AI use cases in the real world

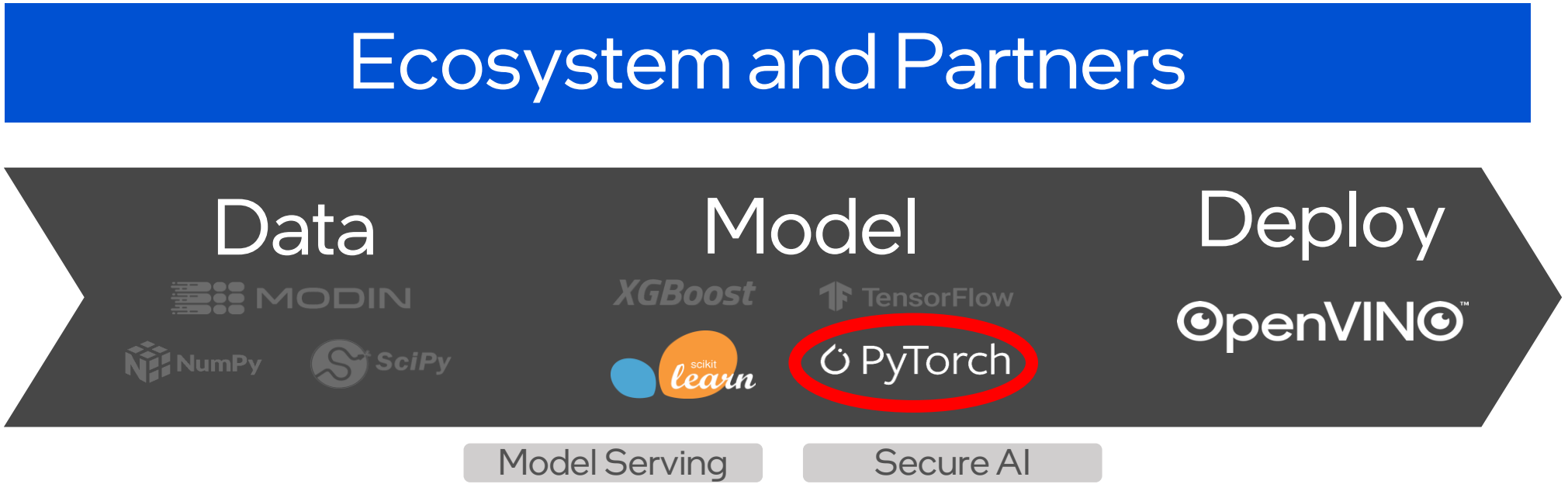
Bob Chesebrough

Thrill of the Hunt

Agenda

- Dinosaur bone hunting with AI
  - Motivation
  - Tools used
  - Why it works
  - How it works
  - Show me the fruit!
- Stable Diffusion briefly
  - Synthesize aerial photos based on an image example and text

