

ディープラーニング最新動向と技術情報

なぜGPUがディープラーニングに向いているのか

エヌビディア

ディープラーニングソリューションアーキテクト 兼 CUDAエンジニア 村上 真奈



AGENDA

ディープラーニングとは

なぜGPUがディープラーニングに向いているか

NVIDIA DIGITS

ディープラーニングとは



Deep Neural
Networks

「ディープラーニングのおかげで、わずか一年で**音声認識**の誤認識率が23%から8%に下がりました。

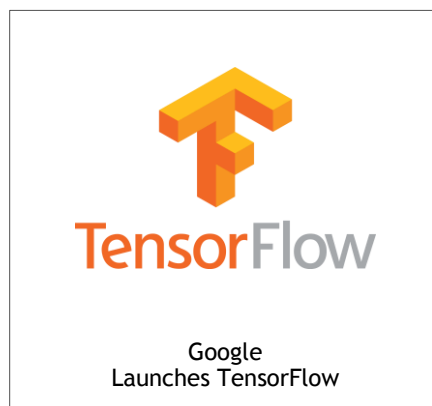
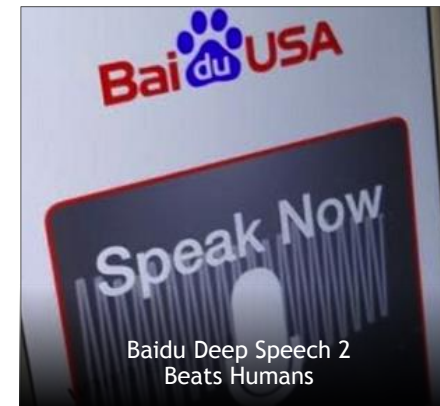
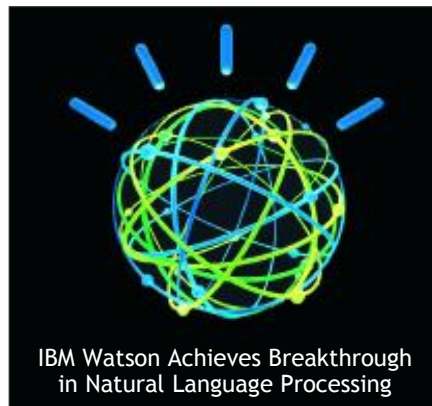
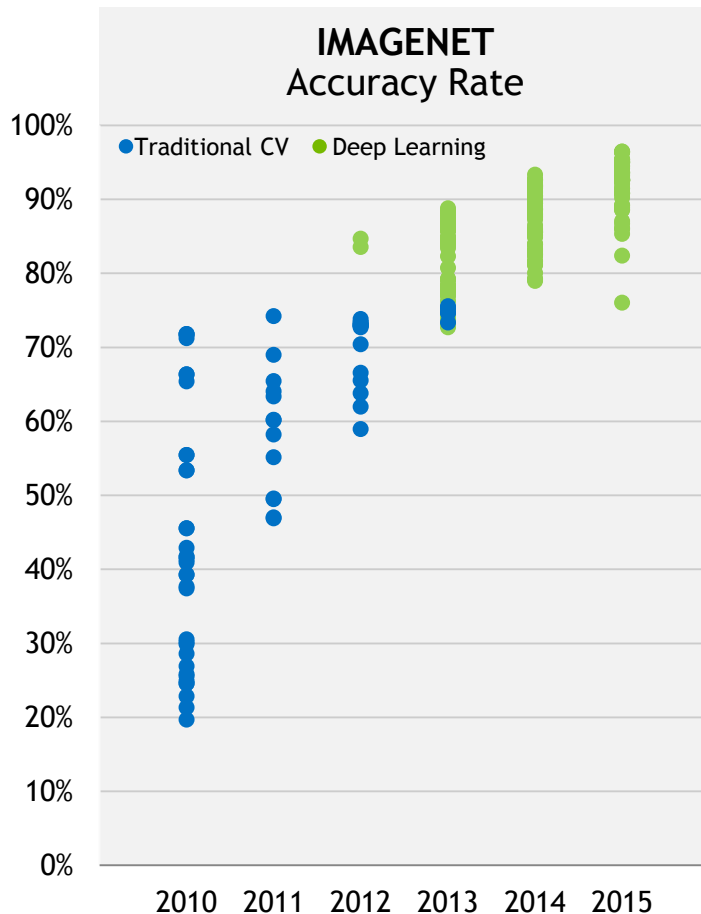
自動写真整理 とインテリジェント**写真検索** によって、まさに探していたイメージを見つけることができます。

文脈を理解し、**自然言語処理**、**言語翻訳** によって、ユーザーに即座に返事ができるのです。」

スンダル・ピチャイ
グーグル上級副社長



ディープラーニングによって加速するAI競争



ディープラーニングの開発環境が増加

大学

TORCH



CAFFE



THEANO



MATCONVNET



MOCHA.JL



PURINE



MINERVA



MXNET*



BIG SUR



TENSORFLOW



WATSON



CNTK



スタートアップ°

CHAINER



DL4J



KERAS



OPENDEEP



NVIDIA GPU プラットフォーム

*U. Washington, CMU, Stanford, TuSimple, NYU, Microsoft, U. Alberta, MIT, NYU Shanghai

幅広いディープラーニングの応用例

応用分野

顔、場所、物体認識

音声認識および翻訳

自然言語処理

ビデオ検索および分析

インデキシングおよび検索

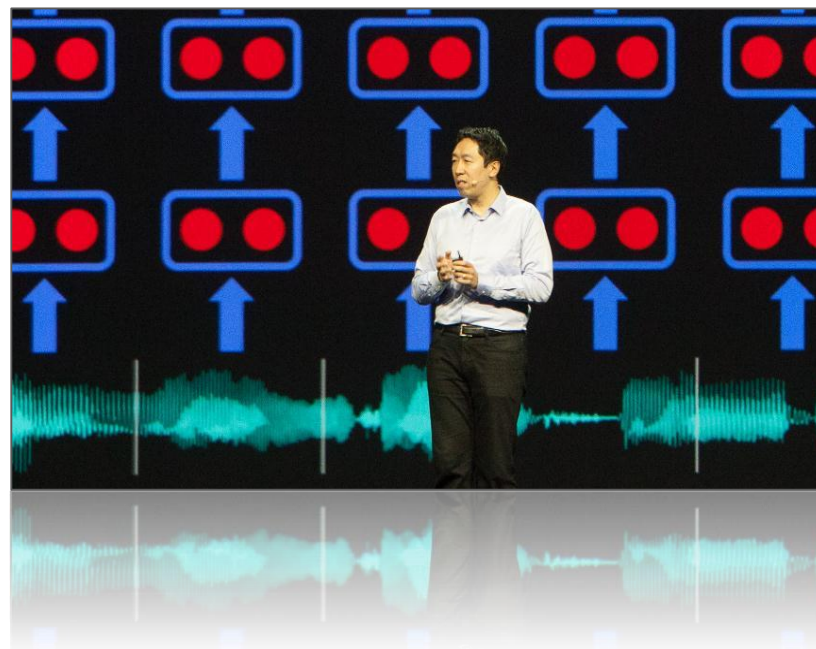
レコメンデーション エンジン

コンピュータ ビジョン

自動運転

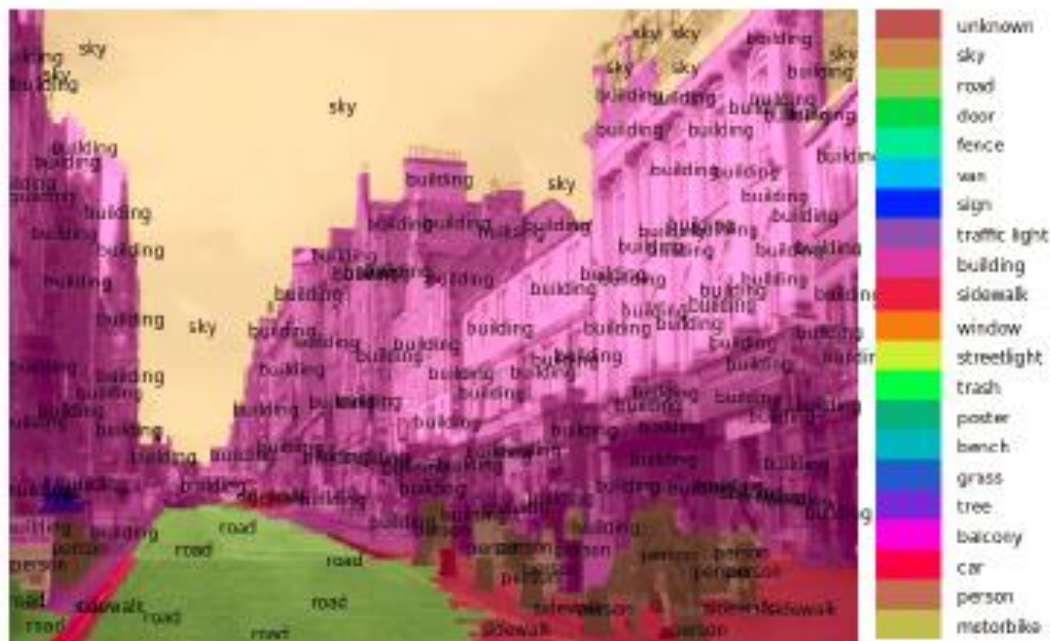
「検索の50%は音声か画像になるだろう」

Andrew Ng, Baidu



ディープラーニング事例

シーンセグメンテーションとシーン分類

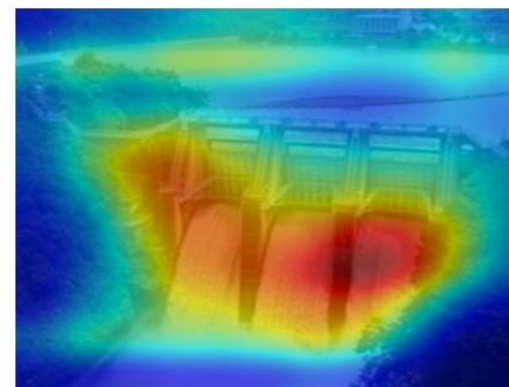


Farabet, PAMI 2013



Predictions:

- **Type of environment:** outdoor
- **Semantic categories:** dam:1.00,
- **SUN scene attributes:** naturallight, openarea, man-made, foliage, leaves, moistdamp, runningwater, vegetation, nohorizon, shrubbery
- **Informative region for the category *dam* is:**



<http://places.csail.mit.edu/> B. Zhou et al. NIPS 14

ディープラーニング事例

絵画様式認識と模倣

VisLAB: RESULTS

experiment setting style
split actual_label predicted_label confidence page



conf: 1.56 | gt: +



conf: 1.51 | gt: +



conf: 1.49 | gt: +



conf: 1.41 | gt: +



conf: 1.41 | gt: +



conf: 1.40 | gt: +

<http://demo.vislab.berkeleyvision.org/>



The Deep Forger
@DeepForger

Follow

#StyleNet #NeuralArt, comission from
@FinnSiegmund and style by Pablo Picasso.



