The Camembert Model and Beyond

Seminar Institut Pasteur June 2022

Benjamin Muller, INRIA Paris @Ben_mlr

Background

Camembert is a **transformer-based language model for French**

In this talk:

- What motivated the design of those models
- How does it work?
- Beyond Camembert?

Acknowledgement

We built Camembert in 2019 as part of a collaborc between **INRIA Paris (ALMANACH** team) and **Facebook AI**

Work done by **Louis Martin, Pedro Ortiz, Benjamin Muller** with the guidance of Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah and Benoît Sagot

Camembert is derived from BERT (Devlin et. al 2018) and ROBERTa (Liu et. al 2019)

Outline

1. **How did we get here? Intuition about BERT-like models**
2. The Camembert Model
3. Camembert in the real-world
4. Beyond the pretraining-finetuning paradigm

How to build NLP models?

NLP Pipeline

Prediction: Tags/Tree/Text

TRAINABLE MODEL

FEATURES/EMBEDDINGS

TOKENS

Input: Text

How to build NLP models?

NLP Pipeline

Prediction: Tags/Tree/Text

TRAINABLE MODEL

FEATURES/EMBEDDINGS

TOKENS

Input: Text

1st Step:
Define our modeling units (e.g. word, character, etc.)

How to build NLP models?

NLP Pipeline

Prediction: Tags/Tree/Text

TRAINABLE MODEL

2nd Step: Represent the tokens into vectors

FEATURES/EMBEDDINGS

TOKENS

Input: Text

How to build NLP models?

NLP Pipeline

Prediction: Tags/Tree/Text

3rd Step: Combine the input information to make a prediction

TRAINABLE MODEL

FEATURES/EMBEDDINGS

TOKENS

Input: Text

Challenge in building Accurate NLP Models

Holy Grail of NLP: Building NLP models that **generalize to many domains, languages, tasks without a lot of annotated data**

- Tokenization that is robust to infrequent words
- A rich vector representation of the input tokens
- Models that are able to combine those representations to do accurate predictions

How to learn good representations of words?

Idea 1: Hand-Crafted Features

➔ E.g. is this word a location? A verb? Is it a synonym with this other word?

How to learn good representations of words?

Idea 1: Hand-Crafted Features

➔ E.g. is this word a location? A verb? Is it a synonym with this other word?

Limits: Costly to collect, task-specific, etc…

How to learn good representations of words?

Idea 1: Hand-Crafted Features

➔ E.g. is this word a location? A verb? Is it a synonym with this other word?

Idea 2: Data-Drive Representations

➔ **Distributional Hypothesis**

"You shall know a word by the company it keeps" Firth (1957)

➔ Model the *context* of a word to build its vectorial representation

Example: What is the meaning of "Bardiwac" ?

- He handed her a glass of bardiwac.

(Evert & Lenci 2009)

Example: What is the meaning of "Bardiwac" ?

- He handed her a glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.

(Evert & Lenci 2009)

Example: What is the meaning of “Bardiwac” ?

- He handed her a glass of **bardiwac**.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.

(Evert & Lenci 2009)

Example: What is the meaning of "Bardiwac" ?

- He handed her a glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- I dined off bread and cheese and this excellent bardiwac

(Evert & Lenci 2009)

Example: What is the meaning of “Bardiwac” ?

- He handed her a glass of **bardiwac**.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- I dined off bread and cheese and this excellent bardiwac
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia’s sunshine.

(Evert & Lenci 2009)

Example: What is the meaning of "Bardiwac" ?

- He handed her a glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- I dined off bread and cheese and this excellent bardiwac
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia's sunshine.
- The drinks were delicious: blood-red bardiwac as well as light, sweet Rhenish.

Thanks to **the distributional hypothesis** we guess that bardiwac is... ?

(Evert & Lenci 2009)

The Distributional Hypothesis for NLP

A large number of methods have been designed

1. Count-Based Approaches: Build **co-occurrence matrices**

Limits: sparse vectors, do not generalize to new words

The Distributional Hypothesis for NLP

A large number of methods have been designed

1. **Count-Based Approaches:** Build **co-occurrence matrices**

2. **Word2vec models:** Continuous fixed vectors by learning to predict the context of words

The Distributional Hypothesis for NLP

A large number of methods have been designed

1. **Count-Based Approaches:** Build **co-occurrence matrices**

2. **Word2vec models:** Continuous fixed vectors by learning to predict the context of words

Limits: each word gets a single vector regardless of its context

e.g.: *I like **cherry pie**, This dress is **cherry** red*

The Distributional Hypothesis for NLP

A large number of methods have been designed

1. **Count-Based Approaches:** Build **co-occurrence matrices**

2. **Word2vec models:** Continuous fixed vectors by learning to predict the context of words

Limits: each word gets a single vector regardless of its context

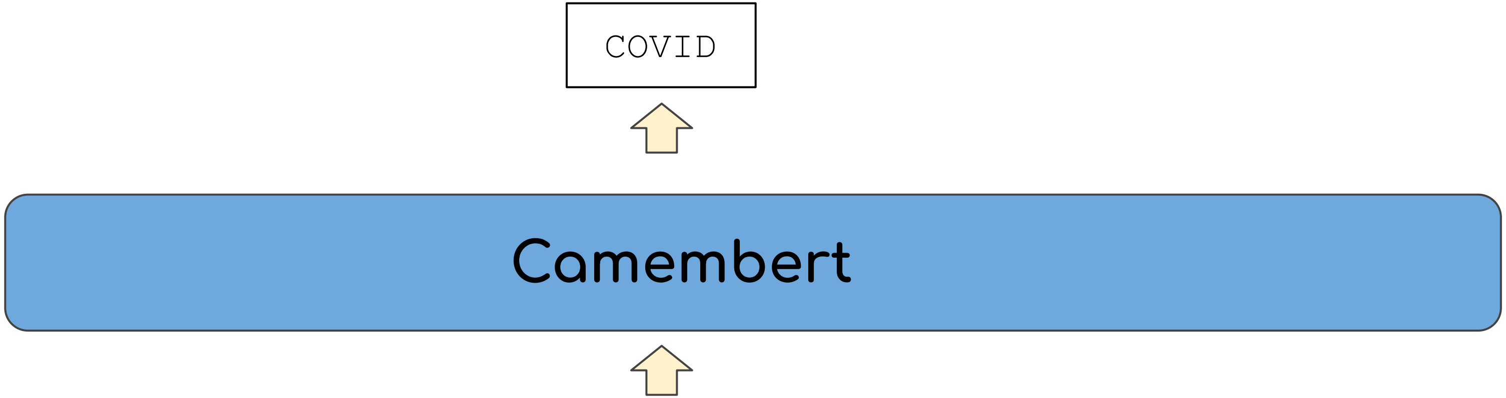
3. Contextualized Representation with Language Models

