

ディープラーニングが変える未来

- PFNの事業戦略とChainer on AWS -

Preferred Networks
研究担当VP
比戸 将平

株式会社Preferred Networks (PFN)

- 設立：2014年3月
- 所在地：東京本社、カリフォルニア州バークレー（米国子会社）
- 社員数：日本約150名、米国7名



Agenda

- ディープラーニングが変える未来
- PFNの事業戦略とそれを支えるChainer
- Chainer on AWS

ディープラーニング（深層学習）とは

- 層が深いニューラルネットワークを利用した機械学習手法およびその応用
- 2012年の大ブレイク以来、研究コミュニティのみならず産業界に多く使われてきた
 - 2014～2015年中に出された関連論文数は1500を超える*
おそらく2018年は10000に達しているだろう
- 画像認識、音声認識などで劇的な精度向上を果たし、その多くが既に実用化されている
- 囲碁でもDeepMindの AlphaGoが大きな成果



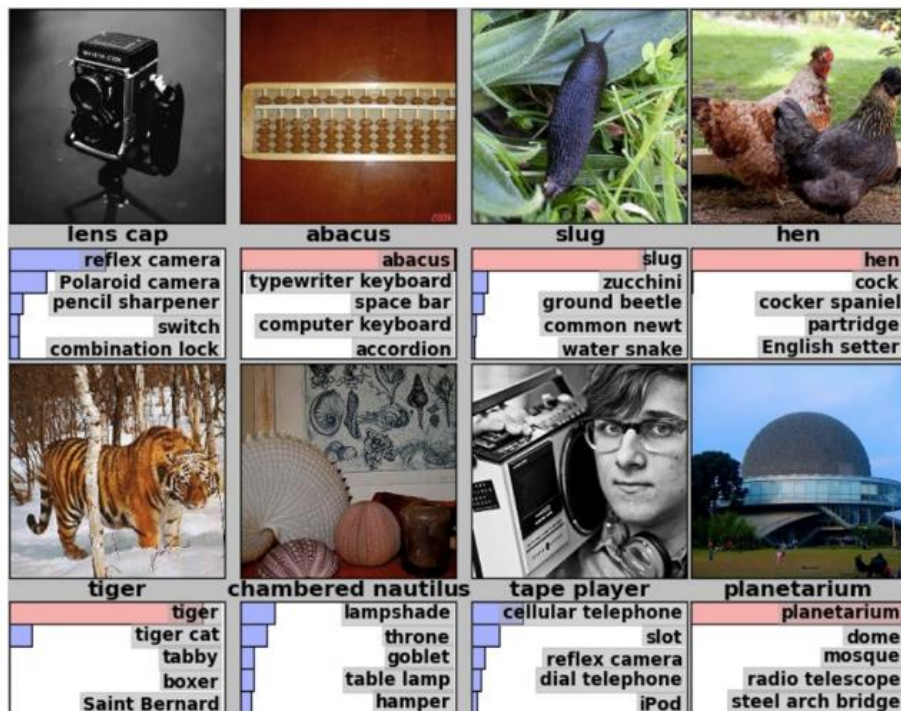
2014年の一般画像認識コンテストで優勝した
22層からなるNNの例 [Google]

2015年の一般画像認識コンテストで優勝した
152層からなるNNの例 [MSRA]

*<http://memkite.com/deep-learning-bibliography/>

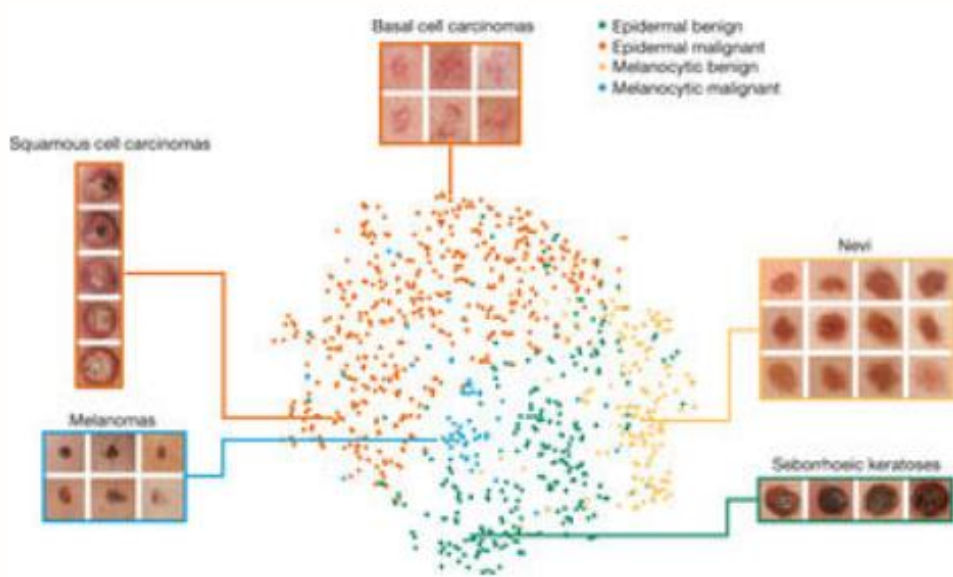
最初のブレイクスルー：画像認識コンテストで圧勝

- ImageNet (ILSVRC, 1000クラス分類) でエラー率が26%→16%に激減

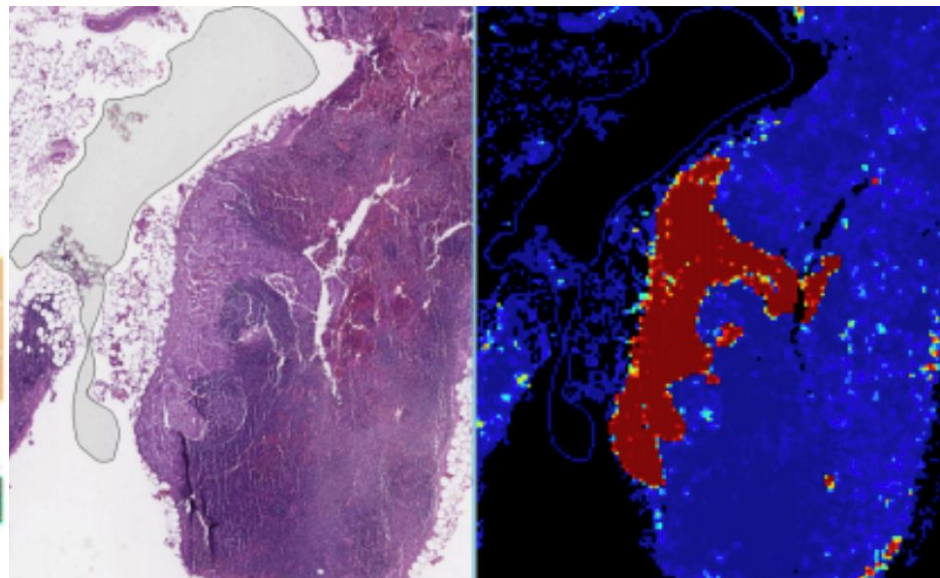


医療応用の現在：医療画像解析では専門医に並ぶ診断精度を達成

- 一般物体の画像認識で発展したディープラーニング技術のストレートな応用
- 専門医を超える検出精度を出し始めている（左：皮膚がん、右：リンパ節転移）



Dermatologist-level classification of skin cancer with deep neural networks [[Esteva+, Nature, 2017](#)]



