DU PROMPTING AU FINE-TUNING

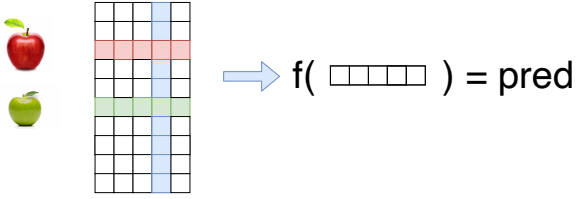
Vincent Guigue & Rémy Découpes
vincent.guigue@agroparistech.fr

INRAE AgroParisTech université PARIS-SACLAY

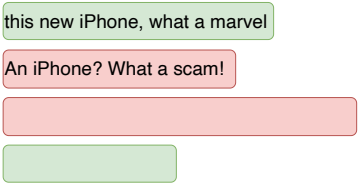
Textual data

From tabular data to text

- Tabular data
  - Fixed dimension
  - Continuous values



- Textual data
  - Variable length
  - Discrete values

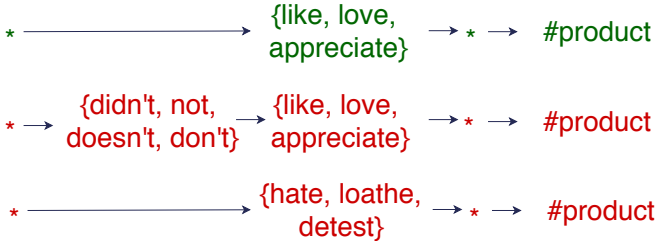


AI + Textual Data: Natural Language Processing (NLP)

NLP = largest scientific community in AI

Linguistics [1960-2010]

Rule-based Systems:

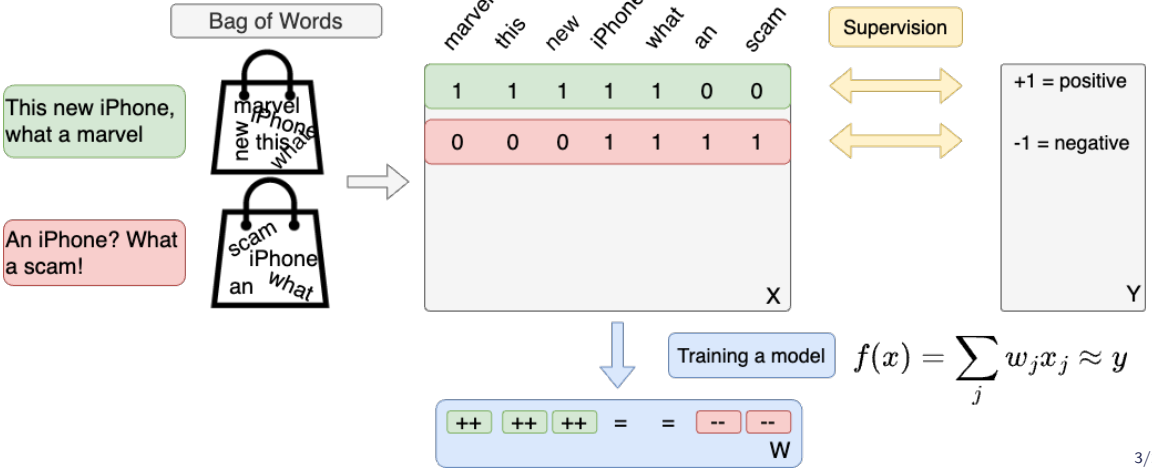


- Requires expert knowledge
- Rule extraction $\Leftrightarrow$ very clean data
- Very high precision
- Low recall
- Interpretable system

AI + Textual Data: Natural Language Processing (NLP)

NLP = largest scientific community in AI

Machine Learning [1990-2015]



AI + Textual Data: Natural Language Processing (NLP)

NLP = largest scientific community in AI

Linguistics [1960-2010]

- Requires expert knowledge
- Rule extraction ⇔ very clean data

\+ Interpretable system
\+ Very high precision
− Low recall

Machine Learning [1990-2015]

- Little expert knowledge needed
- Statistical extraction ⇔ robust to noisy data

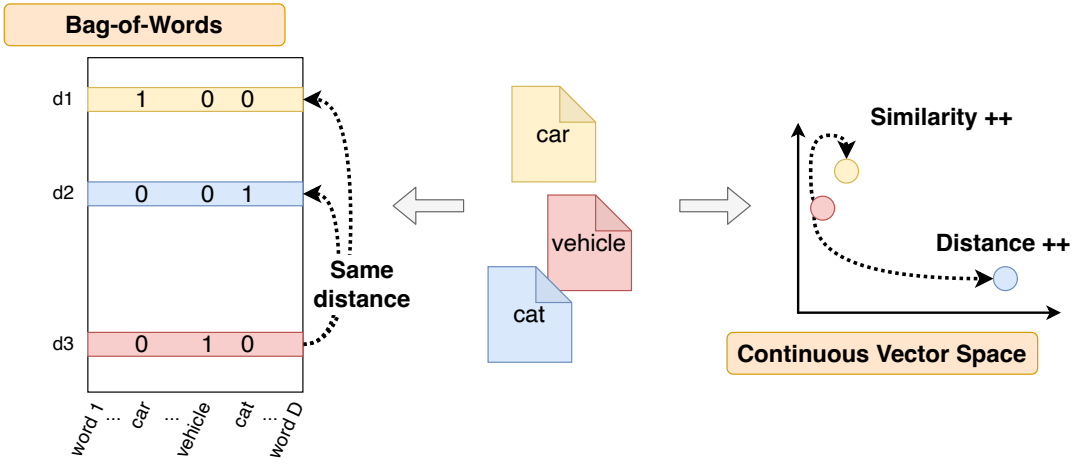
≈ Less interpretable system
− Lower precision
\+ Better recall

Precision = criterion for acceptance by industry

→ Link to metrics

Deep/Representation Learning for Text Data

From Bag of Words to Vector Representations [2008, 2013, 2016]

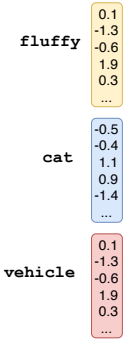


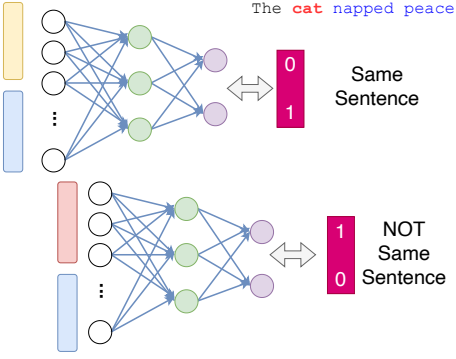
LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

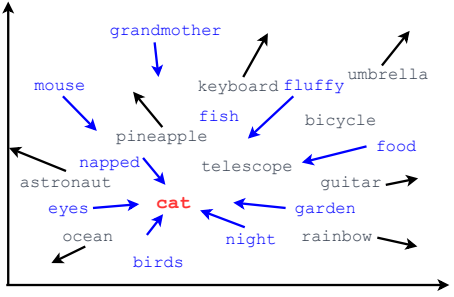
Deep/Representation Learning for Text Data

From Bag of Words to Vector Representations [2008, 2013, 2016]

The fluffy cat napped lazily in the sunbeam.
I adopted a stray cat from the shelter last week.
My cat loves to chase after toy mice.
The black cat stealthily crept through the dark alley.
I often find my cat perched on the windowsill, watching birds.
She gently stroked her cat's fur as it purred contentedly.
Our neighbor's cat frequently visits our backyard.
My cat has a preference for fish flavored cat food.
The cat stealthily stalked a mouse in the garden.
My grandmother has a collection of porcelain cat figurines.
The cat napped peacefully in the warm sunlight.