Outline

1. How did we get here? Intuition about BERT-like models
2. **The Camembert Model**
3. Camembert in the real-world
4. Beyond the pretraining-finetuning paradigm

Masked-Language Modeling

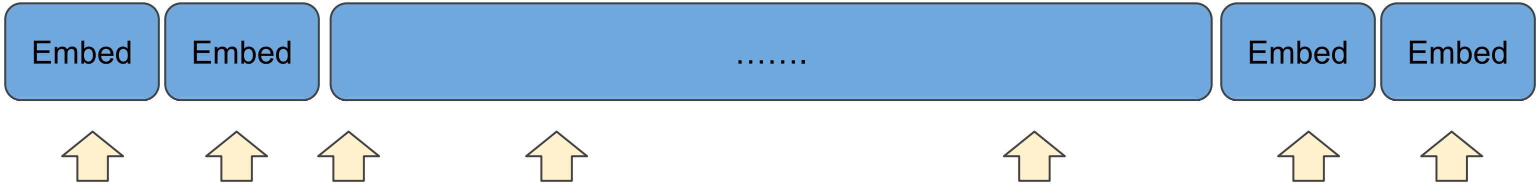
MLM consists in training a model **to guess a word** using both left and right context



`J'ai de nouveau attrapé le` **<mask>**`. Je l'ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.`

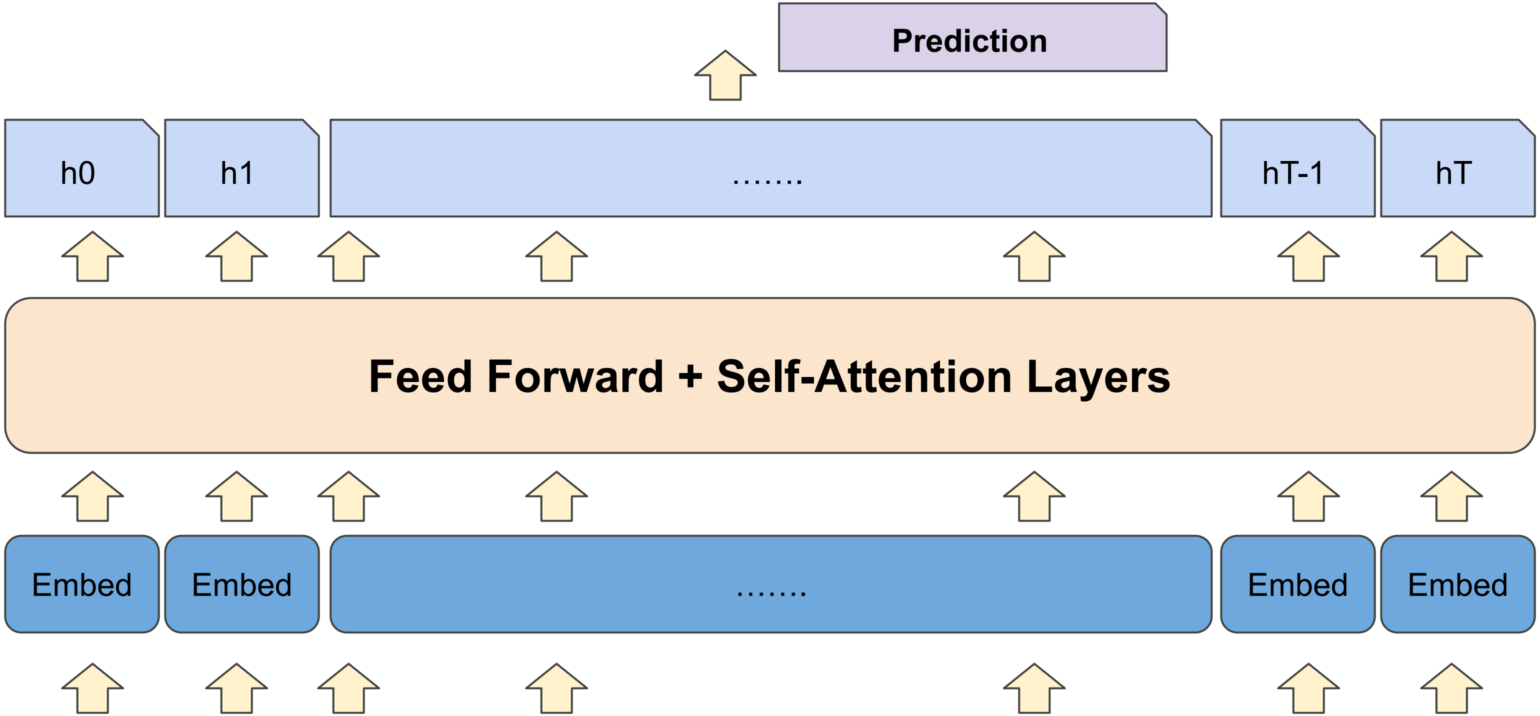
(source)

