

Tổng quan về Pytorch

Giới thiệu về Pytorch

Pytorch là framework được phát triển với giấy phép mã nguồn mở do đó nó tạo được cho mình một cộng đồng rất lớn. PyTorch được phát triển bởi Facebook. Đây là một ông lớn về công nghệ đầu tư rất nhiều nguồn lực cho việc phát triển trí tuệ nhân tạo. Pytorch cùng với Tensorflow và Keras là một trong số những framework phổ biến để xây dựng và huấn luyện các mạng nơ ron nhân tạo hiện nay. Đặc biệt, trong các lĩnh vực nghiên cứu, hầu như các tác giả đều sử dụng pytorch để triển khai bài toán của mình.

Tensor trong Pytorch

Số thực là dữ liệu 0D, vector là dữ liệu 1D, ma trận là dữ liệu 2D còn dữ liệu từ 3D trở đi được gọi là tensors. Pytorch tensors có chức năng và mục đích tương tự Numpy arrays nhưng có một vài ưu điểm hơn:

- Thực hiện tính toán nhanh trên GPU.
- Các Pytorch tensors có thể lưu lại được đồ thị tính toán nên có thể tính đạo hàm nhanh chóng, phục vụ cho thuật toán backpropagation trong Deep Learning.

Autograd

Các thao tác tính toán trong quá trình backward của một mạng nơ ron đều dựa trên một quy tắc đó là Chain Rule hay đạo hàm của hàm hợp. Và module AutoGrad được sinh ra để giải quyết vấn đề tính toán đạo hàm phức tạp đó. Trong quá trình forward sẽ định nghĩa đồ thị tính toán với các node là các Tensors và các cạnh sẽ là các function tính toán giá trị của Tensor đầu ra khi đưa vào Input Tensor. Quá trình Backpropagating cũng được tính toán thông qua biểu đồ này cho phép dễ dàng tính toán giá trị của đạo hàm.

Neural Network Module

Neural Network Module là module sử dụng rất nhiều trong Pytorch. Có thể tạo một mô hình là một lớp bằng cách kế thừa `nn.Module`, điều này cho phép tạo một mô hình học sâu dưới dạng một lớp. Lớp này có 2 hàm bắt buộc phải có đó là `init` và `forward`:

- **init**: Là một hàm khởi tạo nhận vào các biến, tham số giúp ta có thể khởi tạo các biến, hàm của đối tượng được khởi tạo.
- **forward**: Là hàm nhận vào là dữ liệu input ban đầu. Dữ liệu sẽ đi lần lượt qua từng layer của model và trả về output của model.

Một số hàm trong Pytorch

Pytorch hỗ trợ rất nhiều các hàm loss như các bài toán hồi quy, phân loại,... Một số hàm loss phổ biến như là `nn.MSE()`, `nn.CrossEntropyLoss()`, `nn.KLDivLoss()`, `nn.BCELoss`, `nn.NLLLoss()`.

Tương tự như các hàm loss, Pytorch cũng hỗ trợ rất nhiều các hàm tối ưu để sử dụng như là `torch.optim.Adadelta`, `torch.optim.Adagrad`, `torch.optim.RMSprop`.

Sinh viên

Trịnh Xuân Minh