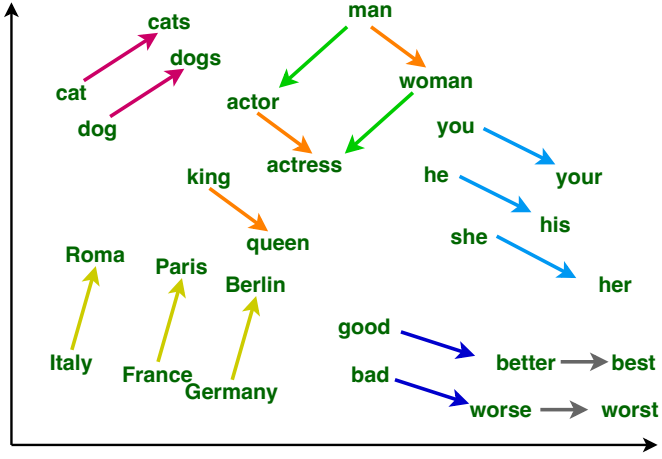






Deep/Representation Learning for Text Data

From Bag of Words to Vector Representations [2008, 2013, 2016]



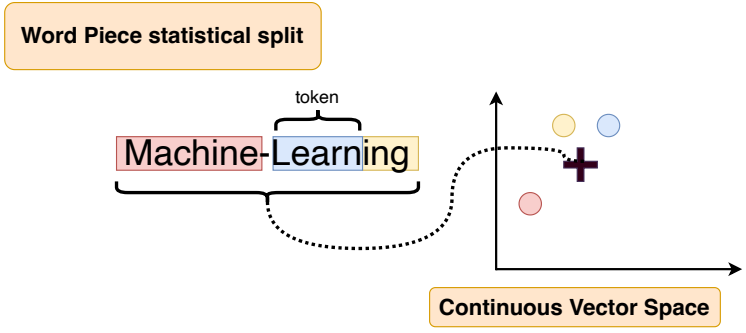
- Semantic Space:
  similar meanings
  $\Leftrightarrow$
  close positions
- Structured Space:
  grammatical regularities,
  basic knowledge, ...

Distributed representations of words and phrases and their compositionality, Mikolov et al. NeurIPS 2013

Deep/Representation Learning for Text Data

From Bag of Words to Vector Representations [2008, 2013, 2016]

From Words to Tokens

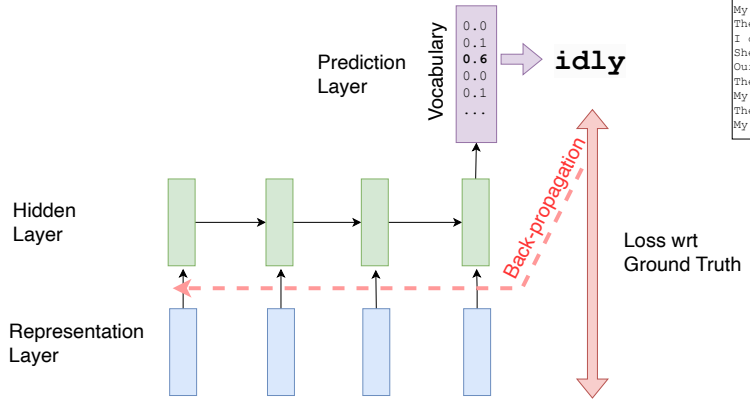


- Representation of unknown words
- Adaptation to technical domains
- Resistance to spelling errors

Enriching word vectors with subword information. Bojanowski et al. TACL 2017.

Aggregating word representations: towards generative AI

- Generation & Representation
- New way of learning word positions



The fluffy cat napped lazily in the sunbeam.
I adopted a stray cat from the shelter last week.
My cat loves to chase after toy mice.
The black cat stealthily crept through the dark alley.
I often find my cat perched on the windowsill, watching birds.
She gently stroked her cat's fur as it purred contentedly.
Our neighbor's cat frequently visits our backyard.
The playful cat swatted at the dangling string with its paw.
My cat has a preference for fish flavored cat food.
The cat stealthily stalked a mouse in the garden.
My grandmother has a collection of porcelain cat figurines.

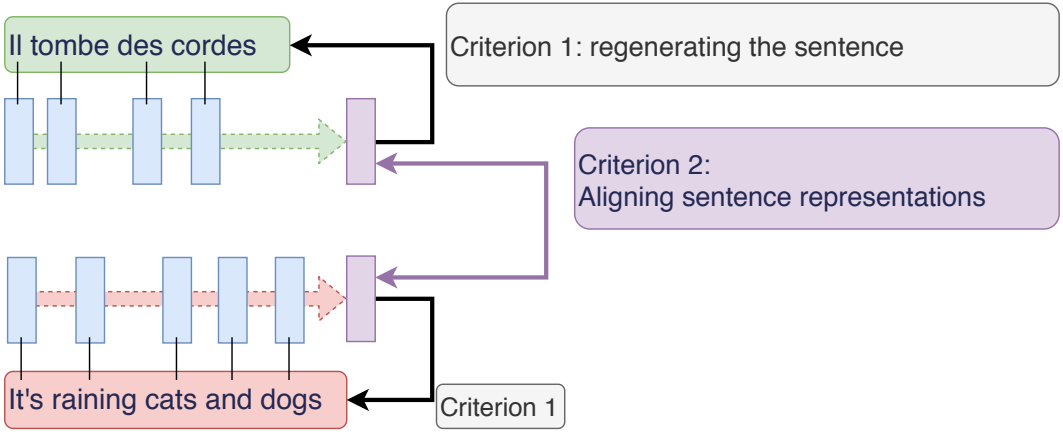
Corpus

Sequence to Sequence Learning with Neural Networks, Sutskever et al. NeurIPS 2014

Use-Case: Machine Translation

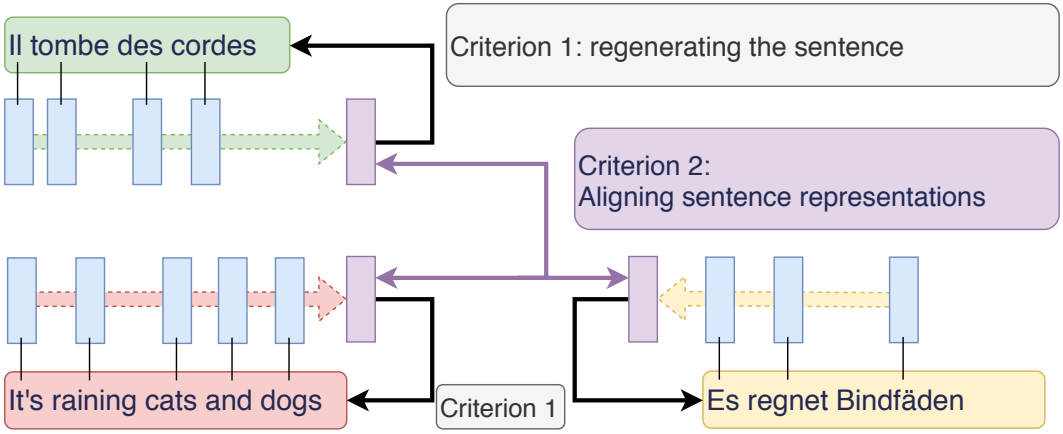
Beyond word-for-word translation, multilingual representation of sentences

Use-Case: Machine Translation



Beyond word-for-word translation, multilingual representation of sentences

Use-Case: Machine Translation

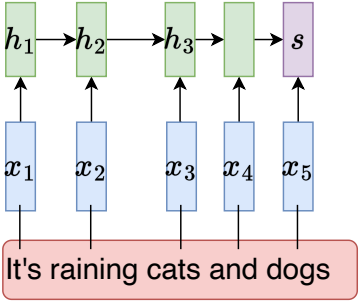


Beyond word-for-word translation, multilingual representation of sentences

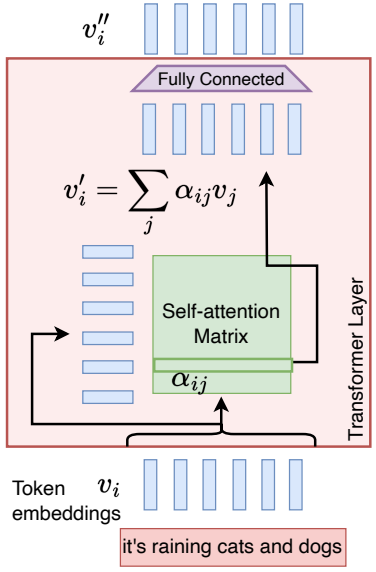
Transformer architecture: state-of-the-art aggregation