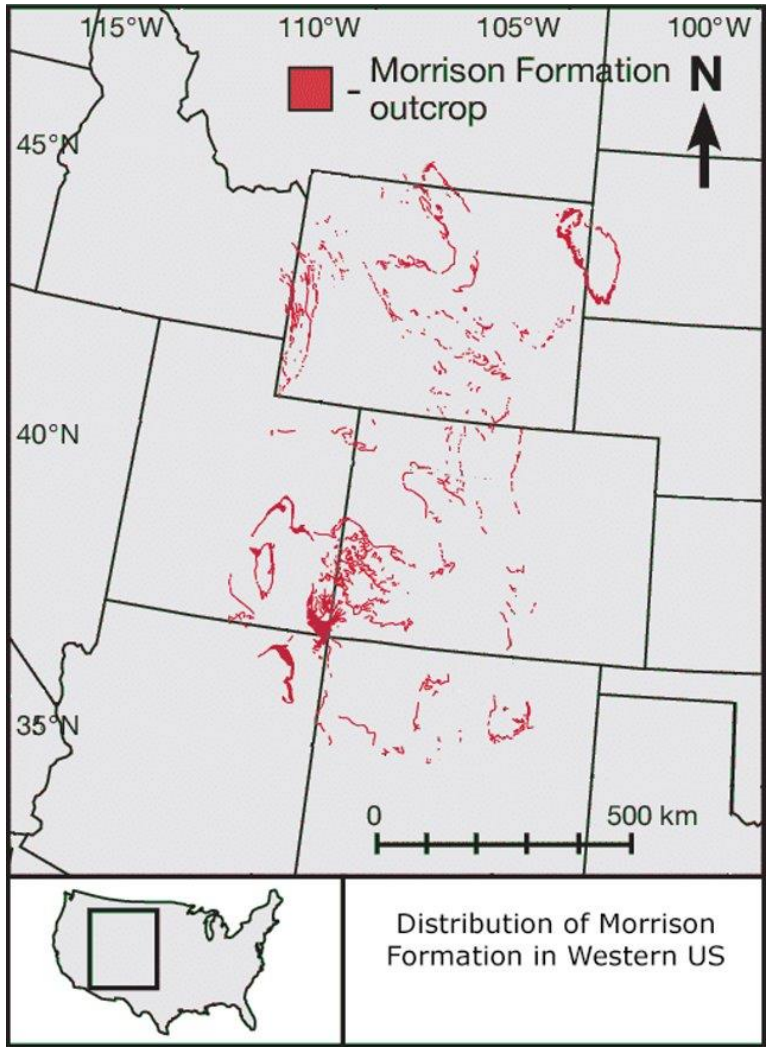
Intel AI Software Portfolio



What Do Dinosaur Bones Look Like & Where Are They Found?

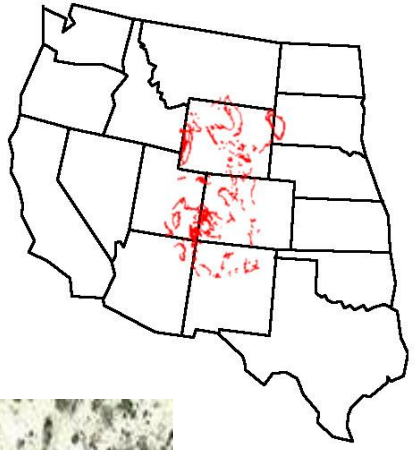
AI: NOT Searching for Bones!

It is Searching for bone bearing depositional environments!



