



Theano in Python

Last Updated : 27 Dec, 2023

Theano is a Python library that allows us to evaluate mathematical operations including multi-dimensional arrays efficiently. It is mostly used in building **Deep Learning Projects**. Theano works way faster on the Graphics Processing Unit (GPU) rather than on the CPU.

This article will help you to understand *What is Theano in Python and how to install and work with Theano in [Python](#)*.

Theano in Python

Theano attains high speeds that give tough competition to C implementations for problems involving large amounts of data. It can take advantage of GPUs, making it perform better than [C](#) on a CPU by considerable orders of magnitude under certain circumstances.

Theano also knows how to take structures and convert them into very efficient code that uses [Numpy](#) and some native libraries. It is mainly designed to handle the types of computation required for large neural network algorithms used in [Deep Learning](#). That is why, Theanos is a trendy library in the field of Deep Learning.

How to install Theano?

Before installing Theano in your system, You need to make sure that [Python](#) and Numpy is already installed in your system. To install Python , You can refer to this article.

[How to download and install Python Latest Version on Windows](#)

To install NumPy in your system , You can refer to these below mentioned articles.

- [How to Install Numpy on Windows?](#)
- [How to Install Numpy on Linux?](#)
- [How to Install Numpy on MacOS?](#)

To install Theano, You need to run this command in your terminal.

```
pip install theano
```

Several symbols we will need to use are in the **tensor** subpackage of Theano. We often import such packages with a handy name, let's say, T.

```
from theano import *  
import theano.tensor as T
```

Why Theano Python Library ?

Theano is a sort of hybrid between numpy and sympy, an attempt is made to combine the two into one powerful library. Some advantages of theano are as follows:

- **Stability Optimization:** Theano can find out some unstable expressions and can use more stable means to evaluate them.
- **Execution Speed Optimization:** As mentioned earlier, theano can make use of recent GPUs and execute parts of expressions in your CPU or GPU, making it much faster than Python.
- **Symbolic Differentiation:** Theano is smart enough to automatically create symbolic graphs for computing gradients.

Basics of Theano

Theano is a Python library that allows you to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays efficiently. Some Theano implementations are as follows.

Subtracting two scalars

Python program showing subtraction of two scalars

Python

```
import theano
from theano import tensor
# Declaring variables
a = tensor.dscalar()
b = tensor.dscalar()
# Subtracting
res = a - b
# Converting it to a callable object
# so that it takes matrix as parameters
func = theano.function([a, b], res)
# Calling function
assert 20.0 == func(30.5, 10.5)
```

It will not provide any output as the assertion of two numbers matches the number given, hence it results into a true value.

Adding two scalars

Python

```
# Python program showing
# addition of two scalars
# Addition of two scalars
import numpy
import theano.tensor as T
from theano import function
# Declaring two variables
x = T.dscalar('x')
y = T.dscalar('y')
# Summing up the two numbers
z = x + y
# Converting it to a callable object
# so that it takes matrix as parameters
f = function([x, y], z)
f(5, 7)
```

Output:

```
array(12.0)
```

Adding two matrices

Python program showing addition of two matrices.

Python

```
import numpy
import theano.tensor as T
from theano import function
x = T.dmatrix('x')
y = T.dmatrix('y')
z = x + y
f = function([x, y], z)
f([[30, 50], [2, 3]], [[60, 70], [3, 4]])
```

Output:

```
array([[ 90., 120.],
       [  5.,  7.]])
```

Logistic function using theano :

Let's try to compute the logistic curve, which is given by:

$$s(x) = \frac{1}{1 + e^{-x}}$$

Logistic Function

Python program to illustrate logistic sigmoid function using theano

Python

```
# Load theano library
import theano
from theano import tensor
# Declaring variable
a = tensor.dmatrix('a')
```

```
# Sigmoid function
sig = 1 / (1 + tensor.exp(-a))
# Now it takes matrix as parameters
log = theano.function([a], sig)
# Calling function
print(log([[0, 1], [-1, -2]]))
```

Output :

```
[[0.5          0.73105858
 0.26894142    0.11920292]]
```

Theano is a foundation library mainly used for deep learning research and development and directly to create deep learning models or by convenient libraries such as Keras. It supports both convolutional networks and recurrent networks, as well as combinations of the two.

Conclusion

In the dynamic landscape of deep learning, Theano shines as a versatile and efficient library. As we continue to delve into the intricacies of AI, Theano remains a steadfast ally, propelling us into a future where innovation knows no bounds.



aish...



