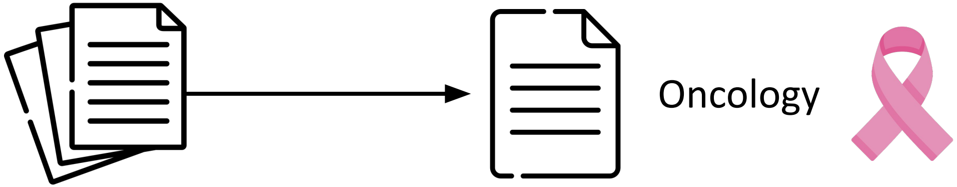
2. Transform words into vectors

3. Train ML algorithm to detect oncology mentions

4. Detect oncology mentions with the trained model

Recap

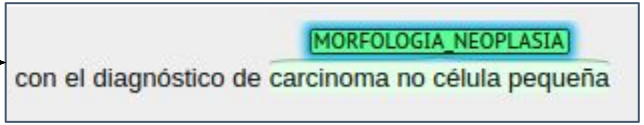
Goal: Find documents about oncology in a collection of unknown documents



Oncology

Approach:

1. Find oncology-related entities in the documents → use Machine Learning to find oncology-related documents



MORFOLOGIA_NEOPLASIA

con el diagnóstico de carcinoma no célula pequeña

2. Every document with 1 oncology-related entity is a relevant document for me.

Step 1: Data preparation

Motivation: real-world text is usually dirty, with bad encodings, text is in PDF and is difficult to parse, etc. To use it in any algorithm, we need to prepare it

Common steps:

- encoding fixer
- cleaning: punctuation and accents, special characters, numeric digits, leading, ending and vertical whitespace, HTML formatting, lowercasing
- tokenization: split sentences into individual tokens

Step 2: Transform words into vectors

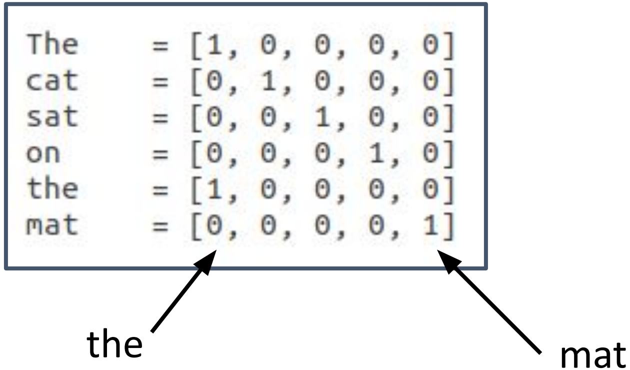
Motivation: Machine learning algorithms cannot work with raw text directly; the text must be converted into numbers: **words must be transformed into vectors**

Different ways:

- One-Hot encoding
- Bag-of-Words (BoW)
- Word embeddings
- Context embeddings
- Language models

Step 2: Transform words into vectors - One-Hot encoding

1. **Description:** each word is represented as a binary vector that is all zero values except the index of the word, which is marked with a 1
2. **Example:**
   "The cat sat on the mat"



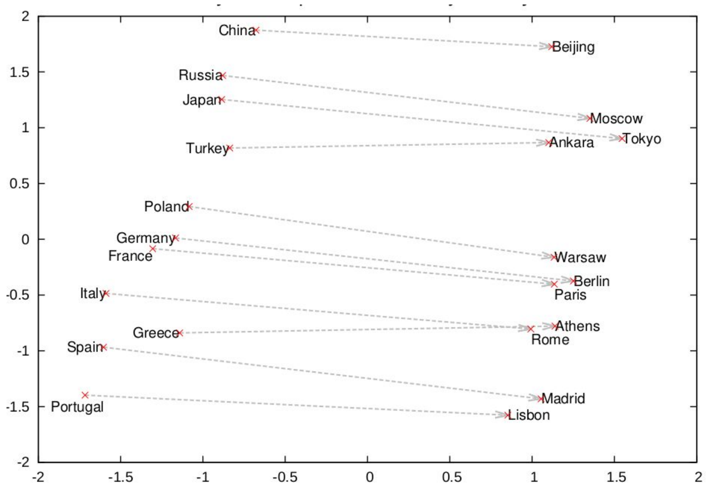
```
The     = [1, 0, 0, 0, 0]
cat     = [0, 1, 0, 0, 0]
sat     = [0, 0, 1, 0, 0]
on      = [0, 0, 0, 1, 0]
the     = [1, 0, 0, 0, 0]
mat     = [0, 0, 0, 0, 1]
```