Parametrization: The Transformer Architecture



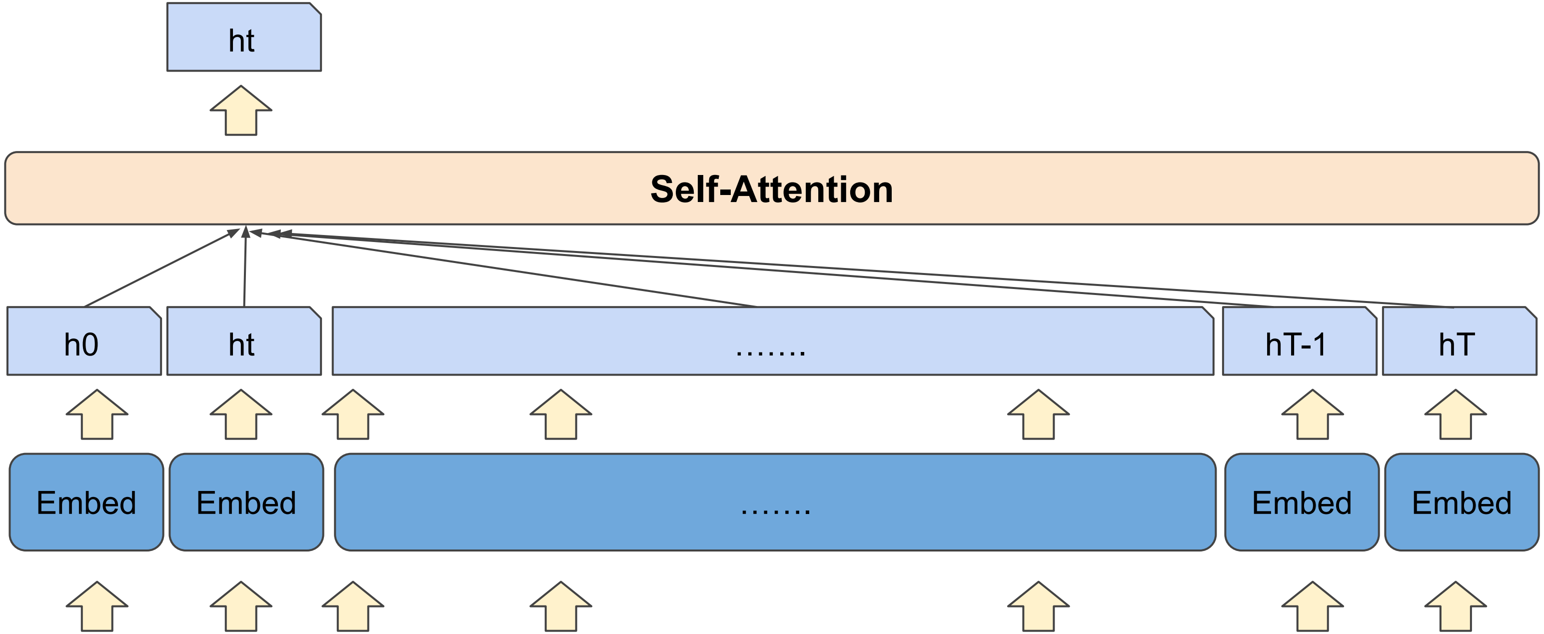
`J'ai de nouveau attrapé le` **<mask>**`. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.`

Parametrization: The Transformer Architecture



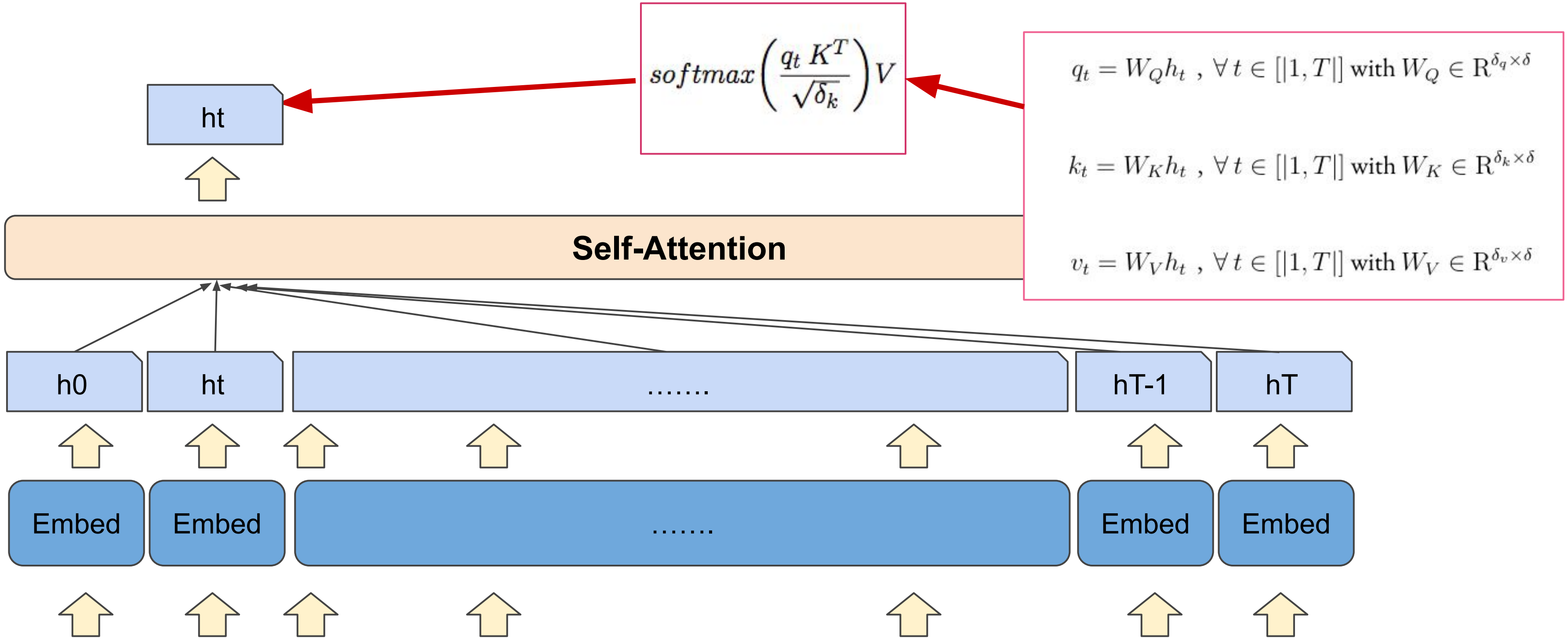
`J'ai de nouveau attrapé le` **<mask>**`. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.`

