ARMY CYBER INSTITUTE AT WEST POINT

Rapid Development of Good Enough Machine Learning Models

MAJ Iain Cruickshank

iain.cruickshank@westpoint.edu

12 JUN 2023

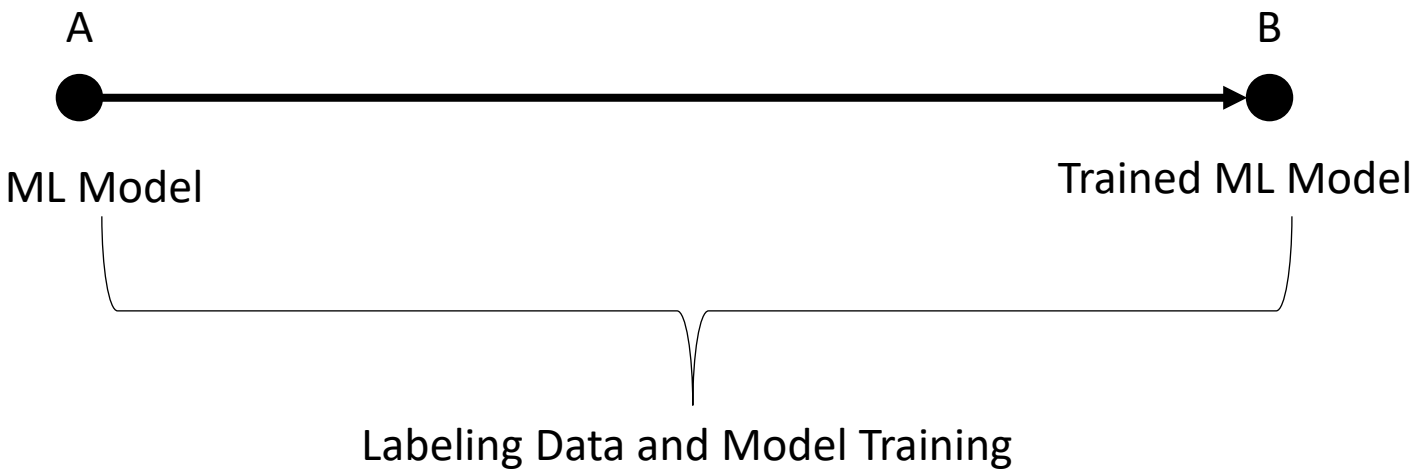
Code Repo
https://github.com/ijcruic/rapid-ml-prototyping

About Me

- Military Background
  - Functional Area 49 (ORSA)
  - Base branch of Military Intelligence
  - Assignments at 101st, 780th, and AI2C
  - Currently a senior research scientist at the ACI
- Academic Background
  - BS in Mathematics from USMA (2010)
  - MS in OR from U of Edinburgh (2011)
  - Ph.D. in Societal Computing from CMU (2020)
- Currently the Deputy Junior Analyst Chair and taking over the Junior Analyst Chair shortly
  - I am looking for a deputy – please contact me if interested!

Opening Thought

What is the shortest distance between two points?



A=B

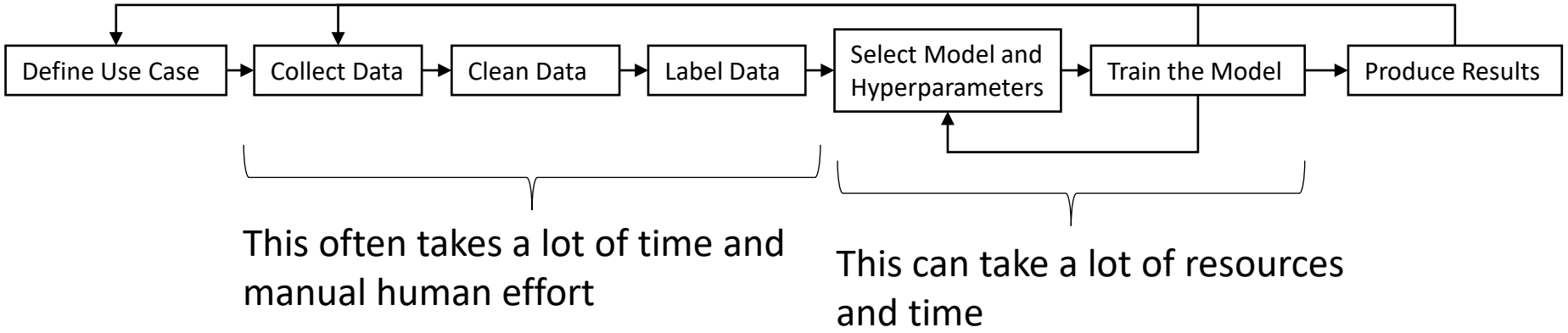
Start with a Trained* Model and Prompt it