Recurrent Neural Network:

$$h_{t+1} = h_t W_1 + x_{t+1} W_2$$



Transformer:



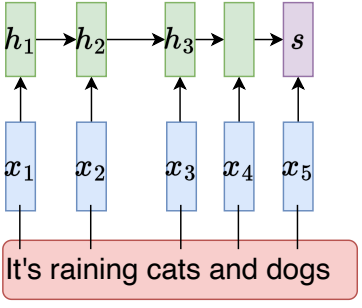
Attention is all you need, Vaswani et al. NeurIPS 2017

Sequence to Sequence Learning with Neural Networks, Sutskever et al. NeurIPS 2014

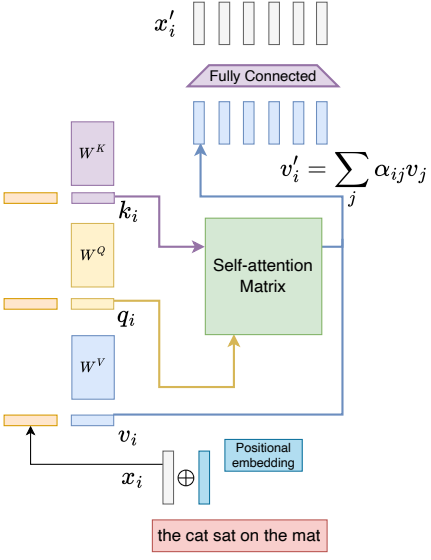
Transformer architecture: state-of-the-art aggregation

Recurrent Neural Network:

$$h_{t+1} = h_t W_1 + x_{t+1} W_2$$



Transformer:



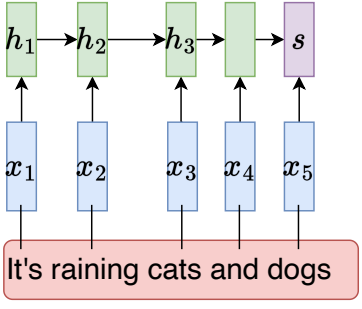
Attention is all you need, Vaswani et al. NeurIPS 2017

Sequence to Sequence Learning with Neural Networks, Sutskever et al. NeurIPS 2014

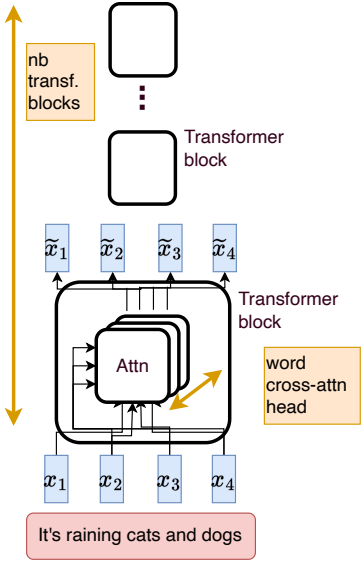
Transformer architecture: state-of-the-art aggregation

Recurrent Neural Network:

$$h_{t+1} = h_t W_1 + x_{t+1} W_2$$



Transformer:

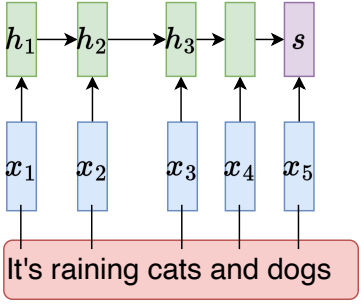


Attention is all you need, Vaswani et al. NeurIPS 2017

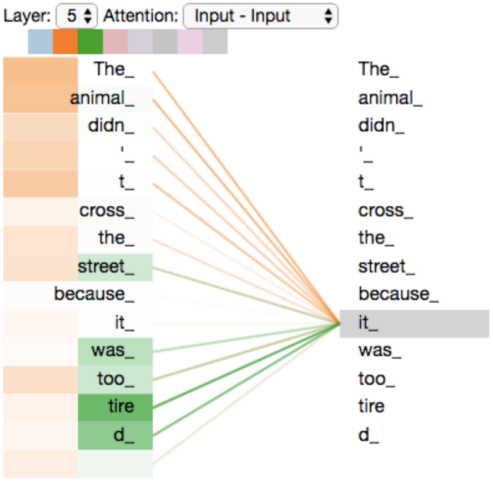
Transformer architecture: state-of-the-art aggregation

Recurrent Neural Network:

$$h_{t+1} = h_t W_1 + x_{t+1} W_2$$



Transformer:

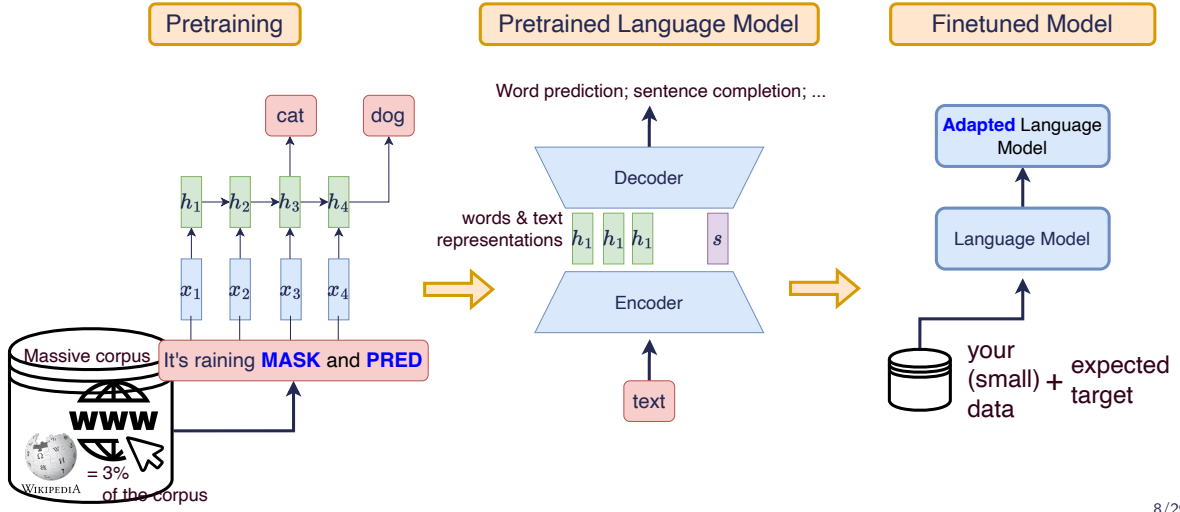


Attention is all you need, Vaswani et al. NeurIPS 2017

Sequence to Sequence Learning with Neural Networks, Sutskever et al. NeurIPS 2014

A new developpement paradigm since 2015

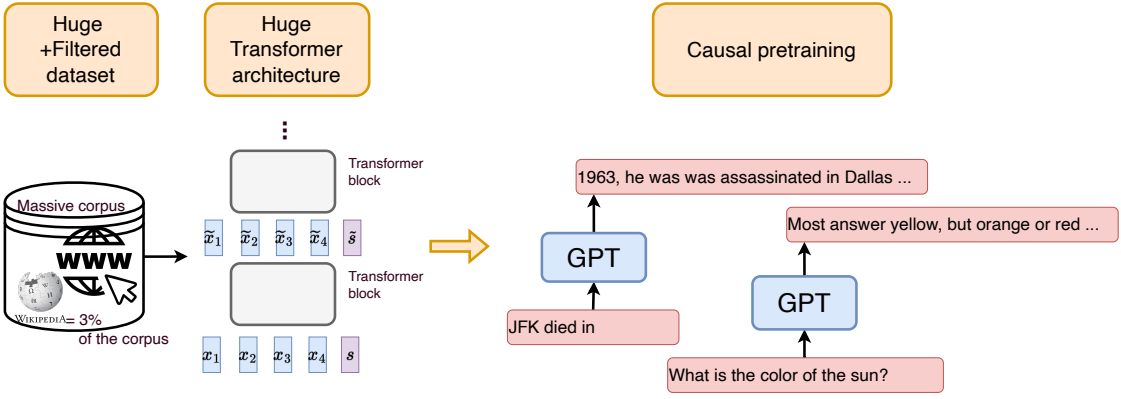
- Huge dataset + huge archi. $\Rightarrow$ unreasonable training cost
- Pre-trained architecture + 0-shot / finetuning



