LLMS ARCHITECTURE & FINE TUNING

آجي تفهم AI

SOFIA AGOURRAM

AKA “THE AI GIRL”

CEO AIMPACTIFY MAROC

SOFTWARE & DATA ENGINEER

AGENDA

1- LARGE LANGUAGE MODELS

2- TRANSFORMERS ARCHITECTURE

3- GENERATIVE AI PROJECT

4- WHAT IS FINE TUNNING

5- FINE TUNING PROCESS

6- FINE TUNING USING HUGGINGFACE

CHAPTER 1

LARGE LANGUAGE MODELS

LARGE LANGUAGE MODELS

شنو هما؟

AI systems trained on vast text data to understand and generate language.

شنو هما قدراتهم ؟

Writing, answering questions, coding, and more.

أمثلة ديال LLMS

GPT series, BERT.

WHY LLMS?

كاين إنفجار ديال البيانات

Abundance of data has provided the raw material for training

كاين تقدم كبير فالقدرت الحسابية

Hardware (e.g., GPUs, TPUs) and software (e.g., TensorFlow, PyTorch)

التقدم في أبحاث الذكاء الاصطناعي

Innovations in neural network design, such as transformer architectures

TRIGGERS FOR LLM DEV

- **Limitations of Previous Models**
- **Success of Transformer Architecture**
- **The Push for AI that Understands and Generates Human-like Text.**

TRANSFORMERS

An architecture popularized by the "Attention is All You Need" paper.

شكرا Google !

TRANSFORMERS

Allow direct connections between any two elements, capturing long-range dependencies more effectively.

Utilize attention mechanisms to weigh the importance of different elements in the sequence.

(Attention is all you need ! Remember ??)

PROS OF TRANSFORMERS

- Parallel processing (Ability to handle larger models and datasets,efficient utilization of computational resources..)
- Attention to Input meaning
- Scale efficiently

كا ينين !!

CHAPTER 2

TRANSFORMERS ARCHITECTURE

TRANSFORMERS

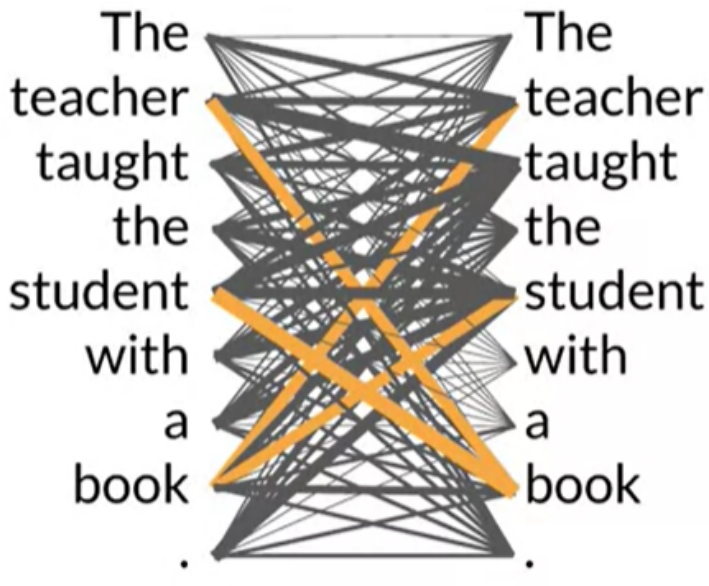
The transformers revolutionized the field of natural language processing (NLP) and became the basis for the LLMs we now know - such as GPT.

و اليوم ندخلو بشكل مبسط

In their architecture that led to an explosion in regenerative capability.

THE POWER OF TRANSFORMERS ARCHITECTURE

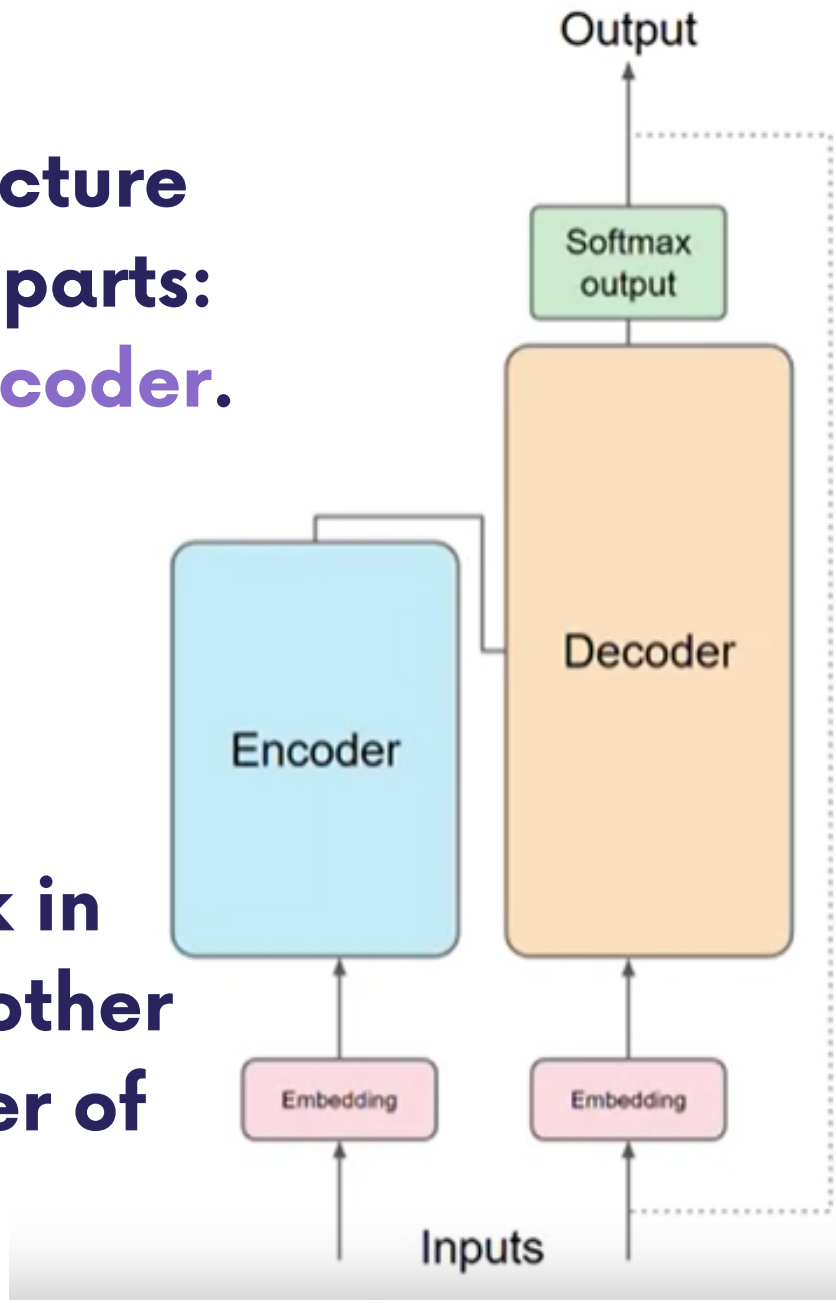
The power of the transformer architecture lies in its ability to learn the relevance and context of all of the words in a sentence.