Parametrization: The Transformer Architecture



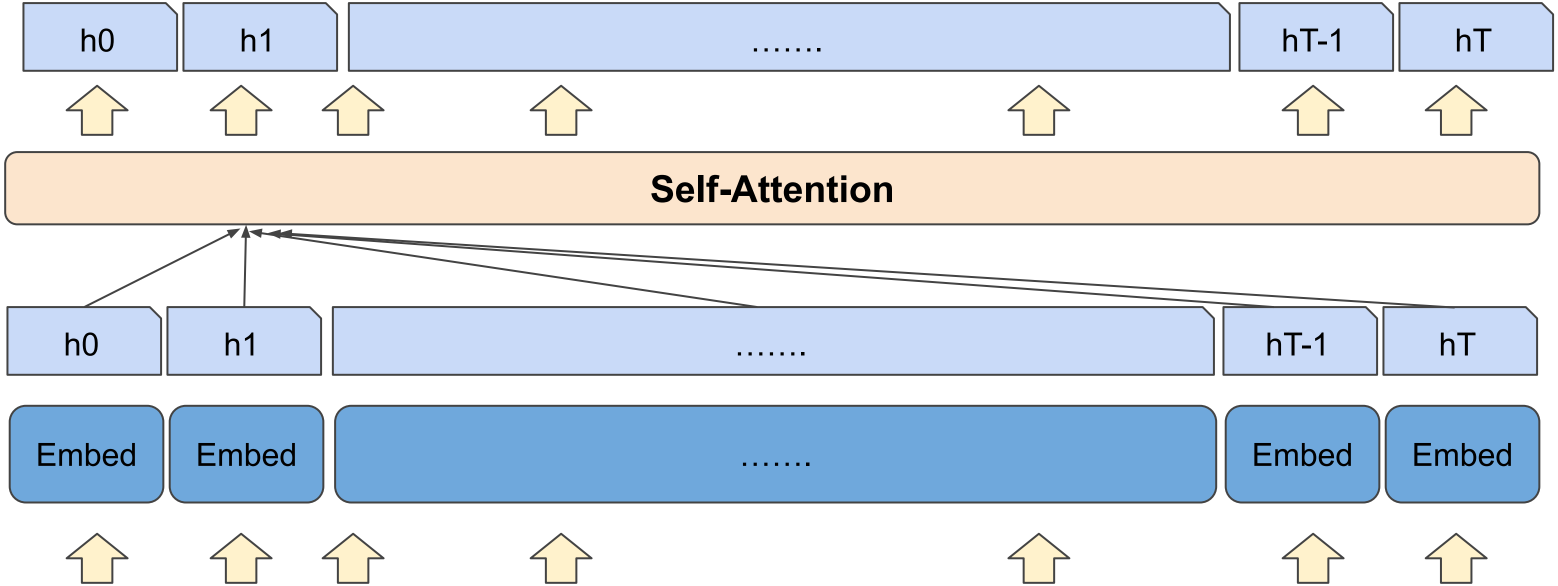
J'ai de nouveau attrapé le **<mask>**. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.

Parametrization: The Transformer Architecture



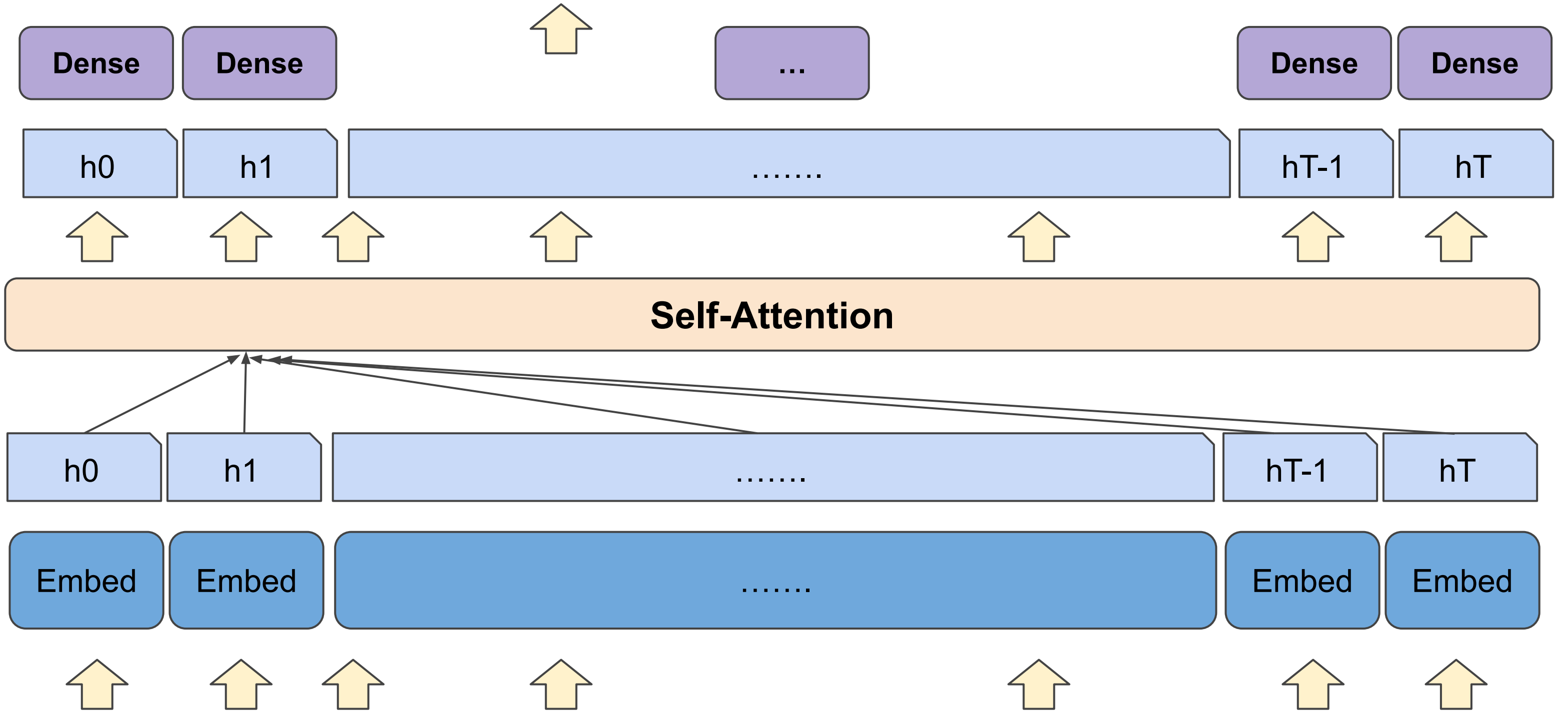
J'ai de nouveau attrapé le **<mask>**. Je l'ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.