Generative AI & finetuning

The genAI wave

- *Causal* architecture / generating relevant answers

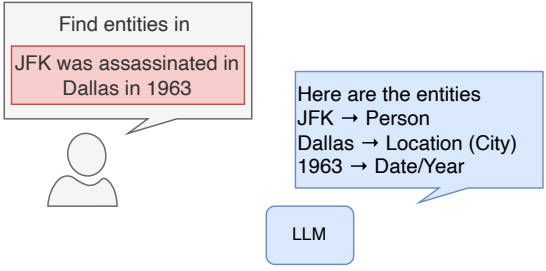


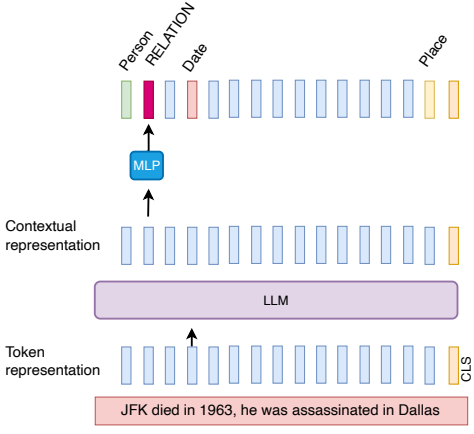
- Grammatical skills: singular/plural agreement, tense concordance
- Knowledges: entities, names, dates, places

Language Models are Few-Shot Learners, Brown et al. 2020

A revolution in NLP: new formulation, new(?) metrics

- input = text... output = text also !!!

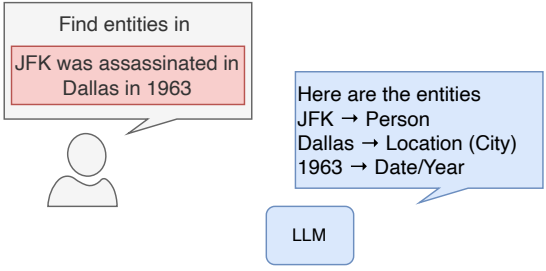




- How to evaluation Entity location?
- What about rephrasing?
- Over detection

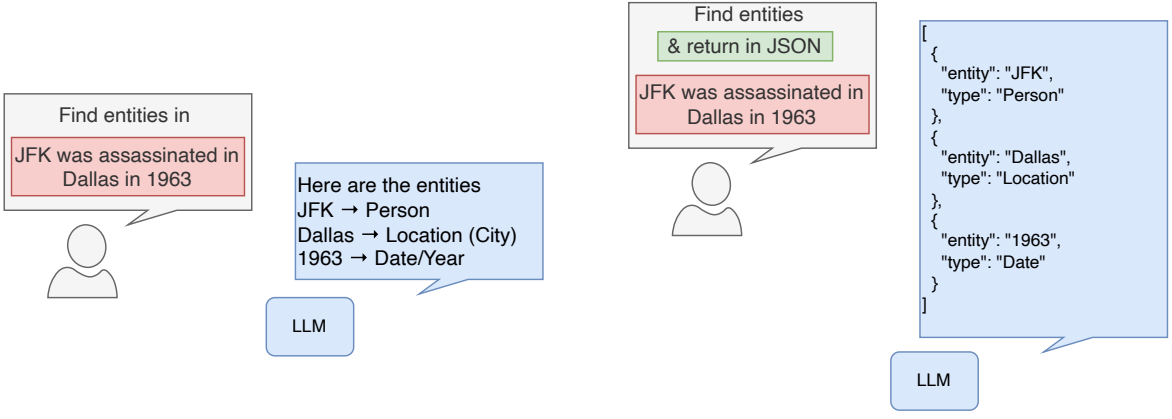
JFK PERSON was assassinated in Dallas GPE in 1963 DATE

Formating constraints



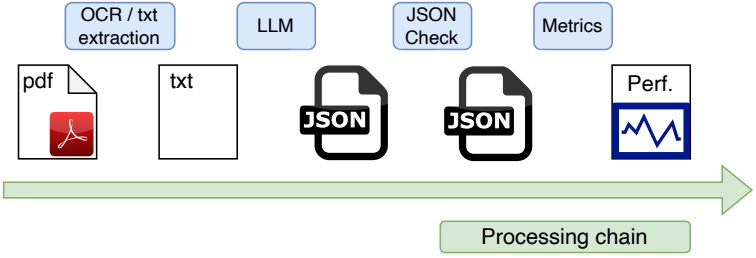
- Prompt optimization: role, task, categories, examples, ...

Formating constraints



- Prompt optimization: role, task, categories, examples, ...
- Adding constraints ⇒ processing chain
- Forcing the system to return JSON... **and only JSON**

Processing chain

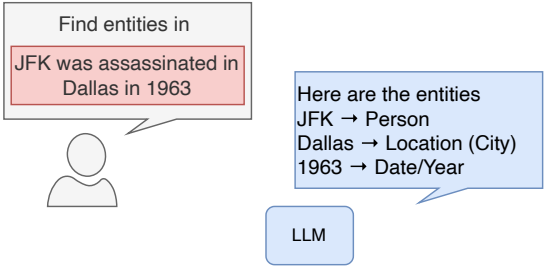


- Famous framework = langchain
- Challenge: all components should be replacable
- OCR : pypdf, tesseract...
- LLM : ollama, HuggingFace, GPT, ...
- JSON : regex, LLM constraint, multiple queries
- Metrics

Zero-shot

Zero-shot

Exploiting a model on a task... (almost) without training



- Is the task really new?
  - to be efficient, LLM are trained on multiple instructions
  - Risk of data contamination
- Which performance to expect?

1st optimization step: Prompting

Prompting = the way to ask a question ⇒ impact on the performance

You are an information extraction system. Your task is to read an English news article and extract named entities of the following types:

- **PERSON**: names of individual people
- **LOCATION**: geographical entities such as cities, countries, landmarks
- **ORGANIZATION**: companies, institutions, government bodies, media outlets, NGOs, etc.
- **DATE**: explicit calendar dates or temporal expressions (e.g., "January 5, 2023", "last Monday")

Return the results in **JSON format** with the following structure:

json Copier le code

```
{
  "PERSON": ["..."],
  "LOCATION": ["..."],
  "ORGANIZATION": ["..."],
  "DATE": ["..."]
}
```

If a category has no entities, return an empty list. Do not include duplicates.

Text to analyze:

1st optimization step: Prompting

Prompting = the way to ask a question ⇒ impact on the performance

You are an information extraction system. Your task is to read an English news article and extract named entities of the following types:

- **PERSON**: names of individual people
- **LOCATION**: geographical entities such as cities, countries, landmarks
- **ORGANIZATION**: companies, institutions, government bodies, media outlets, NGOs, etc.
- **DATE**: explicit calendar dates or temporal expressions (e.g., "January 5, 2023", "last Monday")

Return the results in **JSON format** with the following structure:

json — Copier le code

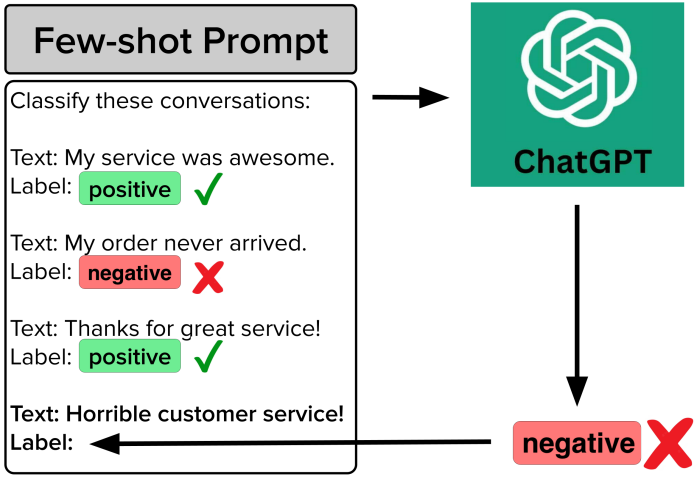
```
{
  "PERSON": ["..."],
  "LOCATION": ["..."],
  "ORGANIZATION": ["..."],
  "DATE": ["..."]
}
```

ROLE
TASK
Class DEF
Format
Supp.