Reason it works: Depositional Environment

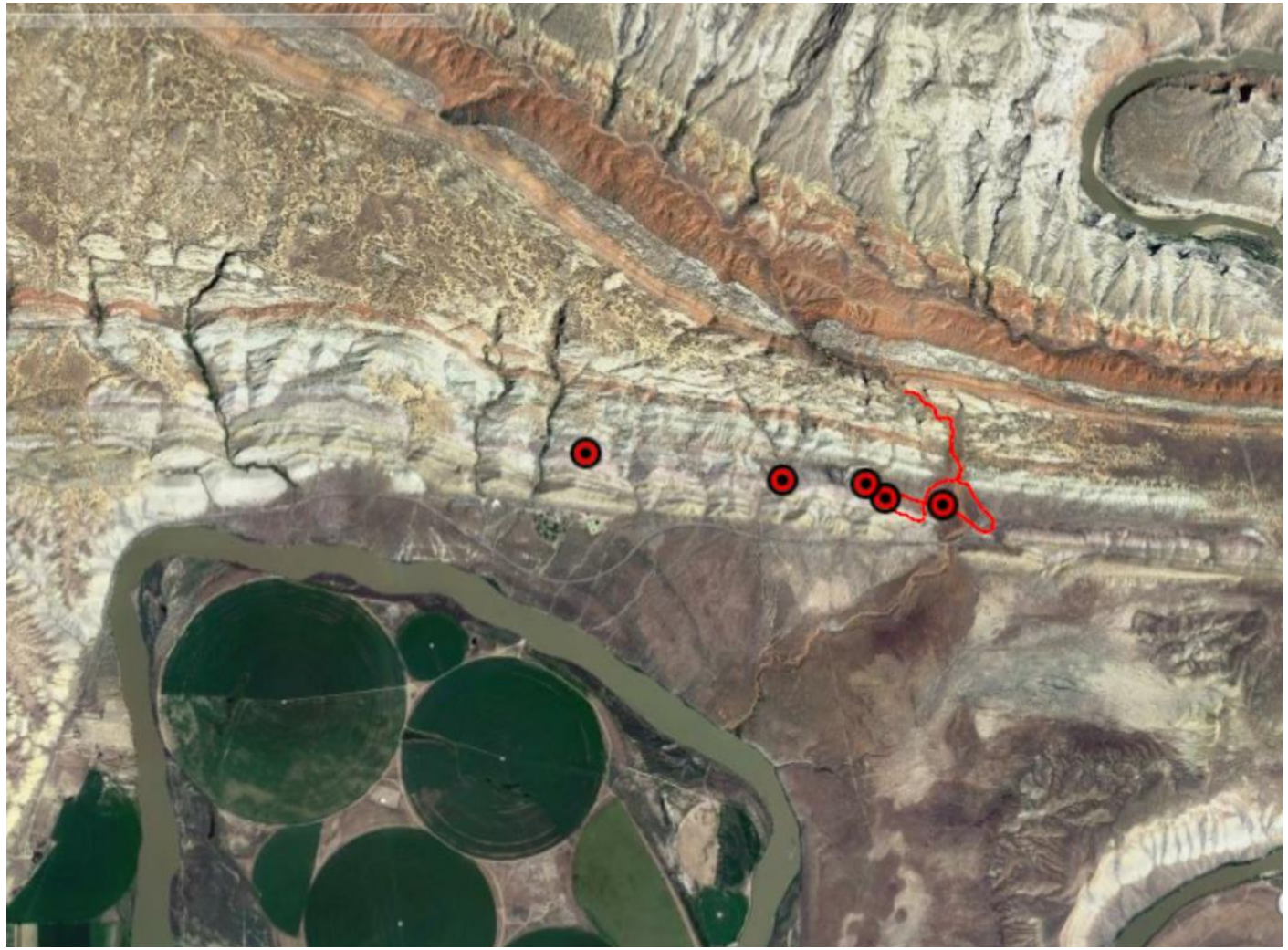
Morrison Formation



Depositional Environment

Where to hunt for dinosaur bones?

