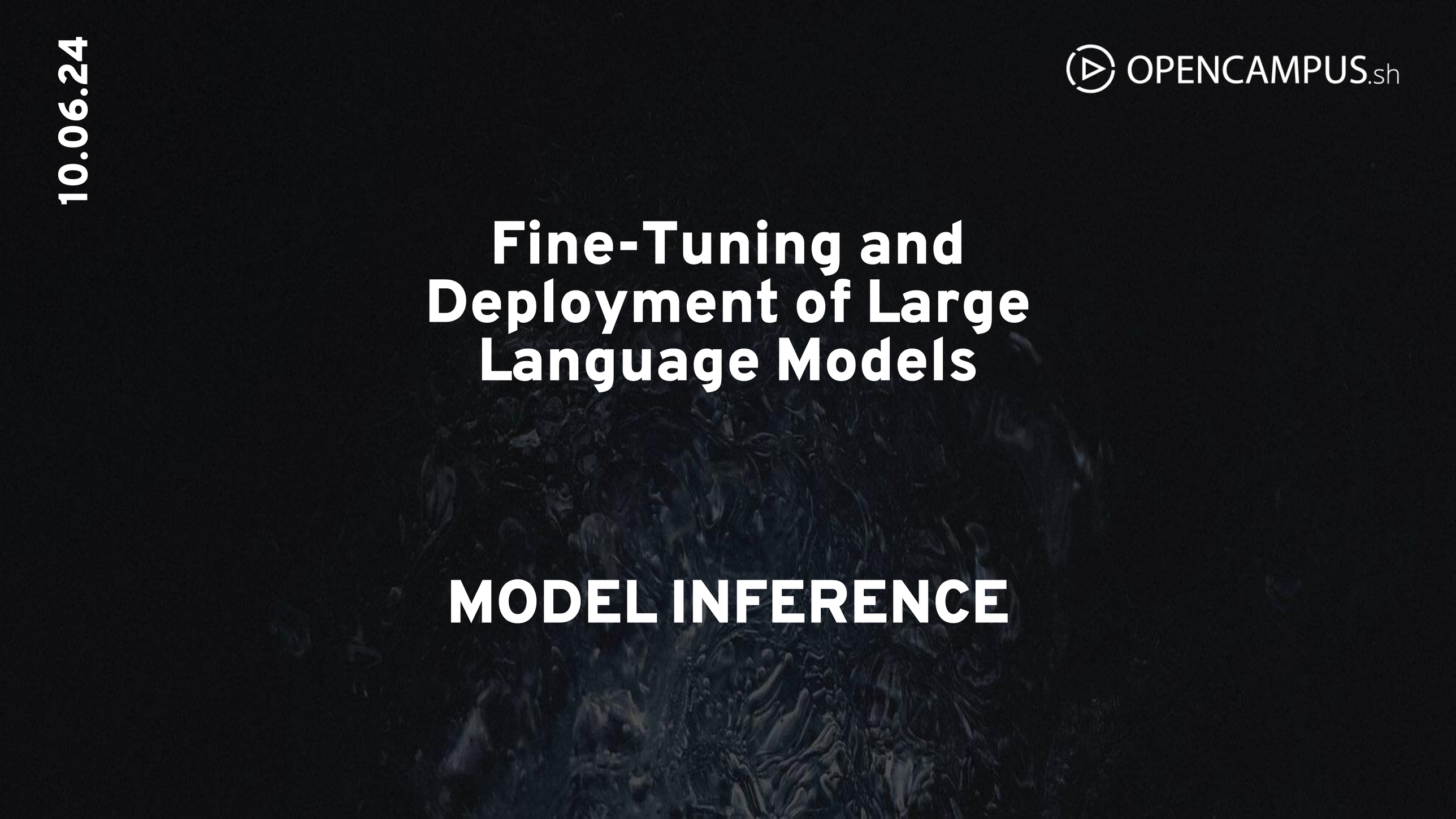
10.06.24

OPENCAMPUS.sh

Fine-Tuning and Deployment of Large Language Models

MODEL INFERENCE



- News
- Efficient Inference
- Hardware Options for Fine-Tuning and Inference
- Project Discussions
- Project Presentation and Documentation

NEWS

- Who is doing the news section next week?