The model learns the relevance of each word to each other words no matter where they are in the input.

SIMPLIFIED ARCHITECTURE

The transformer architecture is split into two distinct parts: the Encoder and the Decoder.



These components work in conjunction with each other and they share a number of similarities.

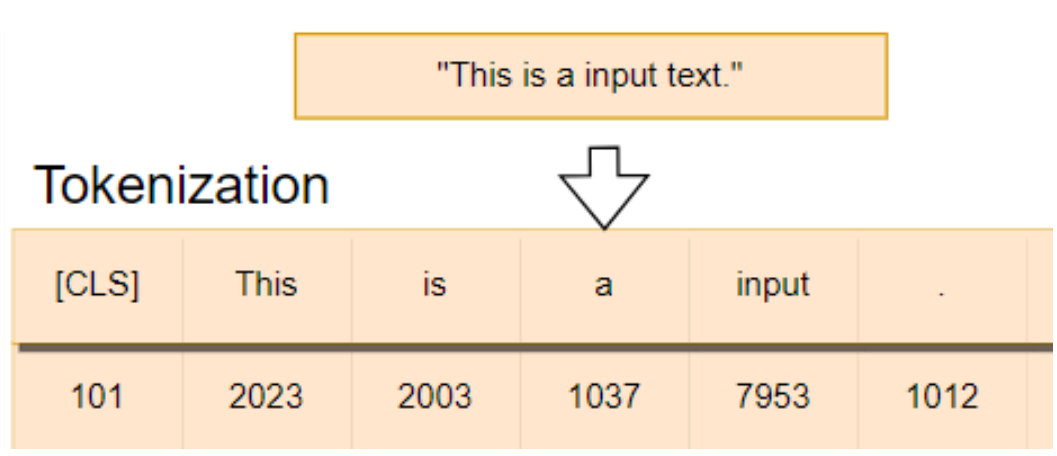
DIFFERENT COMPONENTS & STEPS

- Tokenization
- Vector embedding
- Positional encoding
- Self-attention layer
- Feed-forward network
- Softmax layer

#1- TOKENIZATION

بشكل مبسط tokenization هي

Converting the words into numbers, with each number representing a position in a dictionary of all the possible words that the model can work with.

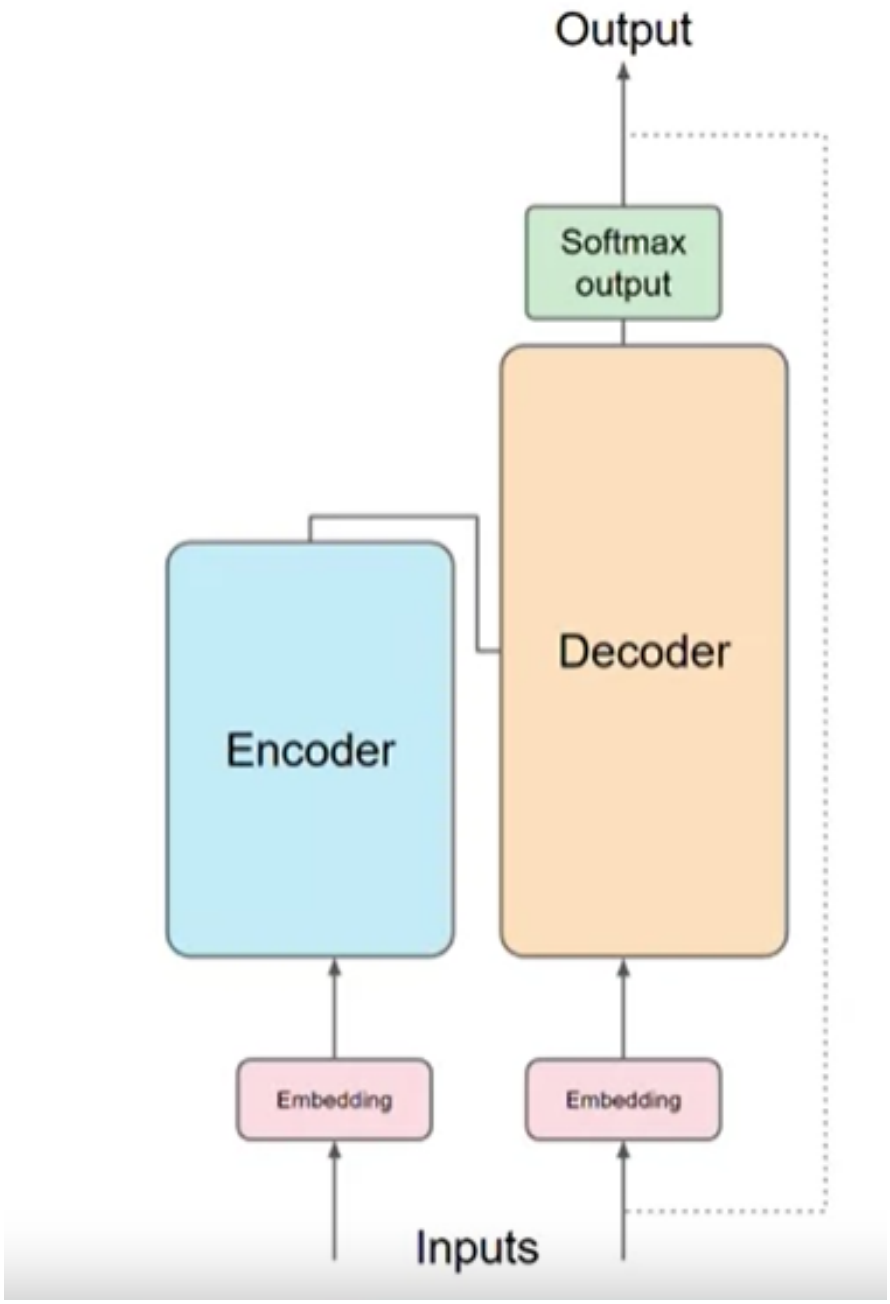


علاش كنحتاجوها ؟

Machine-learning models are just big statistical calculators and they work with numbers.