If a category has no entities, return an empty list. Do not include duplicates.

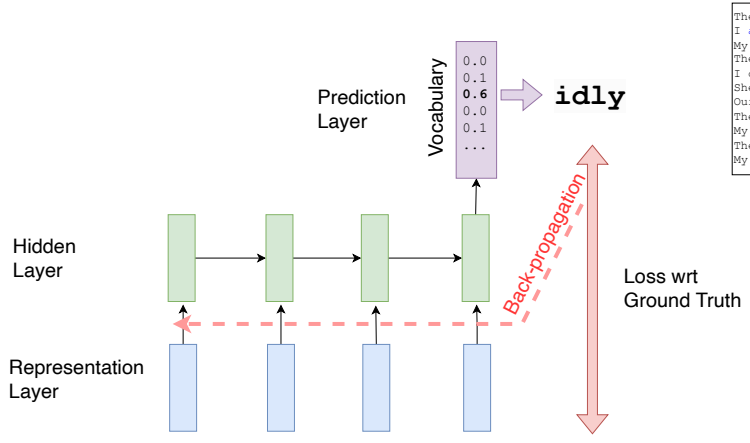
Few-shot learning (no optimization yet!)

Give (input $\Leftrightarrow$ output) on few examples to guide the extraction



Finetuning of genAI

Most generative finetuning = train to generate the ground truth



The fluffy cat napped lazily in the sunbeam.
I adopted a stray cat from the shelter last week.
My cat loves to chase after toy mice.
The black cat stealthily crept through the dark alley.
I often find my cat perched on the windowsill, watching birds.
She gently stroked her cat's fur as it purred contentedly.
Our neighbor's cat frequently visits our backyard.
The playful cat swatted at the dangling string with its paw.
My cat has a preference for fish flavored cat food.
The cat stealthily stalked a mouse in the garden.
My grandmother has a collection of porcelain cat figurines.

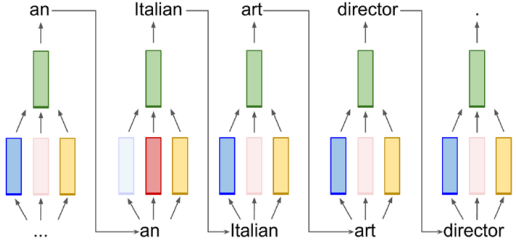
Corpus

The fluffy cat napped lazily in the sunbeam.

Faithfulness optimization : At the token level[1]

Name	Giuseppe Mariani
Occupation	Art director
Years active	1952 - 1992

Giuseppe Mariani was an **Italian** art director.

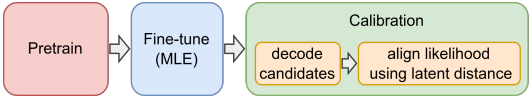


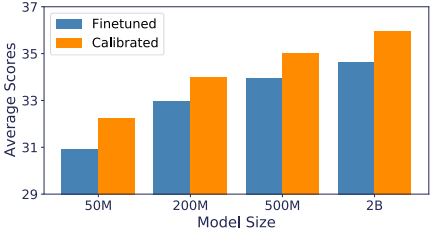
- Require annotation at the token level
- Multi-branch decoder ⇒ find the good balance (fluency, faithfulness, ...)

[1] Rebuffel et al, Data Mining and Knowledge Discovery 2022.
Controlling hallucinations at word level in data-to-text generation.

Faithfulness opti as a post-processing step[3]

- Calibrating the likelihood in the beam-search procedure





- Conditional PMI Decoding[2]: detecting hazard (entropy) + shifting proba

$$\text{score}\left(y \mid \mathbf{y}_{<t}, \mathbf{x}\right) = \log p\left(y \mid \mathbf{y}_{<t}, \mathbf{x}\right) - \lambda \cdot \mathbb{1}_{\{\mathrm{H}(p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x})) \geq \tau\}} \cdot \log p\left(y \mid \mathbf{y}_{<t}\right)$$

[2] van der Poel et al.; EMNLP 2022
Mutual Information Alleviates Hallucinations in Abstractive Summarization

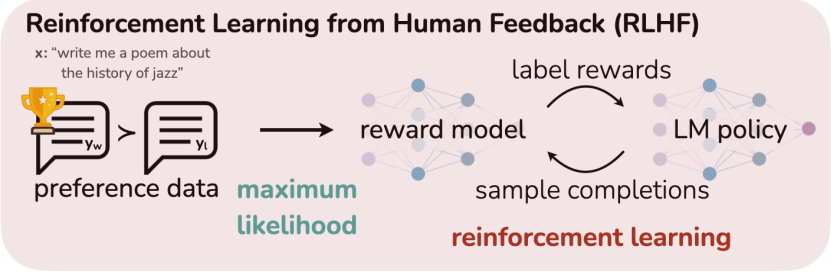
[3] Zhao et al.; ICLR 2023
Calibrating Sequence likelihood Improves Conditional Language Generation

Let's optimize preferences ! [PPO[4]]

(Major) assumption

We have hallucinated *vs* proper sentences in a data-to-text framework

Since instructGPT... We use PPO (Proximal Policy Optimization)



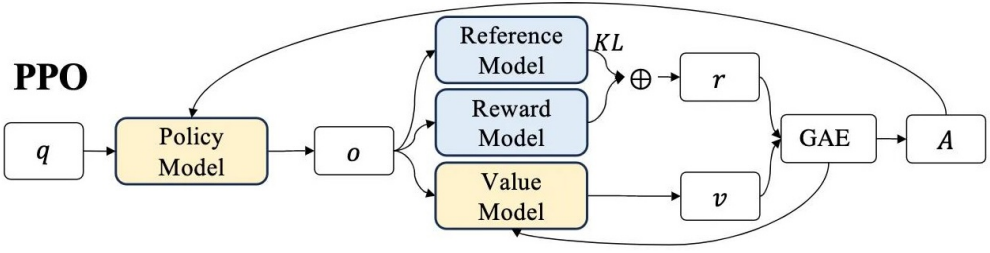
[4]Schulman et al. arXiv 2017. Proximal Policy Optimization Algorithms

Let's optimize preferences ! [PPO[4]]

(Major) assumption

We have hallucinated *vs* proper sentences in a data-to-text framework

Since instructGPT... We use PPO (Proximal Policy Optimization)



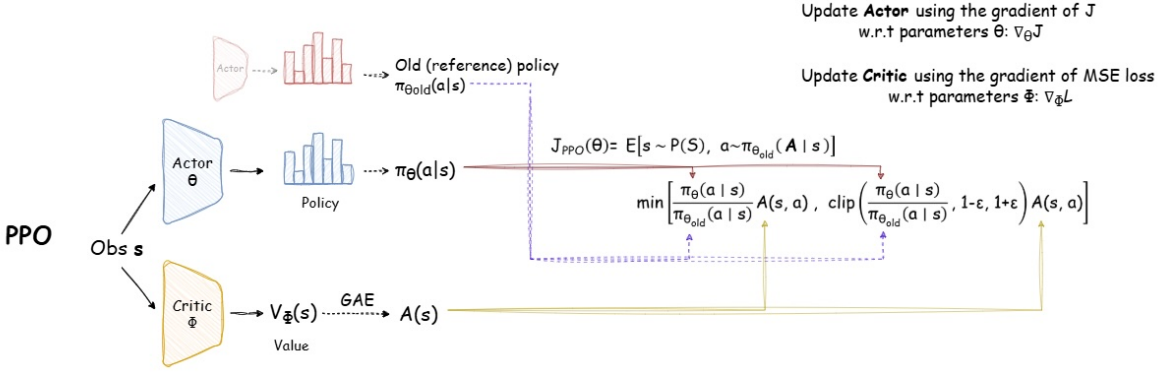
[4]Schulman et al. arXiv 2017. Proximal Policy Optimization Algorithms