ディープラーニング事例

ビデオ中の活動認識

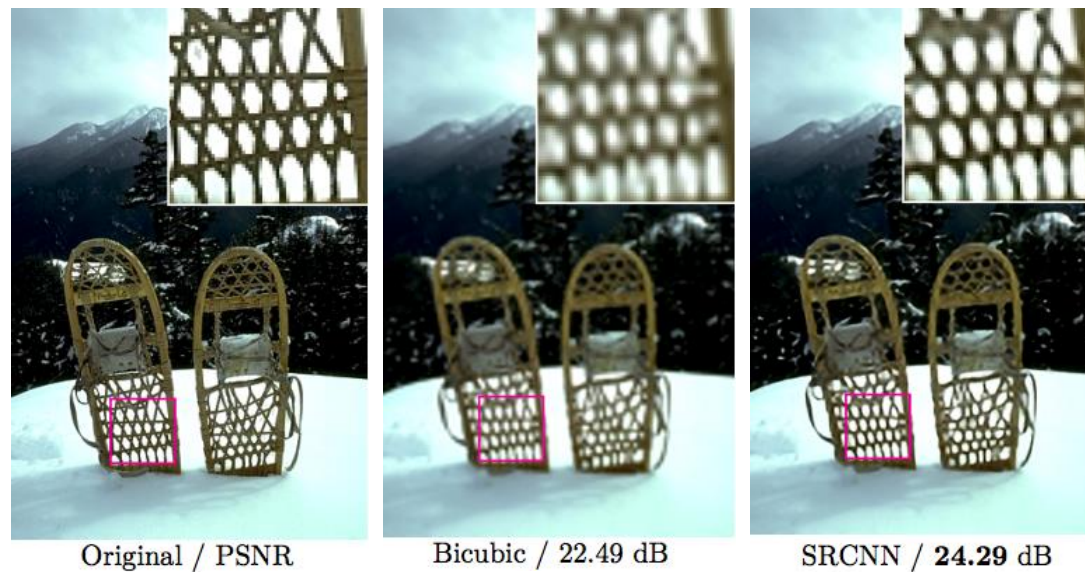


ディープラーニングの事例

画像ノイズ除去および高解像度化



Eigen , ICCV 2010



Dong et al, ECCV 2014

ディープラーニングの事例

エンドツーエンド音声認識

Baidu system is significantly simpler than traditional speech recognition systems, which rely on laboriously engineered processing pipelines.

Deep speech does not need hand-designed components to model background noise, speaker variation etc, but instead directly learns them

System	Clean (94)	Noisy (82)	Combined (176)
Apple Dictation	14.24	43.76	26.73
Bing Speech	11.73	36.12	22.05
Google API	6.64	30.47	16.72
wit.ai	7.94	35.06	19.41
Deep Speech	6.56	19.06	11.85

Table 4: Results (%WER) for 5 systems evaluated on the original audio. Scores are reported *only* for utterances with predictions given by all systems. The number in parentheses next to each dataset, e.g. Clean (94), is the number of utterances scored.

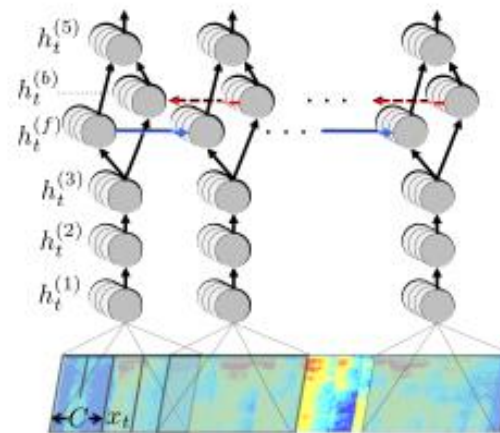


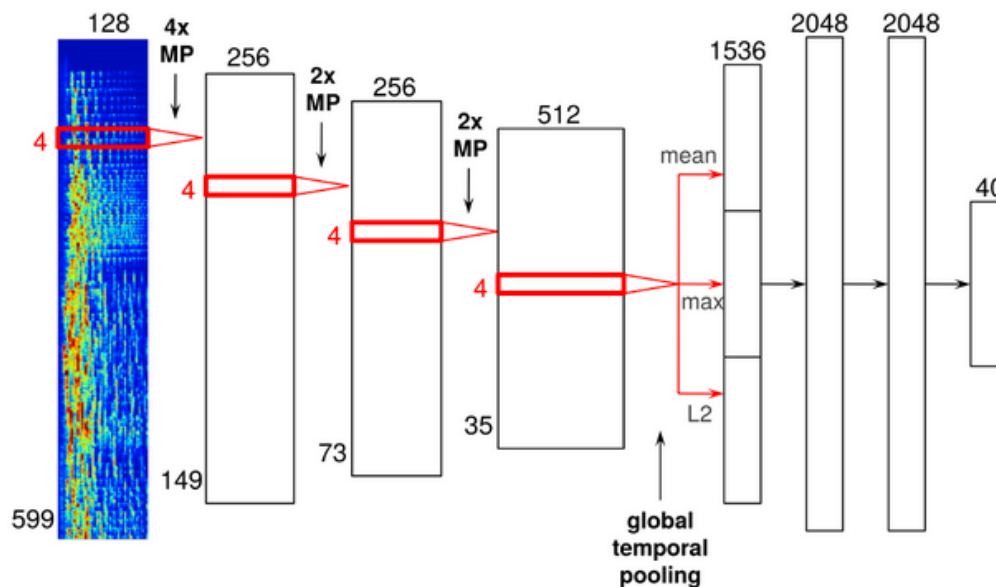
Figure 1: Structure of our RNN model and notation.

ディープラーニングの事例

曲調をベースとした音楽のリコメンデーション

<http://benanne.github.io/2014/08/05/spotify-cnns.html>

音楽のスペクトログラム
を入力

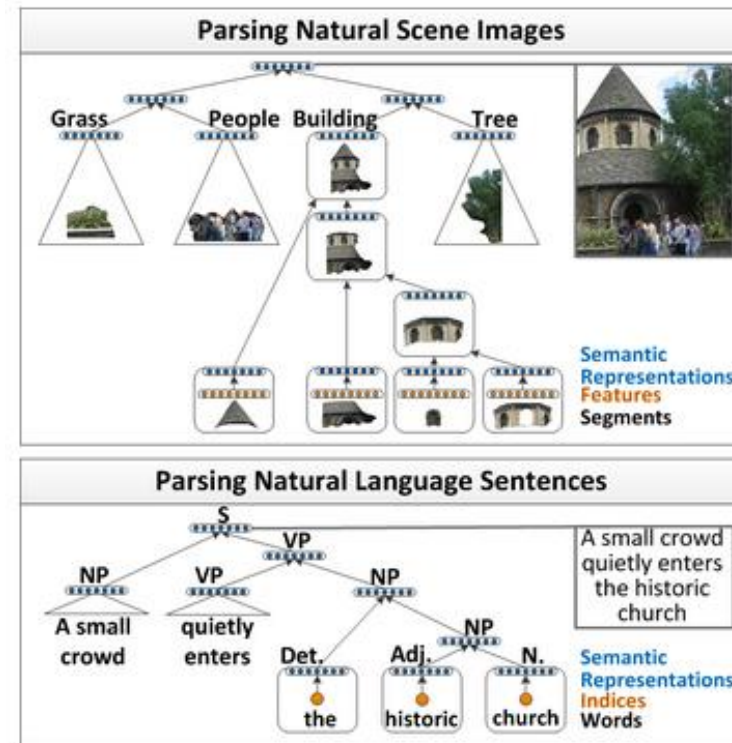
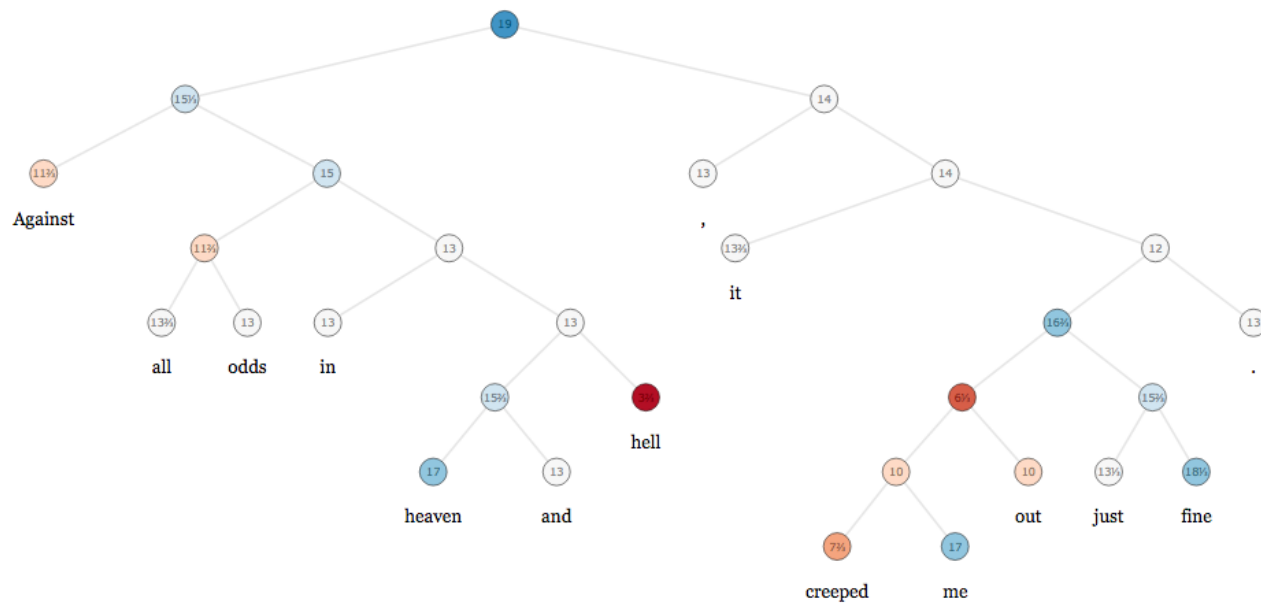