#2- VECTOR EMBEDDING SPACE

A high-dimensional space where each token is represented as a vector and occupies a unique location within that space



#3- POSITIONAL ENCODING

Normally, the model processes each of the input tokens in parallel.

the positional encoding هادشي علاش كنحتاجو

باش نحفاظو على :

the word order and don't lose the relevance of the position of the word in the sentence.

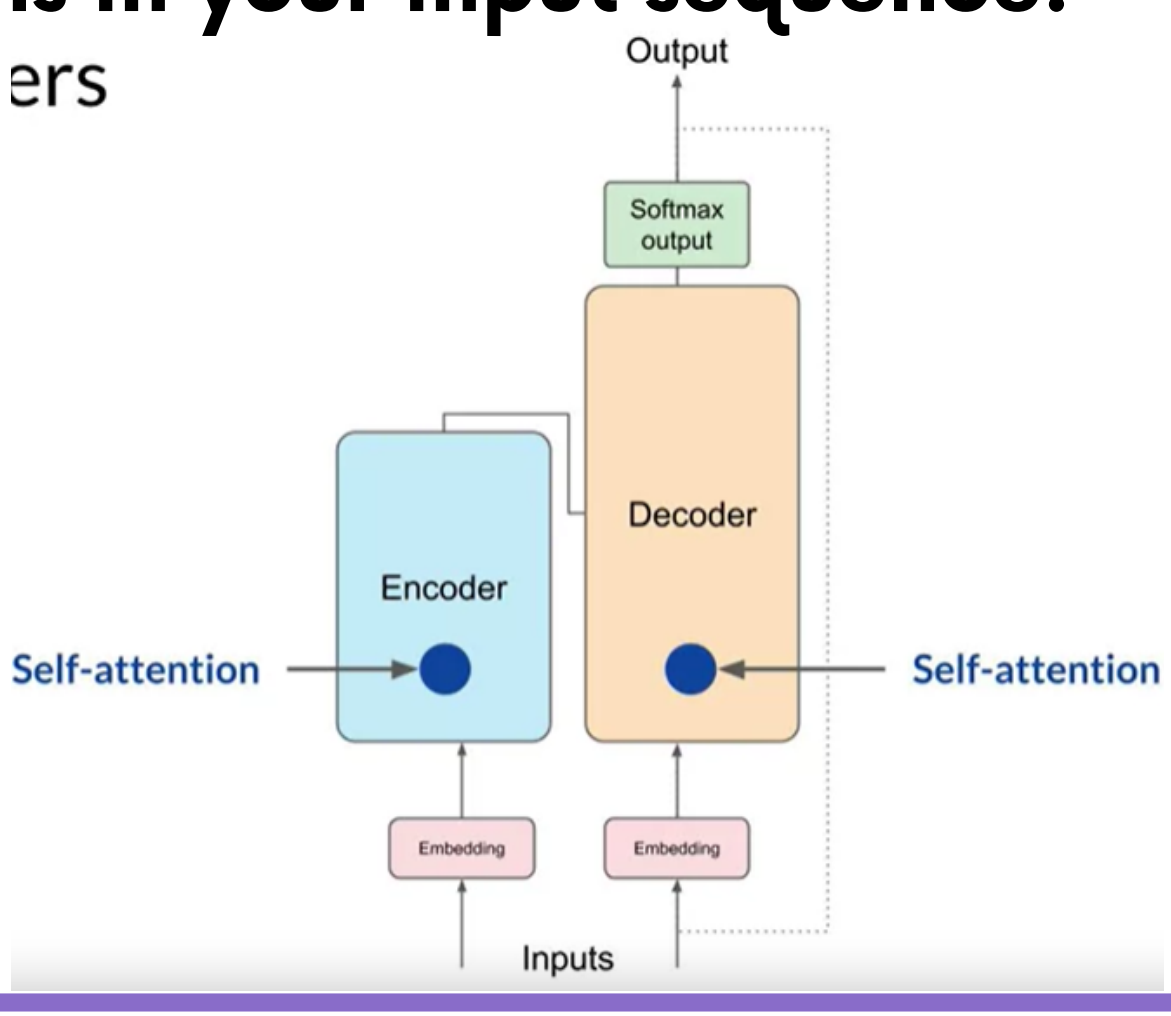
Et Voilà!

Self-Attention Layer كتصيفط كلشي مقاد ل

#4- SELF-ATTENTION LAYER

دابا كايبدا لمعقول !!

We pass the resulting vectors to the self-attention layer. Here, the model analyzes the relationships between the tokens in your input sequence.



#5- MULTI-HEADED SELF-ATTENTION

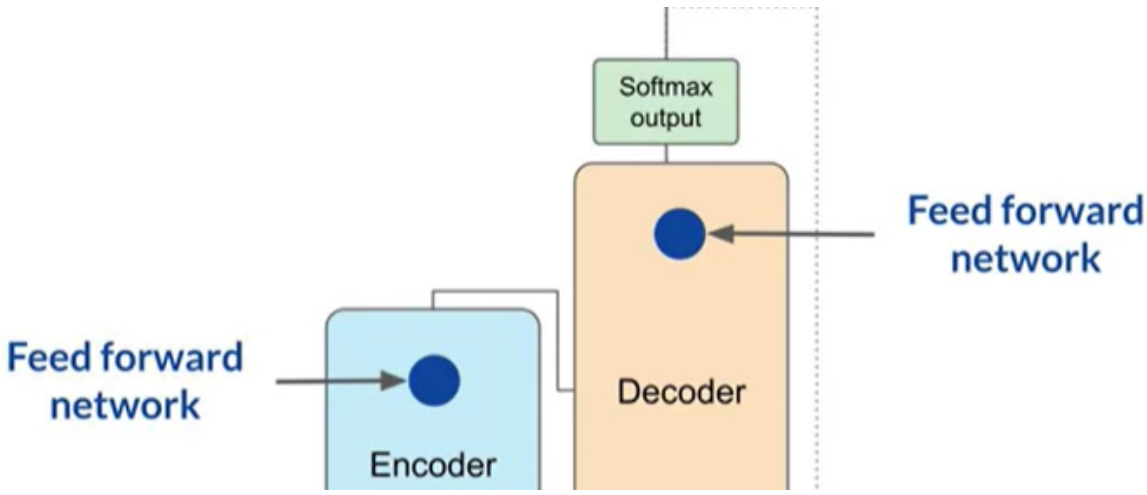
ولكن هذا لا يحدث دقة واحدة !!

Multiple sets of self-attention weights or heads are learned in parallel independently of each other.

The number of attention heads included in the attention layer is 12-100.

Each will learn different aspects of language.

#6- FEED-FORWARD NETWORK



The output of this layer is a vector of logits proportional to the probability score for each and every token in the tokenizer dictionary.