3. **Not very useful:** all word vectors are equally separated, if we have 300K words, our vectors have 300K dimensions, etc.

Step 2: Transform words into vectors - Word embeddings

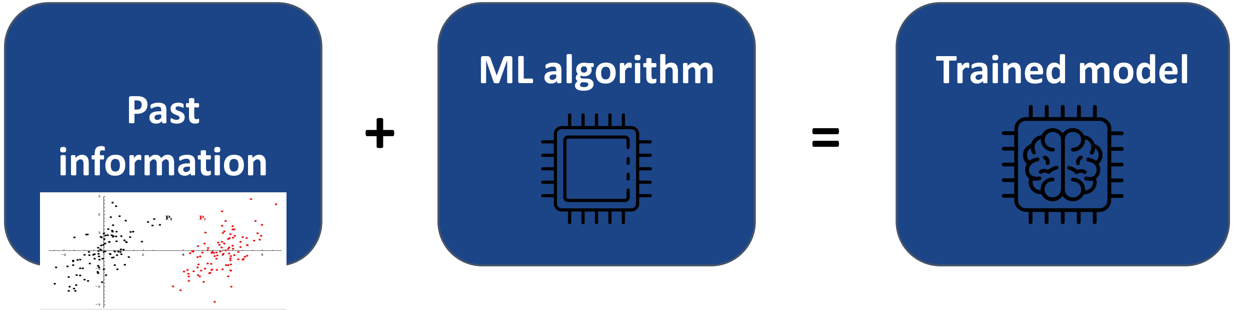
1. **Description:** each word is represented as a real vector. Words with similar meanings are close in the vector space.
2. **How to create them:** the vectors are built with neural networks that use large collections of documents (example: the whole PubMed)
3. **Limitations:** do not take into account the context (e.g. “banco” has different meanings in different contexts)

	X1	X2	...	Xn
apple	0.2	0.0	...	-0.3
doctor	0.5	-0.9	...	0.11
injury	-1.5	0.4	...	-0.3
dog	-0.11	0.6	...	-0.3



Step 3: Train Machine Learning algorithm to detect oncology mentions

Motivation: Obtain a machine learning model ready to detect oncology mentions.



Past information

+

ML algorithm

=

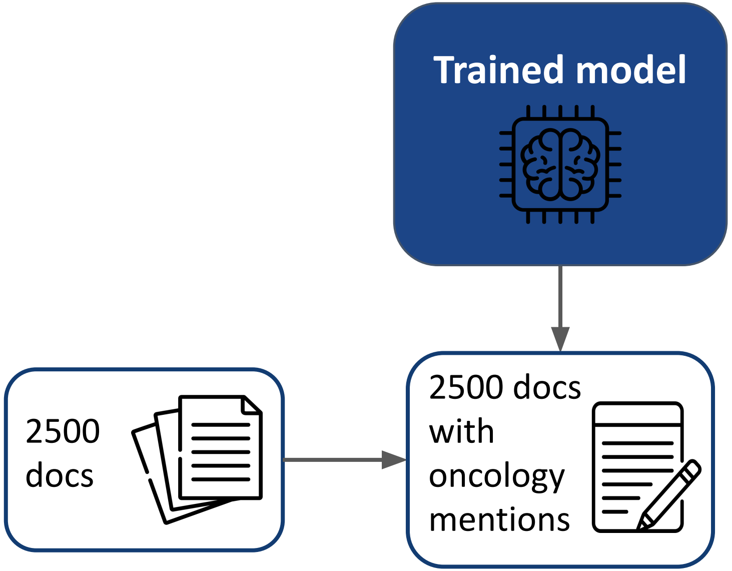
Trained model

Steps:

1. Find a Gold Standard of Spanish documents with oncology mentions
2. Train the algorithm with the Gold Standard → it creates its own rules to detect the oncology mentions in the new 2500 documents.

