

Review of “PyTorch: An Imperative Style, High-Performance Deep Learning Library”

Riyad Bin Rafiq

1. Paper summary

PyTorch is a machine learning library that focuses on both usability and speed. This paper [1] introduces PyTorch and compares the features including performance, implementation, operation with current popular deep learning frameworks such as Caffe, CNTK, TensorFlow and Theano. The authors described the implementation and the architecture of the library in a detailed way. They also represented how the library enables runtime performance. At last, the paper shows the performance of PyTorch in terms of training speed on some common benchmarks.

2. Contribution

PyTorch is a popular library for deep learning nowadays. It's contributions are as follows:

2.1 Modeling via Python class

Any layer of the network is expressed as a class where the constructor creates and initializes their parameters. Models including individual layers are also represented as class. This type of structure helps to implement a new potential neural network architecture easily using PyTorch.

2.2 Flexibility

PyTorch offers a great flexibility as it can leverage the rich ecosystem of Python libraries as part of the user program. For example, it allows conversion between Numpy arrays and PyTorch tensors. Moreover, these conversions are cheap and take a constant amount of time. Users are also allowed to replace any component of the Pytorch that is not needed in their program. This kind of flexibility provides the users a freedom to customize their used components of PyTorch to accelerate the model's performance.

2.3 Performance

There is a problem with the Python interpreter as the global interpreter lock only allows only one thread from a number of concurrent threads to run at a given time. Pytorch solved the problem by optimizing every phase of its execution and the library is written in C++ to obtain high performance. Moreover, the paper provided the overall performance of PyTorch in terms of throughput on some common benchmarks where the library placed in top three for all of the six models. This summary determines to understand the efficiency of PyTorch compared to other frameworks.

3. Critique

3.1 Comparison of Python programming structure is missing

The paper mentioned how layers are expressed as a Python class and how a full model including different layers are also represented easily as a class by PyTorch. The paper mentioned a full basic model and generative adversarial network as examples. But the authors didn't describe how the model is expressed in other frameworks. The comparison of Python programming structure in different frameworks could provide a robust understanding of the implementation.

3.2 Adoption doesn't claim everything

The paper mentioned how PyTorch has been received by the machine learning community. For this, they took account of arXiv papers that mentioned "PyTorch". And from the figure 3 of the paper, we can observe that the popularity increases over time. But it didn't represent what percentage the other frameworks' (TensorFlow, Caffe, etc.) have been used over time.

Reference

1. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An imperative style, high-performance deep learning library. Adv Neural Inf Process Syst. 2019;32. Available: <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>