Introduction to Deep Learning

Amine Kerkeni, Head of Engineering, InstaDeep

ak@instadeep.com

InstaDeep™

History of AI

What’s Artificial Intelligence ?

What's Artificial Intelligence

Machines that mimic human intelligence.

"The theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages."

"AI = Reinforcement Learning + Deep Learning"
David Silver (AlphaZero's inventor) Google DeepMind

The birth

The concept of robot appeared in 1920 in theatre play: Rossum's Universal Robots (R.U.R.) by Karel Čapek.

Alan Turing (1912-1954) introduced the concept of **Turing Machine** and a while after the **Turing Test** (1950).

The Turing Machine is an abstract machine, which can, despite the model's simplicity, construct any algorithm's logic.

The birth

The first checker program and the first chess program in 1951.

Checker programs started beating amateurs in the mid-60s.

In 1955, a program proved 38 of the first 52 theorems in Russell and Whitehead's Principia Mathematica, and found new and more elegant proofs for some.

The birth

The Georgetown experiment is the first attempt to do automatic translation from Russian to English in 1954.

It was based on handcrafted rules, the system was limited to a small subset of words.

The authors claimed that within three or five years, machine translation could be a solved problem.

The birth

In 1956, there was probably the first workshop of Artificial Intelligence and with it the field of AI research was born:

The Dartmouth conference.

“Machines will be capable, within twenty years, of doing any work what man can do.” – Herbert A. Simon (CMU)

“Within a generation … the problem of creating ‘artificer intelligence’ will substantially be solved” – Marvin Minsky (MIT)

The birth

The **Perceptron** algorithm was invented in 1957 at the Cornell Aeronautical Laboratory by Frank Rosenblatt, funded by the United States Office of Naval Research.

Initially the Perceptron was intended to be a machine and it ended up to be a software for IBM 704. It was intended to be used for **image recognition**.

Single layer perceptrons are only capable of learning linearly separable patterns.

The first winter…

In the 70s, the enthusiasm of the 50s and 60s was over and the funding of research in AI started to be scarce.

The UK cut funding for AI, one report specifically mentioned the **combinatorial explosion** problem as a reason for AI's failings.

The hype made expectations very high and the researchers were not able to deliver those promises.

Connectionism was almost dead because of the critics from Marvin Minsky.

The boom

From 1980 till 1987, the paradigm of “expert systems” became popular.

“Expert systems” were rule based systems that were able to answer question in domain specific fields.

Expert systems were applied successfully to design chips (VAX 9000).

In 1982, a new method to train neural net was invented: “back propagation”

The second winter

1987 financial crisis, led to problems in the semiconductor industry and funding for AI research started to go down.

“Expert Systems” quickly faced limitations and were not able to go beyond some specific applications.

AI consolidation

From 1990 until 2011, AI research achieved some success. As compute power was increasing with Moore's Law, connectionism came back to the table.

IBM's Deep Blue was the first program to beat a world champion in Chess (Garry Kasparov) in 1997. It was based on smart tree search.

Connectionism got another set back in early 2000 because researchers did not believe in multi layer neural networks.

The deep learning revolution

In 2006, a publication by Geoff Hinton and some of his students put the foundation of deep learning in practice.

In early 2000, CNN's helped processing between 10 to 20% of cheques in the US.

In 2012, this approach (AlexNet) achieved for the first time superhuman performance in a visual pattern recognition contest (ImageNet).

AI Today

AI Today

MIT Technology Review

Reinforcement Learning

By experimenting, computers are figuring out how to do things that no programmer could teach them.

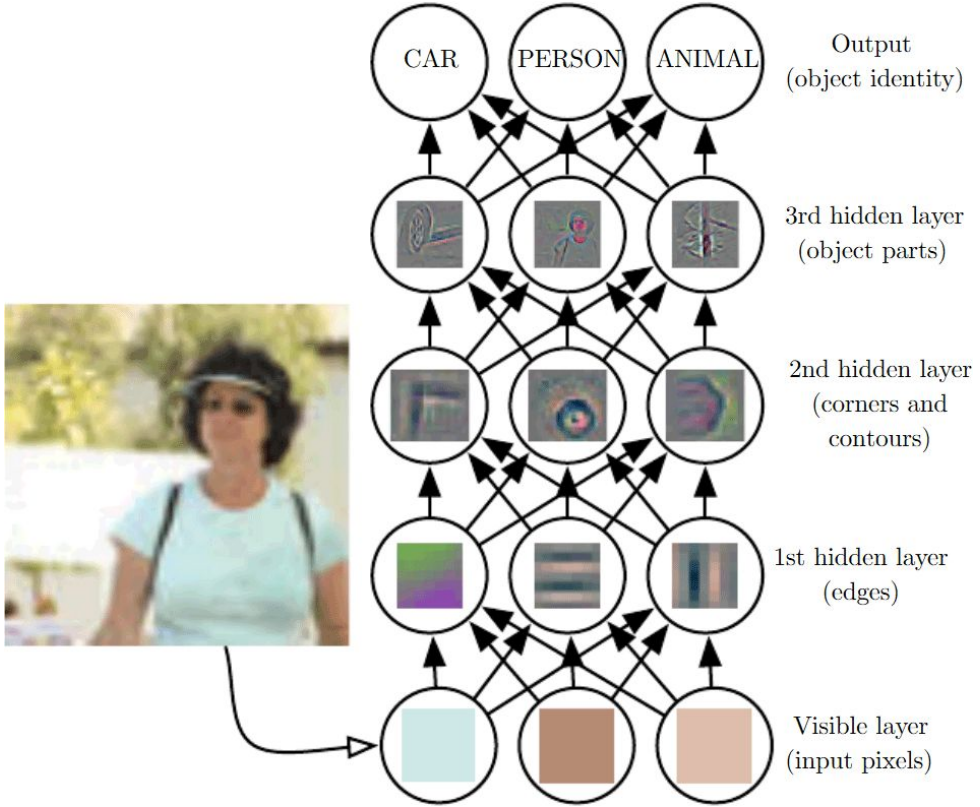
Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

(MIT Tech Review)

Both in MIT TR 10 Breakthrough Technologies (2017 and 2013)