Step 4: Use the trained machine learning algorithm to find oncology mentions

Motivation: I know have a model that finds oncology mentions. I will use it in my 2500 documents, find the oncology mentions and select the documents that have one or more of them.

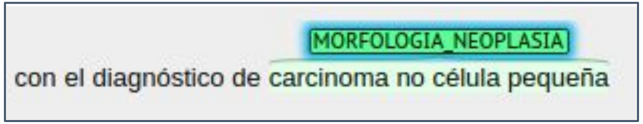


Trained model

2500 docs

2500 docs with oncology mentions

E.g.



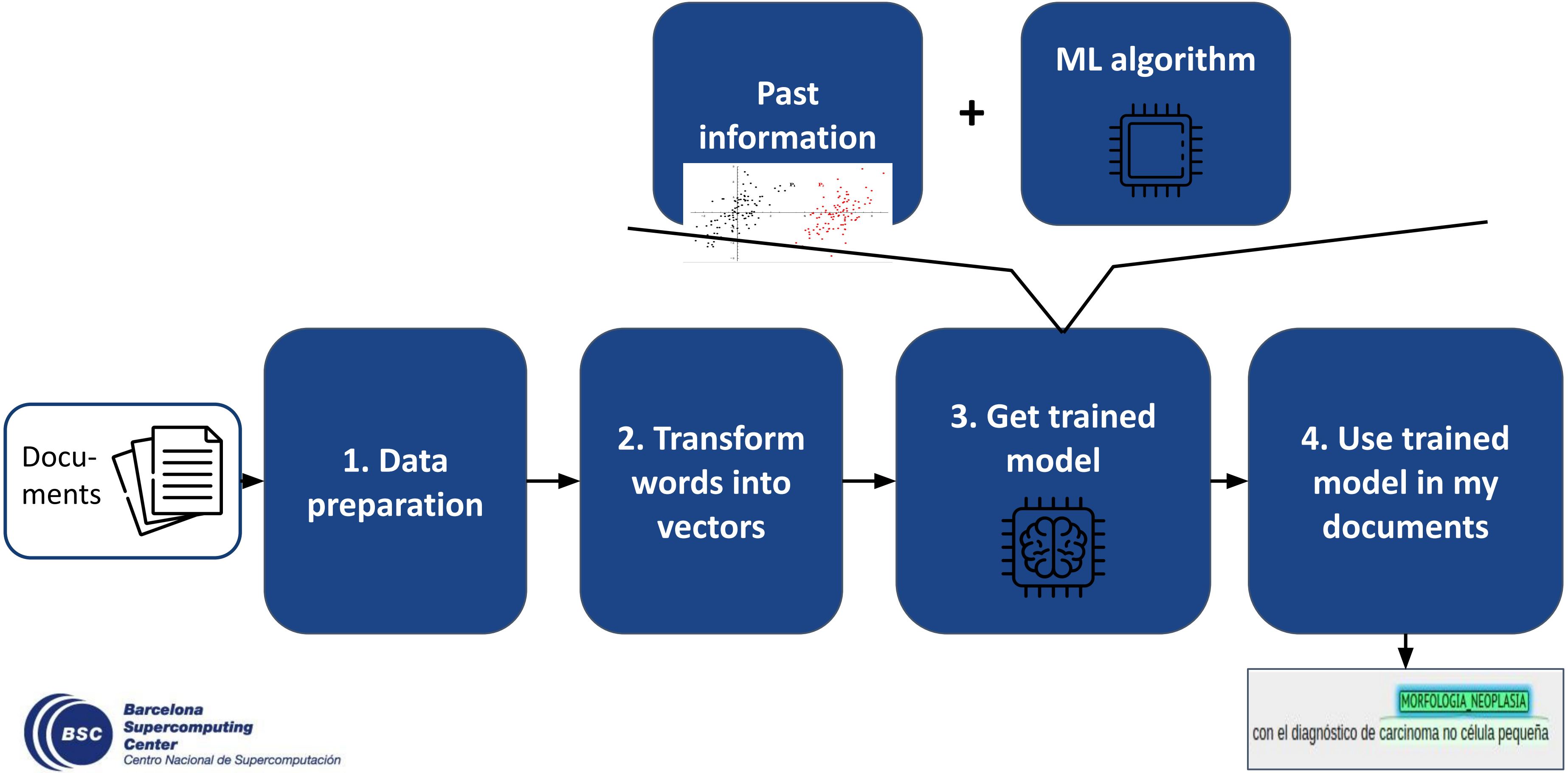
MORFOLOGIA_NEOPLASIA