Detecting Cancer Metastases on Gigapixel Pathology Images [[Lin+, 2017](#)]

自動運転の現在：認識機能にディープラーニング使うのは共通

- 多種センサー：高精細な深度情報 $\longleftrightarrow$ 見た目の課題、高コスト（改善中）
- 画像センサー中心：低コスト $\longleftrightarrow$ 深度・遠方情報が不足し低精度（改善中）

Drive.ai



Lyft and Drive.ai partner on Bay Area self-driving ride-hailing pilot, [Darrel Etherington, TechCrunch](#)
雨天夜間走行のデモ動画あり [\[youtube\]](#)

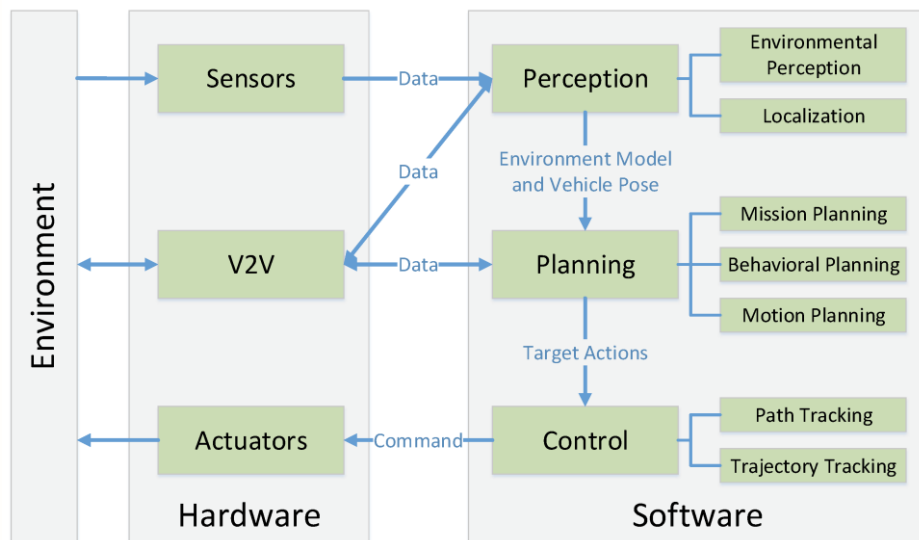


Full Self-Driving Hardware on All Cars

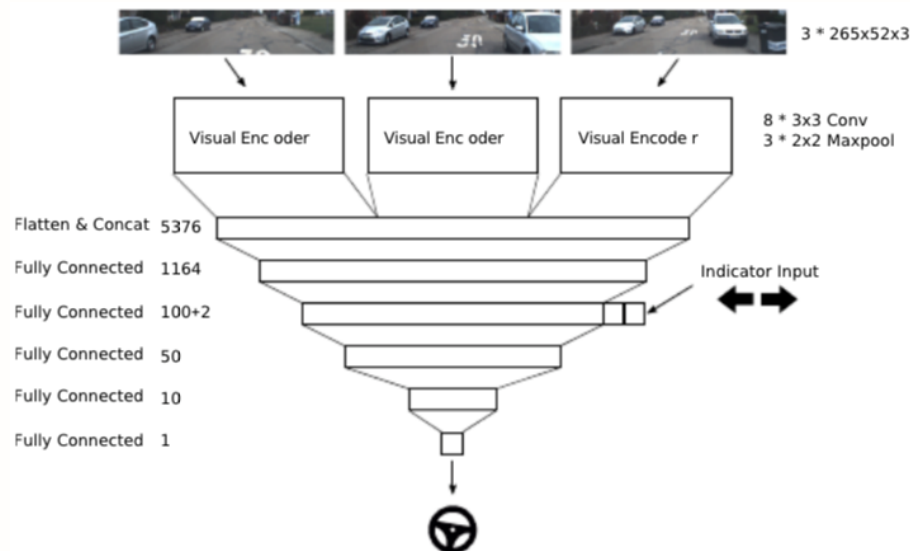
Tesla Autopilot：前後左右遠近8種類のカメラ＋超音波＋レーダー
<https://www.tesla.com/autopilot>

自動運転の未来：プランニングや制御も置き換え or 一気通貫モデルで運転

- 個別モジュール：旧来の認識→経路計画→制御の一部をディープラーニングに
- 一気通貫：センサー入力から直接アクセル・ブレーキ・ハンドル制御を出力



Perception, Planning, Control, and Coordination for Autonomous Vehicles [\[Pendleton+, 2017\]](#)



Adding Navigation to the Equation: Turning Decisions for End-to-End Vehicle Control [\[Hubschneider+, 2017\]](#)

縁の下の力持ちとしてのディープラーニング

- ディープラーニングは本物なのか？またAI冬の時代がやってくるだけか？
 - 一部で極端な楽観や悲観に基づく議論があり、まだ混乱状態
 - ただこれまでの冬の時代と違ってすでに実社会で応用されはじめている
 - 過度の期待を集めすぎなければ徐々に健全な活用は広がっていくはず
 - 普及したAI応用は「AI」とは呼ばれなくなる（レコメンド、エージェント…）
- 特に「認識」については人間と同等あるいはそれを超える能力

Kindleストアのおすすめ商品



Interactively Picking Real-World Objects with Unconstrained Spoken Language Instructions