“タグ”出力

ディープラーニングの事例

言語理解

言語理解



ディープラーニングの事例

ゲームプレイ



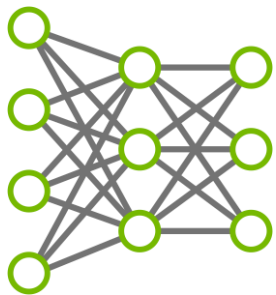
Figure 1: Screen shots from five Atari 2600 Games: (Left-to-right) Pong, Breakout, Space Invaders, Seaquest, Beam Rider



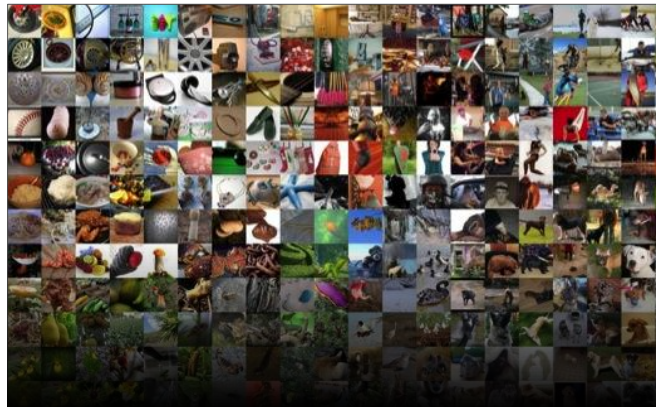
Google DeepMind

なぜGPUがディープラーニングに向いているか

ディープラーニングを加速する3つの要因



DNN



ビッグデータ



GPU

“The GPU is the workhorse of modern A.I.”

POPULAR
SCIENCE

機械学習ソフトウェア



木
トレーニング

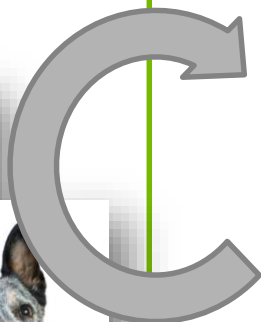


猫

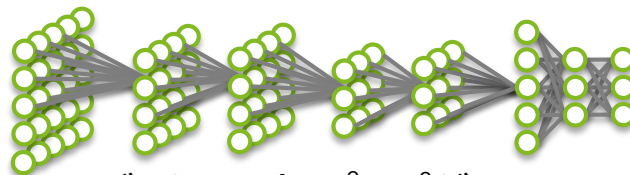


犬

反復



フォワード プロパゲーション



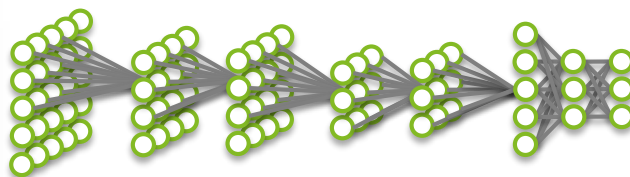
“亀”

バックワード プロパゲーション



“亀” から “犬” へ
計算の重み付けを更新

推論



“猫”

学習済みモデル

多量なトレーニングデータと多数の行列演算

典型的なネットワーク例

目的

顔認識

トレーニングデータ

1,000万～1億イメージ

ネットワークアーキテクチャ

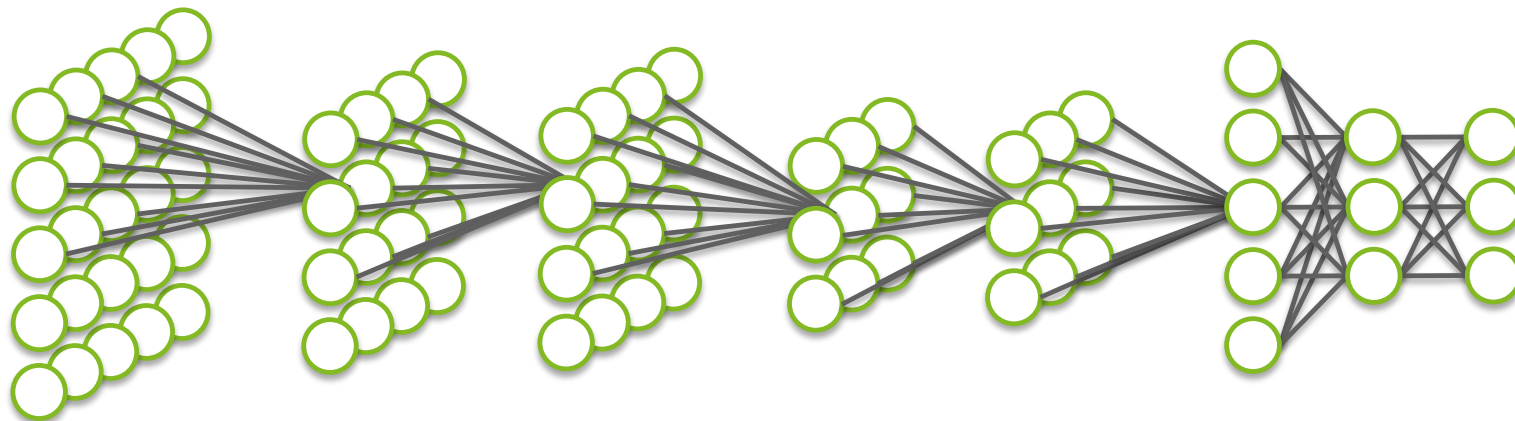
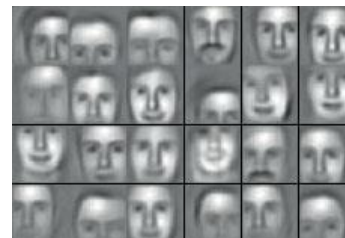
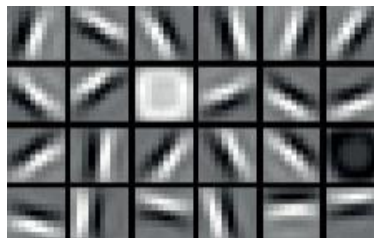
10 層