7

Previous Article

[Python Version History](#)

Next Article

Similar Reads

Important differences between Python 2.x and Python 3.x with examples

In this article, we will see some important differences between Python 2.x and Python 3.x with the help of some examples. Differences between Python 2.x an...

5 min read

Reading Python File-Like Objects from C | Python

Writing C extension code that consumes data from any Python file-like object (e.g., normal files, StringIO objects, etc.). `read()` method has to be repeatedly...

3 min read

Python | Add Logging to a Python Script

In this article, we will learn how to have scripts and simple programs to write diagnostic information to log files. Code #1 : Using the logging module to add...

2 min read

Python | Add Logging to Python Libraries

In this article, we will learn how to add a logging capability to a library, but don't want it to interfere with programs that don't use logging. For libraries that want...

2 min read

Python | Index of Non-Zero elements in Python list

Sometimes, while working with python list, we can have a problem in which we need to find positions of all the integers other than 0. This can have application i...

6 min read

MySQL-Connector-Python module in Python

MySQL is a Relational Database Management System (RDBMS) whereas the structured Query Language (SQL) is the language used for handling the RDBMS...

2 min read

Python - Read blob object in python using wand library

BLOB stands for Binary Large Object. A blob is a data type that can store binary data. This is different than most other data types used in databases, such as...

2 min read

twitter-text-python (ttp) module - Python

twitter-text-python is a Tweet parser and formatter for Python. Amongst many things, the tasks that can be performed by this module are : reply : The usernam...

3 min read

Reusable piece of python functionality for wrapping arbitrary blocks of co...

Context Managers are the tools for wrapping around arbitrary (free-form) blocks of code. One of the primary reasons to use a context manager is resource...

7 min read

Creating and updating PowerPoint Presentations in Python using python -...

python-pptx is library used to create/edit a PowerPoint (.pptx) files. This won't work on MS office 2003 and previous versions. We can add shapes, paragraphs,...

4 min read

Article Tags : [Python](#) [Python-Library](#)

Practice Tags : [python](#)



Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305



Company

[About Us](#)
[Legal](#)
[In Media](#)
[Contact Us](#)
[Advertise with us](#)
[GFG Corporate Solution](#)
[Placement Training Program](#)

Languages

[Python](#)
[Java](#)
[C++](#)
[PHP](#)
[GoLang](#)
[SQL](#)
[R Language](#)

[GeeksforGeeks Community](#)[Android Tutorial](#)[Tutorials Archive](#)

DSA

[Data Structures](#)[Algorithms](#)[DSA for Beginners](#)[Basic DSA Problems](#)[DSA Roadmap](#)[Top 100 DSA Interview Problems](#)[DSA Roadmap by Sandeep Jain](#)[All Cheat Sheets](#)

Web Technologies

[HTML](#)[CSS](#)[JavaScript](#)[TypeScript](#)[ReactJS](#)[NextJS](#)[Bootstrap](#)[Web Design](#)

Computer Science

[Operating Systems](#)[Computer Network](#)[Database Management System](#)[Software Engineering](#)[Digital Logic Design](#)[Engineering Maths](#)[Software Development](#)[Software Testing](#)

System Design

[High Level Design](#)[Low Level Design](#)[UML Diagrams](#)[Interview Guide](#)[Design Patterns](#)[OOAD](#)[System Design Bootcamp](#)[Interview Questions](#)

School Subjects

[Mathematics](#)[Physics](#)[Chemistry](#)[Biology](#)

Data Science & ML

[Data Science With Python](#)[Data Science For Beginner](#)[Machine Learning](#)[ML Maths](#)[Data Visualisation](#)[Pandas](#)[NumPy](#)[NLP](#)[Deep Learning](#)

Python Tutorial

[Python Programming Examples](#)[Python Projects](#)[Python Tkinter](#)[Web Scraping](#)[OpenCV Tutorial](#)[Python Interview Question](#)[Django](#)

DevOps

[Git](#)[Linux](#)[AWS](#)[Docker](#)[Kubernetes](#)[Azure](#)[GCP](#)[DevOps Roadmap](#)

Interview Preparation

[Competitive Programming](#)[Top DS or Algo for CP](#)[Company-Wise Recruitment Process](#)[Company-Wise Preparation](#)[Aptitude Preparation](#)[Puzzles](#)

GeeksforGeeks Videos

[DSA](#)[Python](#)[Java](#)[C++](#)

Social Science	Web Development
English Grammar	Data Science
Commerce	CS Subjects
World GK	

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved