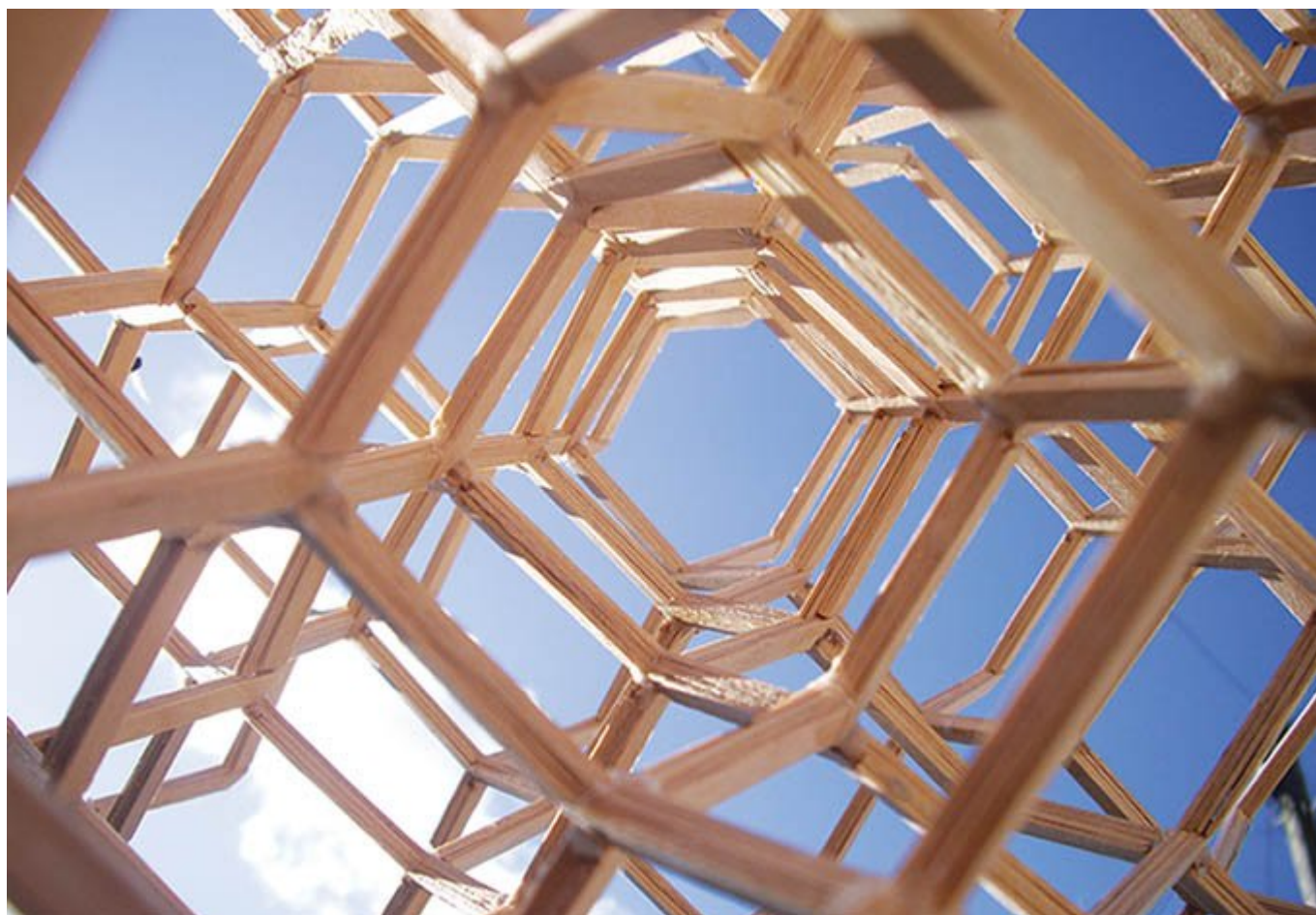


معماری شبکه‌های عصبی عمیق با استفاده از فریم‌ورک TensorFlow (معماری شبکه‌های عصبی عمیق)