#7- SOFTMAX LAYER

و هنا وصلنا آخر مرحلة

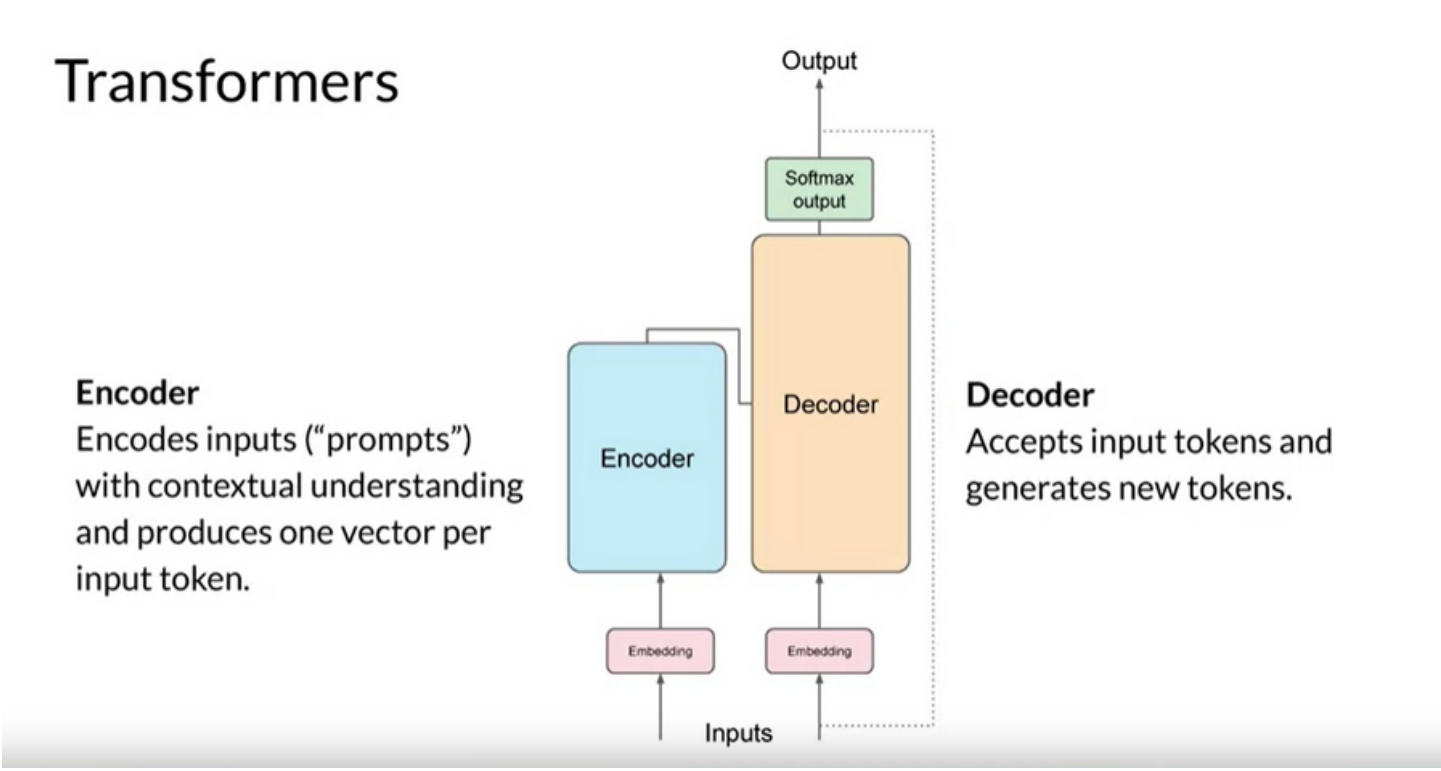
It is the final layer that converts the model's output into a probability distribution.

It ensures that the model's predictions sum to 1, allowing it to select the most likely class or token based on the learned probabilities.

One single token will have a score higher than the rest. This is the most likely predicted token.