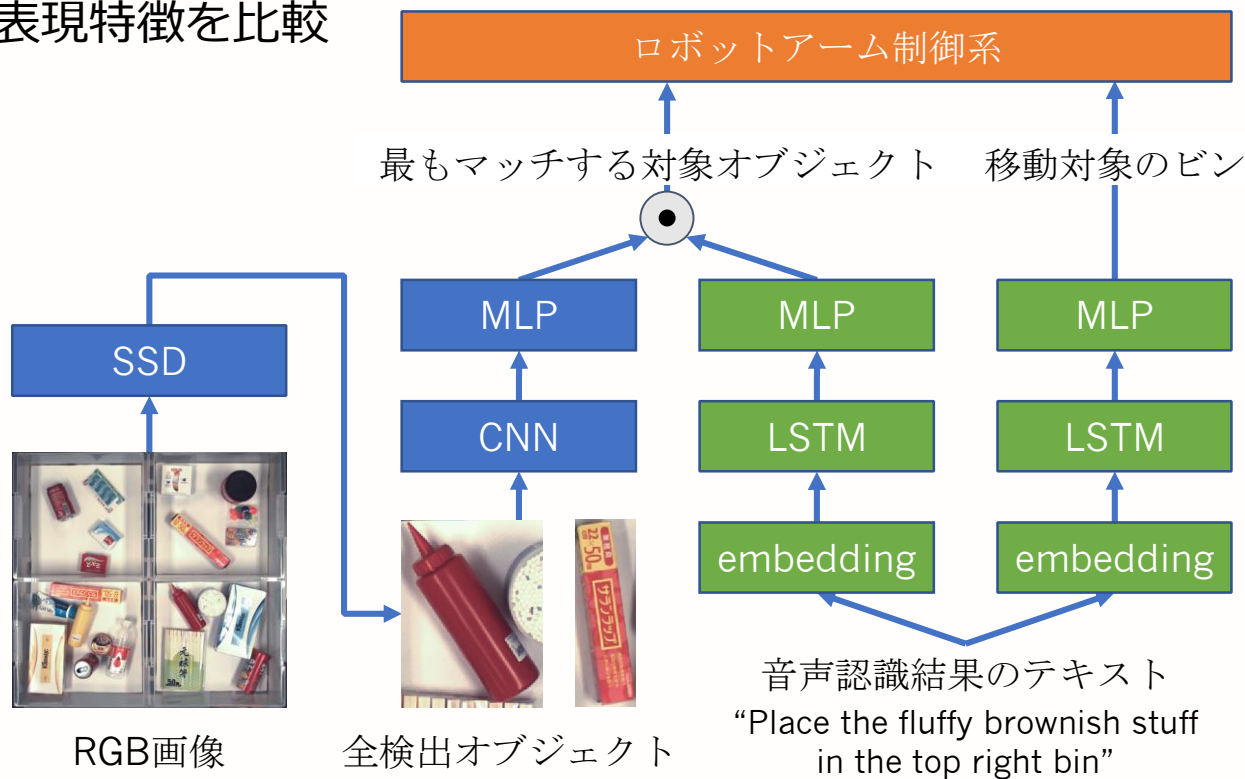
Jun Hatori*, Yuta Kikuchi*, Sosuke Kobayashi*, Kuniyuki
Takahashi*, Yuta Tsuboi*, Yuya Unno*, Wilson Ko, Jethro Tan



* equal contributions

これも最新のディープラーニングモデルで実現されている

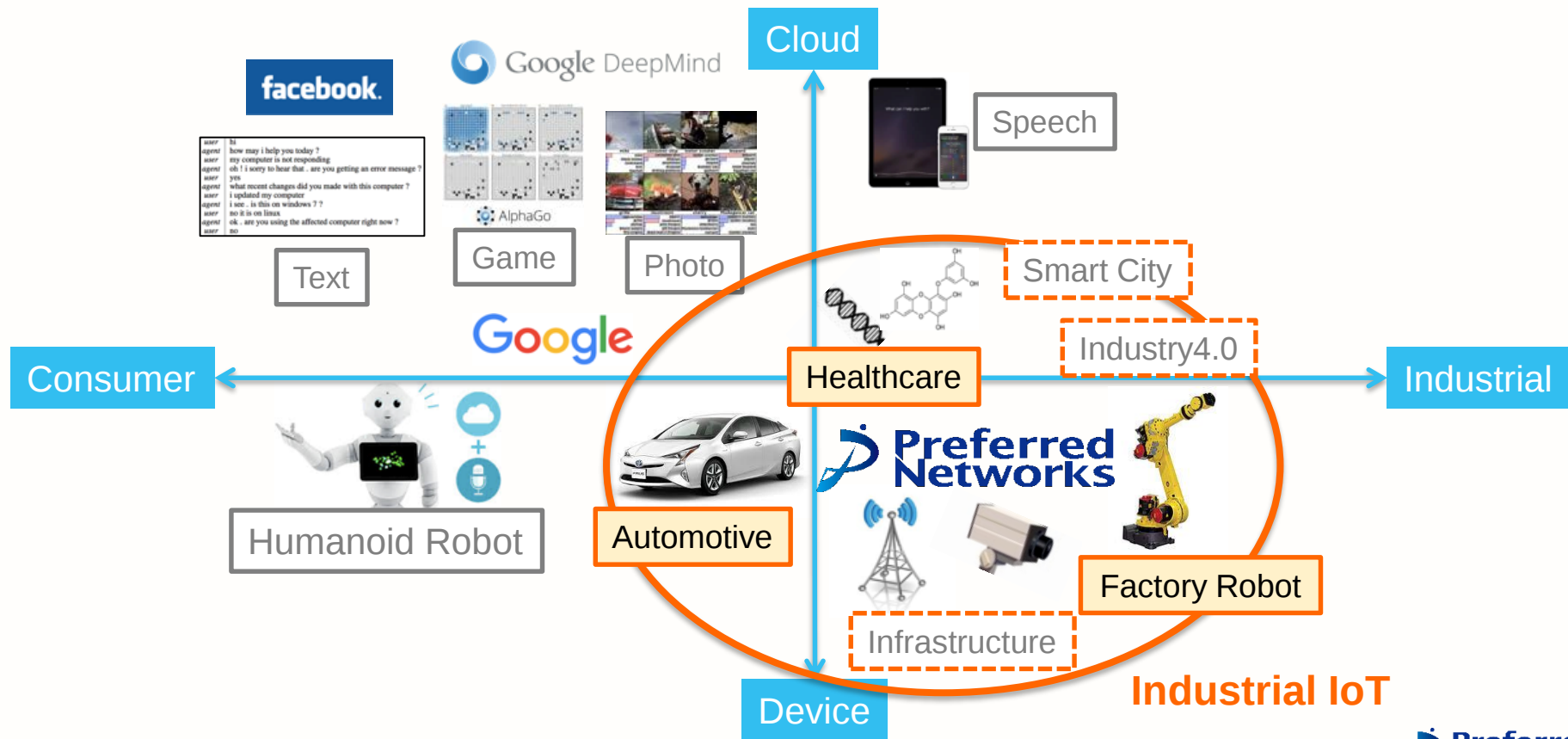
- クラウドソーシングにより単一のオブジェクトの様々な表現を獲得
- 画像特徴と表現特徴を比較



