core_02_00_variable

2019年12月26日 10:04

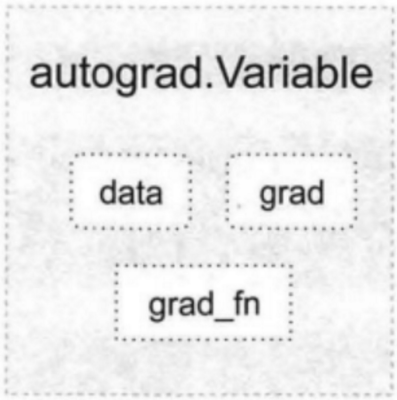

图 2-6 Variable 的数据结构

Variable 主要包含三个属性。

- data：保存 Variable 所包含的 Tensor。
- grad：保存data对应的梯度，grad也是个 Variable，而不是 Tensor，它和data的形状一样。
- grad_fn：指向一个Function对象，这个Function用来反向传播计算输入的梯度，具体细节会在第 3 章讲解。

Variable 的构造函数需要传入 tensor，同时有两个可选参数。

- requires_grad (bool)：是否需要对该 variable 进行求导。
- volatile (bool)：意为“挥发”，设置为 True，构建在该 variable 之上的图都不会求导，专为推理阶段设计。

Variable 支持大部分 tensor 支持的函数，但其不支持部分inplace函数，因为这些函数会修改 tensor 自身，而在反向传播中，variable 需要缓存原来的 tensor 来计算梯度。如果想要计算各个 Variable 的梯度，只需调用根节点 variable 的backward方法，autograd 会自动沿着计算图反向传播，计算每一个叶子节点的梯度。

`variable.backward(grad_variables=None, retain_graph=None, create_graph=None)` 主要有如下参数。

- grad_variables：形状与 variable 一致，对于`y.backward()`，grad_variables 相当于链式法则 $\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y} \times \frac{\partial y}{\partial x}$ 中的 $\frac{\partial z}{\partial y}$。grad_variables 也可以是 tensor 或序列。
- retain_graph：反向传播需要缓存一些中间结果，反向传播之后，这些缓存就被清空，可通过指定这个参数不清空缓存，用来多次反向传播。
- create_graph：对反向传播过程再次构建计算图，可通过`backward of backward`实现求高阶导数。