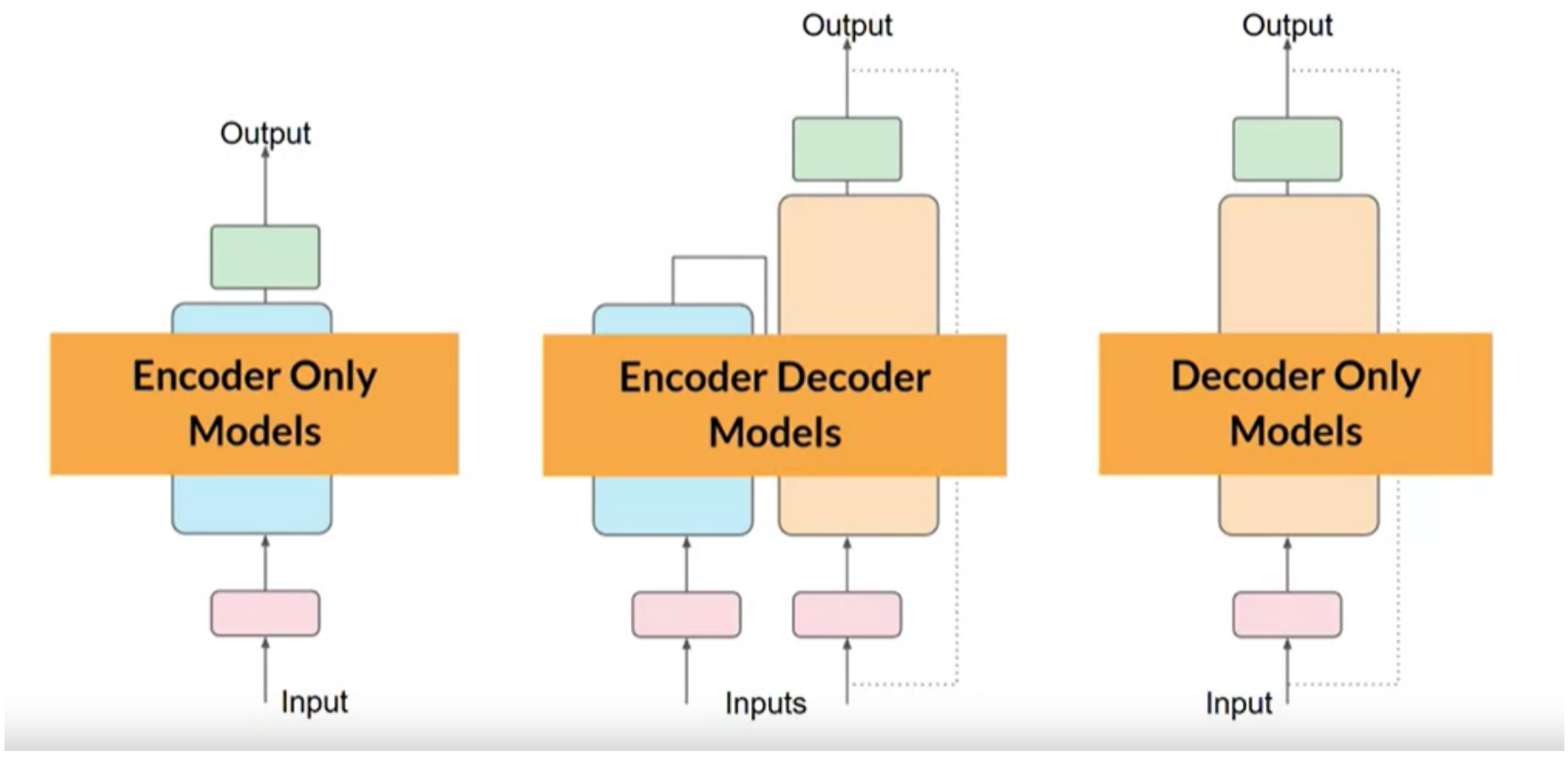
TRANSFORMERS

the global architecture دابا شفنا



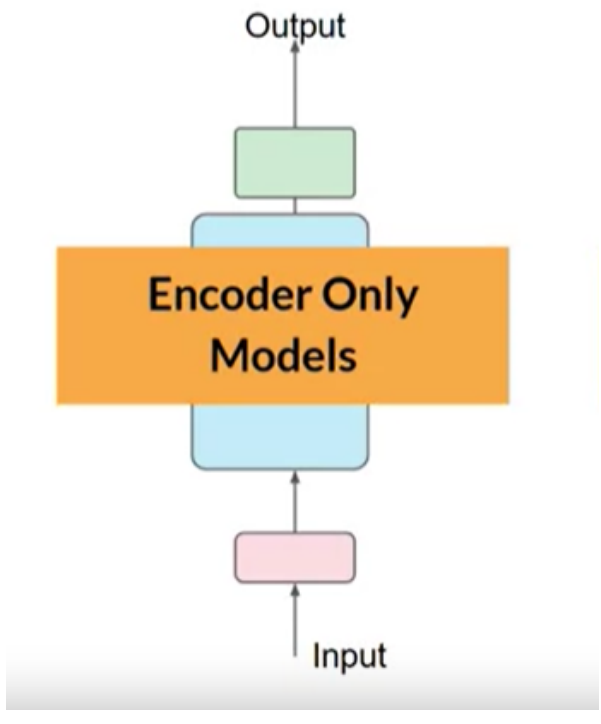
VARIETY OF TRANSFORMERS

و لكن راه كاينين بزاف ديال Models



VARIETY OF TRANSFORMERS

Encoder Only (AutoEncoding models)

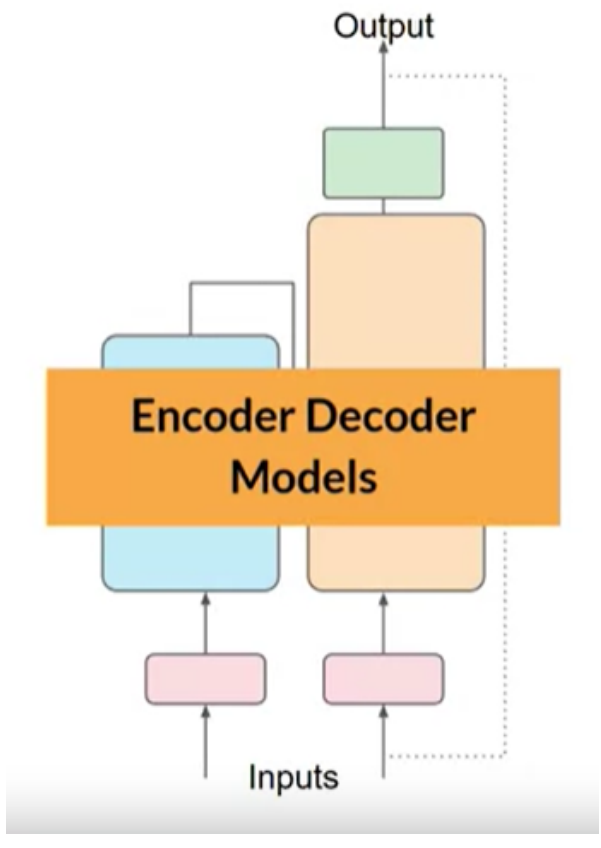


Use Cases:
Sentiment Analysis
Word Classification

Example : BERT

VARIETY OF TRANSFORMERS

Encoder Decoder(Sequence to sequence)



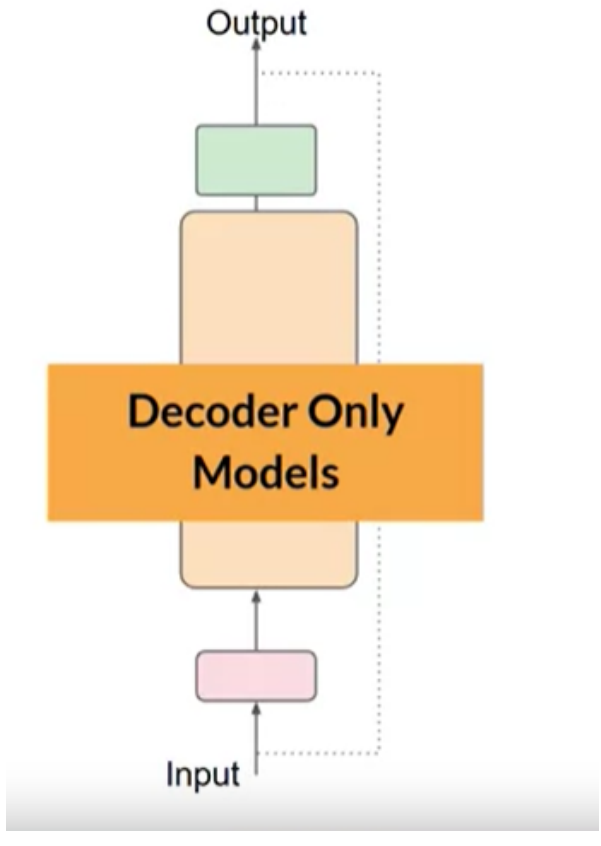
Use Cases:
Translation
Text summarization
Question Answering

Example : BART, T5

VARIETY OF TRANSFORMERS

Decoder Only (AutoRegressive Models)

Objective : Predict the next token



Use Cases:
Text Generation

Example : GPT

CHAPTER 3

GENERATIVE AI PROJECT

فراسك تقدر تقاد واحد

Generative AI project

ولكن خاصك تبع هاد :