Depositional Environment

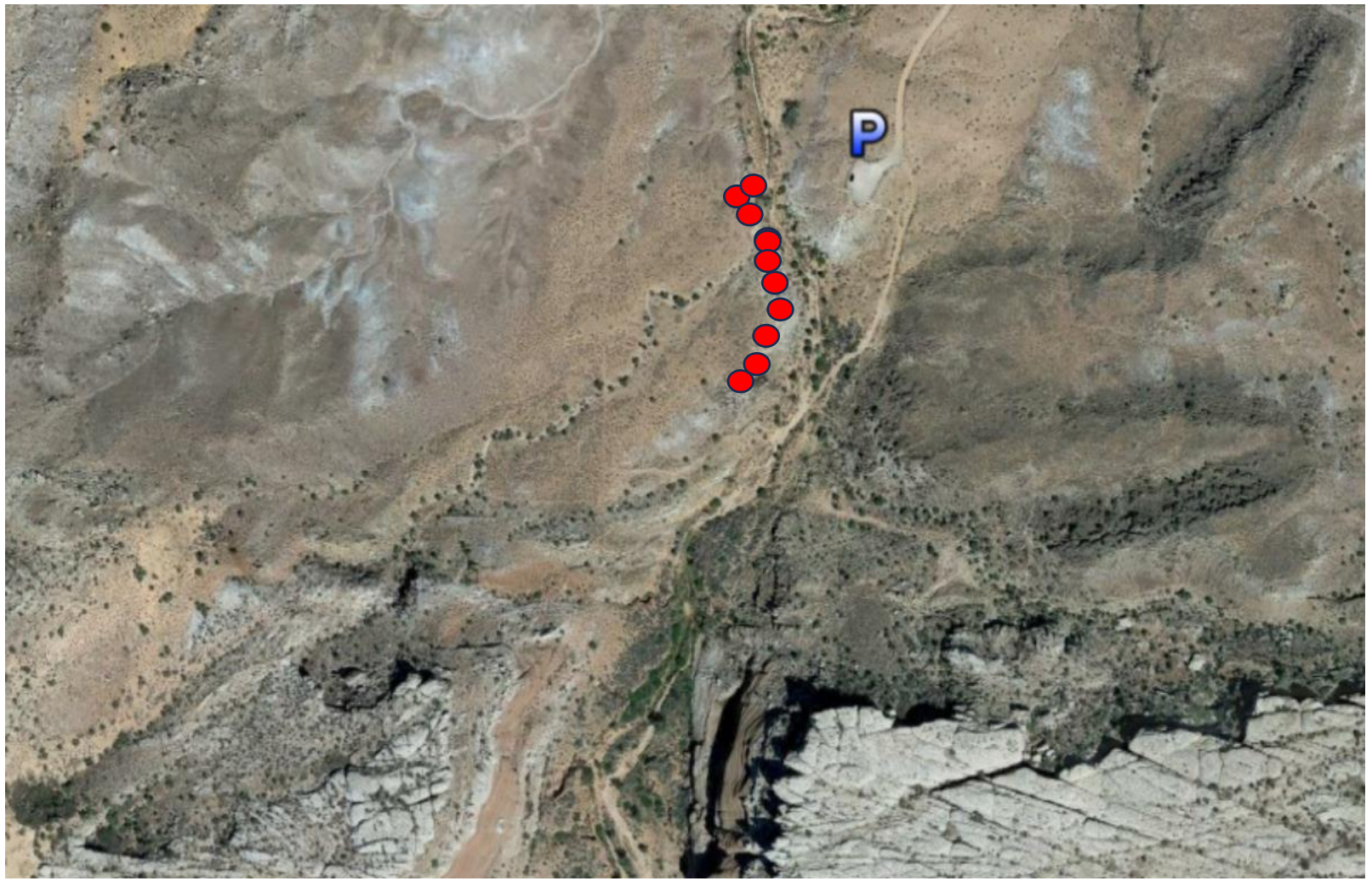


Red dots known bone locations

Short Quiz? (Difficult)

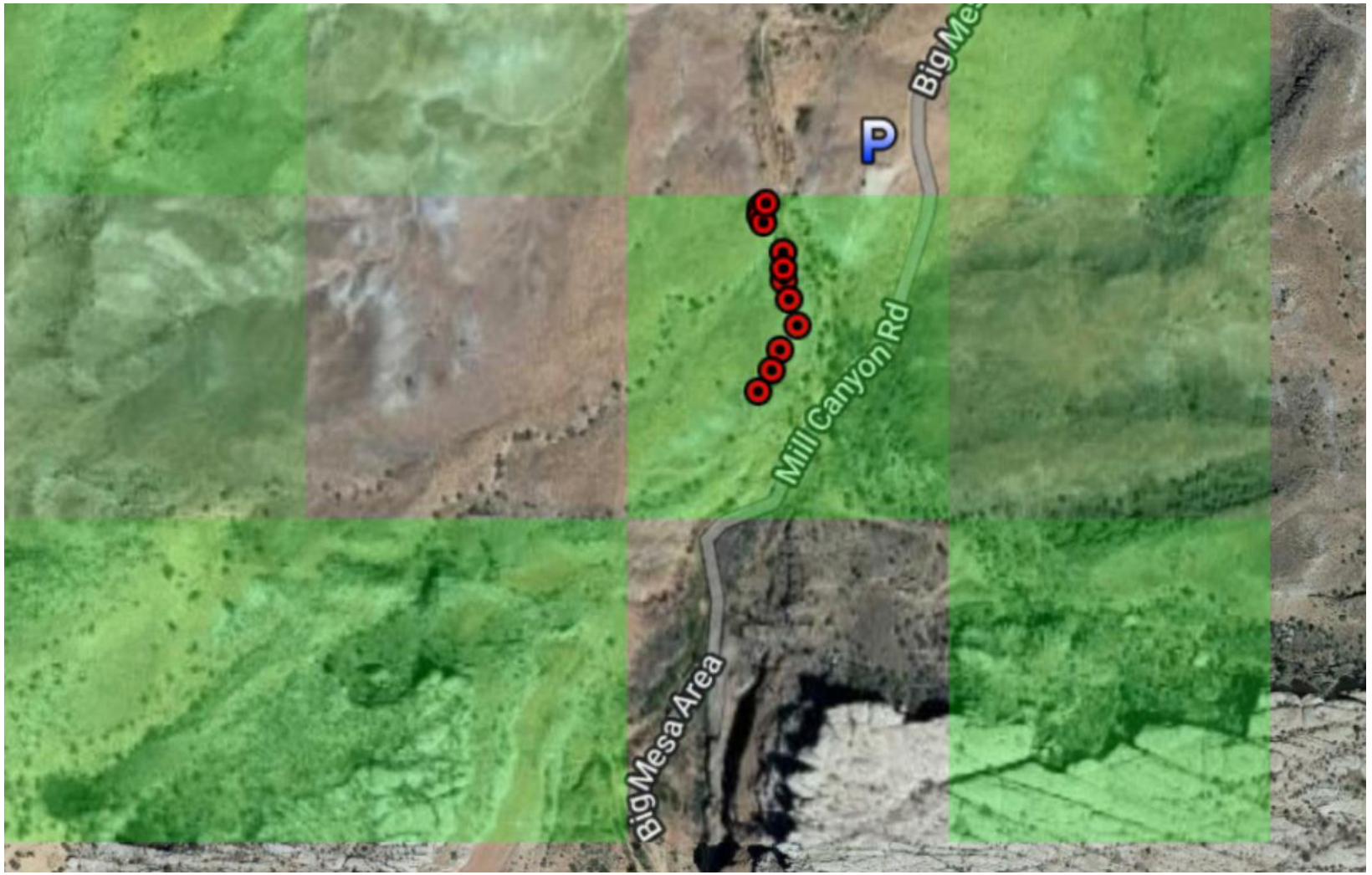
- Make a couple of thumb sized guesses where bones might be found.
- Not so easy right?