Deep Learning: Pattern Recognition

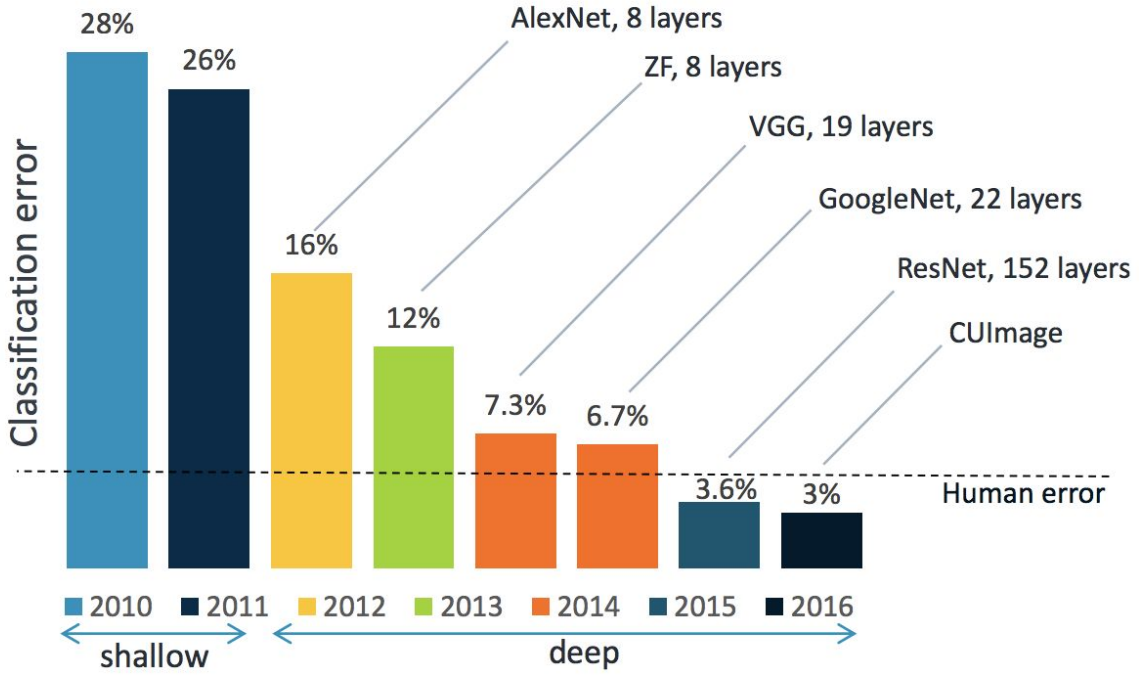


(Deep Learning, Goodfellow, Bengio & Courville 2016)

Nb Parameters:
~50 Million

Mult-Add Ops:
~15 Billion

Deep Learning: Pattern Recognition



(Source: Synopsis)

ImageNet Challenge:
~Million images
1000 categories

Deep Learning:
Top tech since 2012
Superhuman vision

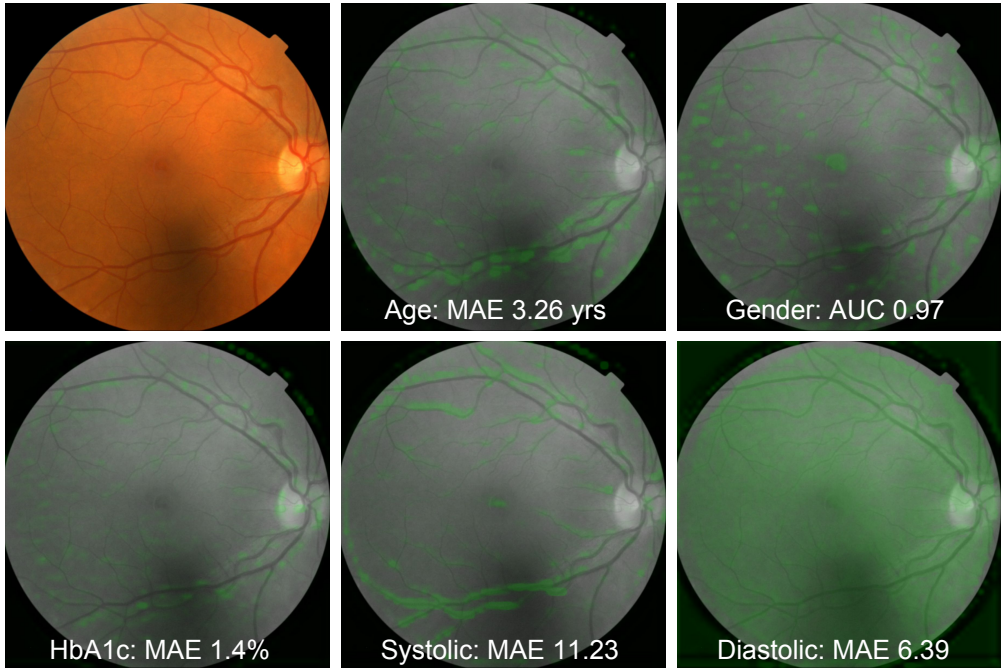
Deep Learning: Scientific Progress

Image of retina

Blood pressure predictions
focus on blood vessels

TensorFlow ™

Deep Learning: Scientific Progress



Predicting things that doctors can't predict from imaging!

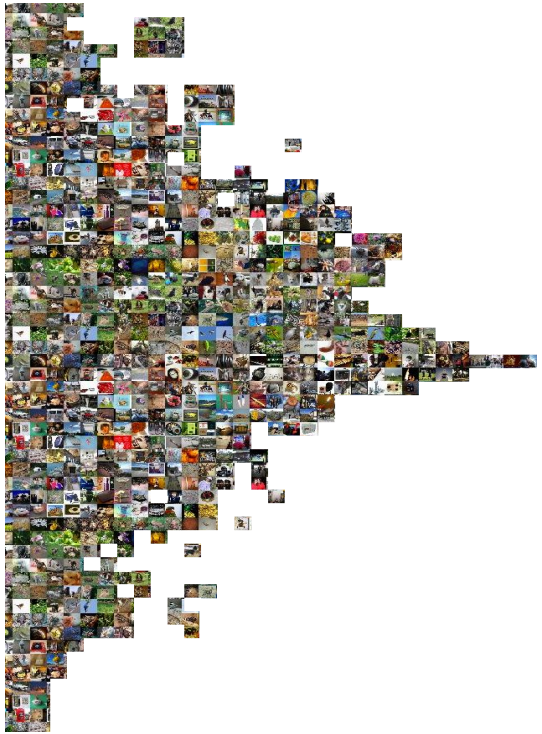
5-yr Cardiovascular risk: AUC 70%

Predict risks in a nice non invasive way!

R. Poplin, A. Varadarajan et al. Predicting Cardiovascular Risk Factors from Retinal Fundus Photographs using Deep Learning. Nature Biomedical Engineering, 2018.