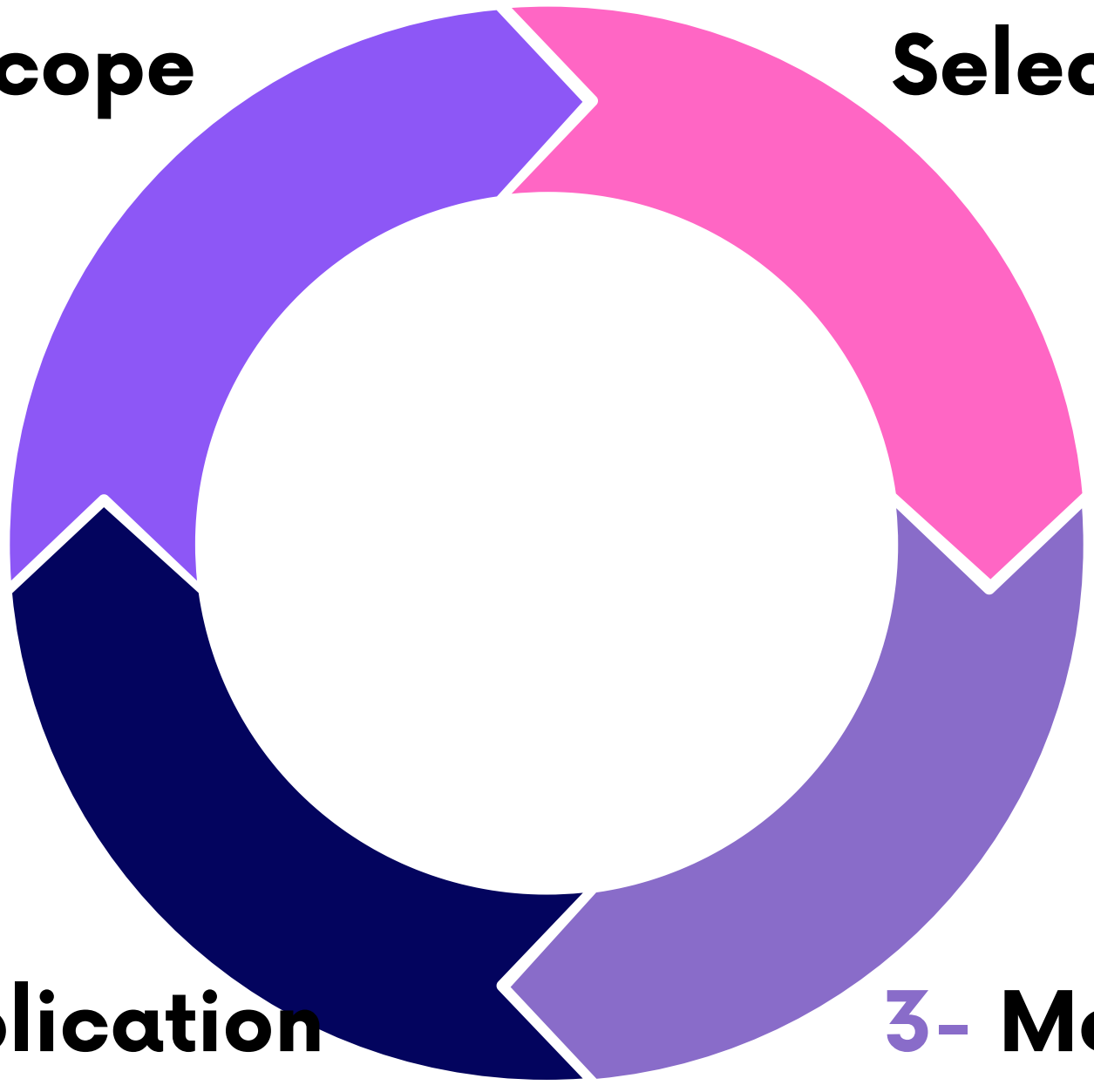
4 Core Principles of the lifecycle

وتحاول تفهم المراحل كاملة

PROJECT LIFECYCLE

1 - Define the Scope

2 - Model Selection



4 -Application Integration

3- Model Alignment

STEP 1: DEFINE THE PROJECT'S SCOPE

- **Identify the specific task or challenge you want to address using the Generative AI.**

- **Clearly outline the objectives and desired outcomes for the project.**

تعرف مزيان الإحتياجات ديالك و النِطَاق ديال المشروع. و ماتنساش !!

- **Consider the dataset availability, project complexity, and target audience.**

STEP 2 :MODEL SELECTION

هنا عندك الإختيار ما بين:

1- Pretraining an existing model

2- Training a model from scratch

بالطبع على حسب:

Your project requirements.

ما تخافوش راه كاينين بزاف ديال:

Open-Source pretrained models available for free usage such us:

- StableLM, Pythia, Falcon AI, LLaMa (by Meta), LaMDA (by Google) .....

STEP 3 : MODEL ALIGNMENT

- **Customize the chosen LLM model to align with the project's specific requirements.**

- **Create or choose suitable prompts that help guide the model's responses**

دابا Model واجد و لكن ماتنساش:

- **Continuously evaluate the model's performance and make adjustments as needed.**

STEP 4 : APPLICATION INTEGRATION

- Integrate the model into the target application or platform for seamless user experience.
  ... web أو mobile يقدر يكون تطبيق

- Deploy the AI-powered application and monitor its performance in real-world scenarios.

CHAPTER 4

FINE TUNING

Some people are confused
about
RAG & fine tuning

ومع ذلك فهما مفهومان مختلفان!!

WHAT IS RAG?

Retrieval-Augmented Generation

It was introduced by Facebook AI in 2020, merges retrieval-based and generative NLP techniques.
It efficiently retrieves information from vast documents and generates responses, enhancing answer relevance and accuracy.

WHAT IS FINE-TUNING?

Fine-tuning adjusts pre-trained models on a specific dataset to tailor the model's performance to a particular task or domain.