Let's optimize preferences ! [PPO[4]]

(Major) assumption

We have hallucinated *vs* proper sentences in a data-to-text framework

Since instructGPT... We use PPO (Proximal Policy Optimization)



[4] Schulman et al. arXiv 2017. Proximal Policy Optimization Algorithms

Let's optimize preferences ! [PPO[4]]

(Major) assumption

We have hallucinated *vs* proper sentences in a data-to-text framework

Since instructGPT... We use PPO (Proximal Policy Optimization)

- 4 models to load in memory (π_θ, π_0, V)
- 2 models with gradients ($\pi_\theta, V/A$)
- Intensive sampling $\propto \pi_\theta(y_t \mid y_{<t})$
- Instable procedure (cf regul. terms)

[4] Schulman et al. arXiv 2017. Proximal Policy Optimization Algorithms

Let's optimize preferences ! [PPO[4]]

(Major) assumption

We have hallucinated *vs* proper sentences in a data-to-text framework

Since instructGPT... We use PPO (Proximal Policy Optimization)

```
(Données humaines)
      |
Comparaisons ↴
     Train Reward Model r_ϕ

      ↓
Prompts → LLM (π_θ) → Réponses
                      ↓
               Eval avec r_ϕ
                      ↓
             Recompense + KL + Avantage
                      ↓
             PPO update π_θ ← π_θ + Δθ
```

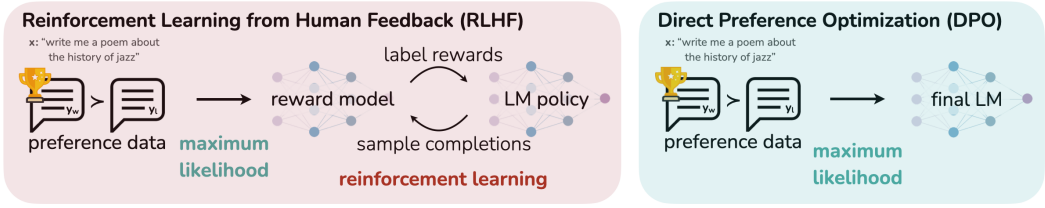
$$L^{\text{CLIP}}(\theta) = -\mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_t\right)\right]$$

Reward:

$$r_t(\theta) = \beta \log \frac{\pi_\theta\left(y_t \mid y_{<t}\right)}{\pi_0\left(y_t \mid y_{<t}\right)}$$

[4] Schulman et al. arXiv 2017. Proximal Policy Optimization Algorithms

Simplifying the procedure [DPO[5]]



Same reward:

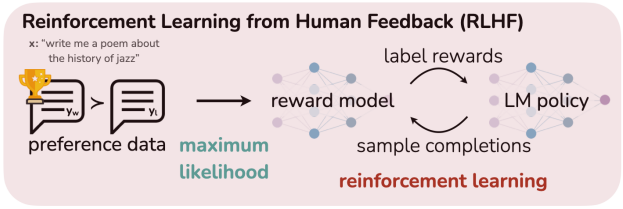
$$\hat{r}_\theta(x, y) = \beta \log \frac{\pi_\theta\left(y_t \mid y_{<t}\right)}{\pi_0\left(y_t \mid y_{<t}\right)}$$

Different cost:

$$\mathcal{L}_{\mathrm{DPO}}\left(\pi_\theta; \pi_{\mathrm{ref}}\right) = -\mathbb{E}_{(y_{t+}, y_{t-}, y_{<t},) \sim \mathcal{D}}\left[\log \sigma\left(\beta \log \frac{\pi_\theta\left(y_{t+} \mid y_{<t}\right)}{\pi_0\left(y_{t+} \mid y_{<t}\right)} - \beta \log \frac{\pi_\theta\left(y_{t-} \mid y_{<t}\right)}{\pi_0\left(y_{t-} \mid y_{<t}\right)}\right)\right]$$

[5] Rafailov et al., NeurIPS 2023.
Direct Preference Optimization: Your Language Model is Secretly a Reward Model

Simplifying the procedure [DPO[5]]



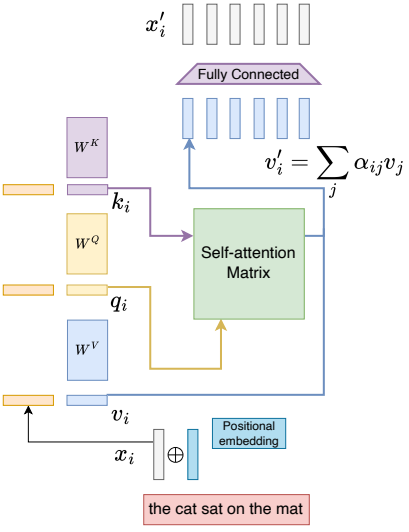
- 2 models to load in memory (π_θ, π_0)
- 1 models with gradients (π_θ)
- Intensive sampling but $\propto \pi_0(y_t \mid y_{<t}) \Rightarrow$ enable precomputing
- Classical (=stable) likelihood optimization

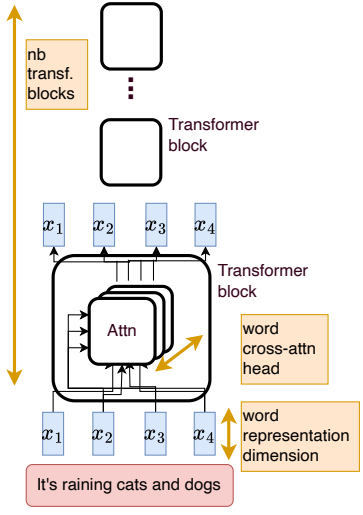
[5] Rafailov et al., NeurIPS 2023.
Direct Preference Optimization: Your Language Model is Secretly a Reward Model