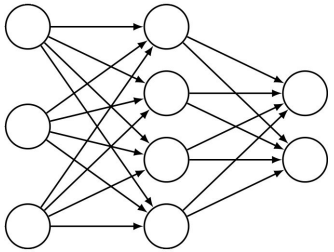
TensorFlow ™

Deep Learning: 3 reasons for success

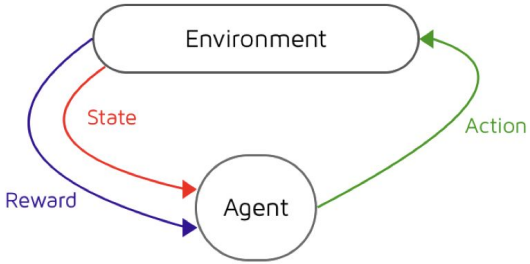


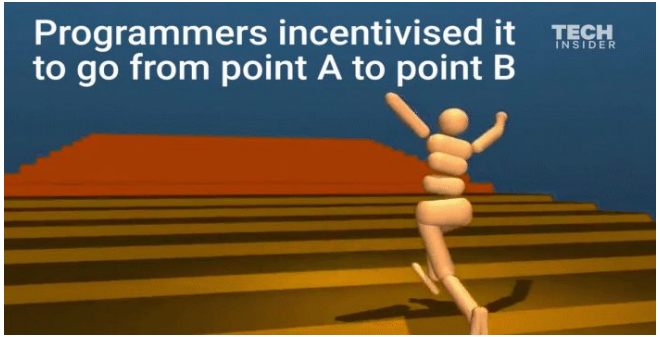
+

+

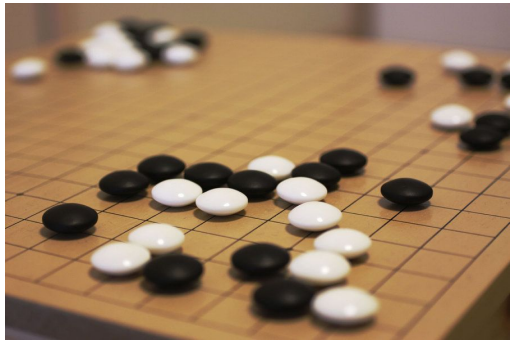


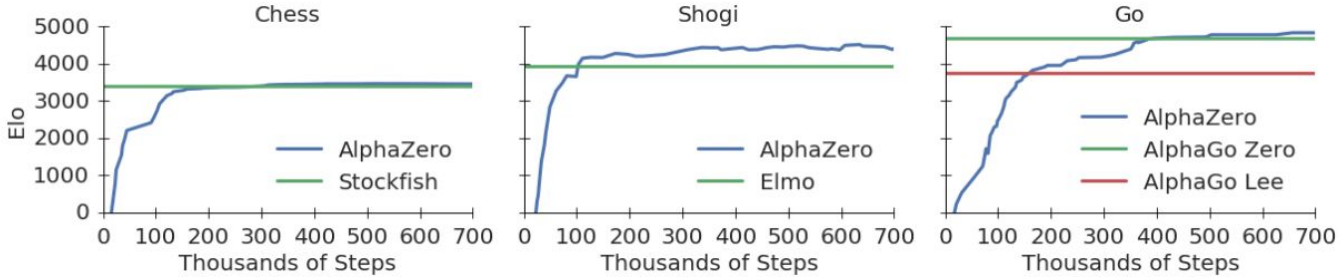
Reinforcement Learning (RL)





A new type of AI Champion





(*Silver et Al. 2017*)

Starting random, **beats world's best Chess and Go programs in hours** using distributed ML

A new type of AI Champion

OpenAI Five: **~1,000** actions to choose from (Chess **~35**, Go **~250**)
State space: **20,000 Dimensions** (Chess **70D**, Go **400D**)

	OPENAI 1V1 BOT	OPENAI FIVE
CPUs	60,000 CPU cores on Azure	128,000 preemptible CPU cores on GCP
GPUs	256 K80 GPUs on Azure	256 P100 GPUs on GCP
Experience collected	~300 years per day	~180 years per day (~900 years per day counting each hero separately)

AI Tomorrow

AI for Data

(*Creative Adversarial Nets*, *Elgammal & al. 2017*)

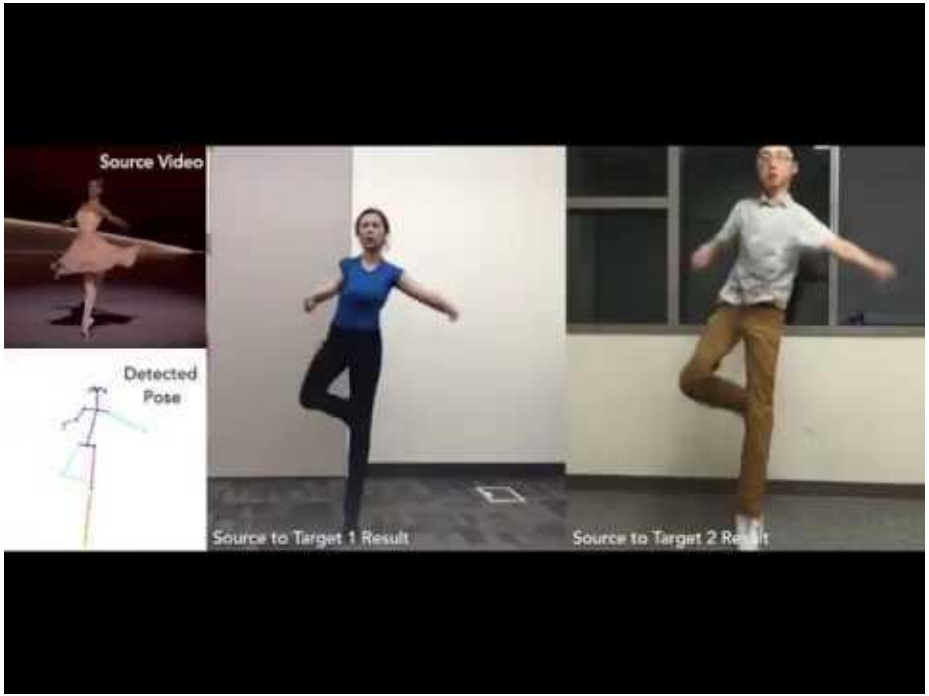
Painting set	Q1 (std)
CAN	53% (18%)†

(*Towards Automatic Anime Characters Creation*, *Jin & al. 2017*)

(*Progressive Growing of GANs*, *Karras & al. 2017*)

AI for Data

(Video-to-Video Synthesis, Wang & al. 2018)



(Everybody Dance Now, Chan & al. 2018)