10 億パラメータ

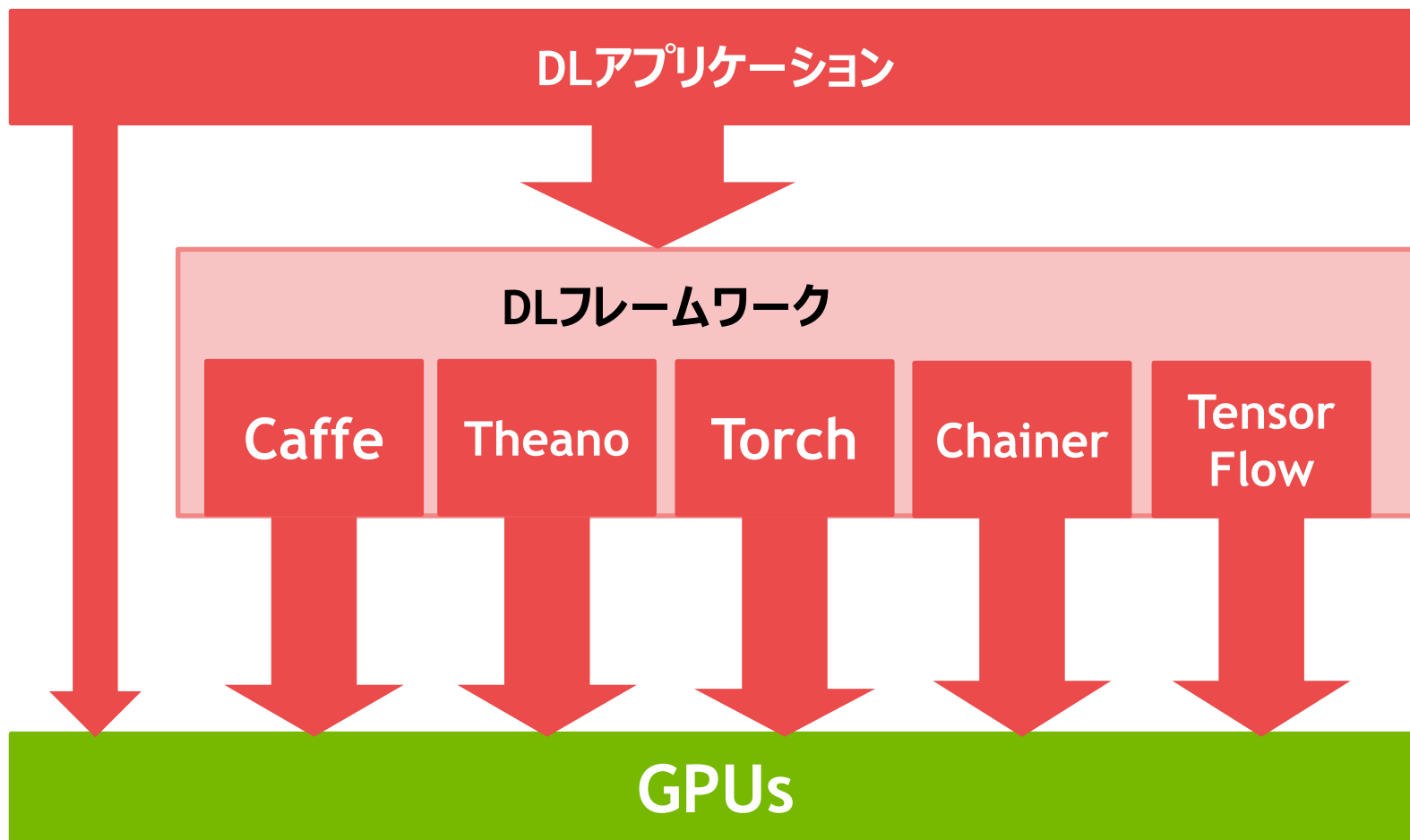
ラーニングアルゴリズム

30 エクサフロップスの計算量

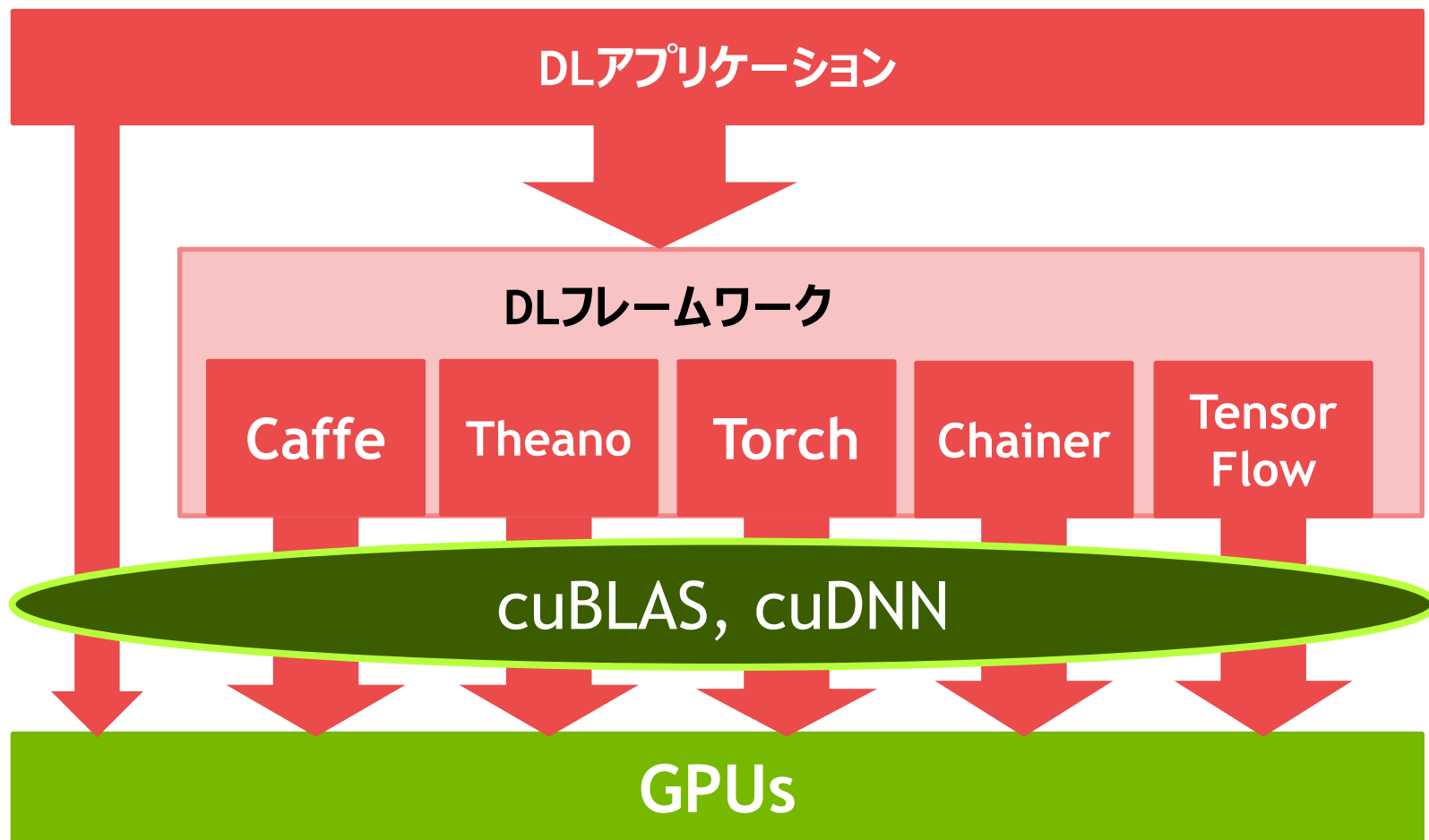
GPU を利用して30日



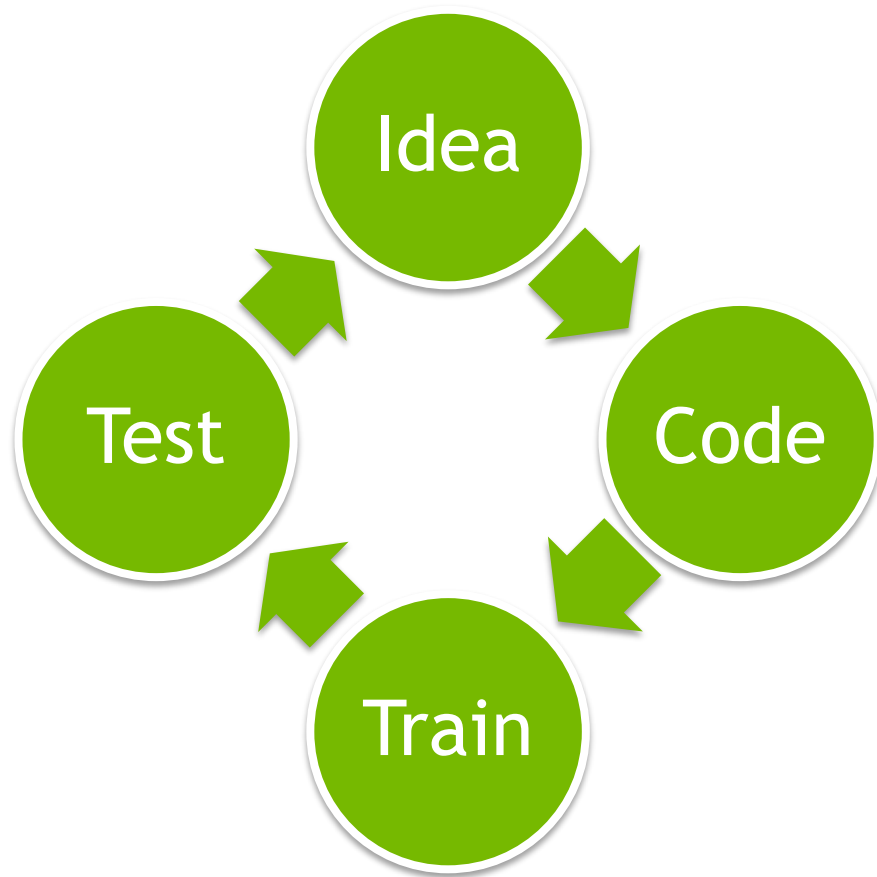
DLトレーニングシステム



DLトレーニングシステム

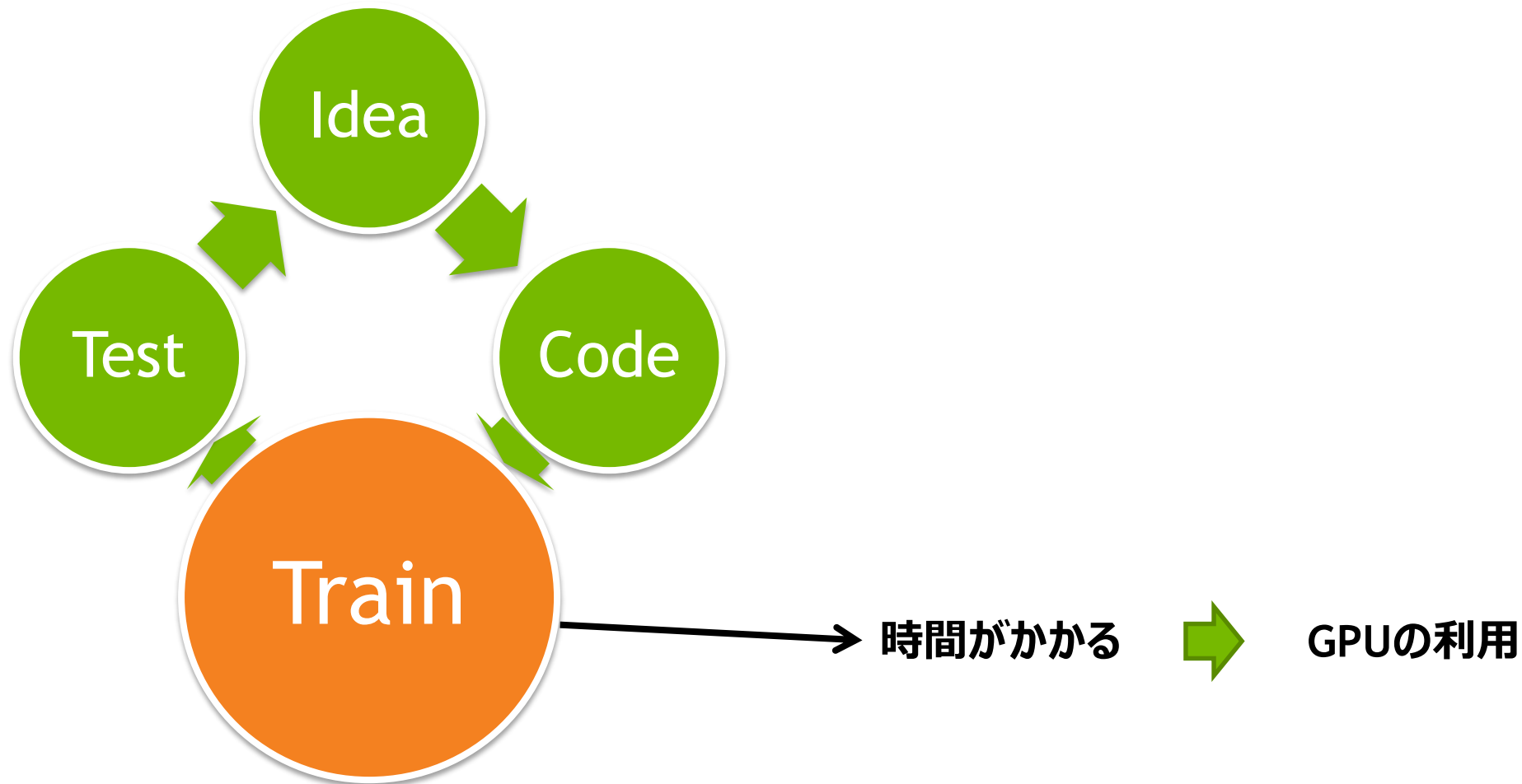


データサイエンティスト R&D サイクル



- ▶ クリティカルパス:
 - ▶ Ideaが沸いてから
 - ▶ Testが完了するまで
- ▶ サイクルを速く回すことが重要

データサイエンティスト R&D サイクル



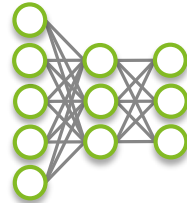
ディープラーニングSDK

ディープラーニングの開発を強力にサポート

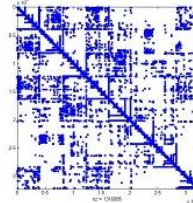
ディープラーニング SDK



DIGITS



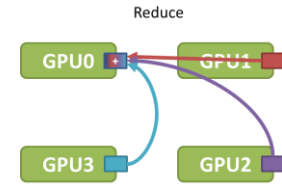
cuDNN



cuSPARSE



cuBLAS

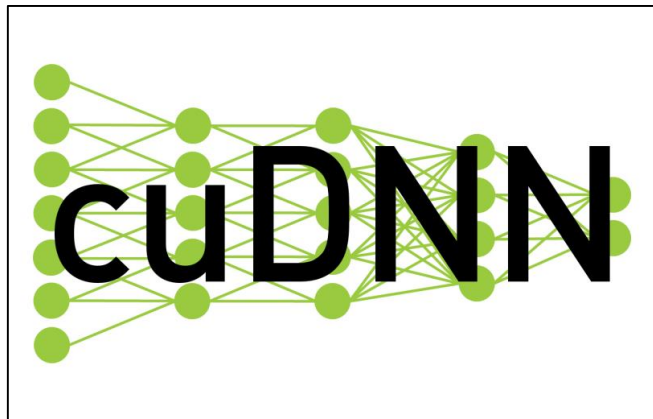


NCCL

DL学習を加速するCUDAライブラリ

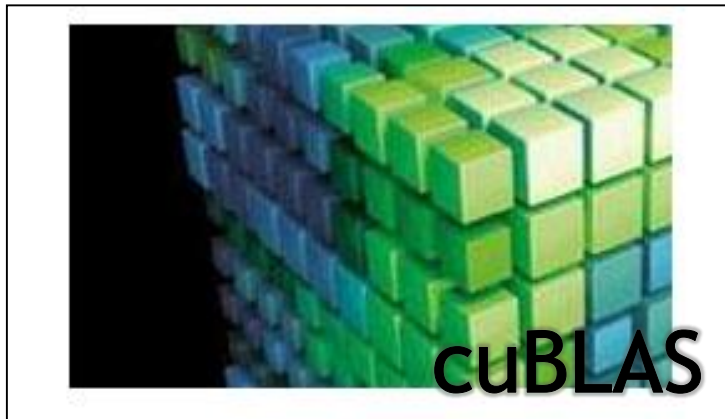
