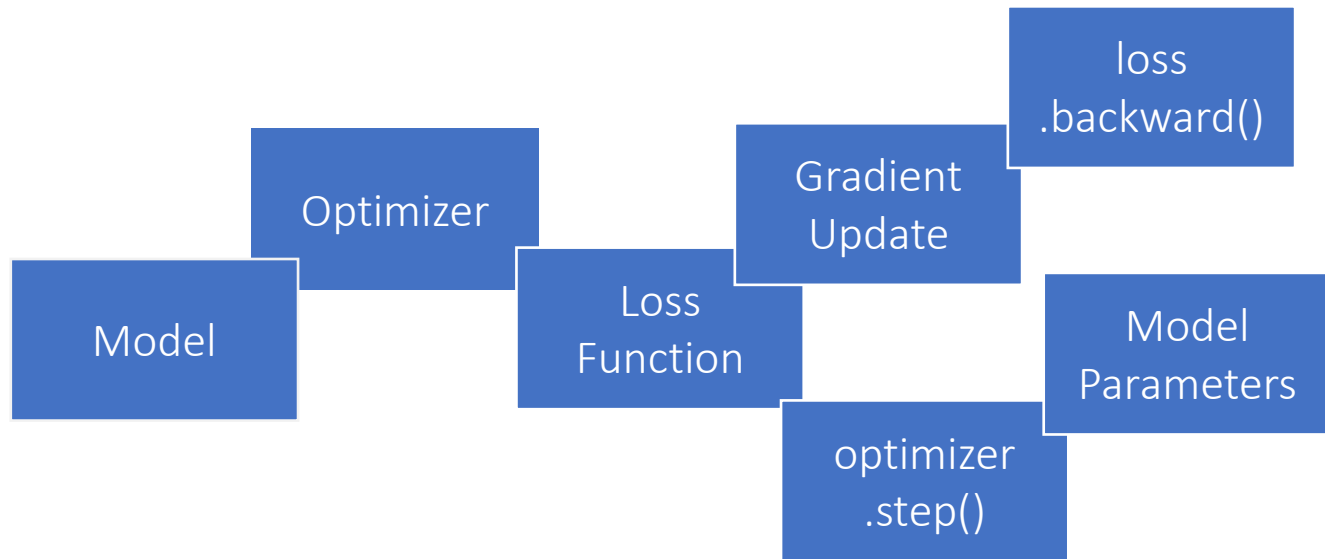


# PyTorch Model Training

# PyTorch Model Training

## Introduction

- Confusing process of interactions

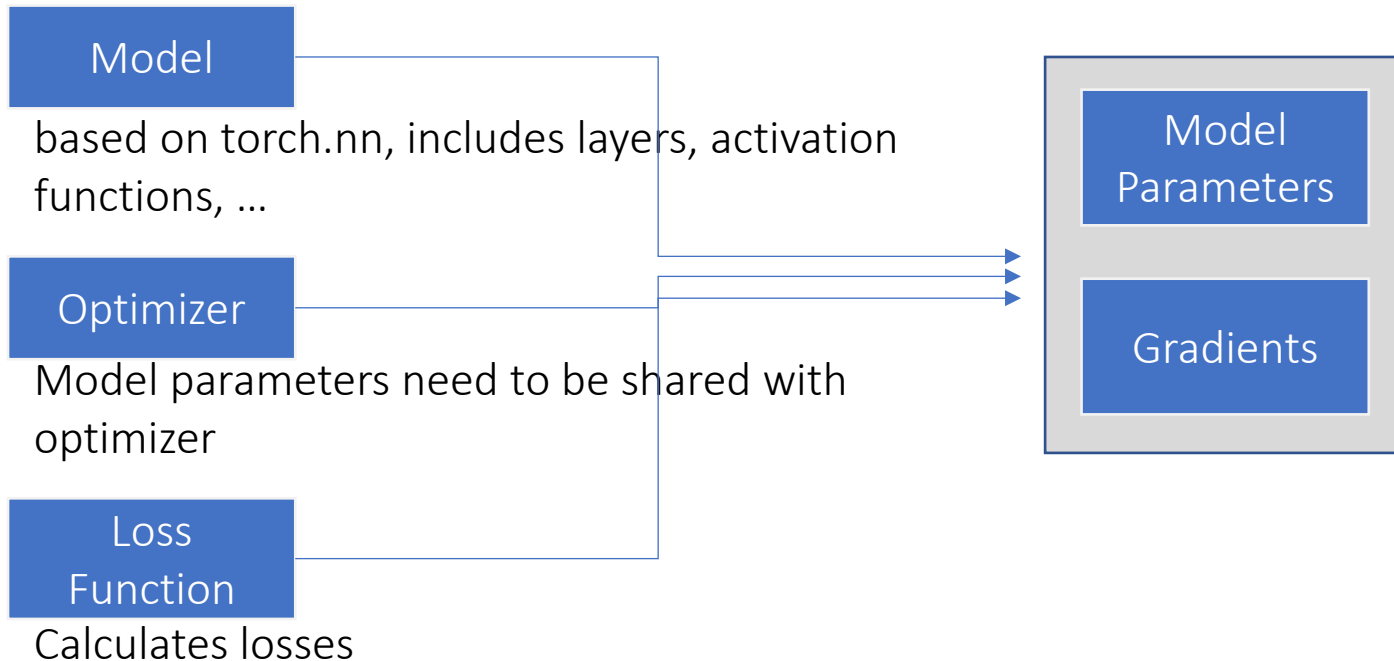


Good article on PyTorch Functional API: <https://jeancochrane.com/blog/pytorch-functional-api>

# PyTorch Model Training

States and objects

- Affected objects and states



# PyTorch Model Training

## Training Loop

- Typical structure of training

```
for epoch in range(number_epochs):  
    for j, data in enumerate(train_loader):  
        # optimization  
        optimizer.zero_grad()  
  
        # forward pass  
        y_hat = model(data[0])  
  
        # compute loss  
        loss = loss_fun(y_hat, data[1])  
        losses.append(loss.item())  
  
        # backprop  
        loss.backward()  
  
        # update weights  
        optimizer.step()
```

# PyTorch Model Training

## Clear gradients

- Optimizer accumulates gradients
- For each new pass (forward/backward) gradients need to be deleted
- Optimizer holds gradients? Not the model?

```
optimizer.zero_grad()
```

# PyTorch Model Training

## Forward Pass

- Predictions are calculated
- Straightforward process

```
# forward pass  
y_hat = model(data[0])
```

# PyTorch Model Training

## Loss calculation

- Also straightforward
- Predictions and true labels used to calculate losses

```
# compute loss  
loss = loss_fun(y_hat, data[1])
```

# PyTorch Model Training

## Gradient calculation

- Loss function object calculates gradients for all nodes
- $grad_{w_1} = \frac{\partial L}{\partial w_1}$  (partial derivative of loss function)
- Gradients are changed inplace
- Implicitly, model layers are used and tensor gradients updated

```
# backprop  
loss.backward()
```



# PyTorch Model Training

## Weight update

- Gradients are known, now weights need to be updated.
- Step() function of optimizer does it.
- Model parameters are updated (although model was never called).

```
# update weights  
optimizer.step()
```

# PyTorch Model Training

Advantages / Disadvantages



- Composable API, which can be used beyond Deep Learning



- Model, loss, optimizer change parameter state, no clear owner