Feature extractors

Print to PDF

Contents

This guide demonstrates how different backbones can be used as feature extractors for anomaly detection models. Most of these models use Timm Feature Extractor except **CSFLOW** which uses TorchFx Feature Extractor. Here we show how to use API and CLI to use different backbones as feature extractors.

> **See also**
>
> For specifics of implementation refer to implementation classes `Timm Feature Extractor` and `TorchFx Feature Extractor`

Available backbones and layers

Timm | **TorchFX**

Available Timm models are listed on Timm GitHub page.

In most cases, we want to use a pretrained backbone, so can get a list of all such models using the following code:

```
import timm
# list all pretrained models in timm
for model_name in timm.list_models(pretrained=True):
    print(model_name)
```

Once we have a model selected we can obtain available layer names using the following

code:

```
import timm
model = timm.create_model("resnet18", features_only=True)
# Print module names
print(model.feature_info.module_name())
>>>['act1', 'layer1', 'layer2', 'layer3', 'layer4']

model = timm.create_model("mobilenetv3_large_100", features_only=True)
print(model.feature_info.module_name())
>>>['blocks.0.0', 'blocks.1.1', 'blocks.2.2', 'blocks.4.1', 'blocks.6.0']
```

We can then use selected model name and layer names with either API or using config file.

Warning

Some models might not support every backbone.

Backbone and layer selection

API **CLI**

When using API, we need to specify `backbone` and `layers` when instantiating the model with a non-default backbone.

```
# Import the required modules
from anomalib.data import MVTec
from anomalib.models import Padim
from anomalib.engine import Engine

# Initialize the datamodule, model, and engine
datamodule = MVTec(num_workers=0)
# Specify backbone and layers
model = Padim(backbone="resnet18", layers=["layer1", "layer2"])
engine = Engine(image_metrics=["AUROC"], pixel_metrics=["AUROC"])

# Train the model
engine.fit(datamodule=datamodule, model=model)
```

Previous
Model Tutorials

Next
Topic Guide