cuDNN

ディープラーニング用ライブラリ



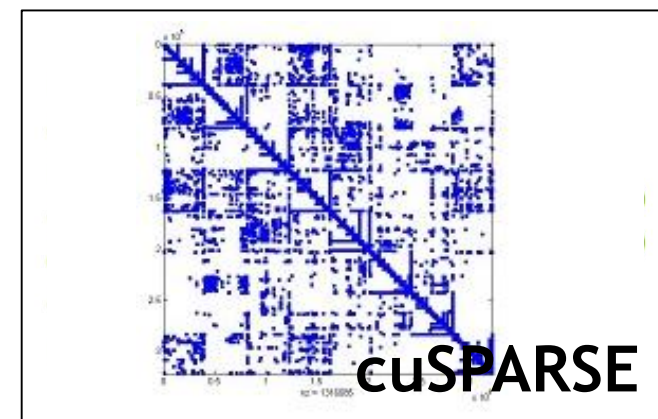
cuBLAS

密行列演算ライブラリ

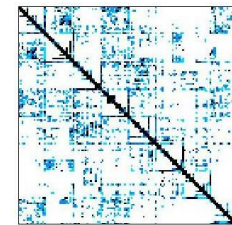


cuSPARSE

疎行列演算ライブラリ



cuSPARSE: 疎行列演算ライブラリ



NVIDIA cuSPARSE

自然言語処理向け機能拡張

密行列 x 疎ベクトル

`cusparse<T>gemvi()`

$$y = \alpha * op(A) * x + \beta * y$$

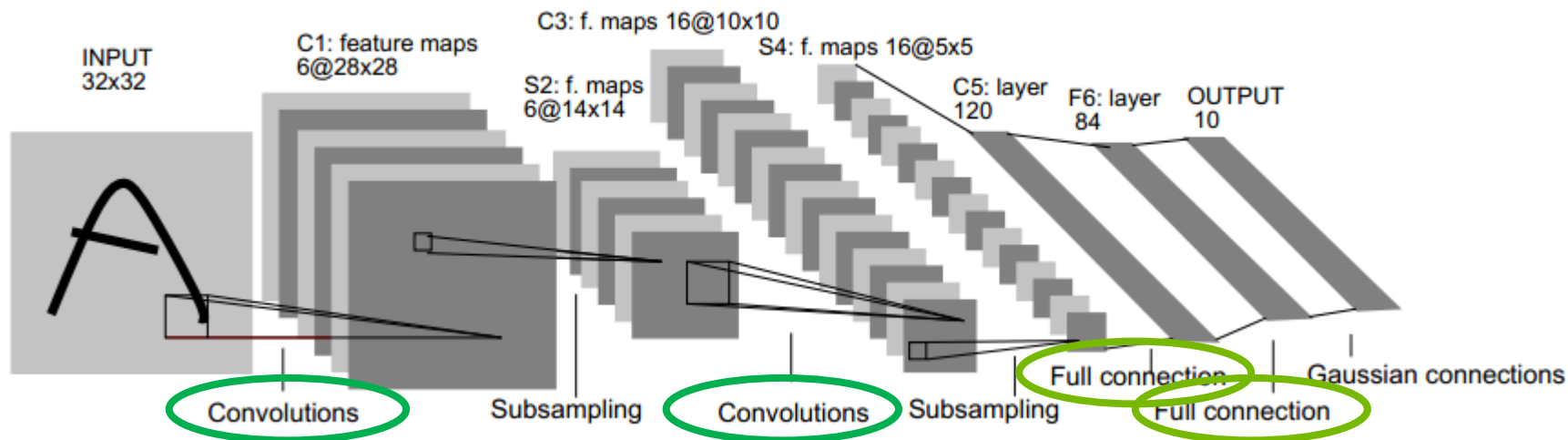
(例) テキスト内の単語の出現頻度

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \alpha \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} \end{bmatrix} \begin{bmatrix} - \\ 2 \\ - \\ - \\ 1 \end{bmatrix} + \beta \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

密ベクトル 密行列 疎ベクトル

CNN:コンボリューショナル・ニューラルネットワーク

LeNet5 [LeCun et al., 1998]