EFFICIENT INFERENCE

Longformer: The Long-Document Transformer

Iz Beltagy* **Matthew E. Peters*** **Arman Cohan***
Allen Institute for Artificial Intelligence, Seattle, WA, USA
{beltagy,matthewp,armanc}@allenai.org

Abstract

Transformer-based models are unable to process long sequences due to their self-attention operation, which scales quadratically with the sequence length. To address this limitation, we introduce the Longformer with an attention mechanism that scales linearly with sequence length, making it easy to process documents of thousands of tokens or longer. Longformer's attention mechanism is a drop-in replacement for the standard self-attention and combines

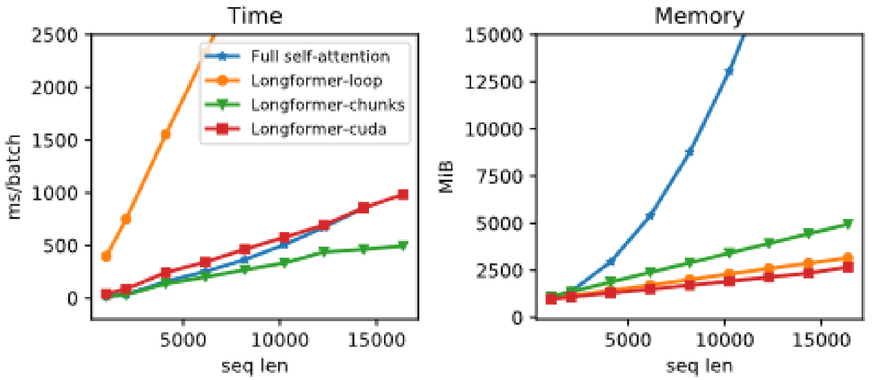
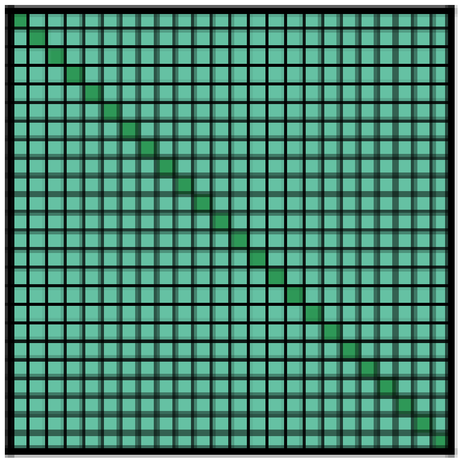
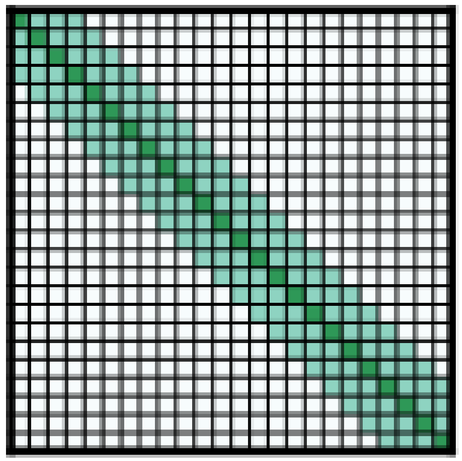


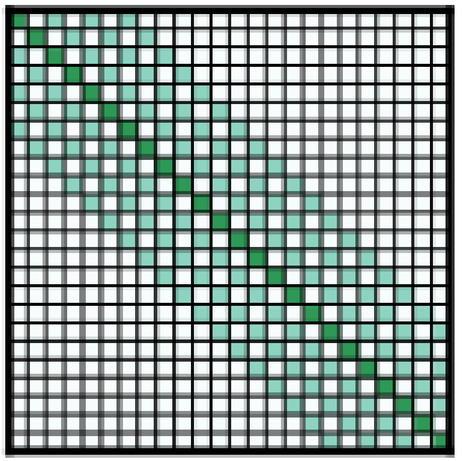
Figure 1: Runtime and memory of full self-attention and different implementations of Longformer's self-attention; `Longformer-loop` is non-