Parametrization: The Transformer Architecture



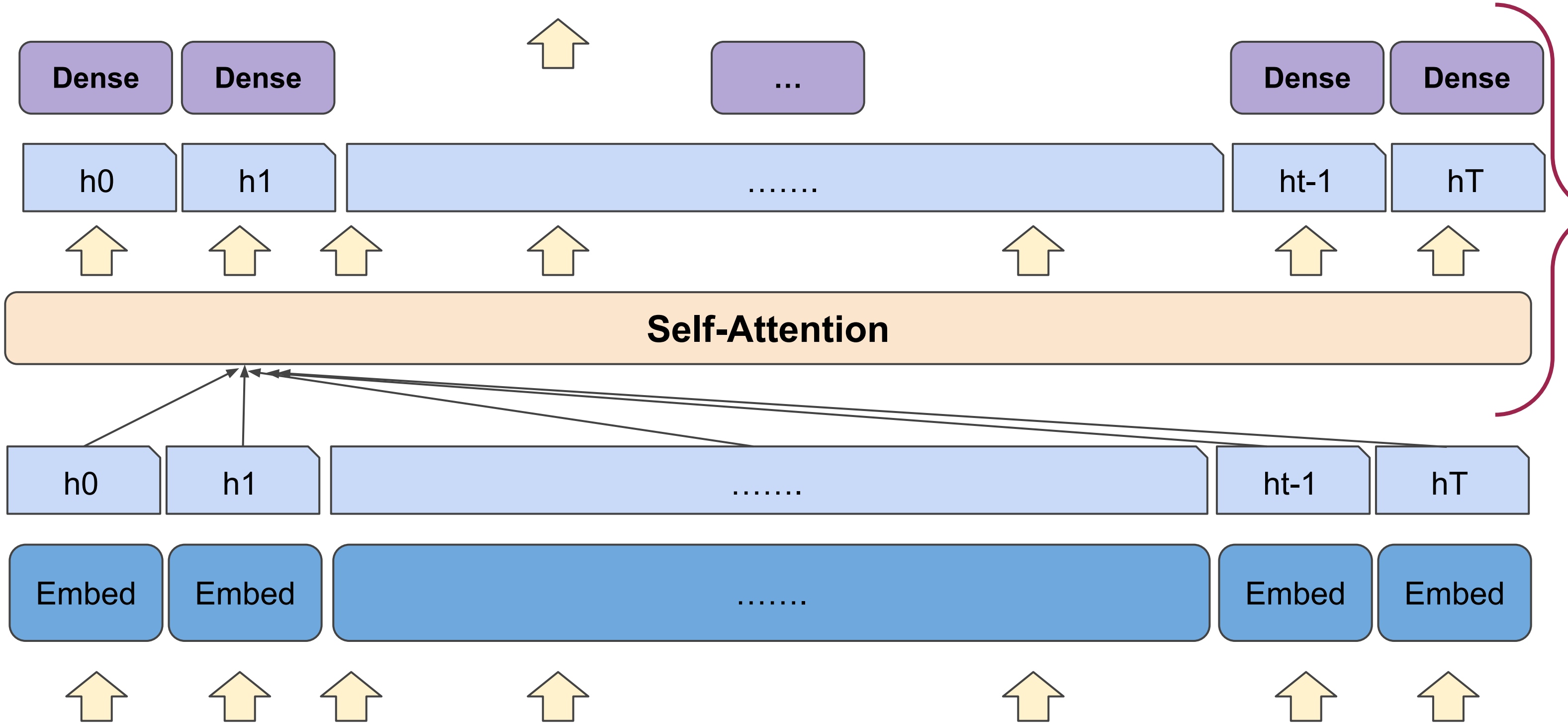
J'ai de nouveau attrapé le **<mask>**. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.

Parametrization: The Transformer Architecture



`J'ai de nouveau attrapé le` **<mask>**`. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.`

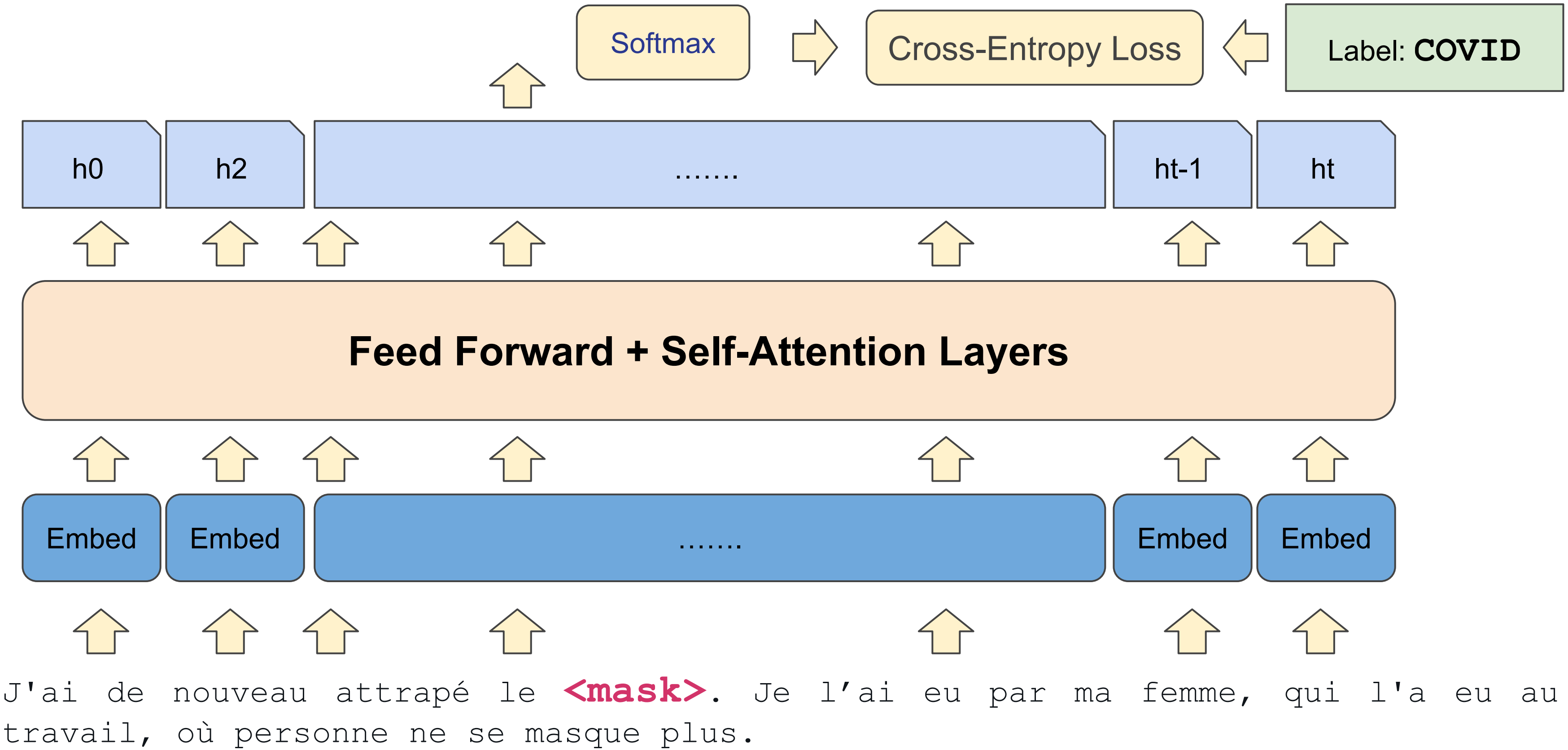
Parametrization: The Transformer Architecture



Repeated x12/24 for Camembert base/large

J'ai de nouveau attrapé le **<mask>**. Je l'ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.