Agenda

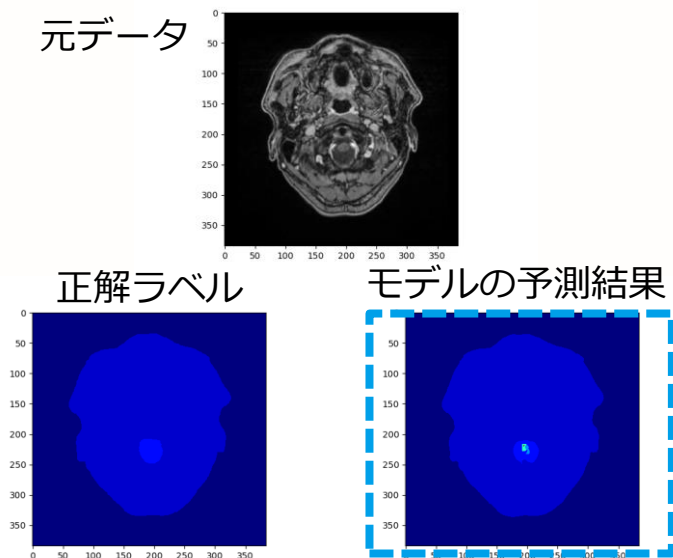
- ディープラーニングが変える未来
- PFNの事業戦略とそれを支えるChainer
- Chainer on AWS

PFNのフォーカス：AI技術の産業応用の最先端を走る



バイオ応用：3次元化、マルチモーダル化

- MRIは連続した輪切り画像として扱うよりも3次元情報として扱ったほうが有用
- 単一の情報源だけでなく多様なデータをいかに収集・統合して解析するか



国立がん研究センター様提供のMRI画像列に対し脳腫瘍を含む部位を領域セグメンテーション(JST CRESTの成果)



JST CREST

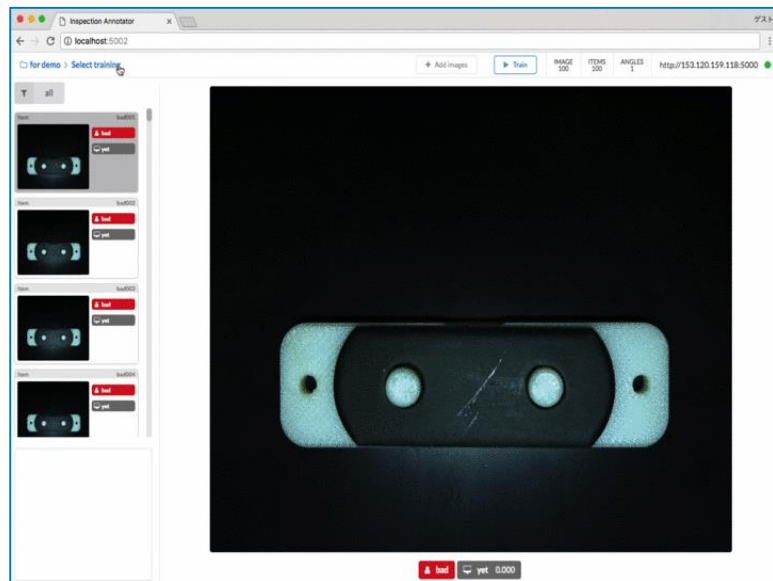
「人工知能を用いた統合的ながん医療システムの開発」

国立がん研、産総研、Preferred Networks



製造業応用：異常検知、予防保全

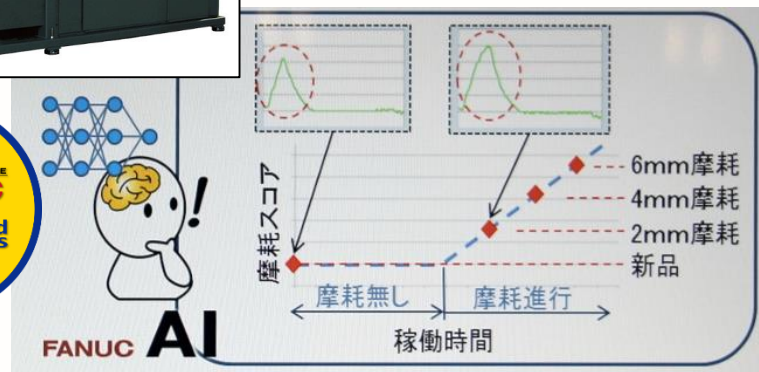
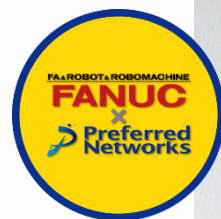
- 外観検査：画像認識技術を応用したソフトウェア・サービス市場の盛り上がり
- 射出成形機の予防保全：逆流防止弁の摩耗をバックフローモニタ時系列から推測



PFNの外観検査：キズの箇所を特定不要、画像単位で正解付け・モデル学習・可視化・精度評価をブラウザで



ロボショット射出成形機：
樹脂を金型に流し込んで成形



射出成形機にAI搭載、ファナックが深層学習技術で予防保全
[\[Monoist\]](#) [\[プレスリリース\]](#)

PaintsChainer: ディープラーニングを用いた自動着色サービス



<https://paintschainer.preferred.tech/>

(参考) エンターテインメント業界における実用化

- 白泉社、博報堂DYデジタル、Preferred Networksによる自動着色マンガ配信



Chainer: ディープラーニング用のPythonライブラリ



Powerful

☑ CUDA

CUDAを用いたGPU計算のサポート

☑ cuDNN

cuDNNによる高速な学習／推論

☑ NCCL

NCCLを用いた高速なマルチGPU学習をサポート

Flexible

☑ Convolutional Network

N次元の入力に対応したConvolution, Deconvolution, Pooling, BN, 等

☑ Recurrent Network

LSTM, Bi-directional LSTM, GRU, Bi-directional GRU, 等のRNNコンポーネント

☑ Many Other Components

ニューラルネットワークで使われる多くのレイヤ定義、各種ロス関数

☑ Various Optimizers

SGD, MomentumSGD, AdaGrad, RMSProp, Adam, 等の最適化手法が選択可能

Intuitive

☑ Define-by-Run

複雑なネットワークの記述が容易

☑ High debuggability

Pythonライブラリであるためエラー箇所の特定が容易：デバッグしやすい

☑ Simple APIs

様々なNNの学習で共通する部分を抽象化、一連の学習フローを簡易に記述可

Chainerを用いたモデル定義・学習実行のPythonスクリプト

画像識別モデルの定義 (LeNet)

```
model = LeNet5()
```

```
model = L.Classifier(model)
```