(a) Full n^2 attention

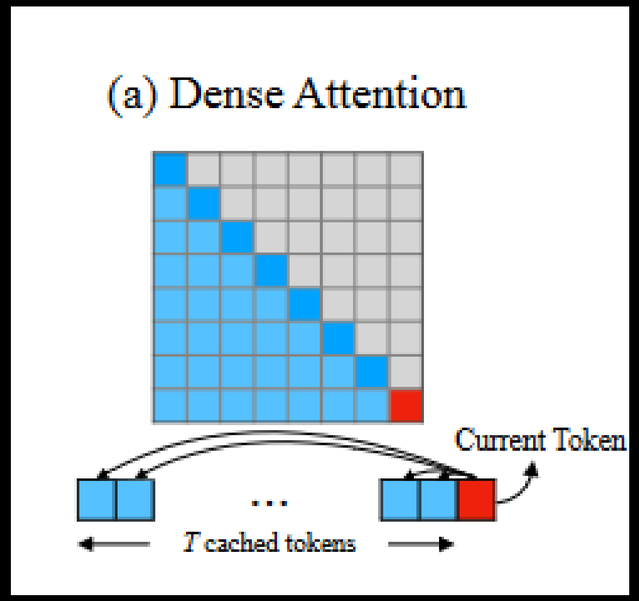


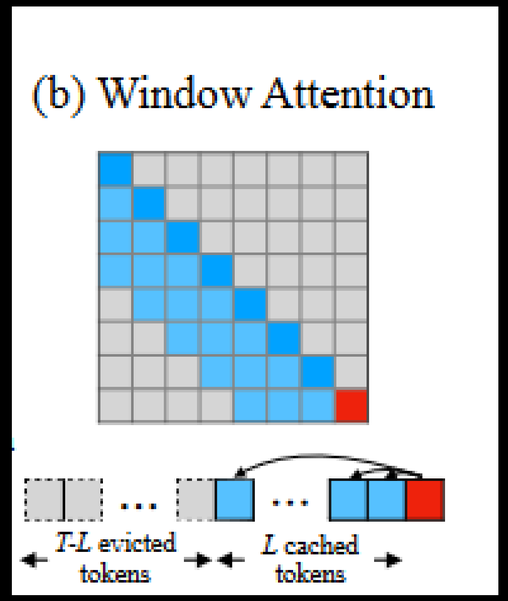
(b) Sliding window attention

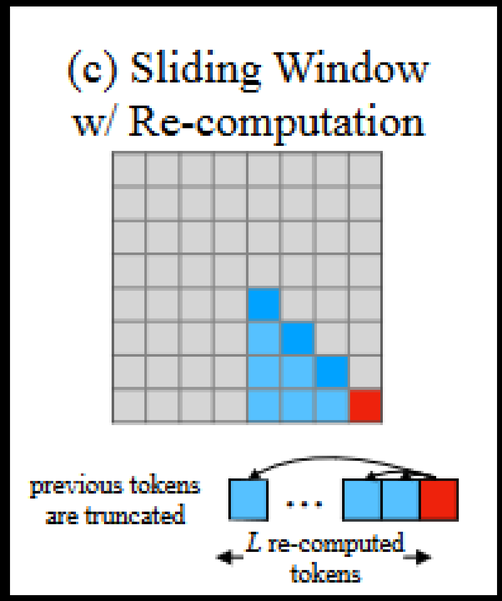


(c) Dilated sliding window

APPROACHES FOR EFFICIENT STREAMING OF LLMS







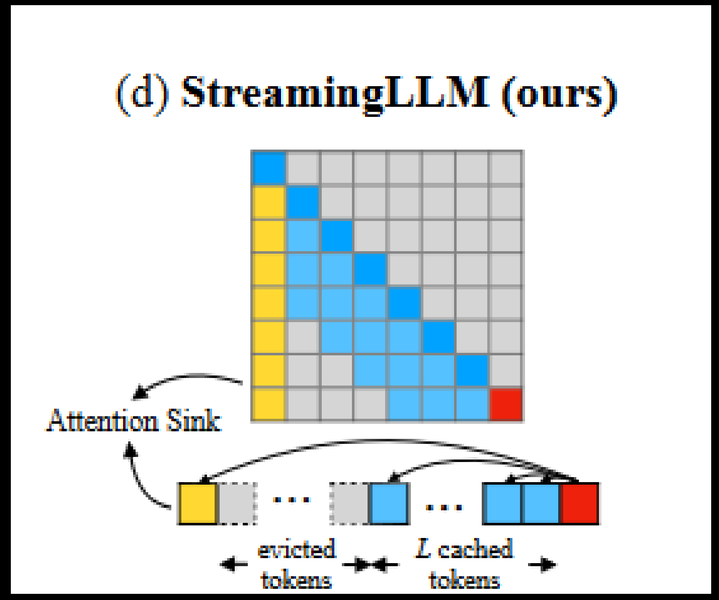
EFFICIENT STREAMING LANGUAGE MODELS WITH ATTENTION SINKS

Guangxuan Xiao[1*] **Yuandong Tian**[2] **Beidi Chen**[3] **Song Han**[1,4] **Mike Lewis**[2]