Parametrization: The Transformer Architecture



Training Transformer

- Transformers are (usually) trained with Stochastic Gradient Descent (or variants like ADAM (Kingma et. al 2014))

- With **Cross-Entropy Loss**

- All the parameters are (usually) trained End-to-End

Outlook on Camembert

Camembert is a language model for French

Pretrained **on 138 GB** of **Web-Crawled Data** (OSCAR) in French

With **Masked Language Modeling**

It is parametrized with a **transformer architecture 12/24 layers (base/large)** leading to about **110M/335M** parameters

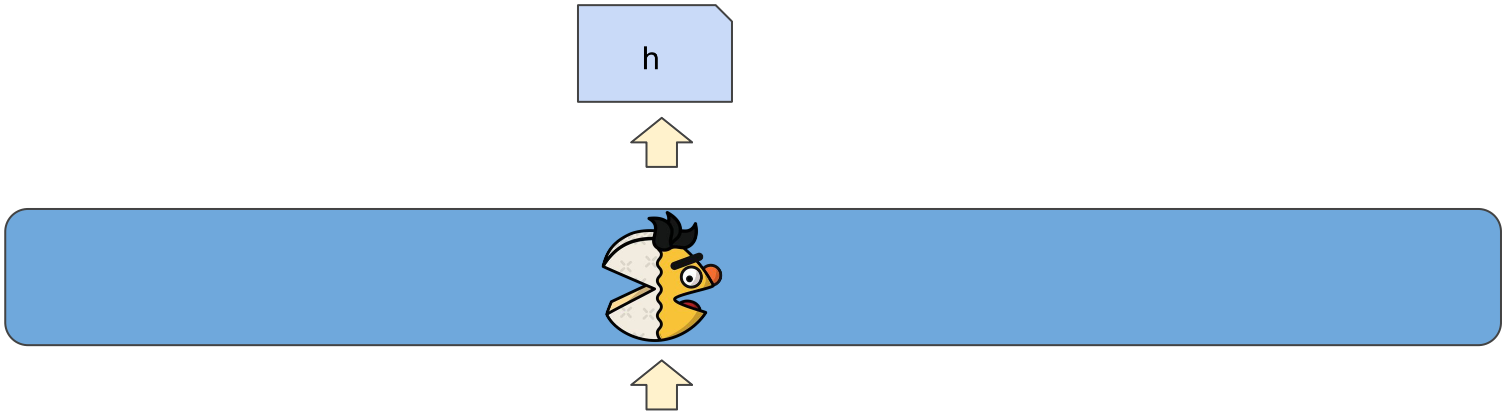
Outline

1. How did we get here? Intuition about BERT-like models
2. The Camembert Model
3. **Camembert in the real-world**
4. Beyond the pretraining-finetuning paradigm

How to use Camembert?

At test time, we can reuse the output vector

- **To represent input tokens (e.g. COVID)**
- **To perform specific tasks** (e.g. Sequence Labelling)



`J'ai de nouveau attrapé le` **COVID**`. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.`

Fine-tuning:

1. Re-use all the parameters of Camembert (except last layer)
2. Appending **a new dense layer** to get the right output space
3. Train end-to-end on the specific-task

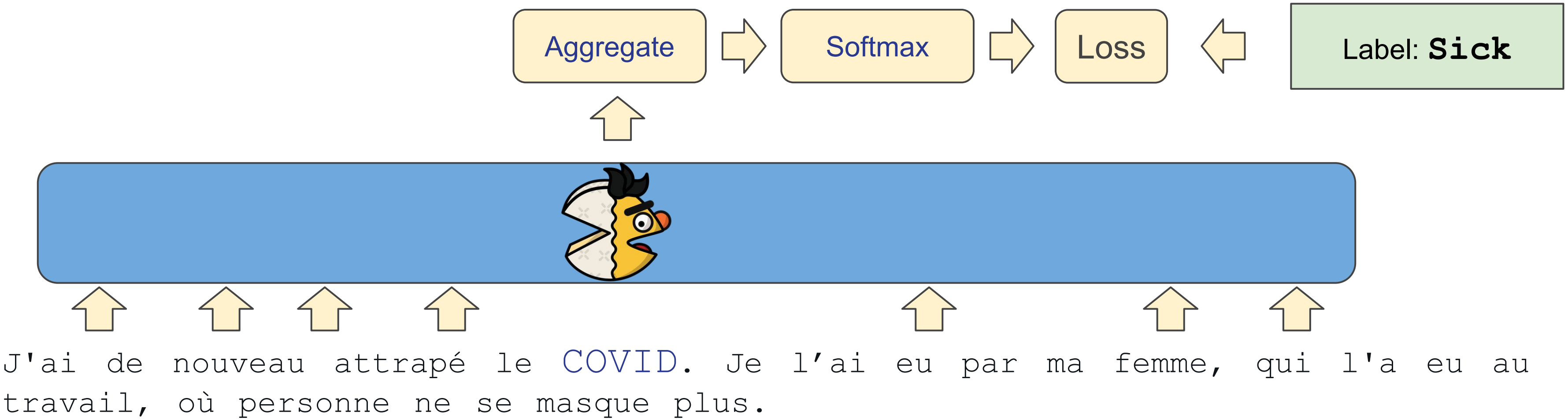
Fine-tuning: e.g. sequence classification

1. Re-use all the parameters of Camembert (except last layer)
2. Appending **a new dense layer** to get the right output space
3. Train end-to-end on the specific-task

J'ai de nouveau attrapé le COVID. Je l’ai eu par ma femme, qui l'a eu au travail, où personne ne se masque plus.

Fine-tuning: e.g. sequence classification

1. Re-use all the parameters of Camembert (except last layer)
2. Appending a new dense layer to get the right output space
3. Train end-to-end on the specific-task



Camembert Performance on standard tasks

After pretraining, we can reuse the entire camembert model and **fine-tune** it on our task

Model	F1
SEM (CRF) (Dupont, 2017)	85.02
LSTM-CRF (Dupont, 2017)	85.57
mBERT (fine-tuned)	87.35
CamemBERT (fine-tuned)	89.08
LSTM+CRF+CamemBERT (embeddings)	**89.55**

Named-Entity Recognition

	FQuAD1.1-test		FQuAD1.1-dev	
Model	F1	EM	F1	EM
Human Perf.	91.2	75.9	92.1	78.3
$\text{CamemBERT}_{\text{BASE}}$	88.4	78.4	88.1	78.1
$\text{CamemBERT}_{\text{LARGE}}$	**92.2**	**82.1**	**91.8**	**82.4**
$\text{FlauBERT}_{\text{BASE}}$	77.6	66.5	76.3	65.5
$\text{FlauBERT}_{\text{LARGE}}$	80.5	69.0	79.7	69.3
mBERT	86.0	75.4	86.2	75.5
$\text{XLM-R}_{\text{BASE}}$	85.9	75.3	85.5	74.9
$\text{XLM-R}_{\text{LARGE}}$	89.5	79.0	89.1	78.9

Question-Answering

Adapting Camembert to New Domains

Camembert was trained on a great diversity of domains (138 GB of Web Crawled text)

+its sentencepiece tokenization make it robust to infrequent words

However, for specific domains it can be helpful to run some adaptation step

How to Adapt Camembert?

