

Chapter 3

Deep Learning With PyTorch

Anmol Chaudhary

Government Engineering College, Ajmer, India

Jyoti Gajrani

Malviya National Institute of Technology, India

Kuldeep Singh Chouhan

Government Engineering College, Ajmer, India

Bhavna Sharma

JECRC University, Jaipur, India

ABSTRACT

In the last decade, deep learning has seen exponential growth due to rise in computational power as a result of graphics processing units (GPUs) and a large amount of data due to the democratization of the internet and smartphones. This chapter aims to throw light on both the theoretical aspects of deep learning and its practical aspects using PyTorch. The chapter primarily discusses new technologies using deep learning and PyTorch in detail. The chapter discusses the advantages of using PyTorch compared to other deep learning libraries. The chapter discusses some of the practical applications like image classification and machine translation. The chapter also discusses the various frameworks built with the help of PyTorch. PyTorch consists of various models that increases its flexibility and accessibility to a greater extent. As a result, many frameworks built on top of PyTorch are discussed in this chapter. The authors believe that this chapter will help readers in getting a better understanding of deep learning making neural networks using PyTorch.

INTRODUCTION

Deep learning is a field of Artificial Intelligence in which a large amount of data is used to train computers to accomplish tasks that cannot be done by simple programming algorithms. With the rise of computational power due to GPUs, which can perform significantly more number of calculations in parallel as compared to traditional CPUs; and the rise of data generated due to the internet in the last decade deep learning techniques have become highly successful to solve problems that were up till now unsolvable by computers.

Deep learning has enabled computers to comprehend and better understand human languages, to understand the content in images and even learn from the environment in a similar fashion that humans do. Traditional machine learning enables the computers to learn and predict from data but deep learning

DOI: 10.4018/978-1-7998-3095-5.ch003

takes it one step forward. Although the foundation of theory and mathematics behind deep learning has existed even long before computers existed but it has become a reality in the 2010s due to computers having enough computational power.

Deep learning has given rise to several new technological innovations like interactive chatbots, better and natural translations, face detection, classifying images for diseases etc.

The field of deep learning is moving at a rapidly fast rate. As per the Artificial Intelligence index 2018 annual report AI papers in Scopus have increased by 7 times since 1996. In the same period, CS papers increased by 5 times. (Shoham, Yoav, et al., 2018) Every few years or even every few months, new neural network architecture or a new deep learning model replaces the previous state of the art.

The same holds for deep learning frameworks. The most commonly used framework is Tensorflow (Abadi et al., 2016) that is developed by Google and launched in 2015. Other frameworks that are less commonly used are Caffe (Jia et al., 2014), MXNet (Apache MXNet, 2017), etc. The recent addition to this is PyTorch (Paszke et al., 2017).

PyTorch is an open-source, python-based deep learning framework developed and backed by Facebook AI Research with strong support for GPU accelerated computations of deep learning models. It is based on Torch, which is written in the Lua programming language (Ierusalimsky et al., 1996). Due to its dynamic nature of computation graphs and close integration with python, it is gaining a lot of attention in the research community.

The pythonic nature of PyTorch allows us to use native python code and other python libraries like Numpy (NumPY 2018) and Scipy (Jones, 2001) to easily integrate with PyTorch code as compared to other deep learning libraries where library-specific methods are used to execute the same task.

The dynamic nature of graphs allows PyTorch to build its deep learning applications to be executed at runtime, as compared to other popular deep learning frameworks that rely on static graphs that have to be built before running the model.

A computational graph is a directed graph in which the nodes correspond to operations or variables. Variables can feed their values into operations, and operations can feed their output into other operations. In this way, each node in the graph defines a function of the variables. The values fed into the nodes and come out of the nodes are called tensors, which is a multi-dimensional array. Hence, it comprises scalars, vectors, and matrices as well as tensors of a higher rank.

The dynamic nature of computation graphs allows the creation and execution of the graph at the same time. This allows easy debugging of a neural network. The ease of debugging becomes even more prominent as the size of the neural network model increases. The dynamic nature also allows for changing the graph at run time which is particularly helpful in sequence modeling where the size of inputs is different. However, dynamic computational graphs do not offer optimization of computations nor static analysis which static computational graphs offer.

This has allowed deep learning researchers and practitioners to quickly create and evaluate new neural network architecture and approaches easily rather than waiting for the entire model to compile and run.

You might have also heard the term machine learning, sometimes even used interchangeably with deep learning. The two terms are although a subset of AI and try to achieve a similar task of making machines intelligent is quite different.

In machine learning, the engineer has to first create numerous features specific to the problem and then extract those features from the data. These features are then fed to a machine learning algorithm. This technique is called feature extraction. Where as in deep learning there is no need for feature extrac-

33 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/deep-learning-with-pytorch/257314

Related Content

Generating an Artificial Nest Building Pufferfish in a Cellular Automaton Through Behavior Decomposition

Thomas E. Portegys (2019). *International Journal of Artificial Intelligence and Machine Learning* (pp. 1-12).

www.irma-international.org/article/generating-an-artificial-nest-building-pufferfish-in-a-cellular-automaton-through-behavior-decomposition/233887

Palmprint And Dorsal Hand Vein Multi-Modal Biometric Fusion Using Deep Learning

Norah Abdullah Al-johani and Lamiaa A. Elrefaei (2020). *International Journal of Artificial Intelligence and Machine Learning* (pp. 18-42).

www.irma-international.org/article/palmprint-and-dorsal-hand-vein-multi-modal-biometric-fusion-using-deep-learning/257270

Breast Cancer Detection Using Random Forest Classifier

Pavithra Suchindran, Vanithamani R. and Judith Justin (2021). *Handbook of Research on Deep Learning-Based Image Analysis Under Constrained and Unconstrained Environments* (pp. 85-98).

www.irma-international.org/chapter/breast-cancer-detection-using-random-forest-classifier/268315

Machine Learning in Video Games

Jayakumar Kaliappan and Karpagam Sundararajan (2020). *Handbook of Research on Emerging Trends and Applications of Machine Learning* (pp. 425-443).

www.irma-international.org/chapter/machine-learning-in-video-games/247575

A Literature Review on Cross Domain Sentiment Analysis Using Machine Learning

Nancy Kansal, Lipika Goel and Sonam Gupta (2020). *International Journal of Artificial Intelligence and Machine Learning* (pp. 43-56).

www.irma-international.org/article/a-literature-review-on-cross-domain-sentiment-analysis-using-machine-learning/257271