con el diagnóstico de carcinoma no célula pequeña

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

all icons are from https://www.flaticon.com/

Recap of the steps performed



Past information

+

ML algorithm

Docu- ments

1. Data preparation

2. Transform words into vectors

3. Get trained model

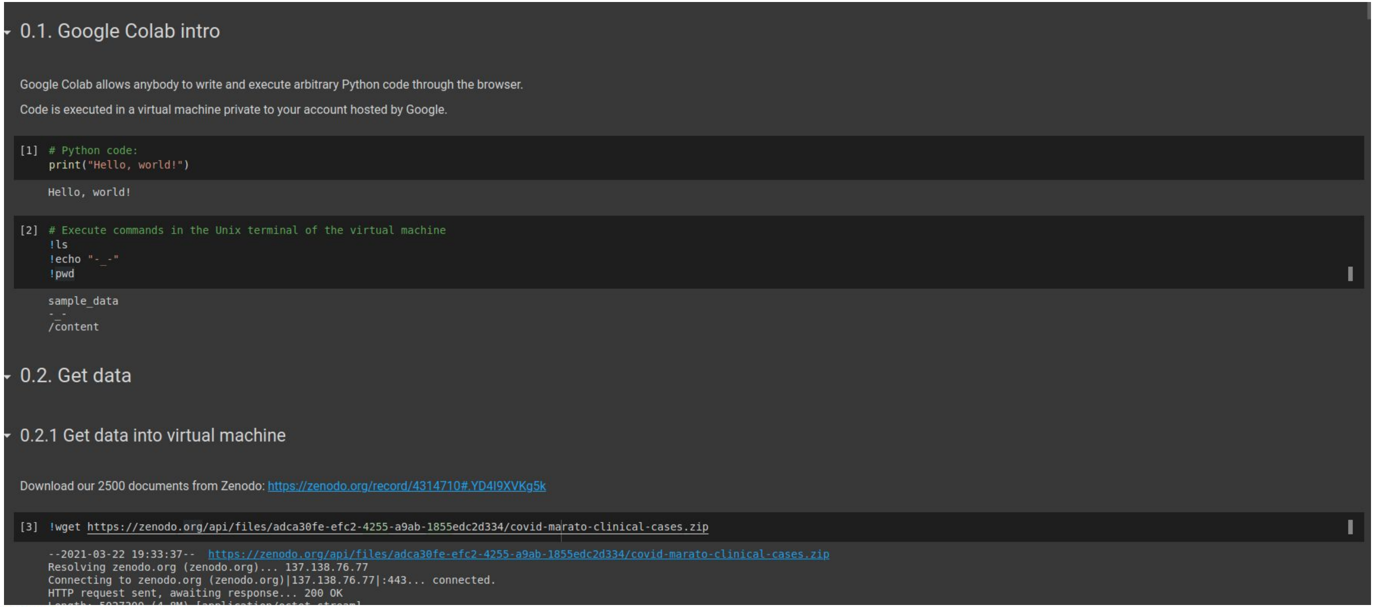
4. Use trained model in my documents

MORFOLOGIA_NEOPLASIA

con el diagnóstico de carcinoma no célula pequeña

Do it with code!