- ▶ 全結合層
- ▶ 畳込み層

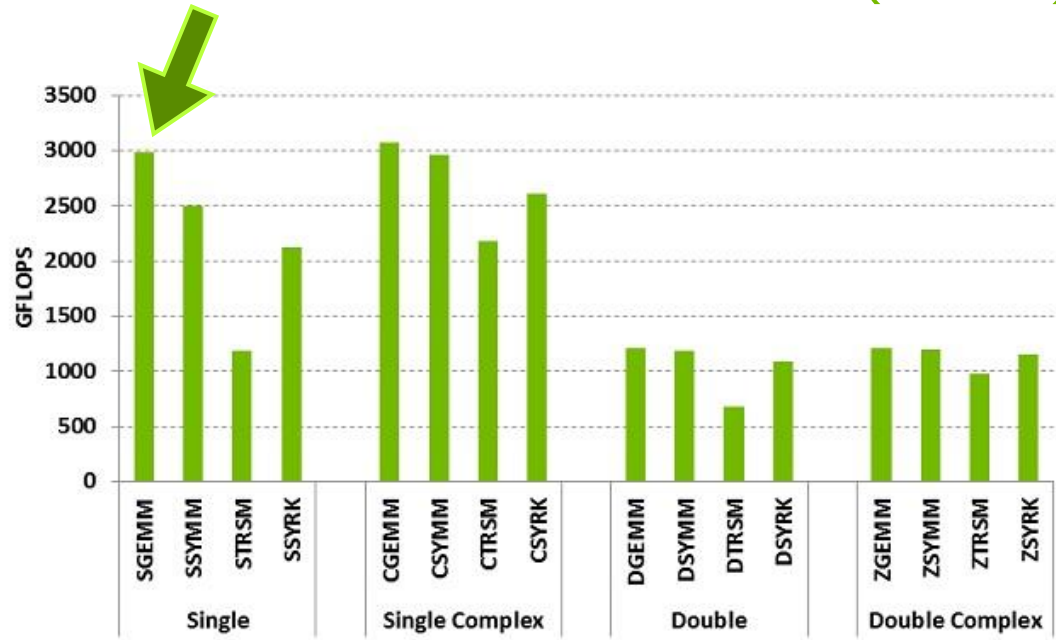
cuBLAS

cuDNN

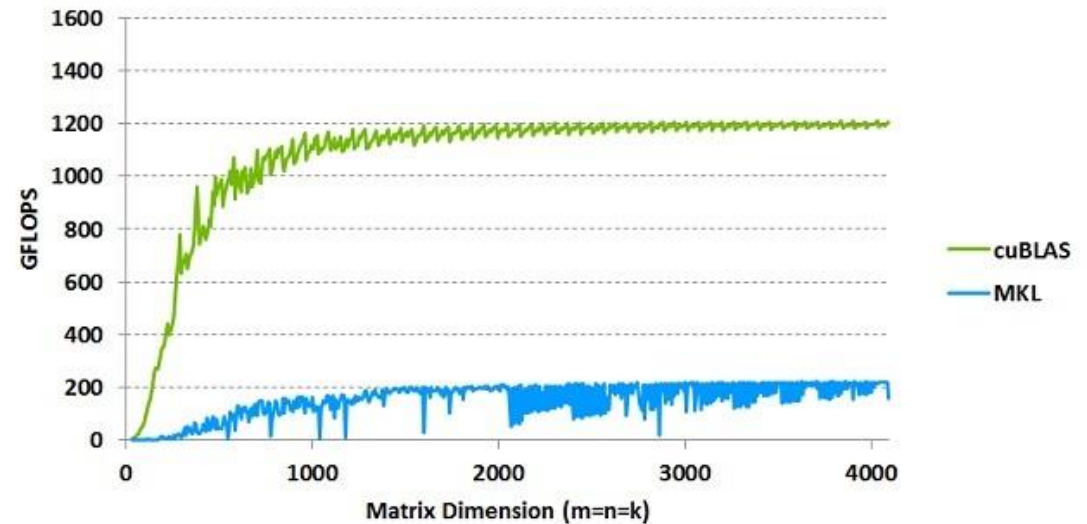
密行列演算ライブラリ: cuBLAS

- ▶ GPU向け行列演算ライブラリ(BLAS完全準拠)

DLで使われるのは単精度の行列積(SGEMM)



cuBLAS 6.5 on K40m, ECC on, input and output data on device.
m=n=k=4096, transpose=no, side=right, fill=lower



cuBLAS 6.5 on K40m, ECC ON, input and output data on device.
MKL 11.0.4 on Intel IvyBridge single socket 12 -core E5-2697 v2 @ 2.70GHz

cuDNN: ディープラーニング用ライブラリ

最新はバージョン4 (2015/12~)

ディープニューラルネットワークの為のGPUプリミティブ群

Caffe, Torch, Theano, Chainer, Tensor-Flow 等の主要ディープラーニングフレームワークが採用

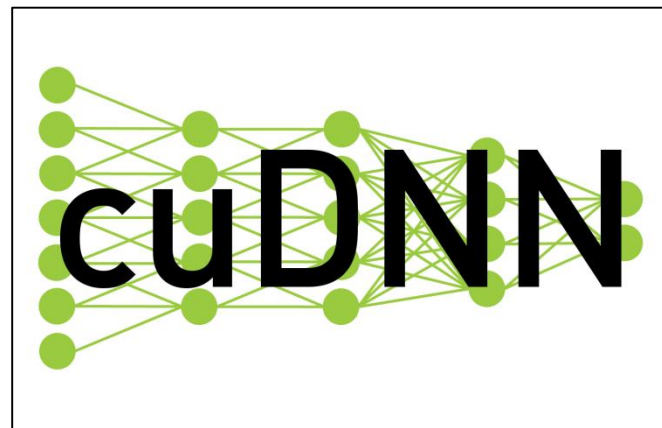
以下のOSをサポート (※2015.12.17現在)

Windows

Linux(x64/ARMv7/ARM64/Power8)

Mac OSX

Android(ARMv7/ARM64)



cuDNN 4 (2015/12~)

学習の高速化

畳み込み演算のアルゴリズムに2D FFT tilingを追加

Batch Normalization処理の追加

normalizationFoward関数、normalizationBackward関数の追加

NHWC format support

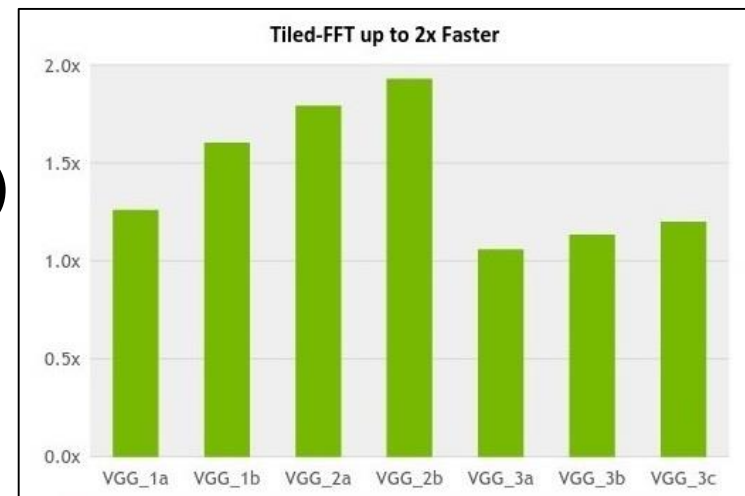
4d-Tensorの入力形式が追加

畳み込み演算のFP16サポート(Tegra X1 only)

Convolutionの計算をFP16に対応

推論処理部の高速化

convolutionFowardのbatchsize=1の場合の最適化



cuDNN Algorithms

- 多種多様なアルゴリズム
 - CUDNN_CONVOLUTION_FWD_ALGO_IMPLICIT_GEMM
 - CUDNN_CONVOLUTION_FWD_ALGO_IMPLICIT_PRECOMP_GEMM
 - CUDNN_CONVOLUTION_FWD_ALGO_GEMM
 - CUDNN_CONVOLUTION_FWD_ALGO_DIRECT
 - CUDNN_CONVOLUTION_FWD_ALGO_FFT
 - CUDNN_CONVOLUTION_FWD_ALGO_FFT_TILING

cuDNN APIs

- ▶ 畳み込み演算

- ▶ `cudannConvolutionForward()`
- ▶ `cudannConvolutionBackward[Bias | Filter | Data]()`

- ▶ 活性化

- ▶ `cudannActivationForward()`
- ▶ `cudannActivationBackward()`

- ▶ プーリング

- ▶ `cudannPoolingForward()`
- ▶ `cudannPoolingBackward()`

- ▶ ソフトマックス

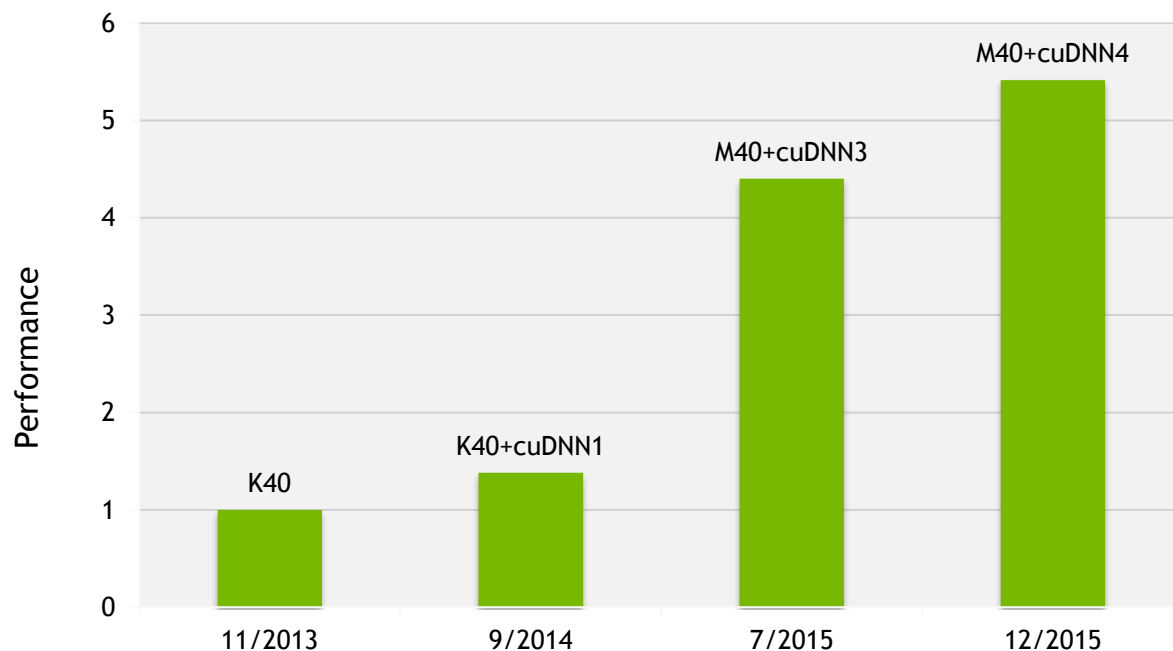
- ▶ `cudannSoftmaxForward()`
- ▶ `cudannSoftmaxBackward()`

- ▶ ...