Short Quiz? (Difficult)



- Green is my AI model prediction in 2022
- Red dots are actual bone finds

Preliminary AI Guesses



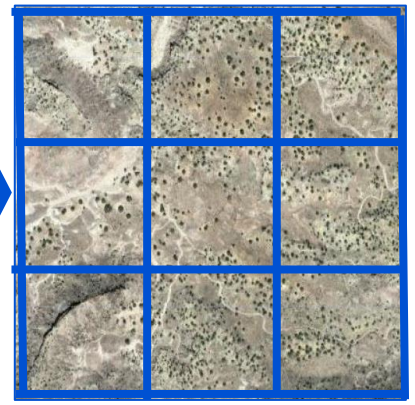
- Start with binary classification model – green is AI bone prediction

Create a Dinosaur Site Treasure Map

Collect

Collecting data

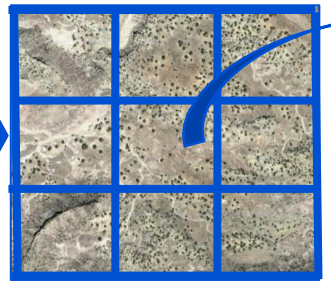
Collect



New Mexico (NM) Aerial Photos

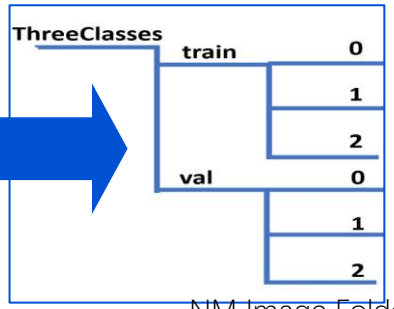
Train a simple Resnet 18

Collect



New Mexico Aerial Photos

NM Labeled Photos



NM Image Folders

Train Resnet18 Model on Intel GPU

NM Trained Model

```python
import intel_extension_for_pytorch as ipex
class Trainer:
    """Trainer class that takes care of training and validation passes."""
    def _to_ipx(self):
        """convert model memory format to channels_last to IPEX format."""
        self.model.train()
        self.model = self.model.to(memory_format=torch.channels_last)
        self.model, self.optimizer = ipex.optimize(
            self.model, optimizer=self.optimizer, dtype=torch.float32
        )
```

