Table of Contents

| | Learn | | Colab | ↓ Notebook | | GitHub |
|---|---|---|---|---|---|---|

# SAVE AND LOAD THE MODEL

In this section we will look at how to persist model state with saving, loading and running model predictions.

```python
import torch
import torchvision.models as models
```

## Saving and Loading Model Weights

PyTorch models store the learned parameters in an internal state dictionary, called `state_dict`. These can be persisted via the `torch.save` method:

```python
model = models.vgg16(pretrained=True)
torch.save(model.state_dict(), 'model_weights.pth')
```

To load model weights, you need to create an instance of the same model first, and then load the parameters using `load_state_dict()` method.

```python
model = models.vgg16() # we do not specify pretrained=True, i.e. do not load default weights
model.load_state_dict(torch.load('model_weights.pth'))
model.eval()
```

> • NOTE
>
> be sure to call `model.eval()` method before inferencing to set the dropout and batch normalization layers to evaluation mode. Failing to do this will yield inconsistent inference results.

## Saving and Loading Models with Shapes

When loading model weights, we needed to instantiate the model class first, because the class defines the structure of a network. We might want to save the structure of this class together with the model, in which case we can pass `model` (and not `model.state_dict()`) to the saving function:

```python
torch.save(model, 'model.pth')
```

We can then load the model like this:

```python
model = torch.load('model.pth')
```

> • NOTE
>
> This approach uses Python pickle module when serializing the model, thus it relies on the actual class definition to be available when loading the model.

## Related Tutorials

Saving and Loading a General Checkpoint in PyTorch

**Total running time of the script:** ( 0 minutes 8.593 seconds)

Rate this Tutorial        ☆ ☆ ☆ ☆ ☆

## Docs

Access comprehensive developer documentation for PyTorch

View Docs

## Tutorials

Get in-depth tutorials for beginners and advanced developers

View Tutorials

## Resources

Find development resources and get your questions answered

View Resources

**PyTorch**

Get Started

Features

Ecosystem

Blog

Contributing

**Resources**

Tutorials

Docs

Discuss

Github Issues

Brand Guidelines

**Stay Connected**