AI for Hardware Computing

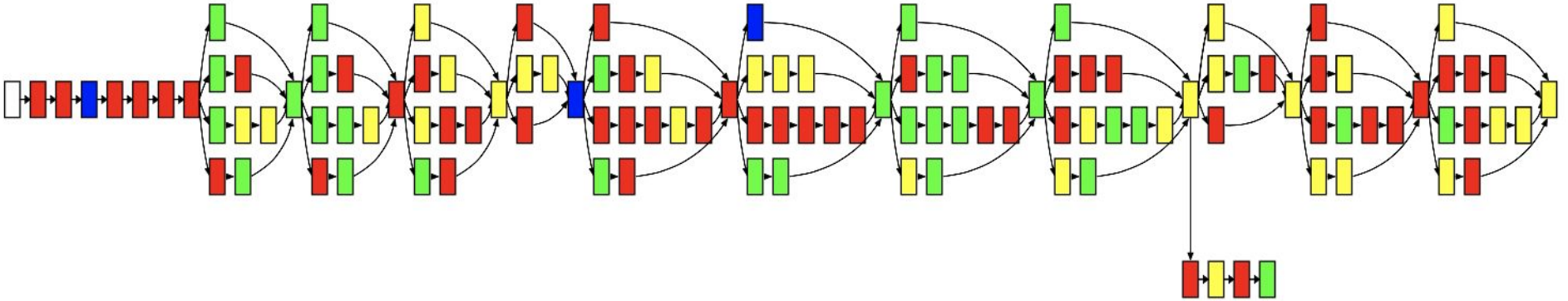


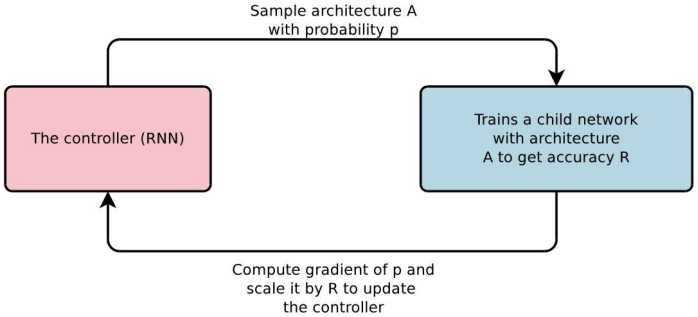
Figure 5. RL-based placement of Inception-V3. Devices are denoted by colors, where the transparent color represents an operation on a CPU and each other unique color represents a different GPU. RL-based placement achieves the improvement of 19.7% in running time

(Device Placement Optimization with Reinforcement Learning, Mirhoseini & al. 2017)

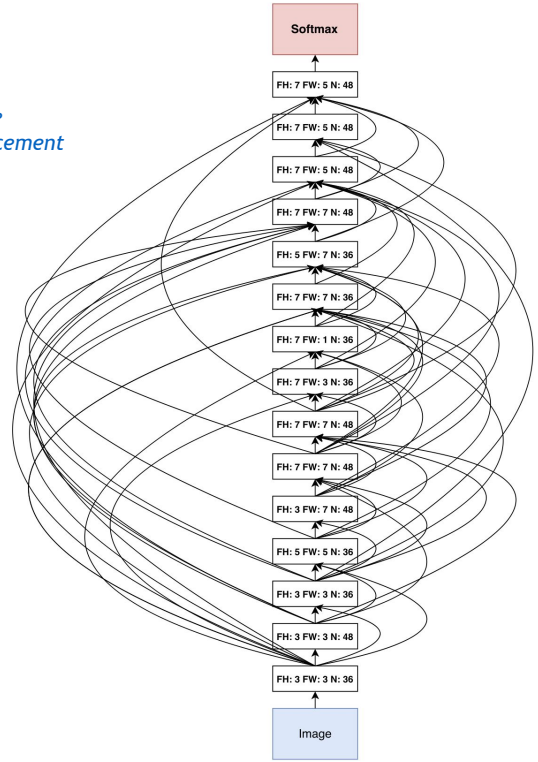
19.7% faster than GPU experts for Deep Learning tasks

AI for Deep Learning

AI builds its own competitive neural network:



(*Neural Architecture Search with Reinforcement Learning*, *Zoph & Le, 2017*)

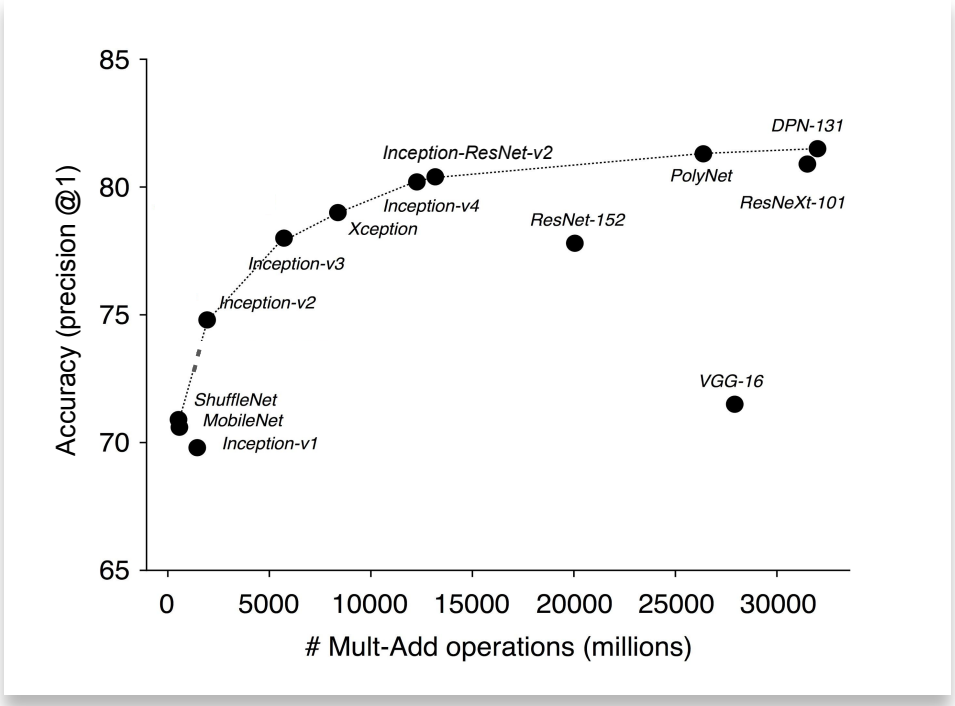


Method	GPUs	Times (days)	Params (million)	Error (%)
NAS (Zoph & Le, 2017)	800	21-28	7.1	4.47
NAS + more filters (Zoph & Le, 2017)	800	21-28	37.4	**3.65**
ENAS + macro search space	1	0.32	21.3	4.23
ENAS + macro search space + more channels	1	0.32	38.0	**3.87**