Using Intel Extension for PyTorch

Inference on CPU

```
import torch
import torchvision.models as models

model = models.resnet50(pretrained=True)
model.eval()
data = torch.rand(1, 3, 224, 224)

import intel_extension_for_pytorch as ipex
model = ipex.optimize(model)

with torch.no_grad():
  model(data)
```

Inference on XPU

```
import torch
import torchvision.models as models

model = models.resnet50(pretrained=True)
model.eval()
data = torch.rand(1, 3, 224, 224)

import intel_extension_for_pytorch as ipex
model = model.to('xpu')
data = data.to('xpu')
model = ipex.optimize(model)

with torch.no_grad():
  model(data)
```

We are Finetuning the ResNet model

- ResNet18 is an early simple CNN classifier – no details here – but we chose it for simplicity
- Just training the last layer from scratch
- Copy & finetune the remaining layers
- We Used a learning rate finder (FastAI) to find a good LR

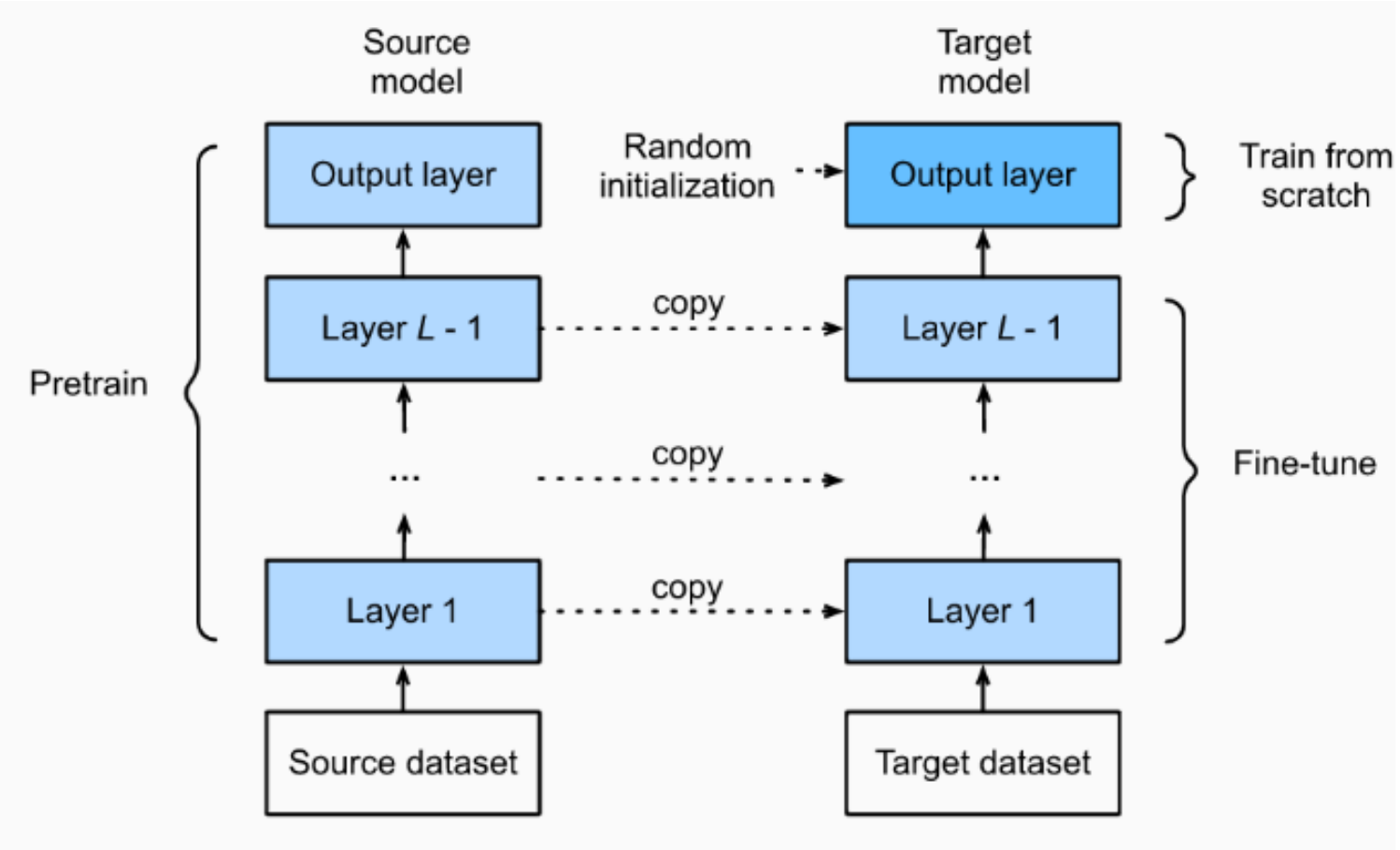
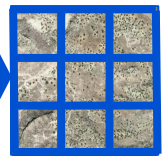


