

# Mini projects

Here below is a list of projects to choose from. You are to work on these projects in groups of 2-3. Please form groups and select a project and write yourself with **full names** into the document, just like I have done as an example below, where you assign the next group index and put in project

## Projects to choose from

- 1. Image+text learning for object detection.** During recent years, image+text models like e.g. the CLIP model have become increasingly popular due to their flexibility in solving different image tasks. In this project, you are given a dataset of almost 8000 images from a football match (one frame per second), where the objective is to find logos, in particular the logos for “Oddset” (appears on banners and near the goals) and “Ceres” (appears on banners and on the white shirts). The usual approach is to label a number of images and train a standard detector like yolov5/yolov8. However, can you find some logos by “asking” and image+text model like e.g. CLIP or OWLv2 (<https://huggingface.co/spaces/merve/owlv2>), or even fine tune it to the specific logos? Dataset: <https://bit.ly/3FxcCJa>.
- 2. Identity recognition from audio.** This project is about classifying between two persons based on the sound of their voice. You are free to come up with a dataset yourself, or you can try my dataset, which is taken from the publicly available Danish podcast ‘Her Går Det Godt’ with the two hosts Peter Falktoft and Esben Bjerre. The dataset consists of 300 audio snippets of 5s split evenly between Peter, Esben and background. It may be that you want to split the data into even shorter segments, e.g. 1s. Remember to hold back some data for validation and/or test. You can either try to build your own classifier network that processes 1D waveforms, or you can try to solve it with few-shot learning, i.e. with an audio embedding model where you embed only a few training instances and then do classification using k-NN. See e.g. pyannote for inspiration: <https://github.com/pyannote/pyannote-audio#tldr> - the call to `pipeline()` can take an extra argument for returning speaker embeddings for each audio snippet, or you can even make pyannote segment the speakers throughout a full audio file (not included in dataset) and then get the embeddings out in the end for the task. Dataset: <https://bit.ly/3QzmliY>.
- 3. Text-to-text translation.** In my company MediaCatch, we are working on automatic text-based translation between different languages. One difficult example is the Greenlandic language (also called Kalaallisut), for which there are limited models and data available. You are given a dataset containing a datadump from the Greenlandic news site <https://sermitsiaq.ag>, which contains 12030 articles in both Kalaallisut (kl) and Danish (da). The task is to train one or two models to translate between both languages, i.e. both kl-da and da-kl. The dataset is in a jsonl file, which is most easily read by reading the file line by line and parsing each line as JSON, e.g. in python using `json.loads(line)`. To get inspiration, look e.g. at a model like mT5 on Huggingface. If you wish to play with larger LLMs like e.g. llama, take a look at the axolotl library for easy training. Dataset: <https://bit.ly/409Iz3A>.
- 4. Your own project - conditions apply!** If you want to define your own project, you should send me the equivalent of ½-1 page of motivation for it before Nov. 25. Here you must motivate the project, what methods you want to use, and expected results.

## Groups

Number	Members	Project
0	Mette Frederiksen, Lars Løkke Rasmussen, Troels Lund Poulsen	2
1	Milo Fournier, William Noviti Edwin Condo & Shenal Devinda	1
2	Bohdan Kopčák, MHD aghyad Jamal eddin	1
3	Martin Kraus, Filip Jašek	1
4	Panus Sawetpiyakul, Filip Lobpreis, Jule Zaunseder	1
5	Isabella Mia Nielsen, Meri Mäkelä, Lorenzo Fagiani	4
6	Anders Johannsen, Nikkie Stella Pedersen	1
7	Ella Røgilds, Lucas Thurnherr, Xuze Cai	1
8	Erik Winkler, Branko Dzabic	2
9	Gian Paolo Curra, Frederike Durow, Guglielmo Borzone	2
10	Zain Ahmed, Henrik Dyrberg Egemose <a href="https://colab.research.google.com/drive/1gcBXWvKuKA82kSHckCow8wYAViM7yn1o?usp=sharing">https://colab.research.google.com/drive/1gcBXWvKuKA82kSHckCow8wYAViM7yn1o?usp=sharing</a>	2
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		