リスト形式のデータセットを仮定

```
dataset = [(x1, t1), (x2, t2), ...]
```

ミニバッチデータセットを返すイテレータ

```
it = iterators.SerialIterator(dataset, batchsize=32)
```

最適化手法

```
opt = optimizers.SGD(lr=0.01)
```

```
opt.setup(model)
```

```
updater = training.StandardUpdater(it, opt, device=0) # device=-1 if you use CPU
```

```
trainer = training.Trainer(updater, stop_trigger=(100, 'epoch'))
```

```
trainer.run()
```

研究開発を加速するChainerファミリー



CuPy: ChainerのGPUバックエンド (NumPy互換GPU Arrayライブラリ)
NumPy-likeに書かれた計算コードをGPUで実行



ChainerMN: 分散深層学習用追加パッケージ
高いスケーラビリティ (128GPUで100倍の高速化)



ChainerRL: 深層強化学習ライブラリ
DQN, DDPG, A3C, ACER, NSQ, PCL, etc. OpenAI Gym サポート



ChainerCV: 画像認識アルゴリズム・データセットラッパーパッケージ
Faster R-CNN, Single Shot Multibox Detector (SSD), SegNet, etc.



- Chainer Chemistry: グラフ構造を持つ化合物データ等でディープラーニング
- Chainer UI: モデル学習の実行状況をモニタリング・スナップショット等

ライトユーザーがすぐにディープラーニングを応用できるように

画像認識

CV

ChainerCV: 画像認識アルゴリズム・データセットラッパーを提供
Faster R-CNN, Single Shot Multibox Detector (SSD), SegNet, etc.



ChainerCV: a Library for Deep Learning in Computer Vision [[Niitani, Ogawa, Saito, Saito, ACM MM 2017](#)]

強化学習

RL

ChainerRL: 深層強化学習ライブラリ
DQN, DDPG, A3C, ACER, NSQ, PCL, etc. OpenAI Gym サポート

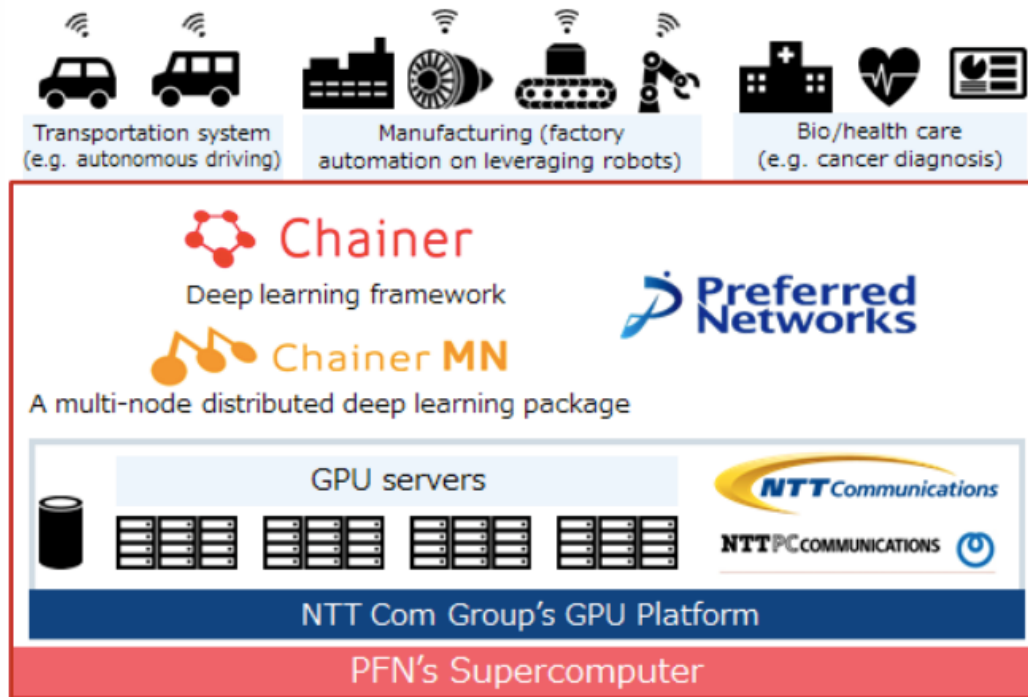


X



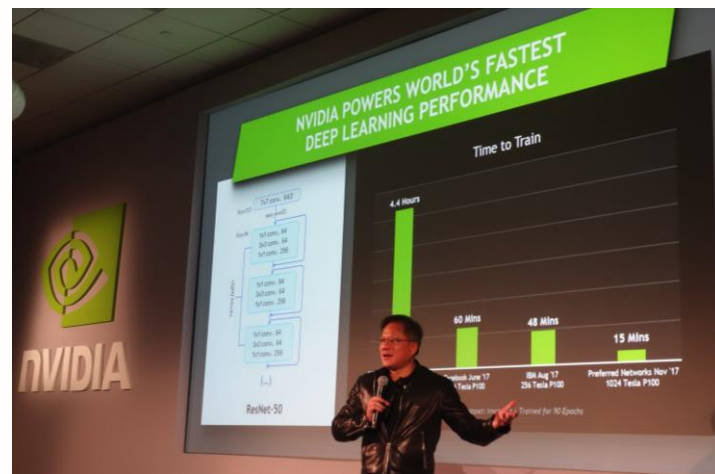
ChainerMNとプライベートGPUクラスタを用いた研究開発環境

- 1024GPU環境を用いて画像認識ベンチマークで世界一を達成（2017年11月）



Overview of the private supercomputer

ImageNet in 15 minutes



NVIDIA CEO Jensen Huang 氏による
学会 SC'17 での講演

<https://twitter.com/NVIDIAAIJP/status/930209150084063232>

前半のまとめ

- ディープラーニングの産業応用
 - 画像認識アプリケーションを中心とした商用化が加速
 - 多種多様なデータを用いた複雑な実問題への取り組みも開始
- それを取り巻く周辺動向
 - ソフトウェアとハードウェア双方の激しい開発競争：高速化、大規模化、標準化、
 - ユーザー層の拡大、研究者だけでなく技術を使いこなせるエンジニアの育成
- 産業応用を支えるPFNの仕組み
 - 研究開発の推進を第一に考え優秀な人の採用と環境整備・サポート