Agenda

- Concept of this Course
- Background on Technical Aspects
- Worked Computer Vision Example
- Worked Text Example
- Concluding Thoughts

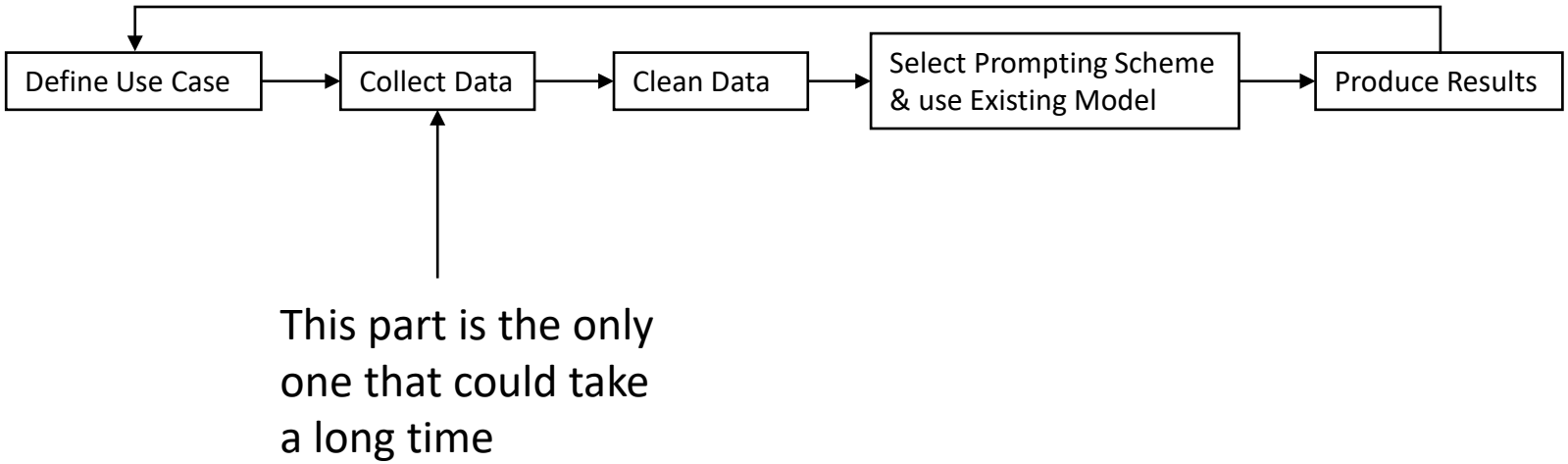
Overarching Concept

The general model development workflow looks something like this:



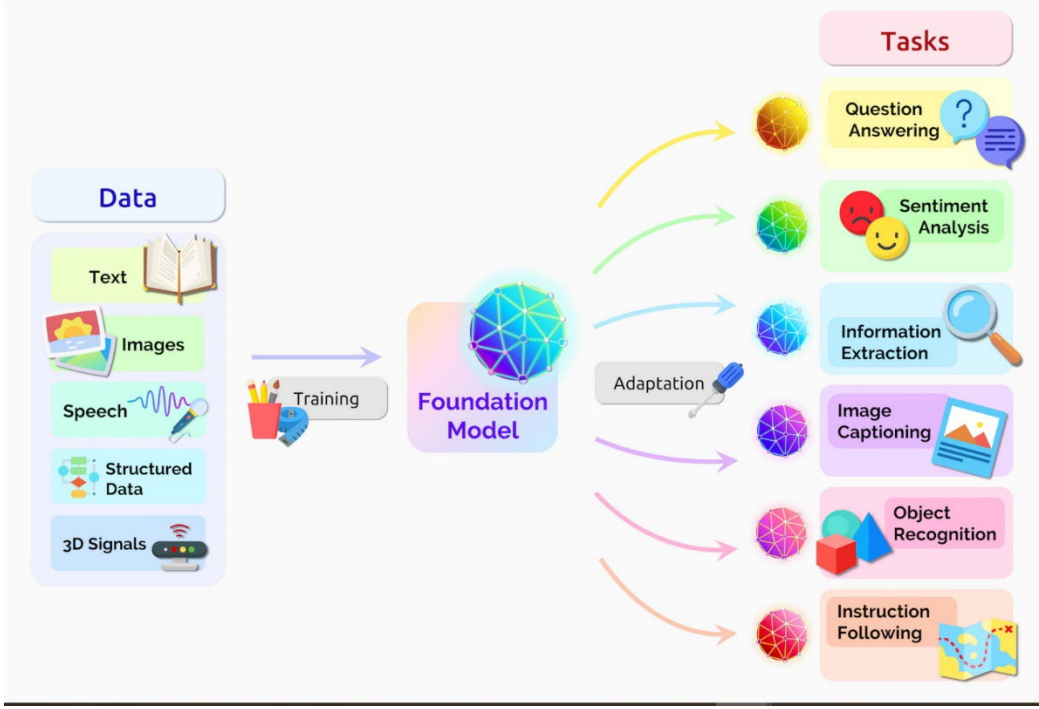
Overarching Concept

The model development workflow for when we rapidly need an ML model looks like this:



What is a Foundation Model

A foundation model (also called base model) is a large machine learning (ML) model trained on a vast quantity of data at scale (often by self-supervised learning or semi-supervised learning) such that it can be adapted to a wide range of downstream tasks.



Merrit, NVIDIA Blog, https://blogs.nvidia.com/blog/2023/03/13/what-are-foundation-models/

What is Zero-shot Learning?

- Zero-shot learning is a subfield of machine learning that enables models to generalize to unseen or unfamiliar classes or tasks without explicit training examples.
- Key Insight: Leveraging knowledge transfer from related domains or auxiliary information to make predictions on novel classes or tasks.
- Zero-Shot Learning Approaches
  - Semantic Embeddings: Learning representations that map input data into a shared semantic space, enabling transfer across domains.
  - Attribute-based Methods: Utilizing semantic attributes or characteristics to describe classes and transfer knowledge.
  - Generative Models: Generating samples of unseen classes by modeling the underlying data distribution.

What is a Language-Vision Model (LVM)?

- Combine a text encoder with a vision encoder
- Leverage captioned images
- Learn a joint representation of the image and text modality

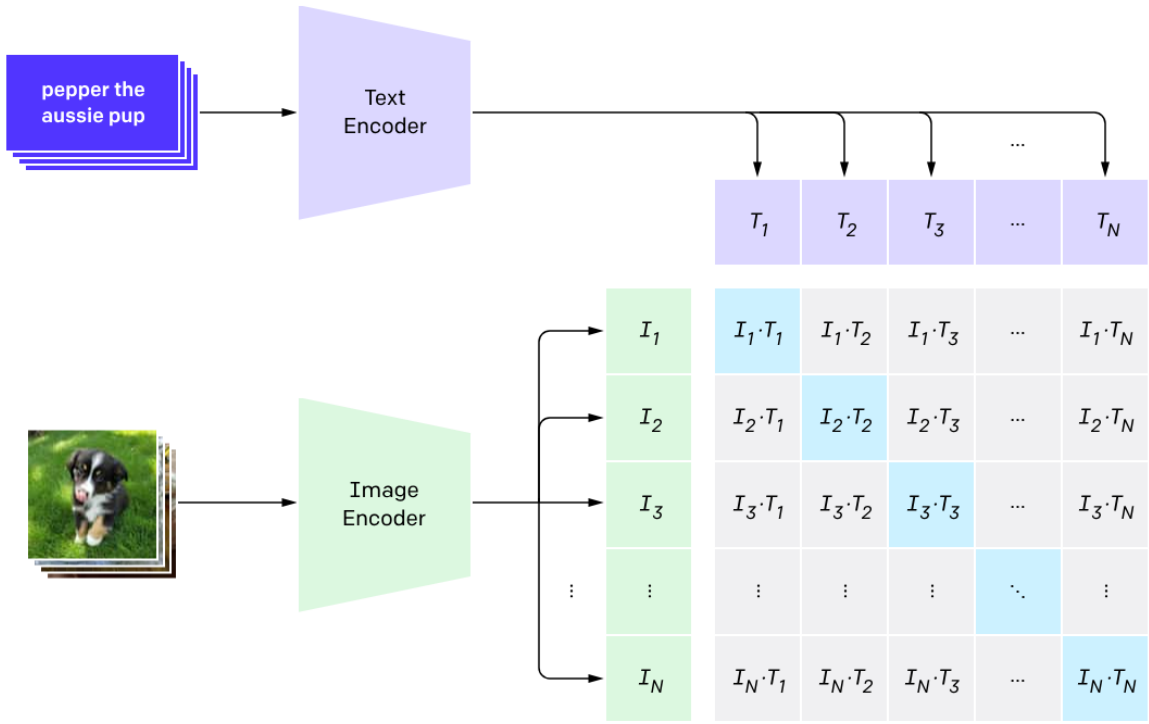


Image from Radford et al. https://openai.com/research/clip

What is Prompting or Prompt Engineering?