[1] Massachusetts Institute of Technology [2] Meta AI
[3] Carnegie Mellon University [4] NVIDIA
https://github.com/mit-han-lab/streaming-llm

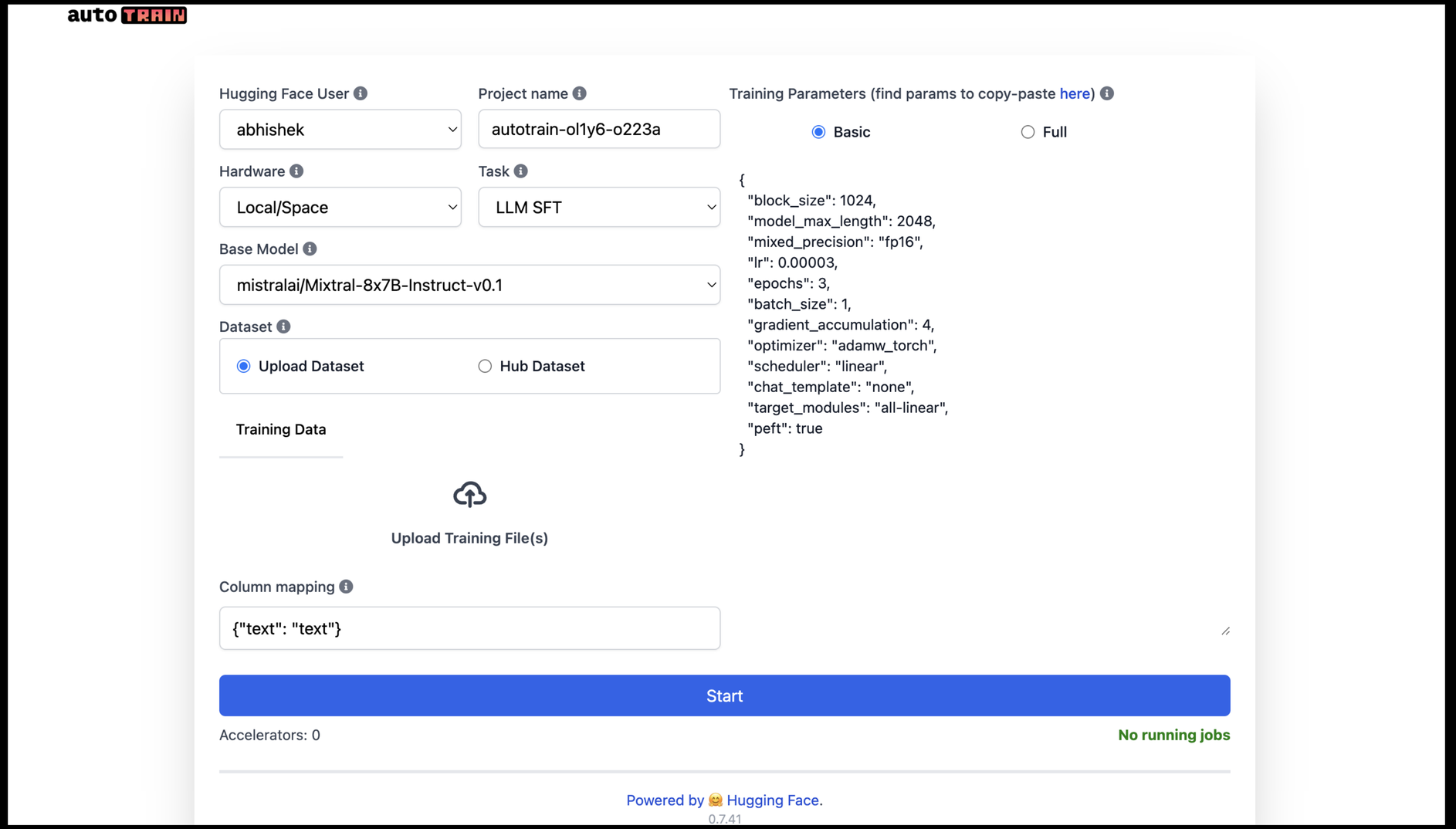
ABSTRACT

Deploying Large Language Models (LLMs) in streaming applications such as multi-round dialogue, where long interactions are expected, is urgently needed but poses two major challenges. Firstly, during the decoding stage, caching previous tokens' Key and Value states (KV) consumes extensive memory. Secondly, popular LLMs cannot generalize to longer texts than the training sequence length. Window