(*Efficient Neural Architecture Search via Parameter Sharing*, *Pham & al. 2018*)

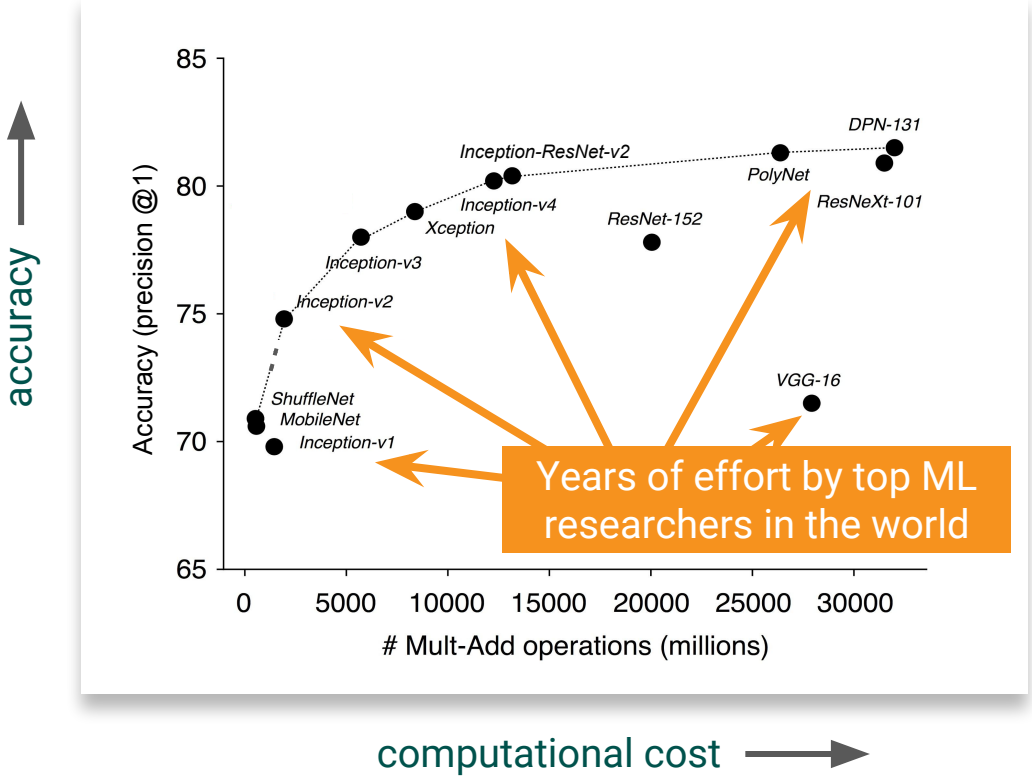
State of the art results with a **1000X speed-up in a year**

AI for Deep Learning



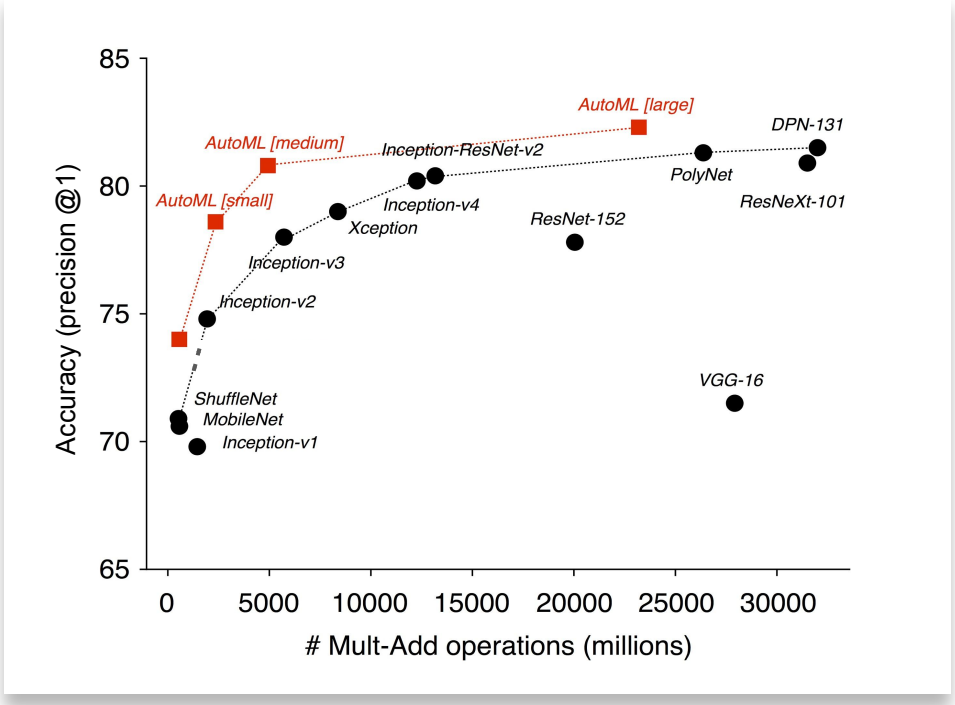
Learning Transferable Architectures for Scalable Image Recognition, Zoph et al. 2017, https://arxiv.org/abs/1707.07012

AI for Deep Learning



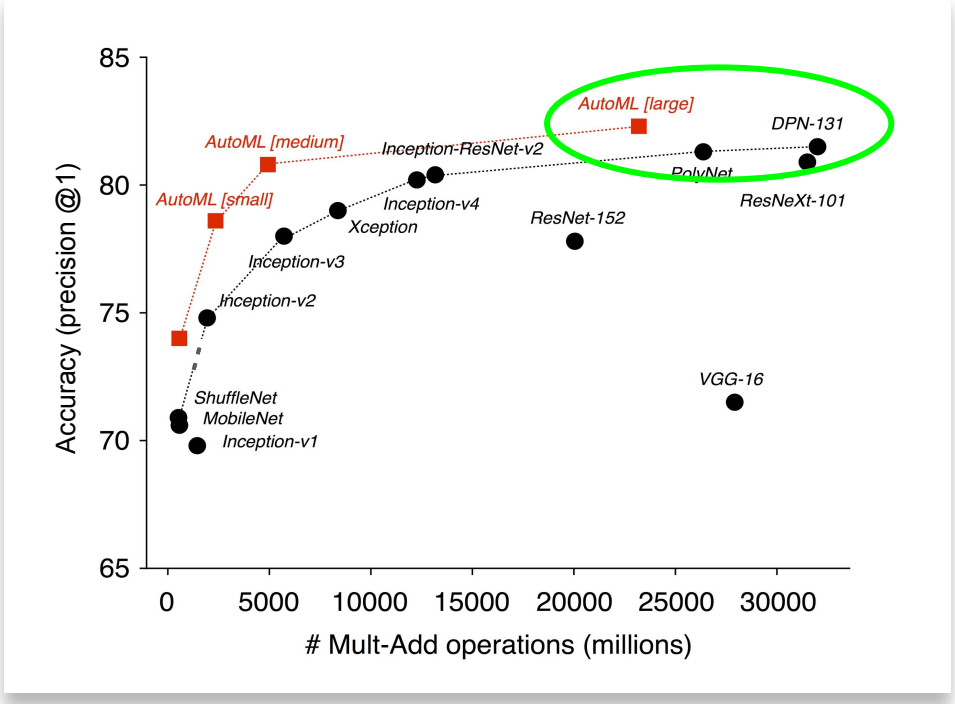
Learning Transferable Architectures for Scalable Image Recognition, Zoph et al. 2017, https://arxiv.org/abs/1707.07012