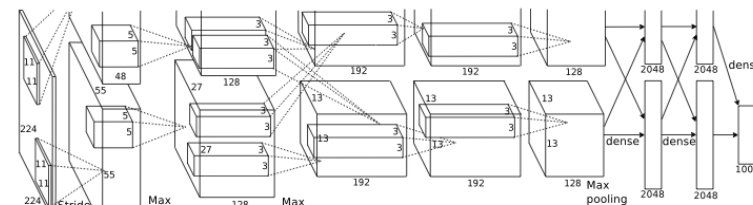
cuDNN: 性能向上

ディープラーニング SDK

Caffe Performance



AlexNet training throughput based on 20 iterations,
CPU: 1x E5-2680v3 12 Core 2.5GHz. 128GB System Memory, Ubuntu 14.04



AlexNet [A. Krizhevsky et al., 2012]

GPU による性能向上

CNN における学習時間

バッチサイズ	学習時間 (CPU)	学習時間(GPU)	GPU／CPU比
64画像	64s	7.5s	8.5X
128画像	124s	14.5s	8.5X
256画像	257s	28.5s	9.0X

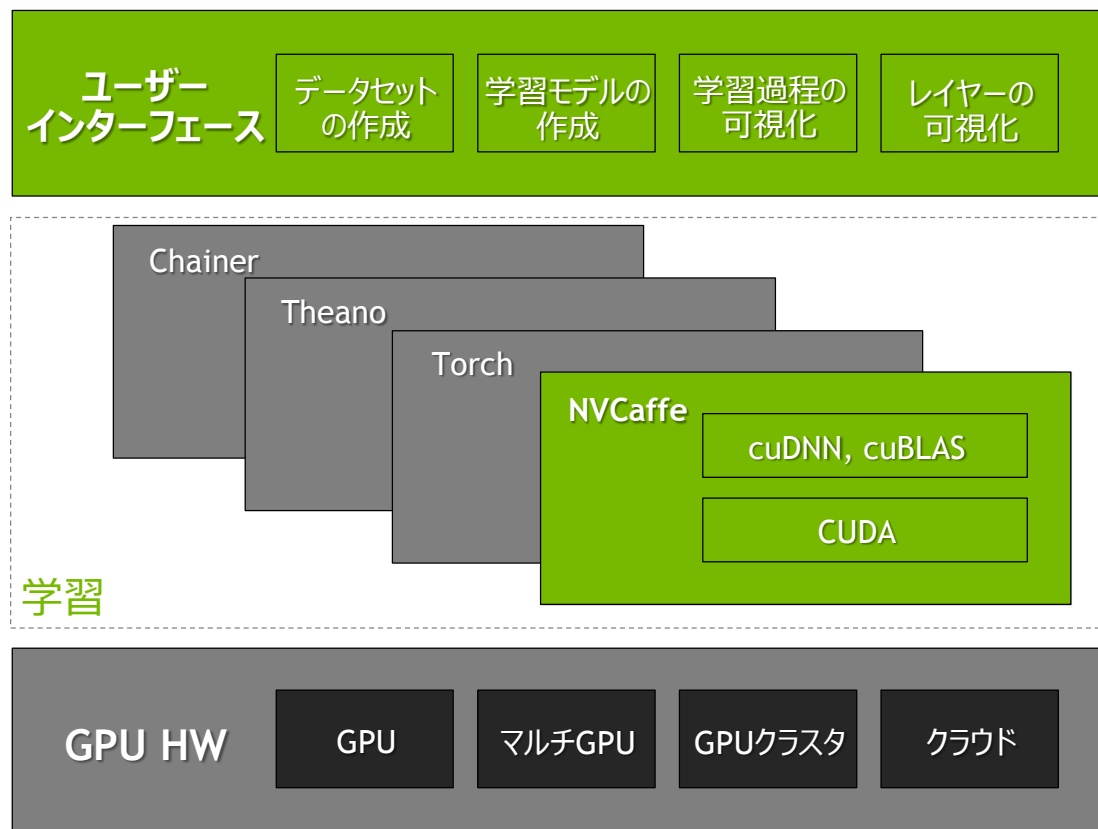
- ▶ ILSVRC12でSupervisionが使用したモデル
- ▶ 7層
- ▶ 5層の畳み込みと2層の全結合
- ▶ ReLU、プーリング、ドロップアウト、正規化
- ▶ Caffeを使用
- ▶ 20反復のトレーニング時間
- ▶ デュアル 10 コア Ivy Bridge CPU
- ▶ 1 Tesla K40 GPU
- ▶ CPU : Intel MKL BLAS ライブリを活用
- ▶ GPU : CUDA行列ライブラリ (cuBLAS) を使用

DIGITS

DIGITS3

Deep Learning SDK

- ▶ WebベースのディープラーニングGPUトレーニングシステム



DIGITS (UI/Server)

ディープラーニングの学習・テストを簡単に行う為のUI
別マシンからDIGITSを操作する為のサーバ機能

NVCaffe

BVLC/cafe(本家)からフォーク
最新のCUDAライブラリを使用
エヌビディアGPUに最適化

DIGITS3

Deep Learning SDK

Torch7に対応

cuDNN4を使った学習/推論

HDF5形式のデータセットのサポート

学習データセットのブラウジング機能

LMDB形式の学習データの中身を確認出来る機能が追加

ジョブマネージメント機能の強化

現在実行中のジョブ一覧を分かりやすく表示する機能が追加

学習結果比較用ブラウザ

学習済みモデルをAccuracyやLoss率など幾つかの項目でソートしたり、分析できる画面が追加

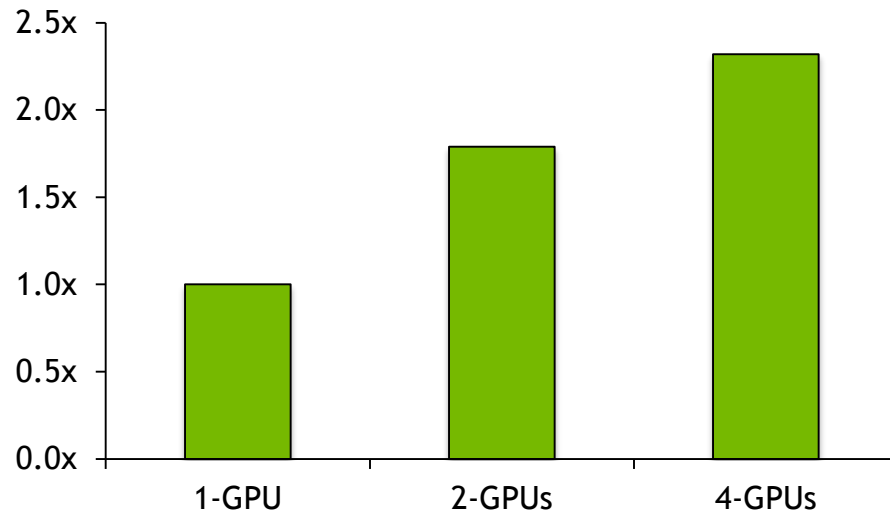
The screenshot displays the DIGITS3 web interface. The top section is the 'Home' dashboard, which includes a table of training jobs and a progress bar for GPU usage. Below this is a section titled 'Exploring imagenet_dog (train_db) images'. It features a list of dog breeds, a pagination control showing 'Items per page: 10 - 25 - 50 - 100', and a gallery of three images of poodles, each labeled 'toypoodle'.

Name	Start Time	Status	Training Loss	Progress
aerial_slayer	08:44:32 PM	Running		
mini_exponent	08:44:05 PM	Done		
aerial_deepnetwork	08:45:47 PM	Running		
aerial_slayer_stepsize	08:45:56 PM	Running		

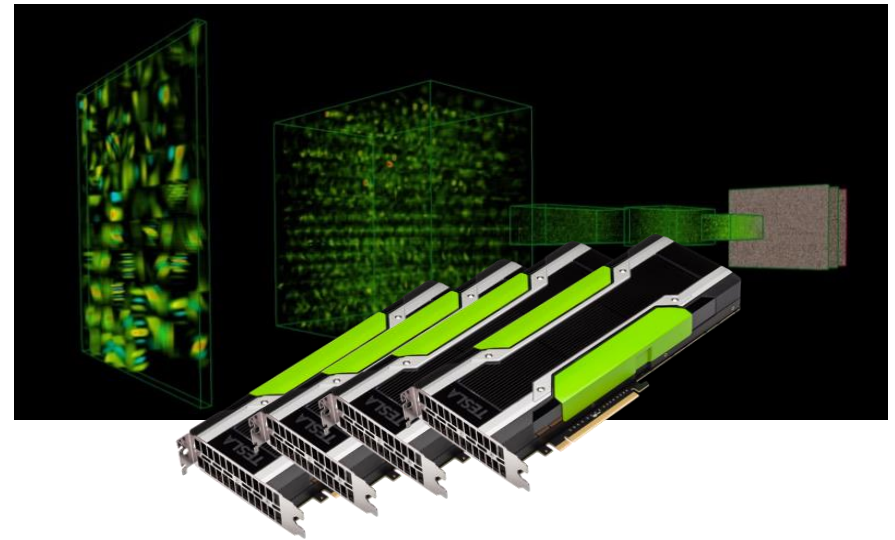
DIGITS2: マルチGPUによる学習の高速化

自動的にマルチGPUにスケールして学習

マルチGPUにスケールする事で従来より高速に学習する事が可能



DIGITS 2 performance vs. previous version on an NVIDIA DIGITS DevBox system



DL のアプリケーション開発を支援

DIGITS Image Classification Dataset

voc_cropped@256x256
Image Classification Dataset

Delete Job

Job Information

Job Directory
/home/michaelo/digits
/jobs/20150311-171431-e0d8

Image Type
Color

Image Dimensions
256x256

Resize Mode
half_crop

Parse Folder (train/val)

Folder
http://sqldata/images/voc_cropped/

Number of categories
20

Training images
26759

Validation images
8917 (25.0%)

Job Status Done

- Initialized at Wed Mar 11, 05:16:57 PM (1 second)
- Running at Wed Mar 11, 05:16:57 PM (2 minutes, 25 seconds)
- Done at Wed Mar 11, 05:16:57 PM (Total: 2 minutes, 26 seconds)

Parse Folder (train/val) Done

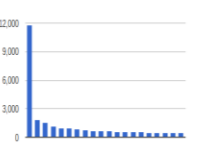
Create DB (train) Done

Create DB (val) Done

Create DB (train)

Input file
train.txt

DB Entries
26759



DIGITS New Model

Select Dataset

PASCAL VOC
ILSVRC 2012
MNIST Dataset

Data Transform

Crop Size
none

☒ Subtract Mean

Solver Options

Training epochs
30

Validation interval (in epochs)
1

(heat progress bar)

