International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 05 | May 2019 www.irjet.net p-ISSN: 2395-0072

PYTHON LIBRARIES AND PACKAGES FOR DEEP LEARNING-A SURVEY

Aishwarya. R. Menon[1], Dr. Sarathambekai[2]

[1]PG Student, Dept of Information Technology, PSG College of Technology, Tamilnadu, India
[2]SL Grade, Dept of Information Technology, PSG College of Technology, Tamilnadu, India

---

Abstract - *Deep learning has different capabilities namely the ability to draw conclusions in a manner that resembles human decision making. It does this by using a layered structure of algorithms inspired by the neural network of the human brain. The result, is a model that can learn multiple levels of representation that correspond to different levels of abstraction. The extensive selection of libraries and frameworks, simplicity and reliability of python helps in ease of developing applications .*

In this paper, the survey of different papers that used python modules and libraries for deep learning are taken and analyzed with metrics like performance, reliability and stability because of using python packages and libraries

Key Words: Deep learning, neural network, performance, reliability, stability, secure, ease of developing

1. INTRODUCTION

This survey paper depicts the various python packages that have been used in deep learning which drastically increases the performance.

1.1 Significance of Python

The significance of python is described as follows,

a. Python is a High -level language

b. Python is an Interpreter level language.

c. Python is an Object-oriented scripting language

d. Python is a portable language

e. Python is scalable in nature.

f. Python is an extendable language

g. Python is an interactive language

h. Python supports GUI Programming language

i. Python supports a broad standard library that can run on Windows, Macintosh, Linux

1.2 Significance of deep learning

The significance of deep learning is described as follows,

a. No need for feature engineering.

b. Best results with unstructured data

c. No need for labelling of data

d. Efficient at delivering high quality results

2. Python libraries and packages for deep learning

a. TensorFlow Python

TensorFlow is an open-source library for numerical computation in which it uses data flow graphs. The Google Brain Team researchers developed this with the Machine Intelligence research organization by Google. TensorFlow is open source and available to the public. It is also good for distributed computing

Installation -pip install tensorflow

b. Keras Python

A minimalist, modular, Neural Network library, Keras uses Theano or TensorFlow as a backend. It makes it easy and faster to experiment and implement ideas into results.

Keras has algorithms for optimizers, normalization, and activation layers. It also deals with Convolutional Neural Networks and lets you build sequence-based and graph-based networks. One limitation is that it doesn't support multi-GPU environments for training a network in parallel

Installation -pip install keras

c. Apache Mxnet

mxnet delivers an amazing number of language bindings for languages like C++, Python, R, JavaScript, and more. It does great with distributed computing and lets us train a network across CPU/GPU machines. The only downside is that we need a little more code to run an experiment in it

Installation -pip install mxnet

d. Caffe

Caffe is a deep learning framework that is fast and modular. This isn't a library but provides bindings into Python. Caffe can process nearly 60 million images per day on a K40 GPU. However, it isn't as easy to turn hyperparameters with it programmatically

e. Theano python

Without NumPy, we couldn't have SciPy, scikit-learn, and scikit-image. Similarly, Theano serves as a base for many. It is a library that will let you define, optimize, and evaluate mathematical expressions that involve multidimensional arrays. It is tightly integrated with NumPy and transparently uses the GPU

Installation -pip install theano

f. Microsoft Cognitive Toolkit

The Microsoft Cognitive Toolkit is a unified Deep Learning toolkit. It describes neural networks using a directed graph in computational steps

Installation -pip install cntk

e. Pytorch

PyTorch is a Tensor and Dynamic neural network in Python. It observes strong GPU acceleration, is open-source, and we can use it for applications like natural language processing

f. Lasagne

Lasagne is a lightweight Python library that helps us build and train neural networks in Theano

g. noLearn

nolearn wraps Lasagna into an API that is more user-friendly. All code it holds is compatible with scikit-learn. We can use it for applications like Deep Belief Networks (DBNs)

h. Pylearn2

PyLearn2 is a machine learning library with most functionality built on top of Theano. It is possible to write PyLearn2 plugins making use of mathematical expressions. Theano optimizes and stabilizes these for us and compiles them to the backend we want

2.1 Related works

A. *Fully convolutional networks for segmenting images from an embedded camera*

[1] describes a Fully Convolutional Network(FCN) to segment images from a compact stereo imaging sensor attached to a robot, i.e., a configurable DUO MR stereo camera embedded onto a robot to provide low-level computer vision functions. This robot is named LEIA-1 and carries a NVIDIA Jetson TK1R platform to execute high-level robotics vision, planning and decision making algorithms. Since the TK1 board is not dedicated solely to robotics vision, the FCN architecture should be light enough to consume limited resources. The FCN described is implemented and prototyped with Python, Keras, and Theano. It is trained and validated using an adaptation of the dataset known as Playing for Data to match the embedded camera specifications. The results reveal the viability of integrating FCNs to processing platforms attached to robots and use their SoC/GPU power to segment indoors/outdoors captured images