AI for Deep Learning



Learning Transferable Architectures for Scalable Image Recognition, Zoph et al. 2017, https://arxiv.org/abs/1707.07012

AI for Deep Learning



Learning Transferable Architectures for Scalable Image Recognition, Zoph et al. 2017, https://arxiv.org/abs/1707.07012

Data

AI

DATA

Example: Visual AI

How to quickly build a visual AI tool?

1000s of products ?

No model ??

No data ???

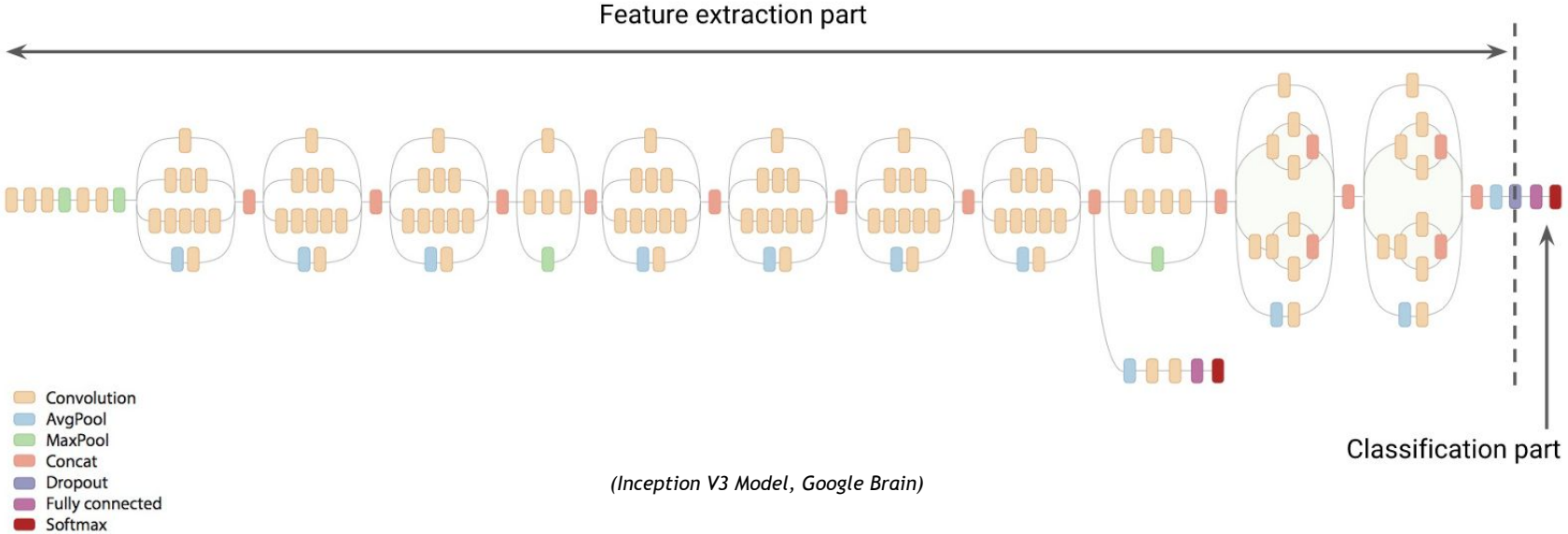
Visual AI: Data Tricks

Take (or collect from the net) ~300 original pics for each product.

Each pic can generate 20 augmented pics (angle, blurring, background etc..). 300 -> 300*20 = **6,000 pics/product.**

Visual AI: Model Tricks

Don’t train from scratch. Use a trained model and **transfer the learning.**



(Inception V3 Model, Google Brain)

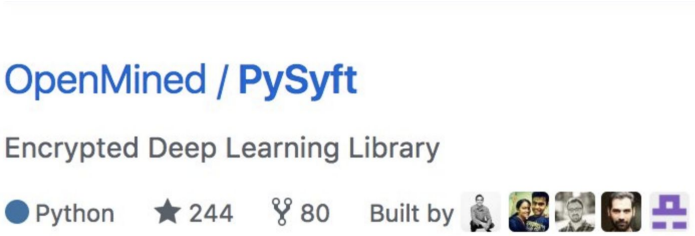
Keep low layers the same. Retrain top layers to classify your own labels
Can be done on the Google Cloud or on TensorFlow.

2 Suggestions for Data:

Simulated/Synthetic Data: when real data is expensive or sensitive to get (self-driving cars, personal medical files, etc)

Decentralized Data gathering with encryption and digital smart contracts:

"You can train my AI model with your data, without you knowing my model or me knowing your data" Andrew Trask, OpenMined



Example:

(Standard Cognition demo, Aug 2017)

Real-time now feasible:

TensorFlow-powered

High-performance edge models.

Real+Augmented Data

Optimized Deep Learning

TensorFlow Lite (Mobile)

The Tools

1. Cloud Based Solutions

Machine Learning in Google Cloud

Pre-trained ML models

Natural Language API

Vision API

Speech API

Translation API

Jobs API

Video Intelligence API

Cloud Based Solutions

Cloud AutoML BETA

Train Custom Machine Learning Models

AutoML nominated in MIT TR 10 Breakthrough Technologies (2018) as an "AI for Everybody" tech.