(d) StreamingLLM (ours)
Attention Sink
evicted tokens
L cached tokens

HARDWARE OPTIONS FOR FINE-TUNING AND INFERENCE

HUGGING FACE AUTOTRAIN



Spaces Hardware

Starting at $0

Spaces are one of the most popular ways to share ML applications and demos with the world.
Upgrade your Spaces with our selection of custom on-demand hardware:

→ Get started with Spaces

Name	CPU	Memory	Accelerator	VRAM	Hourly price
CPU Basic	2 vCPU	16 GB	-	-	FREE
CPU Upgrade	8 vCPU	32 GB	-	-	$0.03
Nvidia T4 - small	4 vCPU	15 GB	NVidia T4	16 GB	$0.40
Nvidia T4 - medium	8 vCPU	30 GB	NVidia T4	16 GB	$0.60
1x Nvidia L4	8 vCPU	30 GB	NVidia L4	24 GB	$0.80
4x Nvidia L4	48 vCPU	190 GB	NVidia L4	96 GB	$3.80
Nvidia A10G - small	4 vCPU	15 GB	NVidia A10G	24 GB	$1.00
Nvidia A10G - large	12 vCPU	46 GB	NVidia A10G	24 GB	$1.50
2x Nvidia A10G - large	24 vCPU	92 GB	NVidia A10G	48 GB	$3.00
4x Nvidia A10G - large	48 vCPU	184 GB	NVidia A10G	96 GB	$5.00
Nvidia A100 - large	12 vCPU	142 GB	NVidia A100	40 GB	$4.00
Nvidia H100	24 vCPU	250 GB	NVidia H100	80 GB	$10.00
Custom	on demand	on demand	on demand	on demand	on demand

Inference Endpoints

Starting at $0.032/hour

Inference Endpoints (dedicated) offers a secure production solution to easily deploy any ML model on dedicated and autoscaling infrastructure, right from the HF Hub.