Adapting Camembert to New Domains

Camembert was trained on a great diversity of domains (138 GB of Web Crawled text)

+its sentencepiece tokenization make it robust to infrequent words

However, for specific domains it can be helpful to run some adaptation step

How to Adapt Camembert?
➔ Don't stop pretraining (Gururangan et. al 2020)

Masked-Language Modeling Adaptation

For domains for which we have a good amount of raw data

- Keep training the model with the MLM Objective
- Only a few thousands sentences are enough

Masked-Language Modeling Adaptation

For domains for which we have a good amount of raw data

- Keep training the model with the MLM Objective
- Only a few thousands sentences are enough

Domain	Task	RoBERTa	Additional Pretraining Phases DAPT	TAPT	DAPT + TAPT
BioMed	ChemProt	$81.9_{1.0}$	$84.2_{0.2}$	$82.6_{0.4}$	$\mathbf{84.4}_{0.4}$
	†RCT	$87.2_{0.1}$	$87.6_{0.1}$	$87.7_{0.1}$	$\mathbf{87.8}_{0.1}$
CS	ACL-ARC	$63.0_{5.8}$	$75.4_{2.5}$	$67.4_{1.8}$	$\mathbf{75.6}_{3.8}$
	SciERC	$77.3_{1.9}$	$80.8_{1.5}$	$79.3_{1.5}$	$\mathbf{81.3}_{1.8}$
News	HyperPartisan	$86.6_{0.9}$	$88.2_{5.9}$	$\mathbf{90.4}_{5.2}$	$90.0_{6.6}$
	†AGNews	$93.9_{0.2}$	$93.9_{0.2}$	$94.5_{0.1}$	$\mathbf{94.6}_{0.1}$
Reviews	†Helpfulness	$65.1_{3.4}$	$66.5_{1.4}$	$68.5_{1.9}$	$\mathbf{68.7}_{1.8}$
	†IMDB	$95.0_{0.2}$	$95.4_{0.1}$	$95.5_{0.1}$	$\mathbf{95.6}_{0.1}$

➔ Leads to significant improvement (Gururangan et. al 2020)

➔ Works even for new languages (Muller et. al 2019, 2021)

Outline

1. How did we get here? Intuition about BERT-like models
2. The Camembert Model
3. Camembert in the real-world
4. **Beyond the pretraining-finetuning paradigm**

Beyond Pretraining-Fine-tuning

Limits of Pretraining Fine-tuning:

- It requires a lot (>1GB) of raw data for pretraining
- Enough annotated data for fine-tuning (1k-100k> samples)

Beyond Pretraining-Fine-tuning

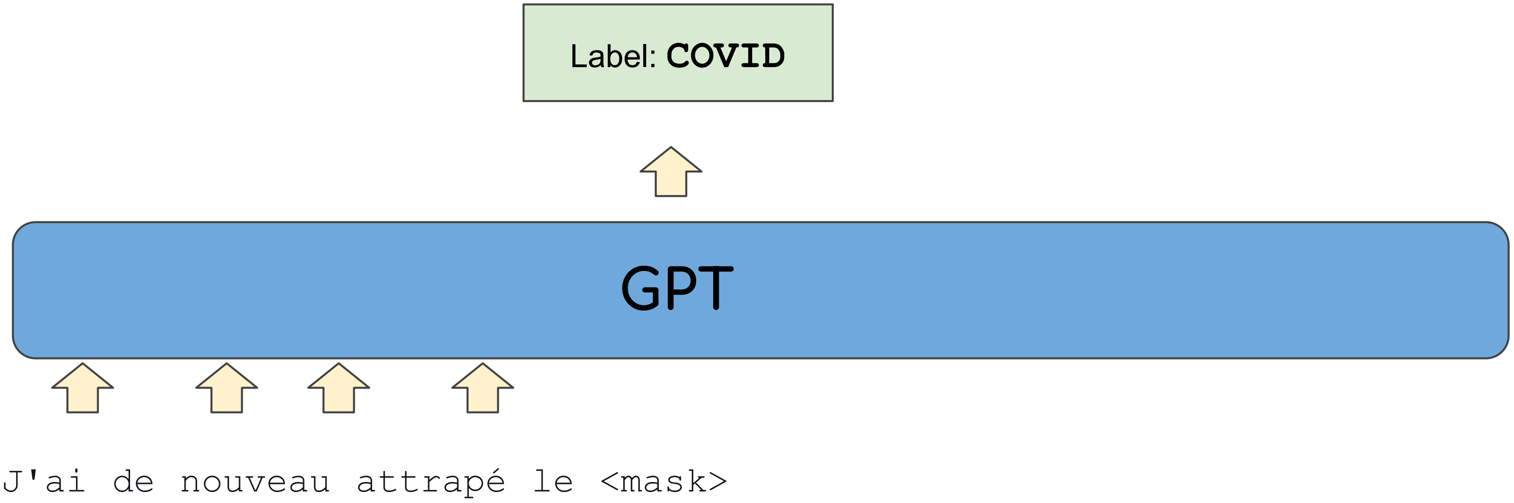
Limits of Pretraining Fine-tuning:

- It requires a lot (>1GB) of raw data for pretraining
- Enough annotated data for fine-tuning (1k-100k> samples)