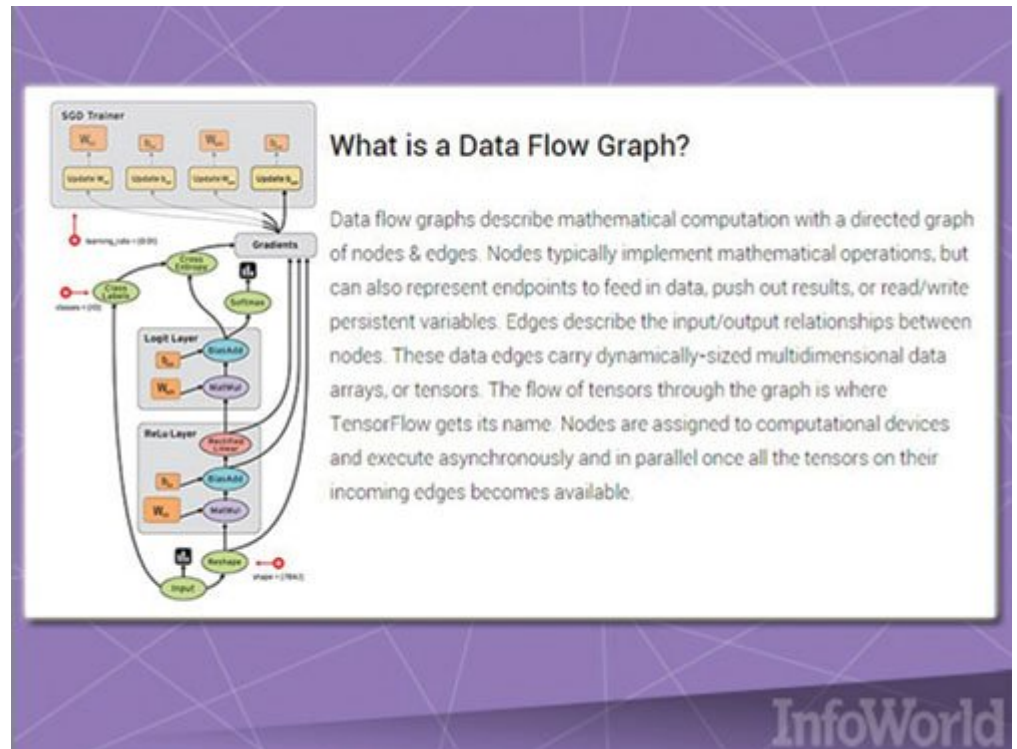


این مطلب یکی از مجموعه مقالات پرونده ویژه «[هوش مصنوعی و اینترنت](#)» شماره ۱۸۱ ماهنامه شبکه است. علاقه‌مندان می‌توانند کل این پرونده ویژه را از روی [QR Code](#) دانلود کنند.

این مطلب یکی از مجموعه مقالات پرونده ویژه «[هوش مصنوعی و اینترنت](#)» شماره ۱۸۱ ماهنامه شبکه است. علاقه‌مندان می‌توانند کل این پرونده ویژه را از روی [QR Code](#) دانلود کنند.

Google TensorFlow

فریم‌ورک TensorFlow یکی از محبوب‌ترین و قدرتمندترین ابزارها برای ساخت و آموزش شبکه‌های عصبی عمیق است. این فریم‌ورک توسط Google توسعه داده شده و از آن برای اجرای مدل‌های یادگیری عمیق استفاده می‌شود. TensorFlow دارای یک رابط کاربری ساده و مستندات جامع است که به توسعه‌دهندگان کمک می‌کند تا به راحتی مدل‌های خود را بسازند و آموزش دهند.