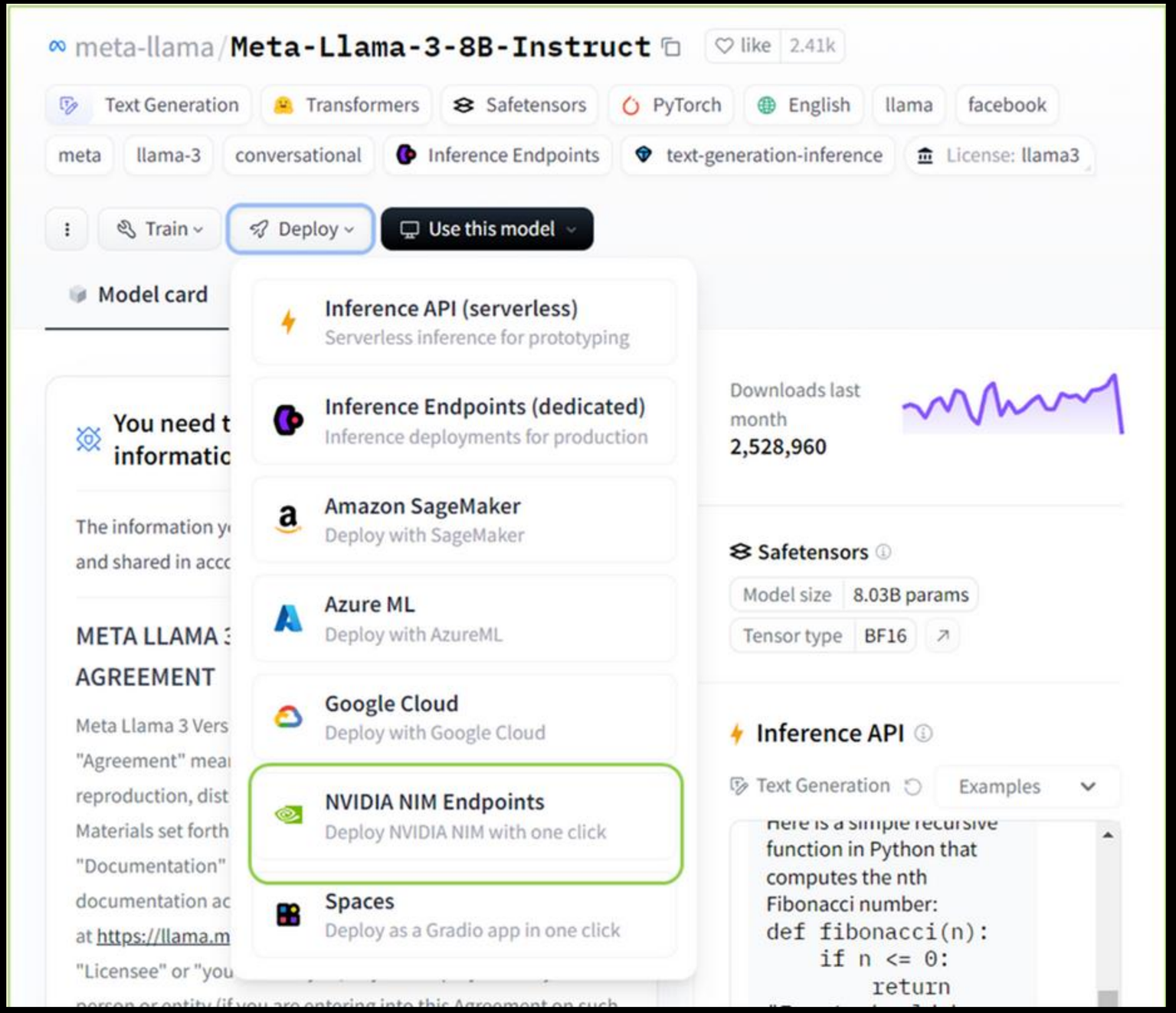
→ Learn more

CPU instances

Provider	Architecture	vCPUs	Memory	Hourly rate
aws	Intel Ice Lake	1	2GB	$0.03
aws	Intel Ice Lake	2	4GB	$0.06
aws	Intel Ice Lake	4	8GB	$0.13
aws	Intel Ice Lake	8	16GB	$0.26
aws	Inferentia2 Neur…	1	14.5GB	$0.75
aws	Inferentia2 Neur…	12	760GB	$12.00
azure	Intel Xeon	1	2GB	$0.06
azure	Intel Xeon	2	4GB	$0.12
azure	Intel Xeon	4	8GB	$0.24
azure	Intel Xeon	8	16GB	$0.48
gcp	Intel Sapphire R…	1	2GB	$0.07
gcp	Intel Sapphire R…	2	4GB	$0.14
gcp	Intel Sapphire R…	4	8GB	$0.28
gcp	Intel Sapphire R…	8	16GB	$0.56

GPU instances

Provider	Architecture	GPUs	GPU Memory	Hourly rate
aws	NVIDIA T4	1	14GB	$0.50
aws	NVIDIA T4	4	56GB	$3.00
aws	NVIDIA L4	1	24GB	$0.80
aws	NVIDIA L4	4	96GB	$3.80
aws	NVIDIA A10G	1	24GB	$1.00
aws	NVIDIA A10G	4	96GB	$5.00
aws	NVIDIA A100	1	80GB	$4.00
aws	NVIDIA A100	2	160GB	$8.00
aws	NVIDIA A100	4	320GB	$16.00
aws	NVIDIA A100	8	640GB	$32.00
gcp	NVIDIA T4	1	16GB	$0.50
gcp	NVIDIA L4	1	24GB	$1.00
gcp	NVIDIA L4	4	96GB	$5.00
gcp	NVIDIA A100	1	80GB	$6.00
gcp	NVIDIA A100	2	160GB	$12.00
gcp	NVIDIA A100	4	320GB	$24.00
gcp	NVIDIA A100	8	640GB	$48.00
gcp	NVIDIA H100	1	80GB	$12.50
gcp	NVIDIA H100	2	160GB	$25.00
gcp	NVIDIA H100	4	320GB	$50.00
gcp	NVIDIA H100	8	640GB	$100.00