KEY DIFFERENCES

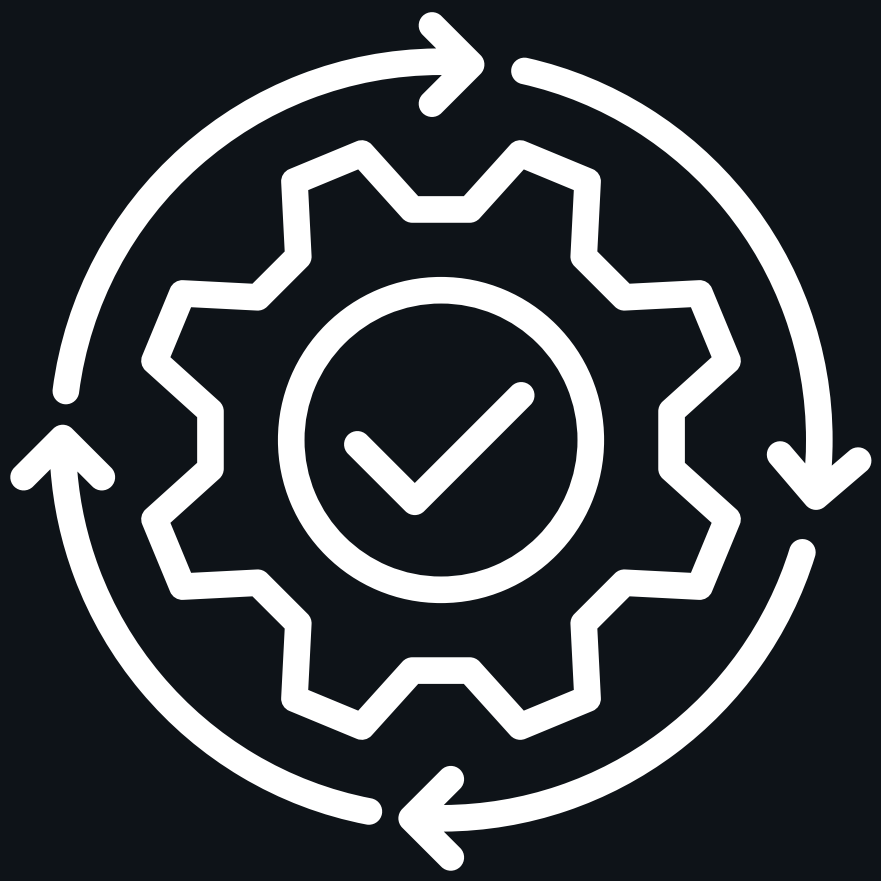
RAG leverages external knowledge dynamically and is great for applications requiring up-to-date information or broad knowledge. While fine-tuning relies on learning from the specific data provided during the training phase. It suits specialized tasks needing domain-specific understanding.

MODEL CUSTOMIZATION

RAG : allows dynamic integration of external knowledge, offering flexibility in updating information without retraining.

Fine-tuning: provides deep customization to a model's responses based on the training dataset, requiring retraining for updates.

CHAPTER 5



PROCESS OF FINE TUNING

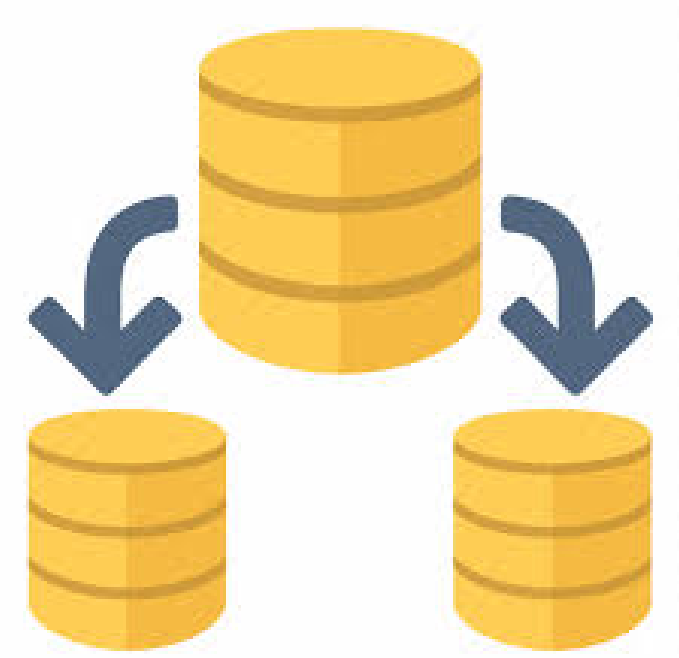
SELECTING A MODEL FOR FINE-TUNING

Choose a pre-trained model that closely aligns with your target task or domain. Consider factors like model size, language, and initial training data.

DATA PREPARATION

Your dataset should be representative of the task at hand. It's crucial to clean and preprocess your data, including tokenization and normalization, to match the model's expected input format.

TRAINING, VALIDATION, AND TEST SETS



Splitting the dataset appropriately to ensure the model is trained effectively and evaluated accurately.

HYPERPARAMETER TUNING

Fine-tuning requires careful selection of hyperparameters. The learning rate is particularly important; too high a rate can lead to rapid divergence, while too low a rate can slow down the learning process.

Experiment with different settings to find the optimal configuration.

EVALUATION

Use a separate validation set to monitor the model's performance during fine-tuning.
This helps in adjusting hyperparameters and avoiding overfitting.
Common metrics include:
accuracy, F1 score, and perplexity, depending on the task.