TensorFlow 7: 7

TensorFlow 7: 7

7: 7



7: 7

7: 7

Microsoft Computational Network Toolkit

Microsoft Computational Network Toolkit


```

In [1]: import veles

In [2]: launcher=veles("veles/znicz/samples/MnistSimple/mnist.py", stealth=True, backend="oci", matplotlib_backend="webagg")

```



```

Version 0.8.9 Tue, 12 May 2015 11:51:57 +0300
Copyright © 2013 Samsung Electronics Co., Ltd.
Released under Apache 2.0 license.
https://velesnet.ml
https://github.com/samsung/veles/issues

INFO:Main:loading workflow "/.../znicz/samples/MnistSimple/mnist.py"...
INFO:Main:Applying the configuration from "/.../znicz/samples/MnistSimple/mnist_config.py...
INFO:Launcher:My Python is CPython 3.4.1
INFO:Launcher:My PID is 32514
INFO:Launcher:My time is 2015-05-14 09:52:16.789127
INFO:Launcher:My ID is 7bc792aa-e93e-4819-88f9-53c64a970ec5
INFO:Launcher:My log ID is 7bc792aa-e93e-4819-88f9-53c64a970ec5
INFO:Main:Created <MnistSimple.mnist.MnistWorkflow object at 0x7f8fc4f18f60> with 21 units
INFO:GraphicsServer:Publishing to inproc://veles-plots; ipc:///tmp/veles-ipc-plots-ws2s718r; epgm://eth2;239.192.1.1:16080; epgm:.
INFO:OpenCLDevice:Selected the following OpenCL configuration:
+-----+
| device | dtype | rating | BLOCK_SIZE | version |
+-----+
| NVIDIA Corporation/GeForce GTX TITAN/4318 | double | 1.000 | 27 | 1.1 |
| NVIDIA Corporation/GeForce GTX TITAN/4318 | float | 1.000 | 30 | 1.1 |
+-----+
INFO:MnistWorkflow:Initializing units in MnistWorkflow...
INFO:Mnistloader:loading from original MNIST files...
INFO:Mnistloader:Minibatch size is set to 88
INFO:Mnistloader:Samples number: test: 0, validation: 10000, train: 60000
INFO:Mnistloader:Normalizing to linear...
INFO:Mnistloader:There are 10 unique labels
INFO:Mnistloader:train label cardinalities: min: 5421 ("5"), max: 6742 ("1"), avg: 6000, o: 322 (5%)
INFO:Mnistloader:validation label cardinalities: min: 892 ("5"), max: 1135 ("1"), avg: 1000, o: 59 (5%)
INFO:Mnistloader:OK: train and validation labels have the same distributions (X-square test's p-value is 1.000)
INFO:Mnistloader:Will try to store the entire dataset on the device
100% [=====] 21 units were initialized
INFO:Launcher:Reactor is running
INFO:Main:Returned the control
INFO:Mnistloader:Served 88 samples (0 epochs, 0.1% current); jobs failed: 0/pending: 1