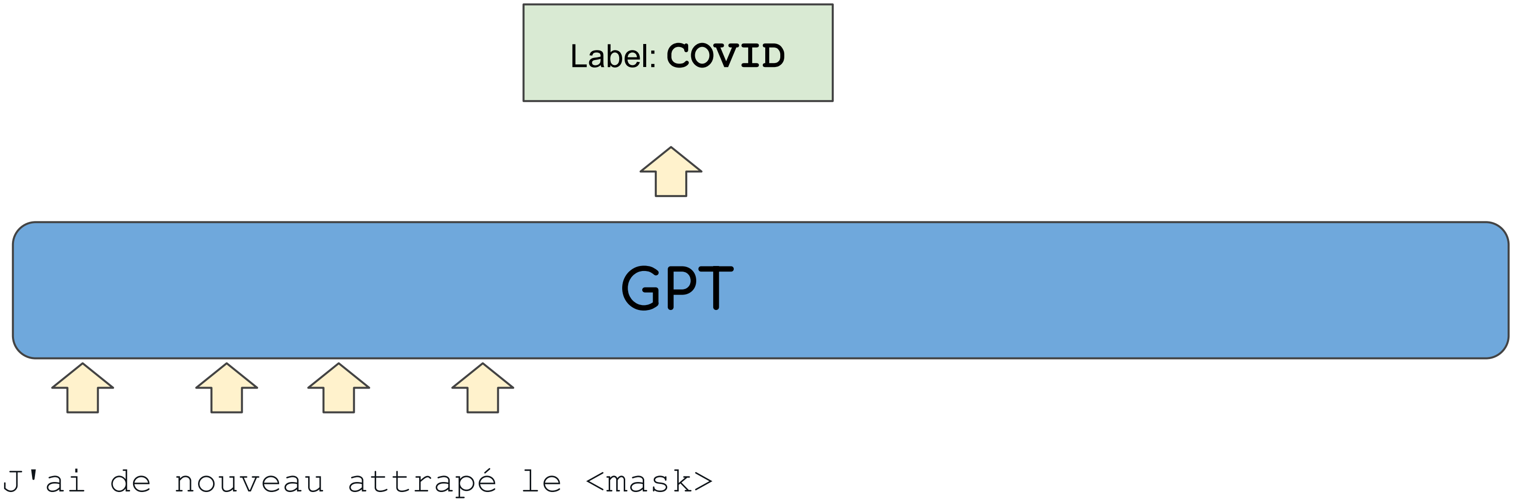
➔ This excludes low-resource languages (i.e. thousands of languages (Joshi et. al 2020))

➔ This makes generalization costly for new tasks

Generative Model (GPT)



Generative Model (GPT)



NB: Instead of MLM, we do left-to-right decoding

In-Context Learning with 10B+ Parameters LM

Generative Language Models (GPT-like) exhibits new learning behavior at scale:

In-context learning: e.g. for Question Answering:

- No gradient and parameter updates
- Only feeding examples to the generative model and predicting next tokens

In-Context Learning with 10B+ Parameters LM

Generative Language Models (GPT-like) exhibits new learning behavior at scale:

In-context learning: e.g. for Question Answering:

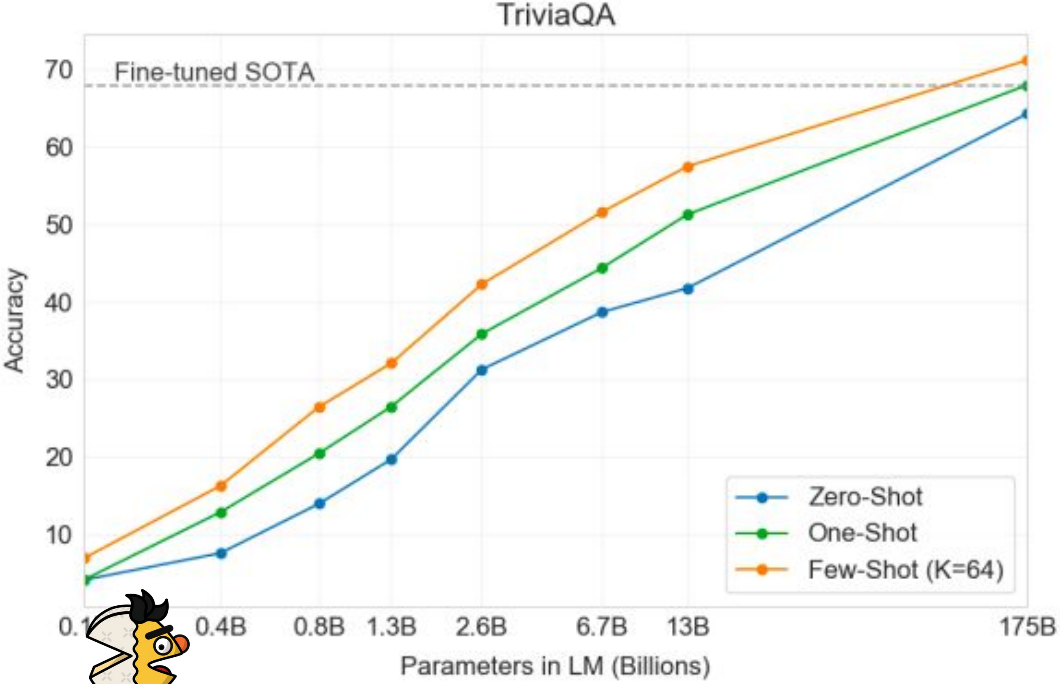
Excerpt: Callan Pinckney was an American fitness professional. She achieved unprecedented success with her Callanetics exercises. Her 9 books all became international best-sellers and the video series that followed went on to sell over 6 million copies. Pinckney's first video release "Callanetics: 10 Years Younger In 10 Hours" outsold every other **fitness** video in the US.

Question: American Callan Pinckney's eponymously named system became a best-selling (1980s-2000s) book/video franchise in what genre?

Answer: Fitness

In-Context Learning with 10B+ Parameters LM

Generative Language Models (GPT-3) exhibits new learning behavior at scale:



(Brown et. al 2020)

However: in the real-world

For less complex tasks with “enough” annotated data:

Pretraining-Fine-tuning remains the most flexible, easy-to-use and reasonably cheap approach

Can still be improved with

➔ **Multimodal** (vision, speech, text) Approaches
➔ **More parameter efficient pretraining** / fine-tuning

Thank you!

Sentencepiece

Tokenization is the first step of everything we do in NLP
It consists in segmenting raw text to define **our modeling units (tokens)**

E.g.: *Il faut que tu trouves l' adresse de l' entreprise de plasturgie*

★ Word level segmentation → **Limit: Out-of-Vocabulary Problem**

★ Character-Level Segmentation → **Limit: Too Long Sequences**

★ **Sentencepiece → Tradeoff between both approaches**
Segment at the word-level except for infrequent words that are segmented at the subword level
['_Il','_faut','_que','_tu','_trouve','s','_l',"'",'_adresse','_de','_l',"'",'_entreprise','_de','_','plast','ur','gie']