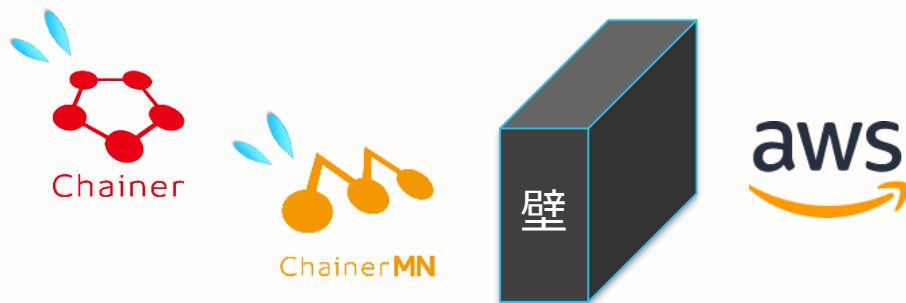


Agenda

- ディープラーニングが変える未来
- PFNの事業戦略とそれを支えるChainer
- Chainer on AWS

「でもChainerってAWSと相性よくないでしょ？」→もはや誤解です！

- 1) ChainerMNはInfiniBand前提だからAWSでは性能が出ない！
- 2) ChainerMN用GPUクラスタをAWSに自前で構築するのは大変！
- 3) Chainerでモデル構築するマネージドサービスがない！
- 4) Chainerモデル学習してもAWSのエッジで推論に使えない！

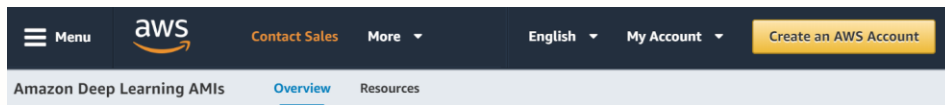


AWSにおけるChainerのサポート



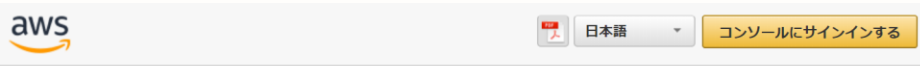
- AWS Deep Learning AMIサポート
 - 現在最新メジャー版のv4.0を搭載

- ドキュメントの充実
 - Chainer on AWSのチュートリアル



Support for deep learning frameworks

The AWS Deep Learning AMIs support all the popular deep learning frameworks allowing you to define models and then train them at scale. Built for Amazon Linux and Ubuntu, the AMIs come pre-configured with [Apache MXNet](#) and [Glueon](#), [TensorFlow](#), Microsoft Cognitive Toolkit, Caffe, Caffe2, Theano, Torch, PyTorch, Chainer, and Keras, enabling you to quickly deploy and run any of these frameworks at scale.



Deep Learning AMI

開発者ガイド

ドキュメント - 本ガイド

検索

[AWS ドキュメント](#) » [Deep Learning AMI](#) » [開発者ガイド](#) » [チュートリアルおよび例](#) » [Chainer](#)

Chainer

Chainer は、簡単に直観的に複雑なニューラルネットワークアーキテクチャを記述するための柔軟性のある Python ベースのフレームワークです。Chainer は、トレーニングにおけるマルチ GPU インスタンスの使用を簡単にします。また、Chainer は自動的に結果、グラフの損失および正確性をログに記録し、[計算グラフ](#) でニューラルネットワークを視覚化するための出力を実行します。これは Deep Learning AMI と Conda (Conda の DLAMI) に含まれています。

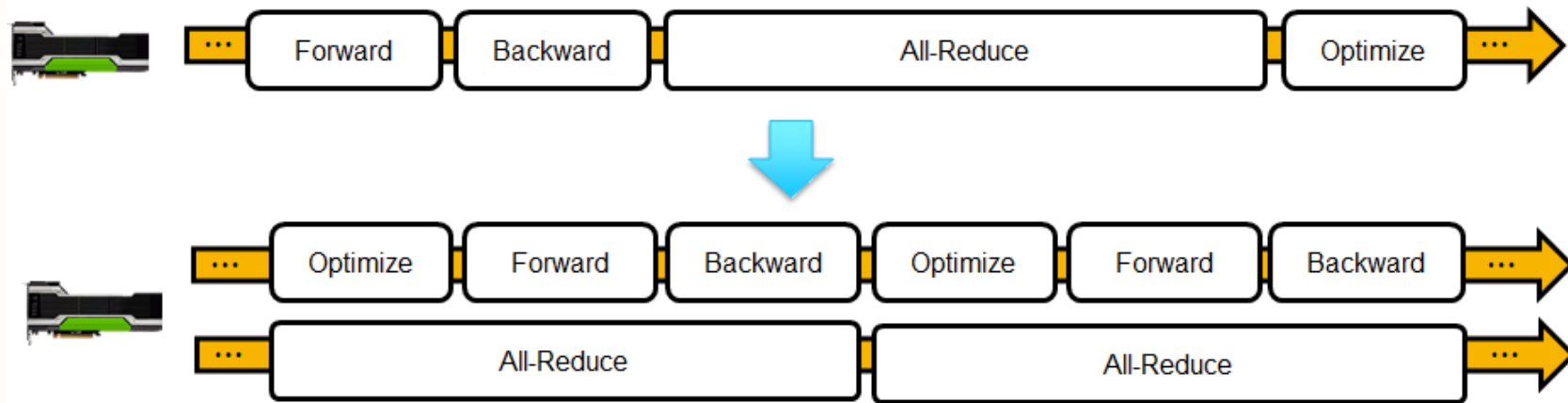
次のトピックでは、複数の GPU、単一の GPU および CPU でのトレーニング、視覚化の作成、および Chainer インストールをテストする方法を説明します。

トピック

- Chainer におけるトレーニングモデル
- Chainer を使用して複数の GPU でトレーニングする
- Chainer を使用して単一の GPU でトレーニングする