Batch size
100

Base Learning Rate
0.01

☐ Show advanced learning rate options

Standard Networks **Previous Networks** **Custom Network**

Custom Network

```
{
  layer {
    name: "conv1"
    type: "Convolution"
    bottom: "data"
    top: "conv1"
    param {
      lr_mult: 1
      decay_mult: 1
    }
  }
}
```

Pretrained model

Model Name
ImageNet

Create

DIGITS Image Classification Model

Solver
solver.prototxt
Network (train/val)
train_val.prototxt
Network (deploy)
deploy.prototxt

Dataset
voc_cropped@256x256
Done Wed Mar 11, 05:16:57 PM

Image Size
256x256

Image Type
COLOR

Create DB (train)
26759 images

Create DB (val)
8917 images

Job Status Running

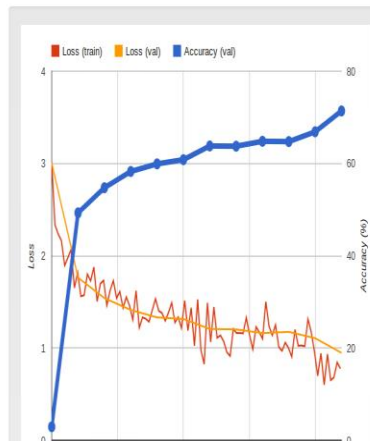
- Initialized at 08:28:15 AM (1 second)
- Running at 08:28:16 AM

Train Caffe Model Running

37%

Estimated time remaining: 30 minutes, 31 seconds

- Initialized at 08:28:15 AM (1 second)
- Running at 08:28:16 AM



50315-091306-191c

Search

Predictions

8
3
0
6
4

Layer Activations

conv1



pool1

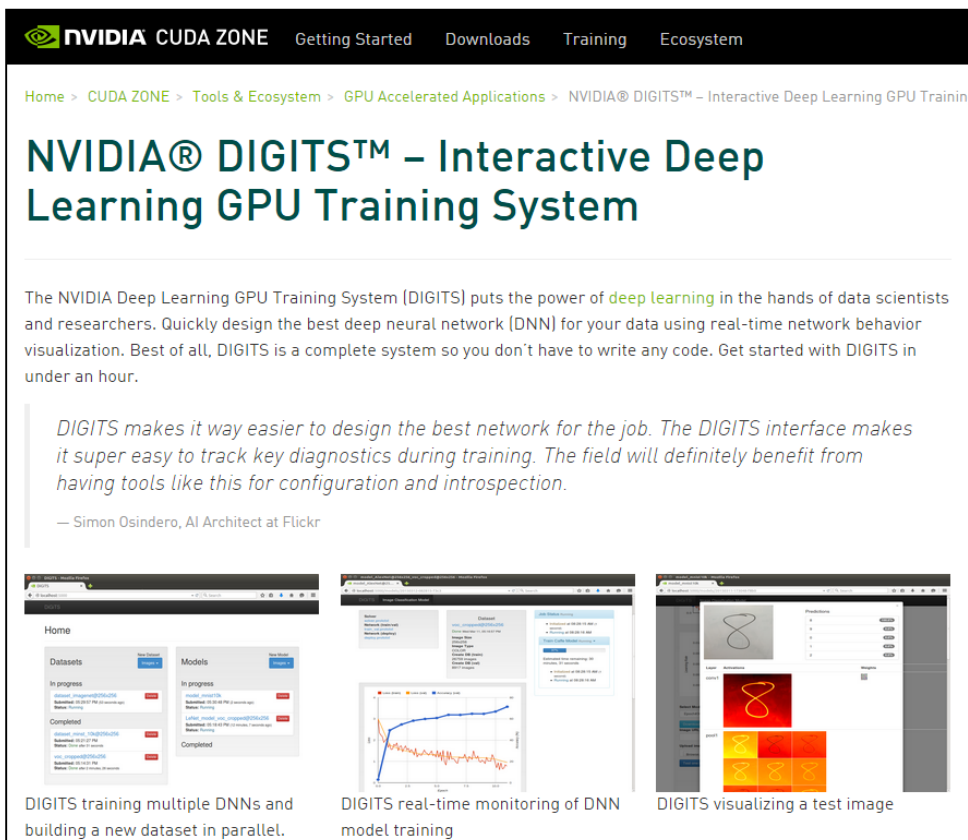


DIGITS DEMO

DIGITS download

developer.nvidia.com/digits

github.com/NVIDIA/DIGITS



The screenshot shows the NVIDIA DIGITS website. The header includes the NVIDIA logo and navigation links: CUDA ZONE, Getting Started, Downloads, Training, and Ecosystem. The main heading is "NVIDIA® DIGITS™ – Interactive Deep Learning GPU Training System". Below this, a paragraph describes the system as a tool for data scientists to design and visualize DNNs. A quote from Simon Osindero is included. At the bottom, three panels show the DIGITS interface: training multiple DNNs, real-time monitoring of a DNN model, and visualizing a test image.

NVIDIA® DIGITS™ – Interactive Deep Learning GPU Training System

The NVIDIA Deep Learning GPU Training System (DIGITS) puts the power of **deep learning** in the hands of data scientists and researchers. Quickly design the best deep neural network (DNN) for your data using real-time network behavior visualization. Best of all, DIGITS is a complete system so you don't have to write any code. Get started with DIGITS in under an hour.

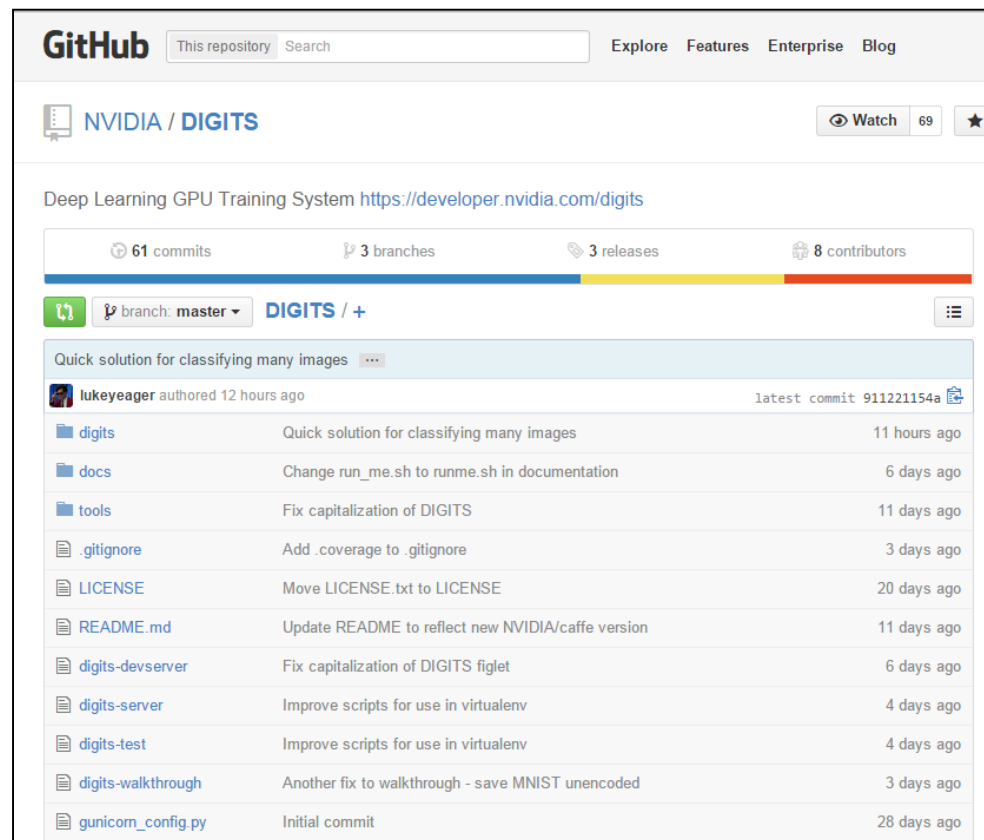
DIGITS makes it way easier to design the best network for the job. The DIGITS interface makes it super easy to track key diagnostics during training. The field will definitely benefit from having tools like this for configuration and introspection.

— Simon Osindero, AI Architect at Flickr

DIGITS training multiple DNNs and building a new dataset in parallel.

DIGITS real-time monitoring of DNN model training

DIGITS visualizing a test image



The screenshot shows the GitHub repository for NVIDIA DIGITS. The repository name is "NVIDIA / DIGITS". It has 61 commits, 3 branches, 3 releases, and 8 contributors. The repository is described as a "Deep Learning GPU Training System". A table lists the repository contents, including folders like "digits", "docs", and "tools", and files like ".gitignore", "LICENSE", and "README.md".

GitHub This repository Search Explore Features Enterprise Blog

NVIDIA / DIGITS Watch 69 Star

Deep Learning GPU Training System <https://developer.nvidia.com/digits>

61 commits 3 branches 3 releases 8 contributors

branch: master DIGITS / +

Quick solution for classifying many images ...

File	Description	Time
digits	Quick solution for classifying many images	11 hours ago
docs	Change run_me.sh to runme.sh in documentation	6 days ago
tools	Fix capitalization of DIGITS	11 days ago
.gitignore	Add .coverage to .gitignore	3 days ago
LICENSE	Move LICENSE.txt to LICENSE	20 days ago
README.md	Update README to reflect new NVIDIA/cafe version	11 days ago
digits-devserver	Fix capitalization of DIGITS figlet	6 days ago
digits-server	Improve scripts for use in virtualenv	4 days ago
digits-test	Improve scripts for use in virtualenv	4 days ago
digits-walkthrough	Another fix to walkthrough - save MNIST unencoded	3 days ago
gunicom_config.py	Initial commit	28 days ago

エヌビディアが加速するディープラーニング フレームワーク

音声認識

画像分析

自然言語処理

エンドユーザ アプリケーション
DIGITS

ディープラーニング フレームワーク (Caffe, Chainer, Torch, Theano, Tensor-Flow)

高度に最適化された cuDNN ライブラリ

CUDA プログラミング ツールキット

GPU ハードウェア

Thank you!