```

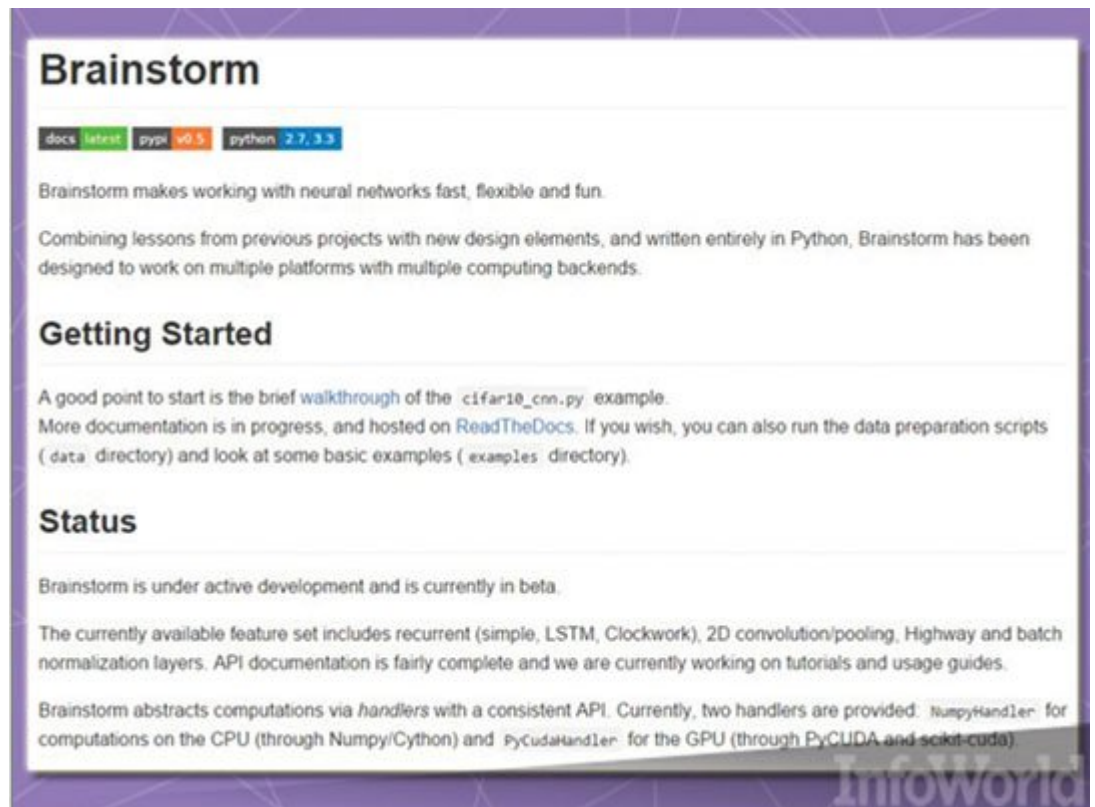


□□□□ □□□□□□□□ □□□□ □□□□□□□□ □□□□□□ *Veles* :9 □□□

[illegible]

Brainstorm

[illegible]