B. *Implementation of Deep-Learning based Image Classification on Single Board Computer*

[2] implements a deep-learning algorithm based on convolutional neural-network using python and tflearn for image classification. A large number of different images which contains two types of animals, namely cat and dog are used for classification. Two different structures of CNN are used, namely with two and five layers. It is shown that the CNN with higher layer performs classification process with much higher accuracy. The best CNN model with high accuracy and small loss function deployed in single board computer This system has the ability to classify two category cat and dog which have many similarity

C*. DeepMEC: Mobile Edge Caching Using Deep Learning*

[3] Uses deep learning to learn and predict the future popularity of contents to support cache decision. First, deep learning models are trained and utilized in the cloud data center to make an efficient cache decision. Then, the final cache decision is sent to each base station to store the popular contents proactively. The proposed caching scheme involves three distinct parts:

1) predicting the future class label of each content;

2) predicting the future popularity score of contents based on the predicted class label;

3) caching the predicted contents with high popularity scores. The prediction models using the Keras and Tensorflow libraries are implemented in this paper. Finally, the performance of the caching schemes is tested with a Python-based simulator. In terms of a cache hit, simulation results show that the proposed scheme outperforms 38%, convolutional recurrent neural network-based scheme outperforms 33%, and convolutional neural network-based scheme outperforms 25% compared to the baseline scheme

D. *High Performance Text Recognition using a Hybrid Convolutional-LSTM Implementation*

[4] describes a new, open-source line recognizer combining deep convolutional networks and LSTMs, implemented in PyTorch and using CUDA kernels for speed. Experimental results compare the performance of different combinations of geometric normalization, 1D LSTM, deep convolutional networks, and 2D LSTM networks.

An important result is that while deep hybrid networks without geometric text line normalization outperform 1D LSTM networks with geometric normalization, deep hybrid

networks with geometric text line normalization still outperform all other networks. The best networks achieve a throughput of more than 100 lines per second and test set error rates on UW3 of 0.25%.

E: *Acceleration of Image Classification with Caffe framework using FPGA*

[5] has developed a platform for the efficient deployment and acceleration of Caffe framework on embedded systems that are based on the Zynq SoC. The most computational intensive part of image classification is the processing of the convolution layers of the deep learning algorithms and more specifically the GEMM (general matrix multiplication) function calls.

In the proposed framework, a hardware accelerator has been implemented, validated and optimized using Xilinx SDSoC Development Environment to perform the GEMM function. The accelerator that was developed achieves up to 98x speed-up compared with the simple ARM CPU implementation. The results showed that the mapping of Caffe on the FPGA-based Zynq takes advantage of the low-power, customizable and programmable fabric and ultimately reduces time and power consumption of image classification

3. CONCLUSIONS

This survey paper comprises of various python libraries and modules that are been used in Deep learning. With the usage of python, it has eliminated various issues such as lack of stability, lack of reliability. The code size has been relatively reduced such that it enhances the throughput and reduces the memory usage.

Importing the various python networking libraries have made the automated and flexible environment

Since python has inbuilt libraries and also provides a provision to include external modules, it is easier to code the project.

Using the python libraries, it is easy to segment the images from LEIA-1 robot's camera[1].

It can also be used to classify two images having many similarities with greater accuracy[2]

Caching method to improve the cache hit probability, backhaul usage, and video contents access delay[3] using a 2 step deep learning based prediction algorithm

Implementation of a text line recognizer for printed text based[4] which would replace the existing text line recognizer in systems like OCRopus, and yields an order of magnitude speedup, allowing around 100 text lines per second to be recognized on a standard desktop machine with a GPU

Thus, python provides various libraries and modules that can be used in making of hardware accelerators with low power consumption[5].

REFERENCES

[1] Bengio, Y. (2009). Learning Deep Architectures for AI. Foundations and Trends® in Machine Learning (Vol. 2). https://doi.org/10.1561/2200000006

[2] D. Liu, B. Chen, C. Yang, and A. F. Molisch, ''Caching at the wireless edge: Design aspects, challenges, and future directions,'' IEEE Commun. Mag., vol. 54, no. 9, pp. 22–28, Sep. 2016, doi: 10.1109/MCOM.2016.7565183.

[3] F Asad et al. "High Performance OCR for CameraCaptured Blurred Documents with LSTM Networks". In: 2016 12th IAPR Workshop on Document Analysis Systems (DAS). ieeexplore.ieee.org, Apr. 2016, pp. 7– 12

[4] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: (Apr. 2014). arXiv: 1409 . 1556 [cs.CV]

[5] S. Ioffe and C. Szegedy. (2015). ''Batch normalization: Accelerating deep network training by reducing internal covariate shift.'' [Online]. Available: https://arxiv.org/abs/1502.03167