- Prompt engineering is the process of designing and crafting effective prompts or instructions to guide the behavior of language models.
- Key Insight: The careful formulation of prompts can influence the output and control the behavior of language models, allowing for specific tasks or desired responses.
- This is an ongoing area of research, but some prompting techniques, like few-shot prompting, chain-of-thought prompting, and others are producing great results from ML models
  - https://www.promptingguide.ai/papers

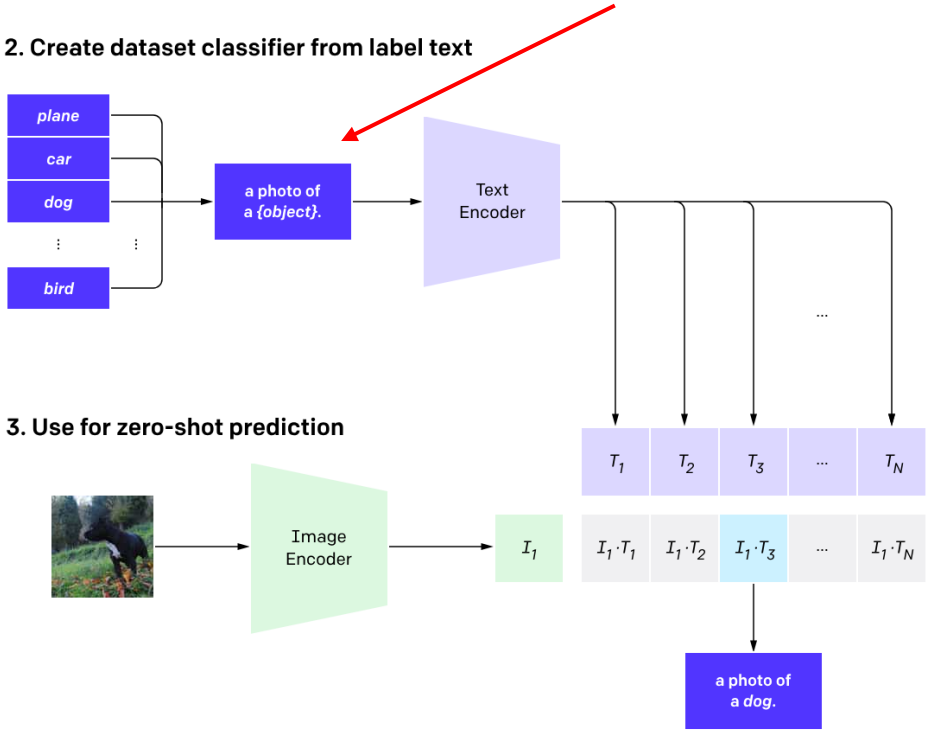


Image from Radford et al. https://openai.com/research/clip

Computer Vision Practical Example

Code base available at:

https://github.com/ijcruic/rapid-ml-prototyping/blob/main/Image/12JUL23_MORSS_Vision.ipynb

Data at:

https://drive.google.com/file/d/175eJk0q19NZAeWzyuWEpYpdo3EBJpPvO/view?usp=drive_link

Computer Vision Summary

- The use of LVMs, especially with prompting, can rapidly provide functional vision models with no data labeling
- Prompting is often an iterative practice
- An LVM can be finetuned to further improve its performance for a specific task/domain
- Questions
  - Say you finalize your prompts and the performance is good enough. How would you distill and productionize an LVM?
  - What other use cases could you use an LVM for?

Text Practical Example

Code base available at:

https://github.com/ijcruic/rapid-ml-prototyping/blob/main/Text/12JUN23_MORSS_Text.ipynb

Text Summary

- LLMs are quickly gaining traction in nearly every domain of ML and are enabling ML to applied to an ever-increasing number of tasks
- A big key to using LLMs successfully is in how you prompt the models, and, once again, prompting can be an iterative practice
- This is an (very) active area of research, so stay tuned for more developments
- Questions
  - Could you use an LLM to design better prompts for the LLM?
  - How could I use an LLM for tabular data? How about image data?

Review

- When seeking to prototype an ML solution
  1. Try to use existing foundational models first
  2. Use prompting techniques
  3. Fine-tune when necessary
- Next Steps
  - Model Distillation
    - https://blog.roboflow.com/autodistill/
  - Augmented Labeling
    - https://towardsdatascience.com/snorkel-ai-programmatic-approach-to-labeling-training-data-11973cf14f70
  - More usability in Foundational Models
    - http://meerkat.wiki/blog
  - Foundational Models for Tabular Data
    - https://arxiv.org/abs/2305.12081
  - Use of Vector Databases with LLMs
    - https://milvus.io/
- Survey feedback
  - https://forms.gle/e2iUyGNWsjKvESrEA