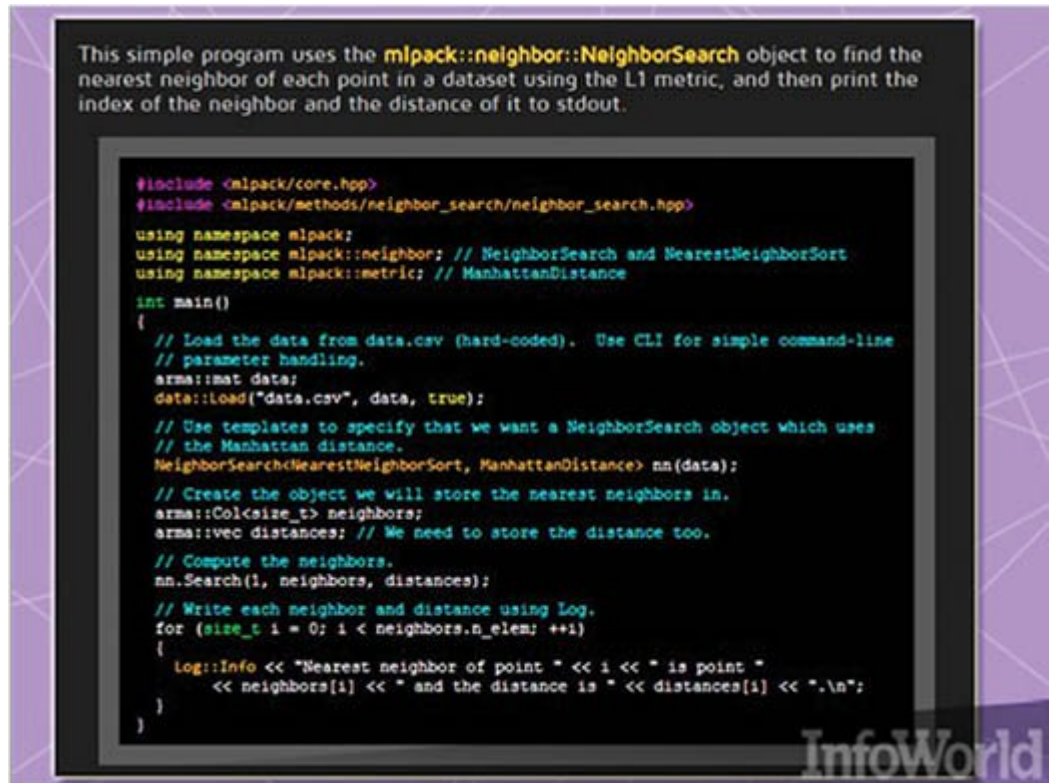


BrainStorm :10

BrainStorm is a Python library for working with neural networks. It is designed to be fast, flexible and fun. It combines lessons from previous projects with new design elements, and is written entirely in Python. BrainStorm has been designed to work on multiple platforms with multiple computing backends. A good point to start is the brief walkthrough of the `cifar10_cnn.py` example. More documentation is in progress, and hosted on ReadTheDocs. If you wish, you can also run the data preparation scripts (`data` directory) and look at some basic examples (`examples` directory). Brainstorm is under active development and is currently in beta. The currently available feature set includes recurrent (simple, LSTM, Clockwork), 2D convolution/pooling, Highway and batch normalization layers. API documentation is fairly complete and we are currently working on tutorials and usage guides. Brainstorm abstracts computations via *handlers* with a consistent API. Currently, two handlers are provided: `NumpyHandler` for computations on the CPU (through Numpy/Cython) and `PyCudaHandler` for the GPU (through PyCUDA and scikit-cuda).

mlpack 2

mlpack 2 is a C++ machine learning library. It is designed to be fast, flexible and fun. It combines lessons from previous projects with new design elements, and is written entirely in C++. mlpack 2 has been designed to work on multiple platforms with multiple computing backends. A good point to start is the brief walkthrough of the `cifar10_cnn.py` example. More documentation is in progress, and hosted on ReadTheDocs. If you wish, you can also run the data preparation scripts (`data` directory) and look at some basic examples (`examples` directory). Brainstorm is under active development and is currently in beta. The currently available feature set includes recurrent (simple, LSTM, Clockwork), 2D convolution/pooling, Highway and batch normalization layers. API documentation is fairly complete and we are currently working on tutorials and usage guides. Brainstorm abstracts computations via *handlers* with a consistent API. Currently, two handlers are provided: `NumpyHandler` for computations on the CPU (through Numpy/Cython) and `PyCudaHandler` for the GPU (through PyCUDA and scikit-cuda).

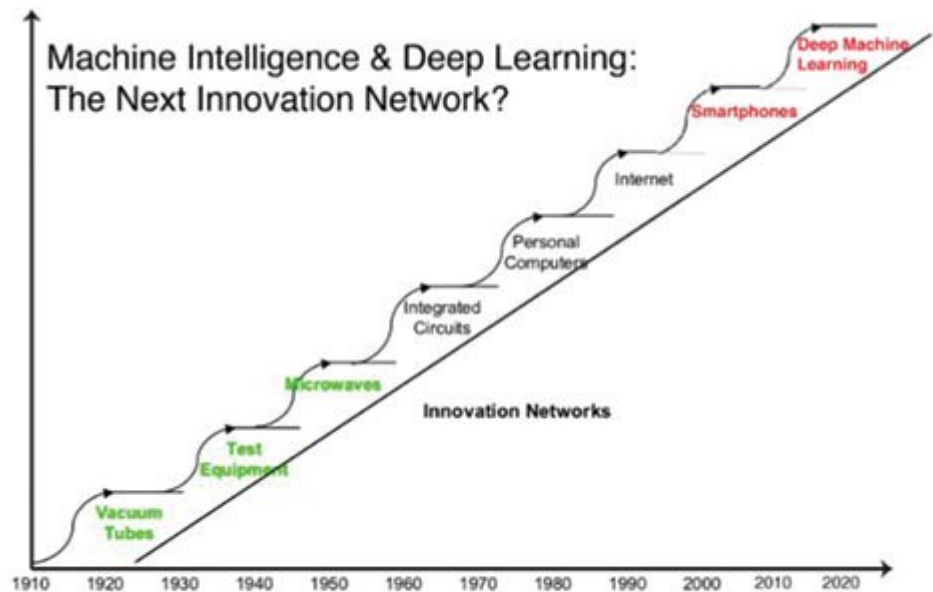


mlpack2 000000 00 00 000000 :11 000

000000 0000 0000 00 .000 0000 0000000000 0 0000 00000 00000 00000 00000 0000 2 0000
 000 000000 0 0000 000000 000 00 00000 00000000000 00 000000 0 0000 0000 00000 0000 0000000000
 .00000 000000000 000 000000 0000 00 000000 00000000 00 000000 000000000 .000000 000000000 000000
 00000000 mlpack 00 00000000000 000000 00 R 000000 000000000000 000 000000000000 00000000 00000000
 0000 .000 000000000 000 0000 0000 00 000000000 000 00 0000 0000 000000000000 000000 000 00000
 0000 .000 000 000000 RcppMLPACK 000 00 00000000 R 00 000000000 000 00 00000000 0000 0000
 0000 000000 00 0000 00000000 000000 00000000 00 0000 000000 000000 000000 000000 000000 000000
 .00000 0000000