1) ChainerMNはInfiniBand前提だからAWSでは性能が出ない！

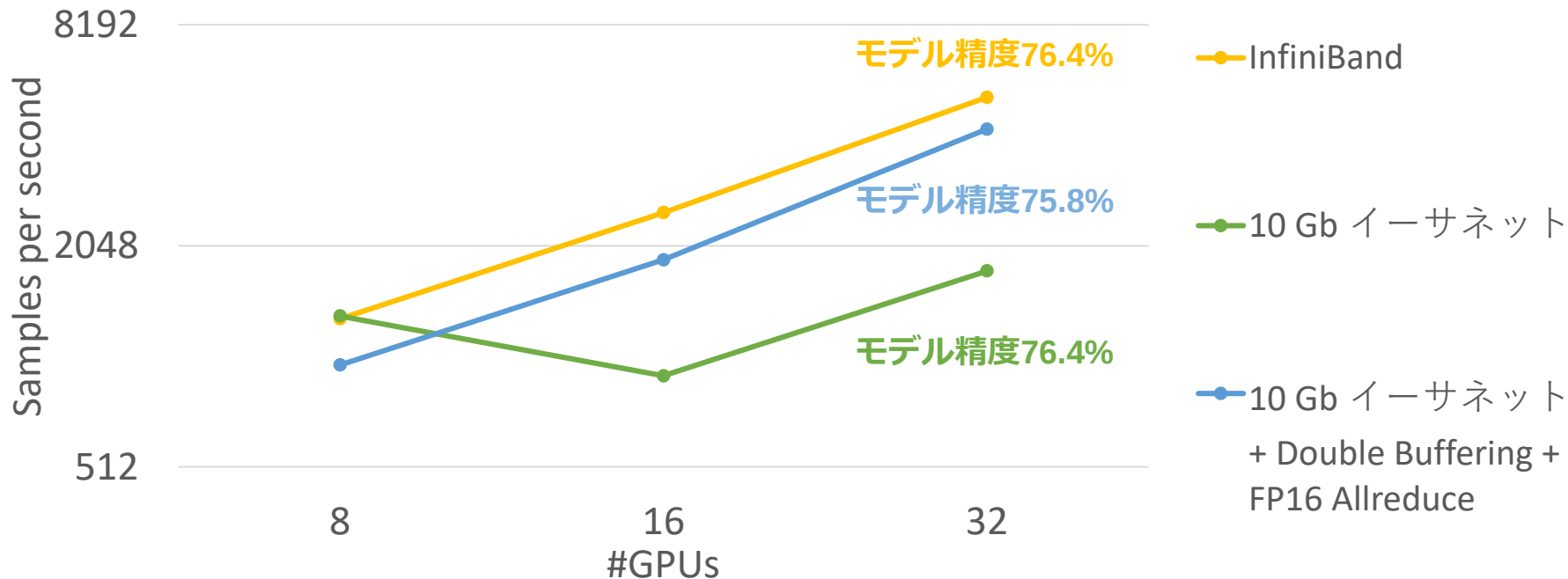
- Chainer v4.1 & ChainerMN v1.3をお使い下さい！
- 追加機能1: Double bufferingで通常のイーサネット環境でも高い分散性能
 - 1ステップ古い勾配を利用してモデル計算を同時に行い通信遅延を隠蔽



- 追加機能2: FP16 All-Reduce: 半精度浮動小数点数を利用し通信量を半減

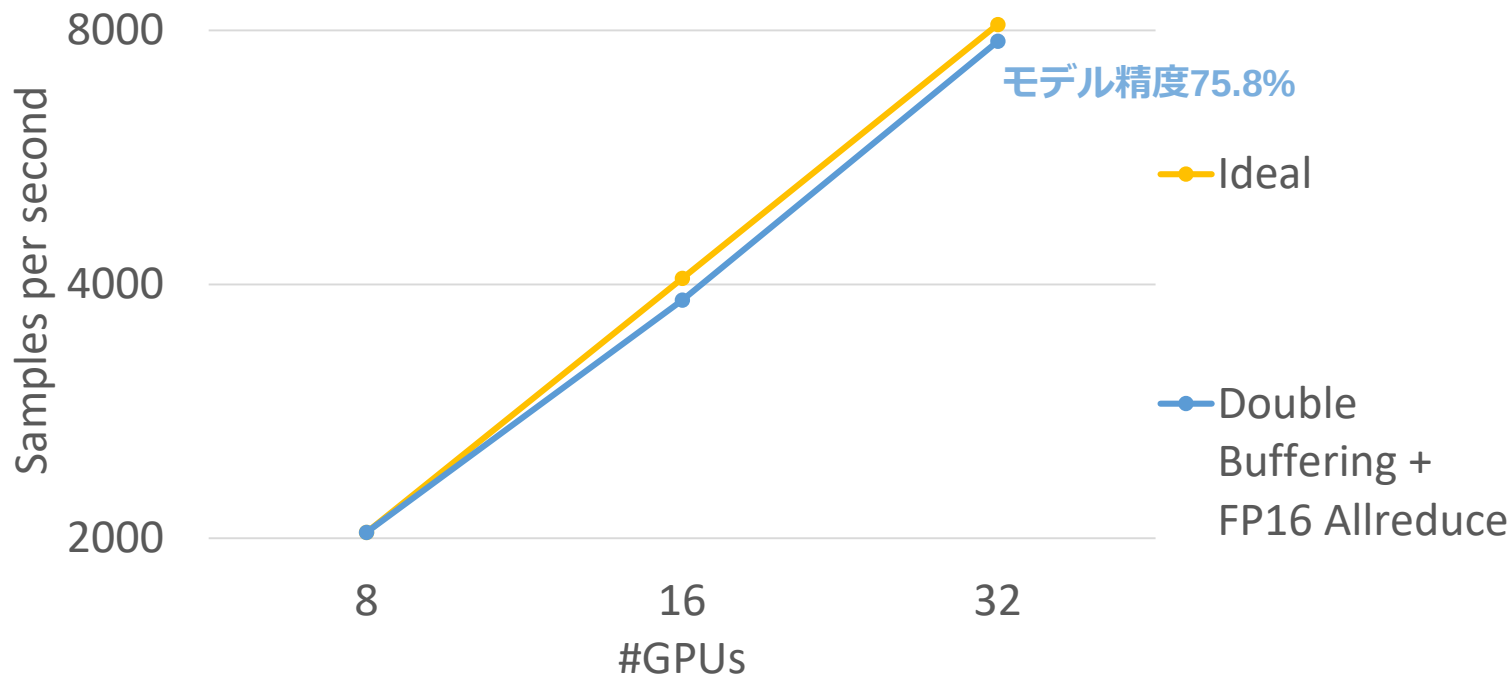
Infinibandの無い環境におけるImageNet学習速度の向上

- 10GbイーサネットにおいてもInfiniBand+FP32時と遜色ない学習速度
- FP16と1ステップ古い勾配を用いることによるモデル精度への影響は微少



AWSのp3インスタンスにおける性能評価実験

- 25Gbpsイーサネット / 32GPU環境で95%の分散性能を達成



2) ChainerMN用GPUクラスタをAWSに自前で構築するのは大変！



- →Chainer/ChhainerMNのCloudFormationテンプレート公開！



chainer-cfn: Cloudformation Template for ChainerMN on AWS

This template automates to build ChainerMN cluster on AWS. The overview of AWS resources to be created by this template are below:

- VPC and Subnet where cluster places (you can configure existing VPC/Subnet)
- S3 Bucket for sharing ephemeral ssh-key which is used to communicate among MPI processes in the cluster
- Placement group for optimizing network performance
- ChainerMN cluster which consists
 - 1 master EC2 instance
 - N (≥ 0) worker instances (via AutoScalingGroup)
 - `chainer` user to run mpi job in each instance
 - `hostfile` to run mpi job in each instance
- (Option) Amazon Elastic Filesystem (you can configure existing filesystem)
 - This is mounted on cluster instances automatically to share your code and data.
- Several required SecurityGroups, IAM Role

Launch Stack



CloudFormationで簡単にマルチノードのChainerMN環境を実現

- テンプレートで定義: ChainerMN入りAMI + クラスタ、MPI...
- ブラウザやCLIからテンプレートを使ってStackの作成を実行
- AWSがプロビジョニング&Stackとリソースを設定



3) Chainerでモデル構築するマネージドサービスがない！

- →Amazon SageMakerがChainerに対応！
- GPUインスタンスを指定してNotebookを立ち上げすぐに利用可能
 - sagemaker-chainer-containerがChainerMNもサポート（MPI、etc）



Chainer on SageMaker (1/3): Estimatorを使って学習Job実行



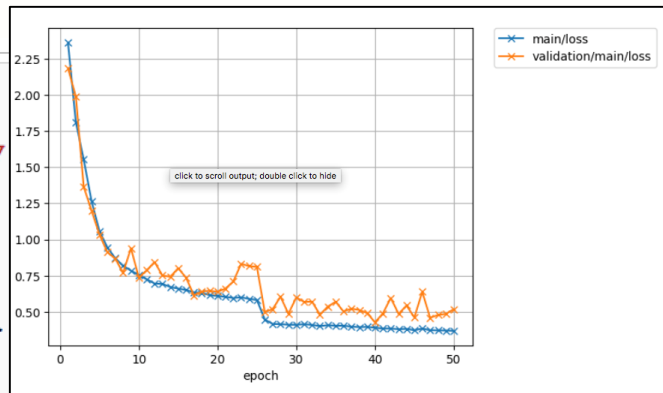
- Notebook上でsagemaker.chainer.estimatorをimport
- 学習スクリプト、IAM role、インスタンス種類／数、パラメータ等をセットしたestimatorのインスタンスを作成
- estimator.fit()で学習Jobを実行

```
] from sagemaker.chainer.estimator import Chainer

chainer_estimator = Chainer(entry_point='chainer_cifar_vgg_single_machine.py',
                             source_dir="src",
                             role=role,
                             sagemaker_session=sagemaker_session,
                             train_instance_count=1,
                             train_instance_type='ml.p3.2xlarge',
                             hyperparameters={'epochs': 50, 'batch-size': 64})

chainer_estimator.fit({'train': train_input, 'test': test_input})
```

INFO:sagemaker:Creating training-job with name: sagemaker-chainer-2018-05-23-20-58-58-434



Chainer on SageMaker (2/3): Deployで予測インスタンス作成

- 学習Jobが完了→学習済みモデルを予測に利用できる
- `estimator.deploy()`で学習用とは別の予測インスタンスを作成
 - 学習はGPUインスタンス→予測はCPUインスタンスなども可能



Deploying the Trained Model

After training, we use the Chainer estimator object to create and deploy a hosted prediction endpoint. We can use a CPU-based instance for inference (in this case an `ml.m4.xlarge`), even though we trained on GPU instances.

The predictor object returned by `deploy` lets us call the new endpoint and perform inference on our sample images.

```
predictor = chainer_estimator.deploy(initial_instance_count=1, instance_type='ml.m4.xlarge')
```

```
INFO:sagemaker:Creating model with name: sagemaker-chainer-2018-05-23-20-58-58-434
INFO:sagemaker:Creating endpoint with name sagemaker-chainer-2018-05-23-20-58-58-434
```

-----!

Chainer on SageMaker (3/3): Predictで予測タスク実行

- estimator.predict()でentry_point中のpredict_fn関数を実行
- 出力の精度評価や可視化などをNotebook上で実行
 - 例 : ChainerCVの可視化機能

```
from chainercv.visualizations import vis_bbox
from chainercv.datasets import voc_bbox_label_names
import matplotlib.pyplot as plt
vis_bbox(image, bbox, label, score, label_names=voc_bbox_label_names)
plt.show()
```



```
] : bbox, label, score = predictor.predict(image)
    print('bounding box: {} \nlabel: {} \nscore: {}'.format(bbox, label, score))
```

```
bounding box: [list([46.97834396362305, 608.4739379882812, 484.9611511230469, 1168.9505615234375])]
label: [11]
score: [0.9990977048873901]
```



4) Chainerモデル学習してもAWSのエッジで予測に使えない！

- AWS GreengrassにChainerプリコンパイル済みライブラリ追加！
- AWS DeepLens、Raspberry Pi、NVIDIA Jetson TX2をサポート



Chainer

powered
by

Machine Learning ライブラリ



機械学習ライブラリは Greengrass Machine Learning リソースの機能で必須になっています。サポートされるプラットフォーム用のプリコンパイル済みライブラリはダウンロード可能です。

ソフトウェア設定

Chainer ▼

すべてのプラットフォームを表示 ▼

モデルタイプ ▼	プラットフォーム ▼	バージョン ▼	License	
Chainer	Intel Apollo Lake	4.0.0	MIT License	📄 ダウンロード
Chainer	Nvidia Jetson TX2	4.0.0	MIT License	📄 ダウンロード
Chainer	Raspberry Pi	4.0.0	MIT License	📄 ダウンロード

まとめ

- ディープラーニングを用いた応用は着実に広がっており、すでに実用化されているサービスや製品が出回っている分野も増えています
- PFNは最先端の研究を行うとともに、その応用に置いて各分野で世界的パートナーと組むことでディープラーニングをいち早く商用化する戦略をとっています
- Chainer/ChainerMNはより多くのユーザーに使って頂けることを目指しておりイーサネット環境向け高速化、CloudFormation/SageMaker/Greengrass対応を達成しましたが、さらにクラウドユーザー向けの開発を進めてまいります