meta-llama/Meta-Llama-3-8B-Instruct
like 2.41k
Text Generation
Transformers
Safetensors
PyTorch
English
llama
facebook
meta
llama-3
conversational
Inference Endpoints
text-generation-inference
License: llama3
Train
Deploy
Use this model
Model card
Inference API (serverless)
Serverless inference for prototyping
Inference Endpoints (dedicated)
Inference deployments for production
Amazon SageMaker
Deploy with SageMaker
Azure ML
Deploy with AzureML
Google Cloud
Deploy with Google Cloud
NVIDIA NIM Endpoints
Deploy NVIDIA NIM with one click
Spaces
Deploy as a Gradio app in one click
You need t
informatic
The information y
and shared in acc
META LLAMA 3
AGREEMENT
Meta Llama 3 Vers
"Agreement" mea
reproduction, dist
Materials set forth
"Documentation"
documentation ac
at https://llama.m
"Licensee" or "you
Downloads last month
2,528,960
Safetensors
Model size 8.03B params
Tensor type BF16
Inference API
Text Generation
Examples
Here is a simple recursive function in Python that computes the nth Fibonacci number:
def fibonacci(n):
if n <= 0:
return

Create a new Dedicated Endpoint

Model Repository

Import a model from the Hugging Face hub

meta-llama/Meta-Llama-3-8B-Instruct

Endpoint Name

Input a name for your new endpoint

meta-llama-3-8b-instruct-rox

Instance Configuration

Contact us if you need a custom solution or if the instance type cannot be selected.

Amazon Web Services | Microsoft Azure | Google Cloud Platform

N. Virginia us-east-1

CPU | GPU | INF2

Nvidia T4 1x GPU · 16 GB 3 vCPU · 15 GB $ 0.5/h	**Nvidia L4** 1x GPU · 24 GB 7 vCPU · 30 GB $ 0.8/h	**Nvidia A10G** 1x GPU · 24 GB 6 vCPU · 30 GB $ 1/h	**Nvidia T4** 4x GPU · 64 GB 46 vCPU · 192 GB $ 3/h
Nvidia L4 4x GPU · 96 GB 47 vCPU · 185 GB $ 3.8/h	**Nvidia A100** 1x GPU · 80 GB 11 vCPU · 145 GB $ 4/h	**Nvidia A10G** 4x GPU · 96 GB 46 vCPU · 186 GB $ 5/h	**Nvidia A100** 2x GPU · 160 GB 22 vCPU · 290 GB $ 8/h
Nvidia A100 4x GPU · 320 GB 44 vCPU · 580 GB $ 16/h	Nvidia A100 8x GPU · 640 GB 88 vCPU · 1160 GB $ 32/h		

✓ An optimized **Nvidia NIM** container has been selected for your endpoint.

A product for every workload

Cloud infrastructure pricing that's simple, predictable and transparent.

Metal

Global, high-performance bare metal servers

Server per hour from

General purpose	$0.14
Storage-optimized	$1.00
Memory-optimized	$1.40

View all pricing >

Accelerate

Dedicated GPU clusters for ML and HPC workloads.

GPU per hour from

L40S	$1.04
A100	$2.02
H100	$3.07

View all pricing >

Launchpad

Container-based GPU cloud for model fine-tuning and inference.

GPU per hour from

L40S	$1.32
H100	$2.10

Promotional pricing during beta

Get started >

All products

PROJECT PRESENTATION AND DOCUMENTATION

15.04.2024 18:00 - 19:30	Introduction	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
22.04.2024 18:00 - 19:30	Project Definition and Introduction to Fine-Tuning	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
29.04.2024 18:00 - 19:30	Characteristics of Fine-Tuning LLMs	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
06.05.2024 18:00 - 19:30	Model Evaluation	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
13.05.2024 18:00 - 19:30	Project Work	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
20.05.2024 18:00 - 19:30	Project Work	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
27.05.2024 18:00 - 19:30	Project Work	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
03.06.2024 18:00 - 19:30	Tokenization for Instruction Tuning	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
10.06.2024 18:00 - 19:30	Model Inference and Deployment	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
17.06.2024 18:00 - 19:30	Project Presentations	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE
24.06.2024 18:00 - 19:30	Project Presentations Extra	Starterkitchen, Kuhnkestr. 6, 24118 Kiel + ONLINE