MIT Technology Review

2. Custom ML

http://tensorflow.org/

Open, standard software for general machine learning

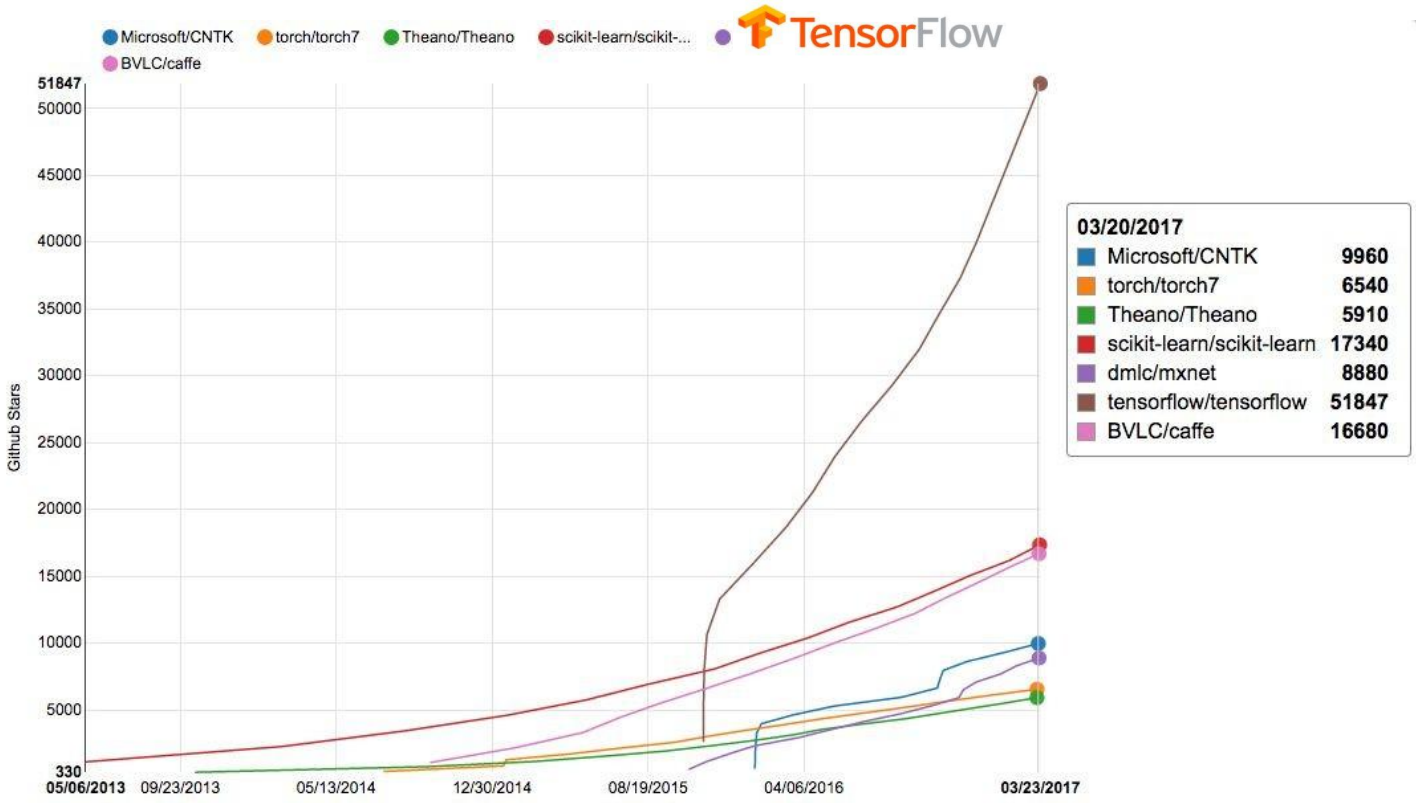
Great for AI/Deep Learning in particular

First released Nov 2015

Version 1.11 released

Apache 2.0 license

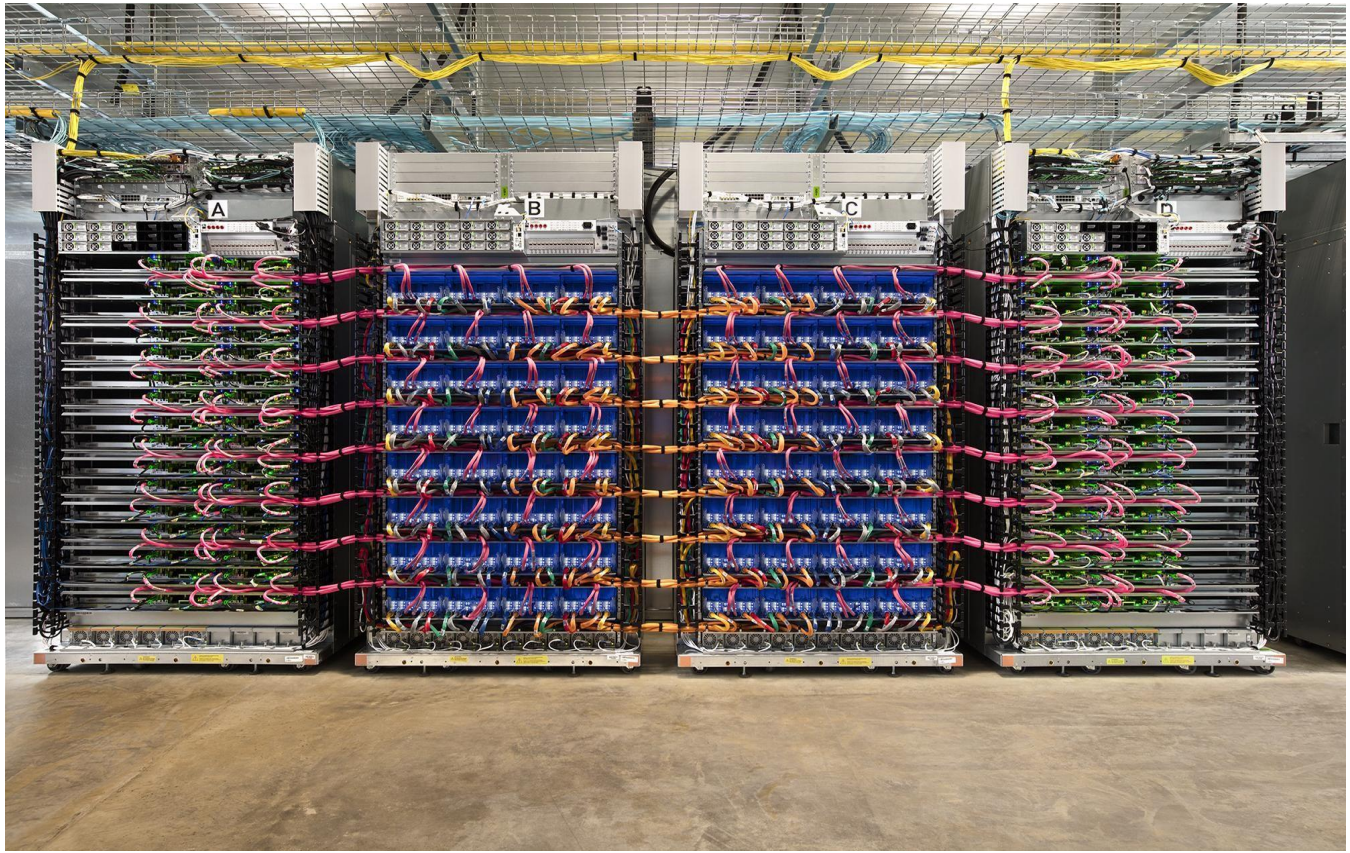
ML software Github Stars



(Jeff Dean, Google Brain 2017)