image src: https://d2l.ai/_images/finetune.svg

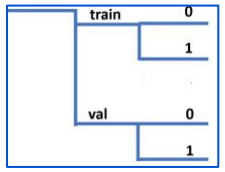
Score Images at New Locations

Collect



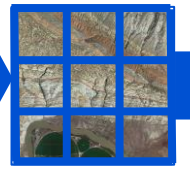
NM
Aerial
Photos

NM
Labeled
Photos

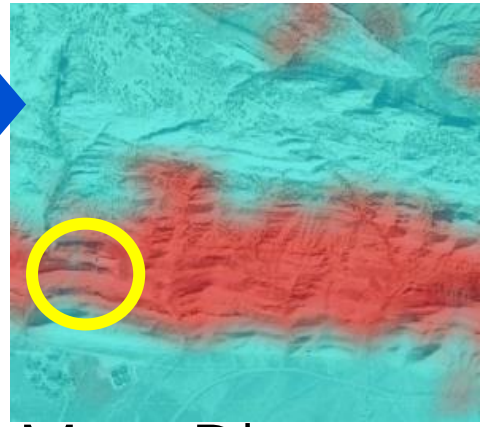


Inference
Model on
CPU

NM
Trained
Model



Score
Utah
Photos



Map: Dinosaur
National
Monument,
Utah
Red = Likely!

Results!


Bone InSitu
Bone InSitu
Ai bone nice limb
Ai bone frag
Ai bone frag
AI bone frags!

AI Bone!

Latest Dinosaur Bone Likelihood Map: Dinosaur National Mounument

DNM

Bones off trail we found

Stable Diffusion Example

- Intel Developer Cloud (register for free)
- Powerful image generators
- Try out Stable Diffusion Example
  - No coding required
  - Launch Jupyter lab
  - Run all cells
  - When prompted:
    - https://github.com/intelsoftware/ForestFirePrediction/blob/main/data/real_USGS_NAIP/train/Fire/m_3912105_sw_10_h_20160713.png?raw=true
    - “similar aerial photo just translate location”

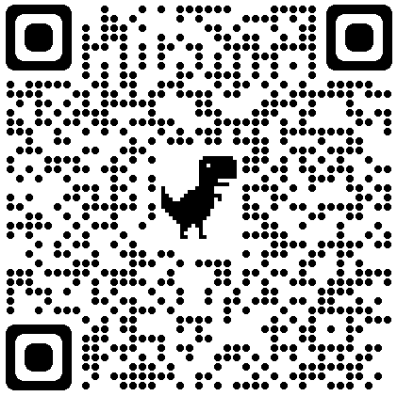
https://console.idcservice.net/

Actionable AI Projects in Computer Vision & ML

- Large Chocolate maker in Europe
  - Estimate Fat/Sugar blooms in chocolate bars with computer vision
  - Teak manufacturing to minimize these blooms
  - Regression example
- Railroad high speed car inspection:
  - Find potential defects
  - Object Detection & Classification
- Movie analysis:
  - Detect violence using sound/ spectorgrams
  - Detect phrases spoken
- Power line inspections
  - Component inspections by drone – locate and classify
  - Foliage encroachment
  - Anomaly Detection

Learn more!

- Code: https://github.com/intelsoftware/jurassic
- Try the Intel Extension for PyTorch https://intel.github.io/intel-extension-for-pytorch/#introduction
- Read about these use cases:



21st Century Paleontology
with Machine Learning

- Q & A

Thank you!

Thank You