Back to the Encoders

BERT : 2018 milestone

- Efficient transformer architecture





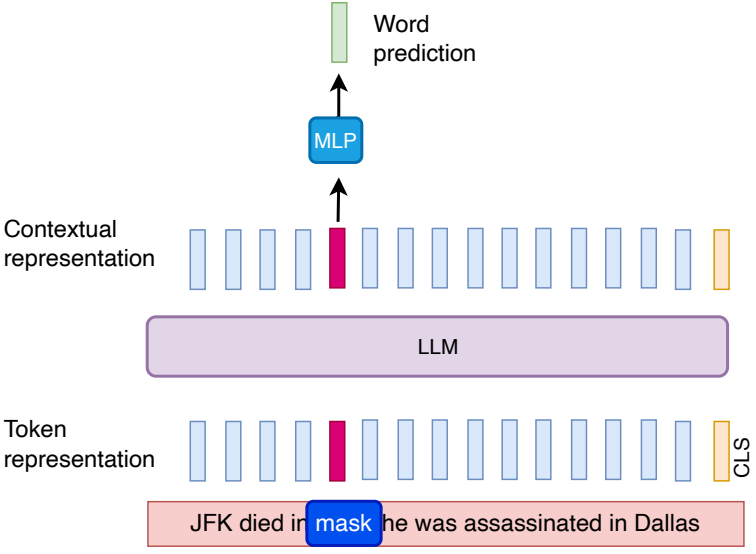
Several sizes

Nice implementation / pretraining

HuggingFace Integration

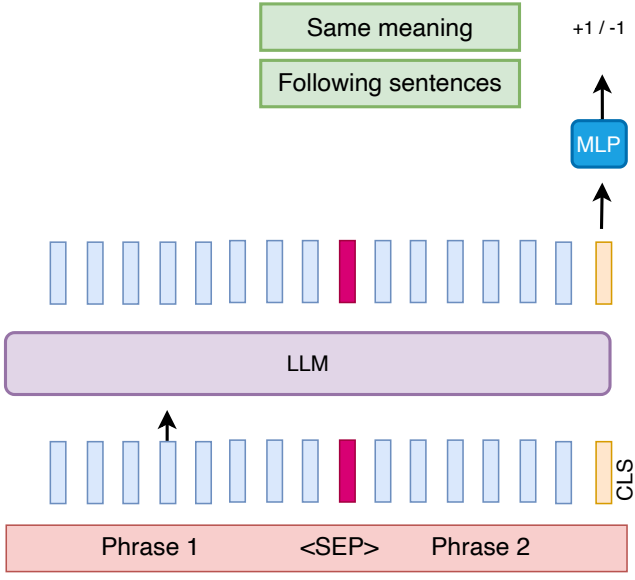
BERT Pre-training

- Masking words



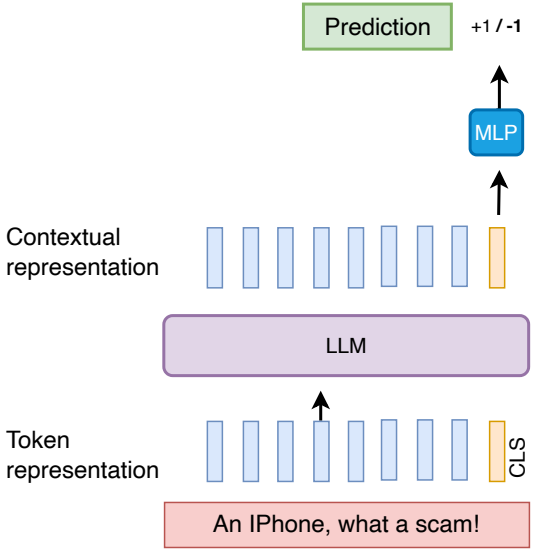
BERT Pre-training

- Predicting coherent / successive sentences



Then tackling various tasks

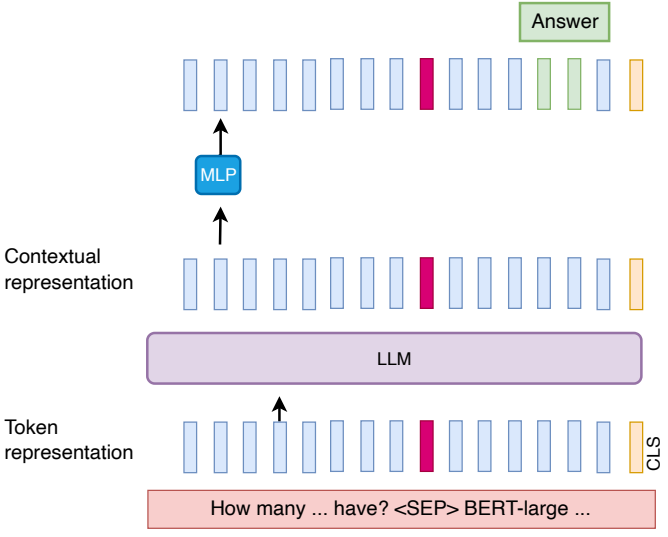
- Opinion mining



Then tackling various tasks

- Question Answering
- Often couple with Info Retrieval

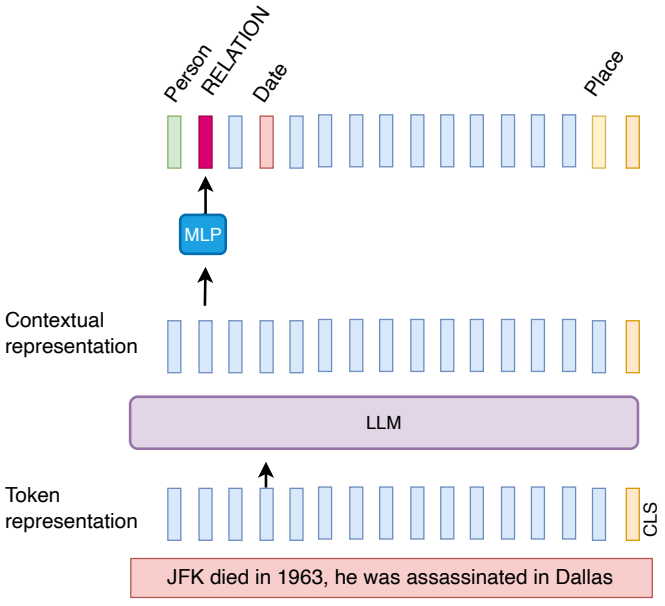
$\Rightarrow$ finding short passages to analyse



Then tackling various tasks

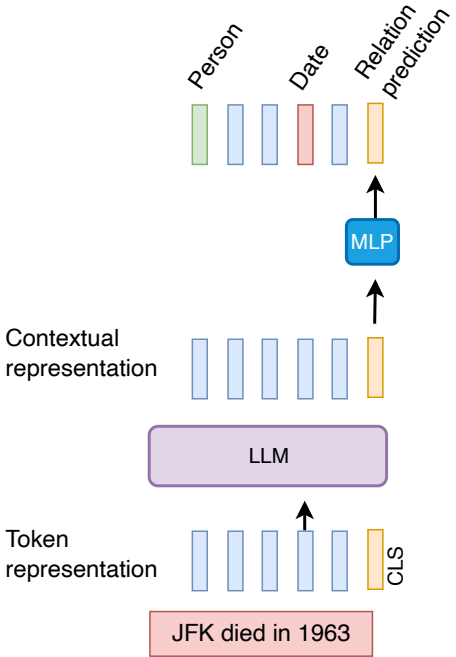
- Information extraction
- Named Entity recognition
- Relation extraction

  ... but on which word ??



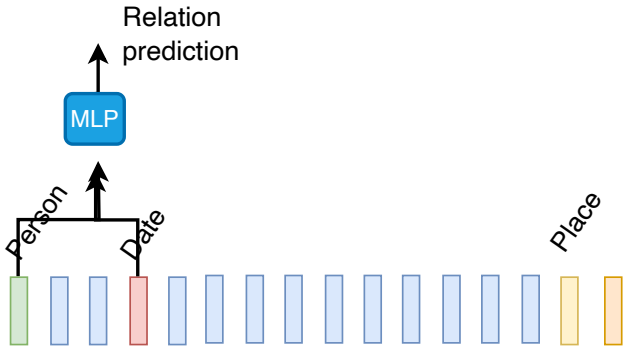
Then tackling various tasks

- Information extraction
- Named Entity recognition
- Relation extraction
  ... but on which word ??
- Hyp : unique relation in the sentence



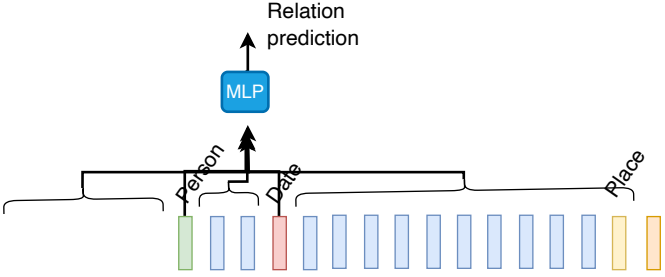
Then tackling various tasks

- Information extraction
- Named Entity recognition
- Relation extraction
  ... but on which word ??
- Hyp : unique relation in the sentence
- Between entities + all couple tested



Then tackling various tasks

- Information extraction
- Named Entity recognition
- Relation extraction
  ... but on which word ??
- Hyp : unique relation in the sentence
- Between entities + all couple tested
- Piecewise representation (PCNN)



Efficient Tuning

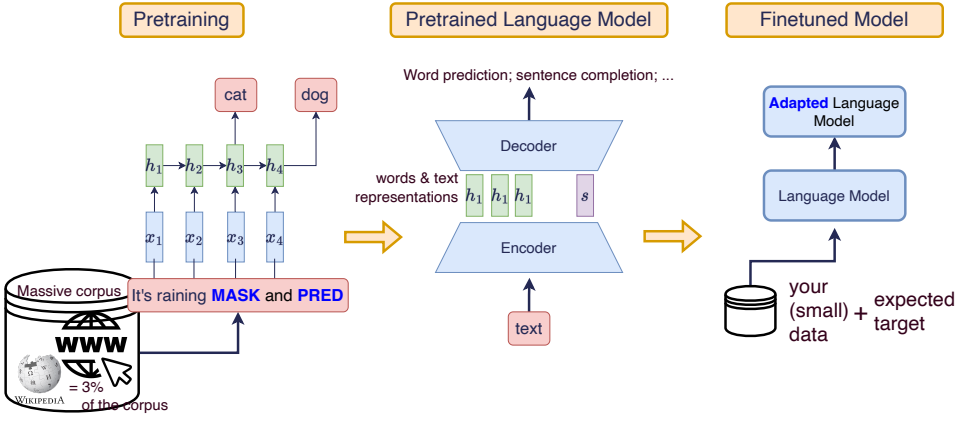
Once the pretraining done

How to update/upgrade a model for a specific domain/task?