Now, you will do all the steps with code!
Copy to open Google Colab (you must be logged in with a Google account):
https://tinyurl.com/h2hfrdv8



0.1. Google Colab intro

Google Colab allows anybody to write and execute arbitrary Python code through the browser.

Code is executed in a virtual machine private to your account hosted by Google.

```
[1] # Python code:
    print("Hello, world!")

    Hello, world!
```

```
[2] # Execute commands in the Unix terminal of the virtual machine
    !ls
    !echo "-_-"
    !pwd

    sample_data
    -_-
    /content
```

0.2. Get data

0.2.1 Get data into virtual machine

Download our 2500 documents from Zenodo: https://zenodo.org/record/4314710#.YD4I9XVKg5k

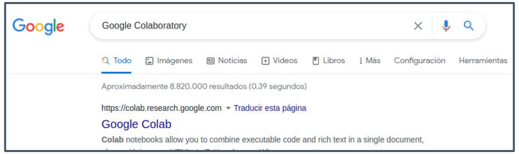
```
[3] !wget https://zenodo.org/api/files/adca30fe-efc2-4255-a9ab-1855edc2d334/covid-marato-clinical-cases.zip

    --2021-03-22 19:33:37--  https://zenodo.org/api/files/adca30fe-efc2-4255-a9ab-1855edc2d334/covid-marato-clinical-cases.zip
    Resolving zenodo.org (zenodo.org)... 137.138.76.77
    Connecting to zenodo.org (zenodo.org)|137.138.76.77|:443... connected.
    HTTP request sent, awaiting response... 200 OK
```

Do it with code!

If the URL does not work:

1. Open Google Colaboratory from Google



Google Colaboratory

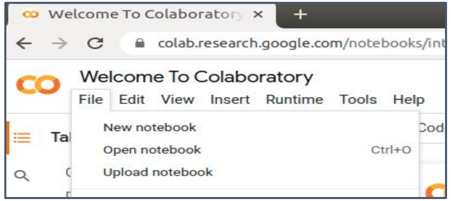
Aproximadamente 8.820.000 resultados (0,39 segundos)

https://colab.research.google.com ▾ Traducir esta página

Google Colab

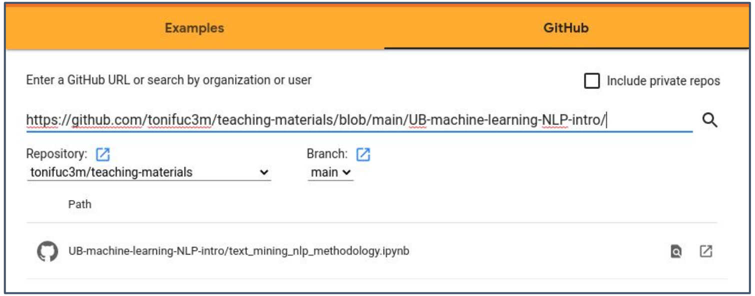
Colab notebooks allow you to combine executable code and rich text in a single document,

2. File > Open Notebook



Welcome To Colaboratory

3. On the GitHub window, search for https://github.com/tonifuc3m/teaching-materials/blob/main/UB-machine-learning-NLP-intro/



Enter a GitHub URL or search by organization or user

□ Include private repos

https://github.com/tonifuc3m/teaching-materials/blob/main/UB-machine-learning-NLP-intro/

Repository: tonifuc3m/teaching-materials

Branch: main

UB-machine-learning-NLP-intro/text_mining_nlp_methodology.ipynb

4. Select UB-machine-learning-NLP-intro/text_mining_nlp_methodology.ipynb