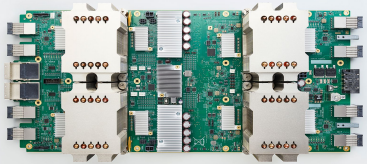
3. Hardware

CLOUD TPU

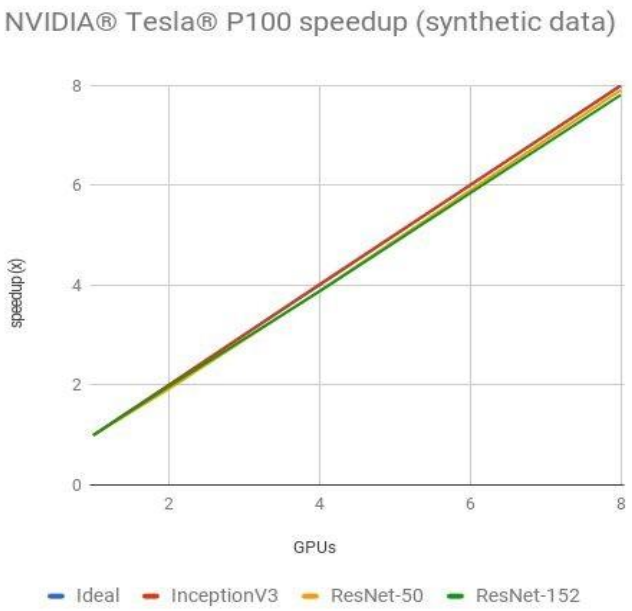


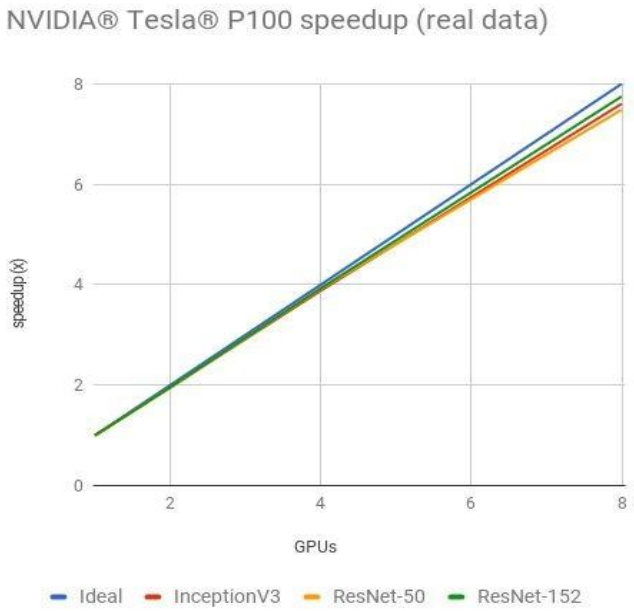
TPU Pod
64 2nd-Gen TPUs
11.5 Pflops
4 TB RAM

2nd-Gen TPU:
180 Tflops 64 GB Ram



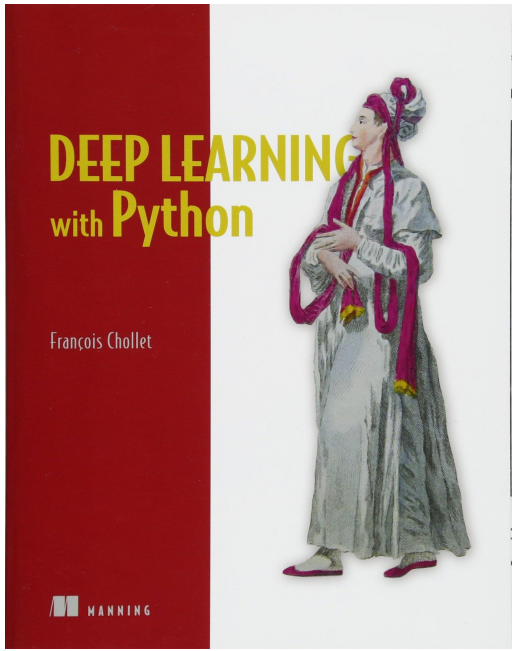
TensorFlow scales on GPUs too





(Jeff Dean, Google Brain 2017)

Learning Resources



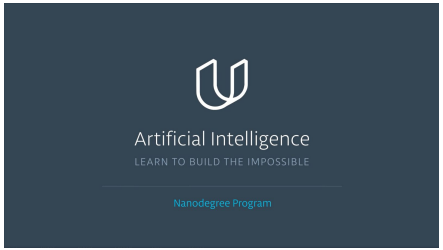
Deep Learning for coders :
http://fast.ai

Deep Learning without a Phd by Martin Gorner:
https://cloud.google.com/blog/products/gcp/learn-tensorflow-and-deep-learning-without-a-phd

Learning

Books: *Deep Learning (Goodfellow)*, *RL: An Introduction (Sutton)*

AI Online Classes: UDACITY coursera



How to challenge yourself? Global AI/Data Science competitions:

kaggle

Course Structure

1. Introduction to Deep Learning
   - History and key factors of the deep learning revolution
   - Infrastructure of deep learning
   - Training you first CNN
   - Tensorflow basics
2. Multi layer perceptron with Tensorflow:
   - Softmax
   - Activations: Relu, Sigmoid
   - Learning Rate
   - Dropout
   - Batch Normalization
3. Backpropagation:
   - Implementing the backpropagation algorithm
4. CNN:
   - Diving in convolutional neural networks
   - Implementing a CNN in Tensorflow from scratch
5. RNN:
   - The theory behind RNN
   - LSTM
   - Implementing an RNN in tensorflow
6. GAN:
   - Generative adversarial networks, the theory behind
   - Implementing a GAN in tensorflow
7. Application of deep learning and its future:
   - Deep learning in computer vision
   - Deep learning in NLP
   - Deep learning in speech processing
   - Deep learning in business
   - Research trends

Our course

Course material are in:

https://github.com/ak-instadeep/deeplearning-enit

https://github.com/deep-learning-indaba/indaba-2018

We will use Google Collab for our tutorials: https://colab.research.google.com

You can contact me for any question on my email, please prefix the email title with **ENIT-DL**: ak@instadeep.com