Yes, if we upgrade it!

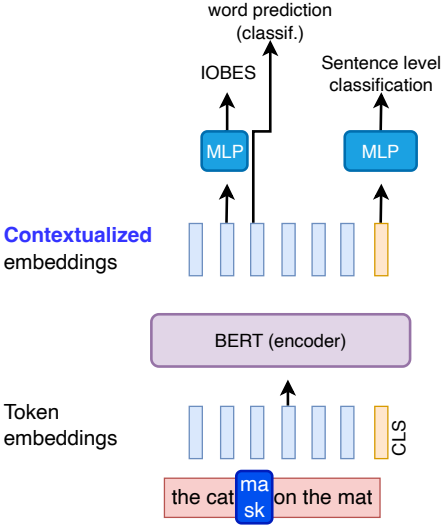
$\Rightarrow$ (Relatively) cheap update

$\Rightarrow$ Few data required



Refining the encoder

- Assuming we have textual data & supervision ⇒ optimizing the models



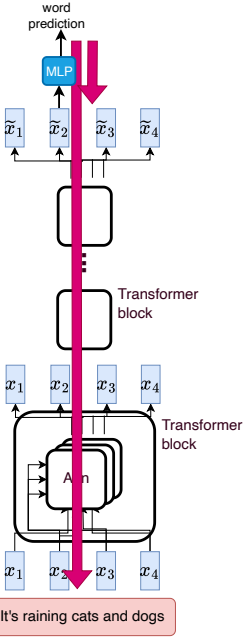
1. Supervision: X: texts; Y: targets; ($+$ W: parameters)
2. Finetuning: refining W s.t. $f_W(X) \approx Y$

May a **small** model become more efficient than a **big** LLM through refinement?

How to make the refinement process cheap?

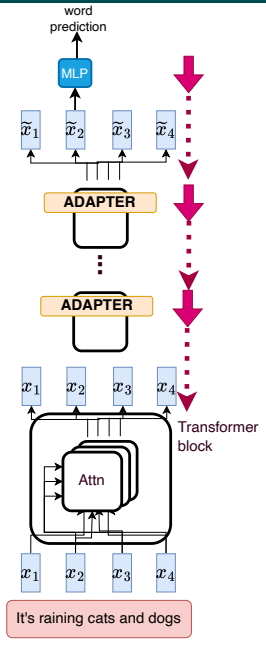
Efficient Tuning

- Limiting the number of epoch
  - Running on few data
- Only optimize the last layer



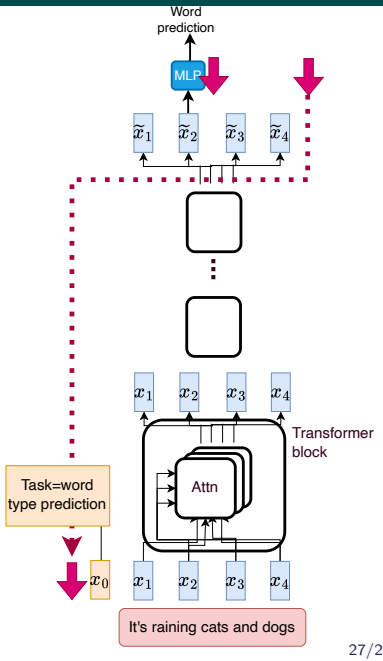
Efficient Tuning: main strategies

1. Adapter
2. Prompt Tuning
3. Low Rank Adaptation



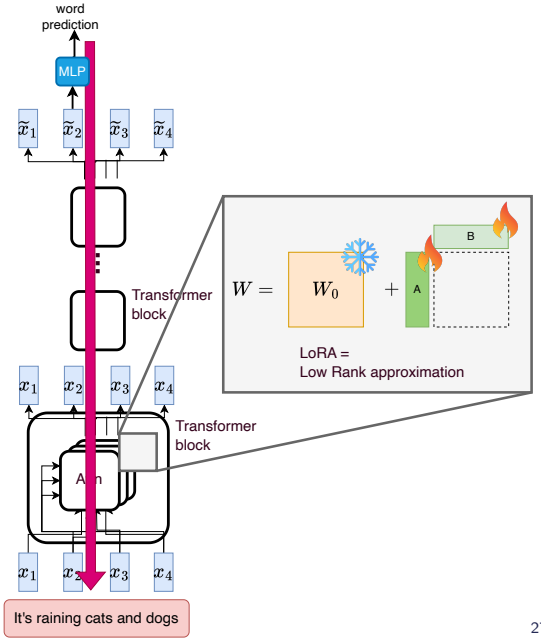
Efficient Tuning: main strategies

1. Adapter
2. Prompt Tuning
3. Low Rank Adaptation



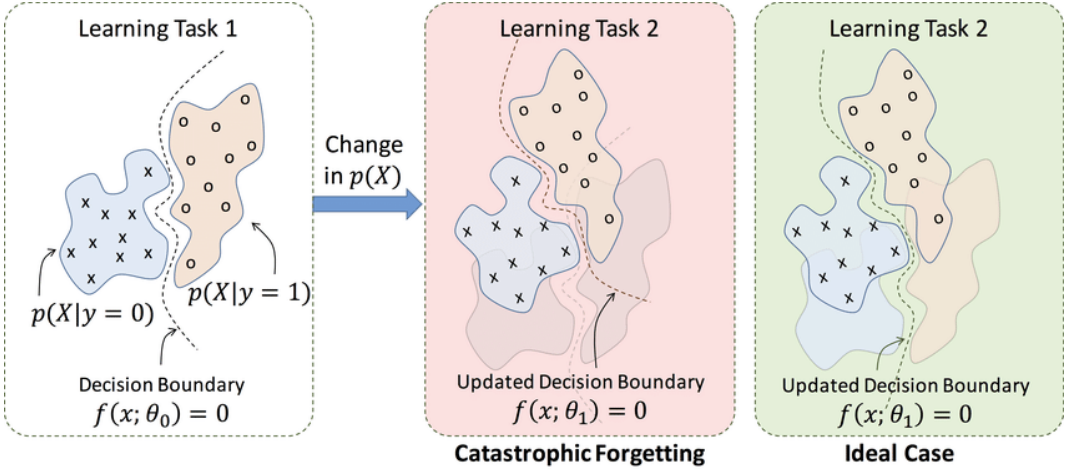
Efficient Tuning: main strategies

1. Adapter
2. Prompt Tuning
3. Low Rank Adaptation



Catastrophic forgetting

Learning new skills... Forgetting the bases?



Conclusion

Generative AI : new opportunities at a cost

- **LLM cost** >> encoders (>> CRF / trad. ML >> classical lexical rules)
  - Paying for extraction *or* for development?
- New formulations ⇒ new metrics ⇒ **how to evaluate properly?**
- Above which threshold the systems are **useful**? Autonomously useful?
- What about a system that would be able to extract *statements*?

Transformer le texte suivant en une liste de statements élémentaires:

Des drones non identifiés ont été observés au-dessus de la plus grande base militaire du Danemark vendredi soir, a annoncé la police, après plusieurs survols d'aéroports cette semaine dans le pays nordique.

Voici une décomposition en statements élémentaires :

1. Des drones non identifiés ont été observés.
2. L'observation a eu lieu au-dessus de la plus grande base militaire du Danemark.
3. L'observation a eu lieu vendredi soir.
4. La police a annoncé cette observation.
5. Des survols d'aéroports ont eu lieu cette semaine.
6. Les survols d'aéroports ont eu lieu dans le Danemark.
7. Le Danemark est un pays nordique.