▪ Web3 Coding Assistant	CodeLlama2, StarCoder // Julien, Kristian B., Anna-Valentina
▪ Socratic Assistant	Llama3 8B Chat // Ben, Julian
▪ Synthetic Data Generation for Event Data	Llama3 8B, GPT-3 .5 // Yorck, Kaan, Dikshyant, Khan
▪ Minimal Size Model for Conversations with Movie Characters	Phi2 // Christopher, Tural
▪ Training a Model for Diagnostics Based on Manuals	Llama3 8B // Christian W., Christian R., Dilip, James, Yildiz
▪ Financial Data Extraction	LeoLLM 7B // Nicolas
▪ Genome Chatbot	BioBERT? // Muhammad
▪ Small Size Language Learning Assistant	Phi3 Mini, LeoLLM, Sauerkraut// Rafael, Ilhay, Philip, Sina
▪ Small, open-source, multilingual function-calling agents	Phi3 Mini, RWKI, Tiny Llama // Jeremy, Boran

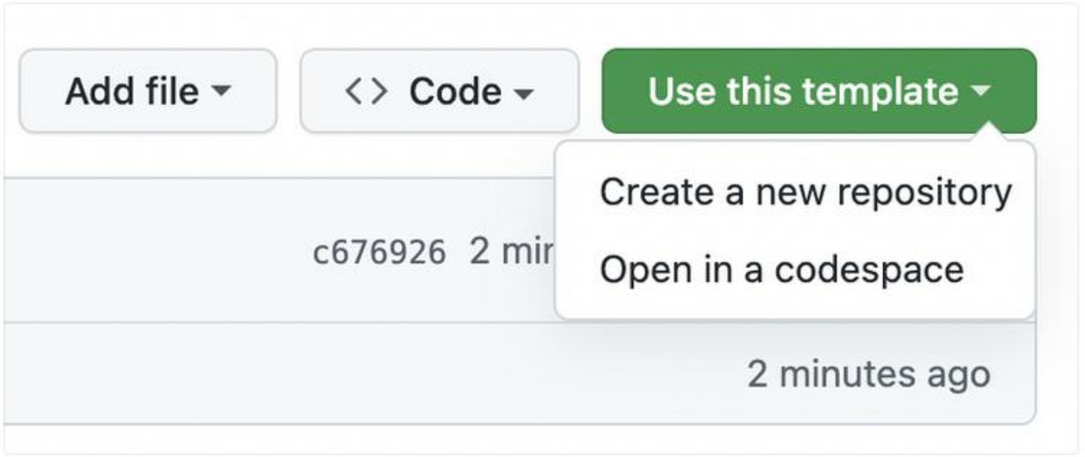
How to Start, Complete, and Submit Your Project

In all Machine Learning courses you have:

- to complete a machine learning project in a team of up to 4 participants,
- attend at least all but 2 sessions of the course, and
- use the provided project template repository for documentation (unless otherwise instructed).

Starting Your Project

1. **Navigate to the Template Repository**
2. **Use this Template**: Above the file list, click the "Use this template" button.



Use this template button

3. **Create Repository from Template**: You'll be prompted to name your new repository and you can choose whether it should be public or private. You'll also have the option to include all branches in the template repository, if there are more than one.
4. **Create Repository**: Click "Create repository from template" to create the new repository.
5. **Clone the New Repository**: You can now clone the new repository to your local machine using `git clone` and start working on your project.

Working on Your Project

Follow the `INSTRUCTIONS` file in each folder of the template repository to complete each section of your project.

Submitting Your Project

Note: Only ONE team member needs to submit the project.

1. **Slides**: Create your presentation slides. Save them in `4_Presentation` as a PowerPoint, Google Slides, or PDF file.
2. **Cover Image**: Replace the placeholder image in `CoverImage` with an image from your slides.
3. **README**: Update the main README with project details.
4. **Link to Slides**: Modify the link in the README of the folder `04_presentation` according to the file name including your presentation slides.
5. **edu.opencampus.sh Submission**:
   - Log in at edu.opencampus.sh.
   - Navigate to your course and go to the 'achievements' section.
   - Select your project title and download possible project specific documentation instructions.
   - Upload the main README via the upload dialog.
   - Include co-authors in the upload dialog as applicable.

Deadline: All submissions will be reviewed after the deadline and certificates will be issued accordingly.