Marvin

0000 00000 000 00 0000 0000 00000000 0 00000 00000 00000 000000 0000» :000 0000 000000000 00000
 00 00 000000 0000 00000 000 00 00000 0000000000 00000 000000 0 0000 00000 0000000 00 0000 «.000
 0000000 2000 000 000000 00 0000000 90 000 000 .000 0000 00 00000 000 00 0000 00000 00000 000 00
 00 000 00 000000 00000000 000000000 2010 000 00 2000 000 00 0000 00000000 0000000000 00000
 00 00 000000 .000 000 0000 000000 00000000 0000000 00000 000 00 000 000 00 0000 0000000000 00000
 Marvin (12 000).000 00000 00000 00000000 000000 00 00 0000 00000 0 0000 00000 000000 2020 000
 Princeton 00 000 000000 00000000 0000 0000 000000 .000 0000 000 0000 0000000 0000000 00 0000 000
 .000 0000 00000 00 00 Vision Group



from www.computervisionblog.com

2020 年 10 月 19 日 星期日 12 月 12 日

[illegible]

□□□□□□ □□□□ □□ □□□□□□□□□□ □□□□ □□□□ □□□□ □□□□ Marvin :13 □□□

AlexNet, VGG, GoogLeNet 等等。 这些模型在 ImageNet 数据集上取得了优异的成绩，为深度学习在计算机视觉领域的应用奠定了基础。

Neon



! 0000 000 00 000000 000000 0000



0000 00000 00000 00 0000 0000000 00000 0000000 0000000 0000



00000000 0 0000000 00000000 :000000 00000000



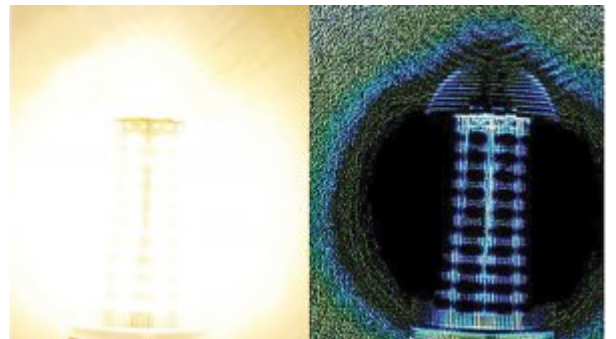
000000 00000000 00 0000 000000 0 0000000 00 0000 0 00 00000000 00000000



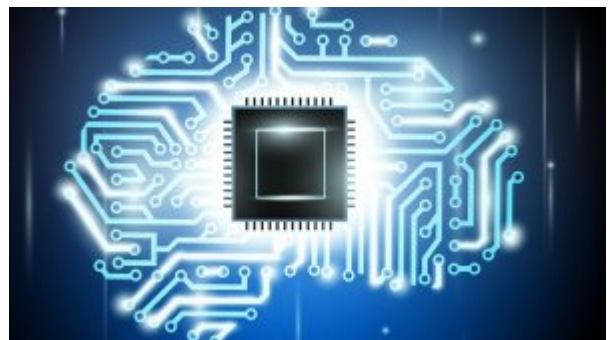
SalesPredict



1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009



!~~~~~!

[illegible]

:☐☐☐☐☐☐

شماره پنجم - فصل دوم

:مجموعه آثار

شماره پنجم - فصل دوم

:مجموعه آثار

11:40 - 05/05/1395

:مجموعه آثار

مجموعه آثار - مجموعه آثار - مجموعه آثار - مجموعه آثار - مجموعه آثار

<https://www.shabakeh-mag.com/cover-story/3813>:مجموعه آثار