CHAPTER 5

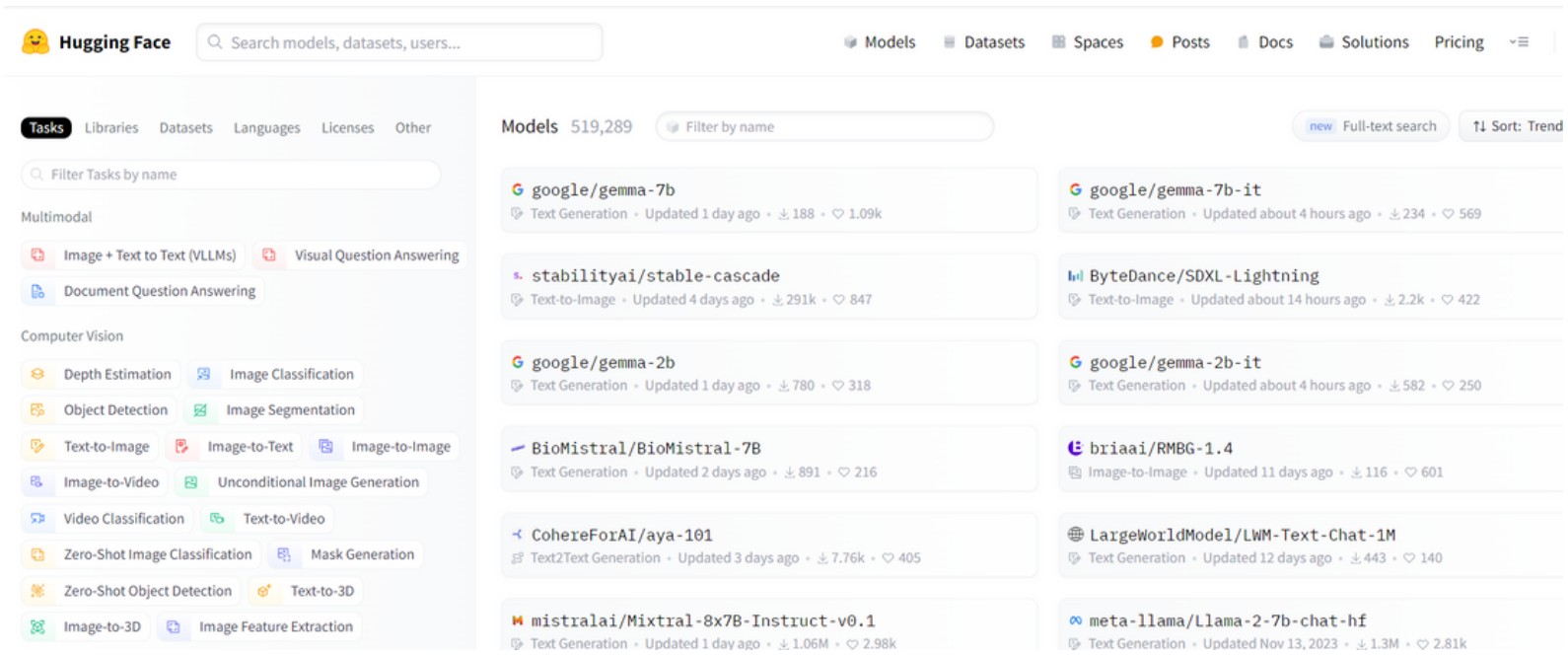
FINE TUNING USING HUGGING FACE

WHAT IS HUGGINGFACE

Hugging Face

Hugging Face offers a comprehensive platform for working with Large Language Models (LLMs) like GPT, BERT, and others, providing several free resources and tools for AI practitioners.

WHAT HUGGING FACE OFFERS FOR FREE:



Transformers Library

Datasets Library

Model Hub

Spaces

HOW TO USE HUGGING FACE FOR FINE-TUNING

1- Select a Model (pre-trained model from the Model Hub)

2- Prepare Your Data (Use the Datasets library to find relevant data or upload your dataset)

3- Fine-Tune the Model (Use the Transformers library to fine-tune the selected model on your dataset.)

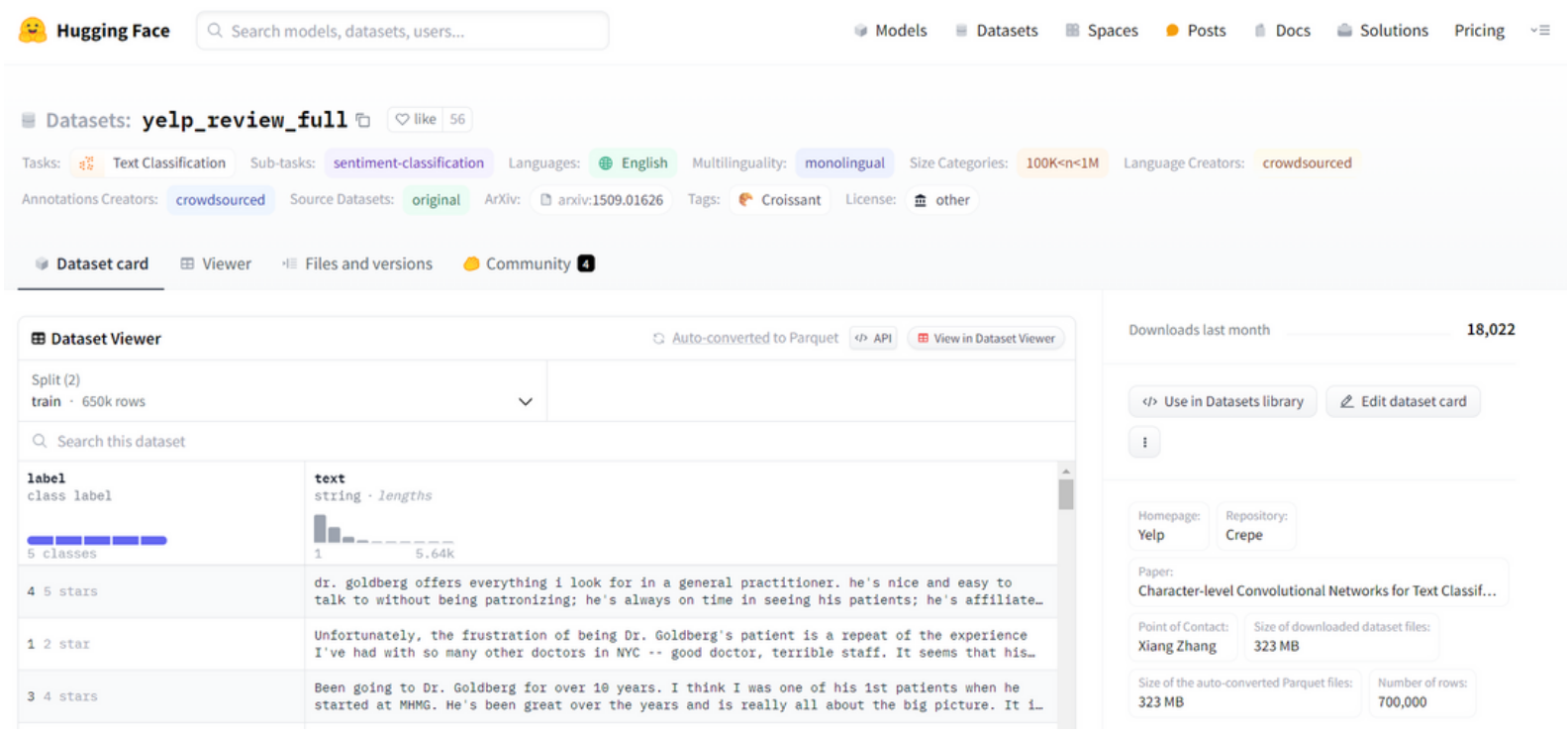
4- Evaluate and Share

EXAMPLE : SENTIMENT ANALYSIS

Pretrained Model to use : BERT

Bidirectional Encoder Representations from Transformers (BERT) is a language model based on the transformer architecture.
It was introduced in October 2018 by researchers at Google.

EXAMPLE : SENTIMENT ANALYSIS



DataSet to use : Yelp

Yelp an American company that publishes crowd-sourced reviews about businesses.

EXAMPLE : SENTIMENT ANALYSIS

Framework to use : TenserFlow & Keras

TensorFlow is an open-source ML library developed by Google. It provides a comprehensive ecosystem of tools, libraries, and community resources. Keras is now TensorFlow's high-level API for building and training deep learning models.

EXAMPLE : SENTIMENT ANALYSIS

Link to the article with explanation:

https://huggingface.co/docs/transformers/en/training

THANKS FOR YOUR TIME :)

Q & A !

AI By Sophia