Machine Learning in imaging and more!

22 Feb, 2018

Varun Gulshan
Research Scientist@Google Brain
g.co/Brain

Selfie introduction

B Tech, Computer Science IIT Delhi, 2007

PhD, Computer Vision and Machine Learning, University of Oxford, 2011

Research Engineer @ Flutter (A computer vision startup, acquired by Google in 2013)

Research Scientist at Google [current]

Outline

- Deep dive into a specific Medical imaging problem (Diabetic retinopathy screening)
- Overview of Neural networks (specifically Convolutional neural network) and optimization.
- The zoo of ML models and applications

Diabetic retinopathy (DR) screening

Trained technician

or a consumer device (eye-selfie)

Fundus image

Automated Diagnosis

- Healthy, no follow up needed.
- Follow up with Ophthalmologist.

Build an automated fundus image reading algorithm to refer 'referable' cases to an Ophthalmologist.

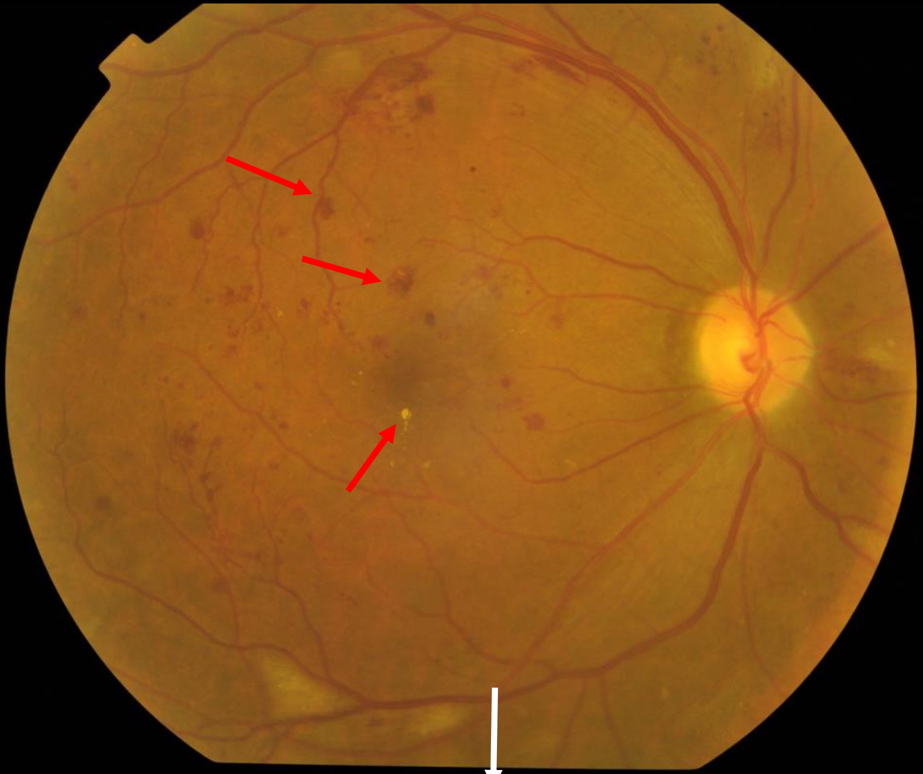
Fundus imaging to diagnose DR

=

DR: 5 point scale

No DR	Mild DR	Moderate DR	Severe DR	Proliferative DR

DR: 5 point scale

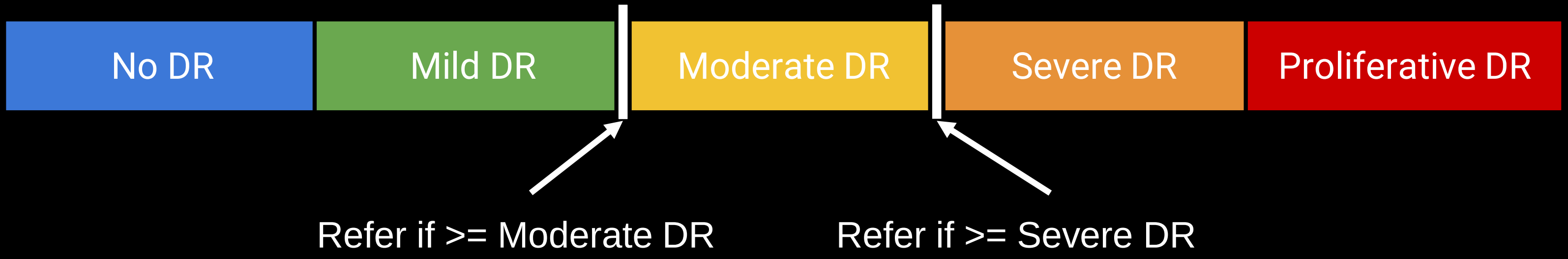


No DR	Mild DR	Moderate DR	Severe DR	Proliferative DR

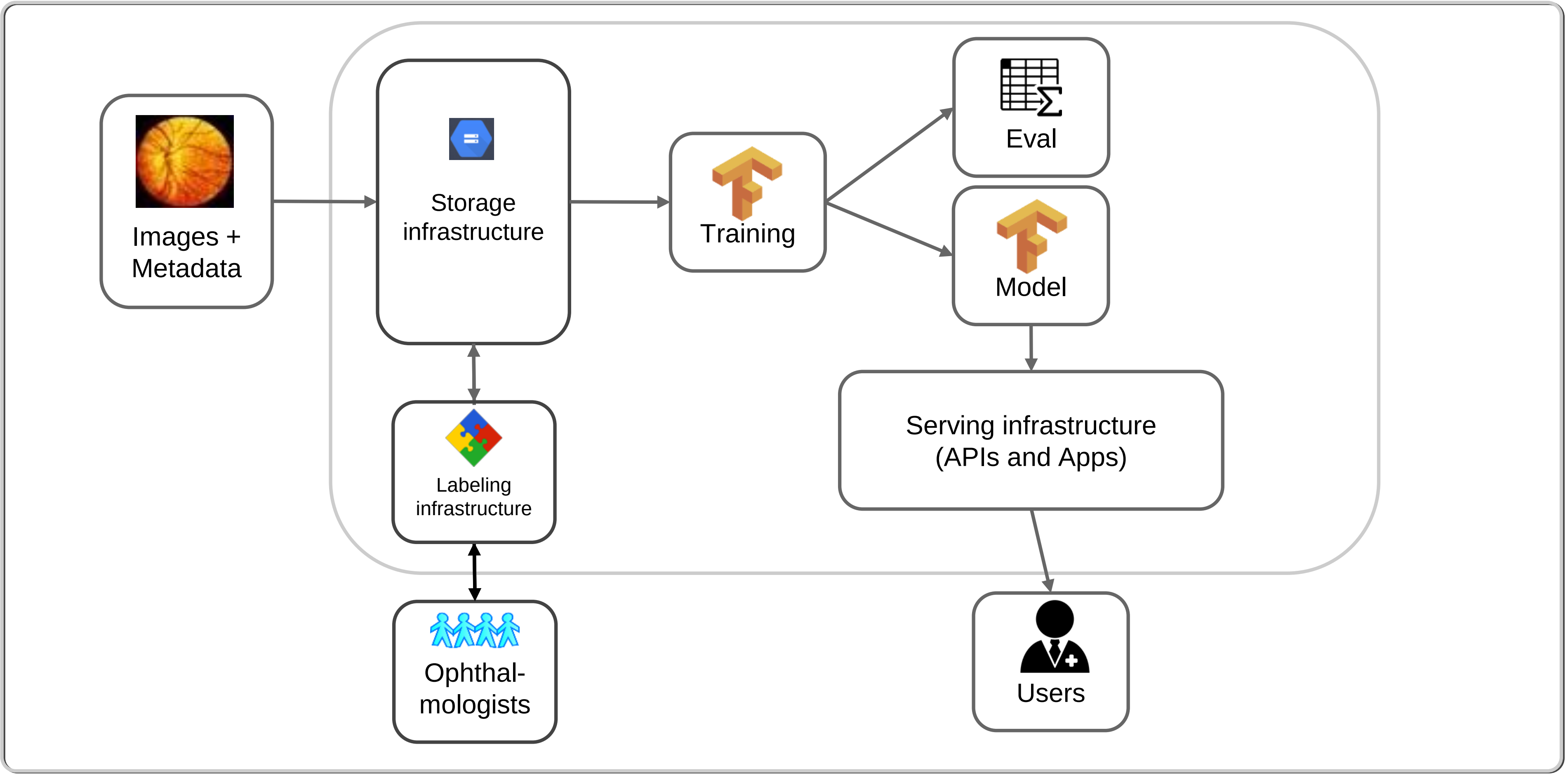
DR: 5 point scale

No DR	Mild DR	Moderate DR	Severe DR	Proliferative DR

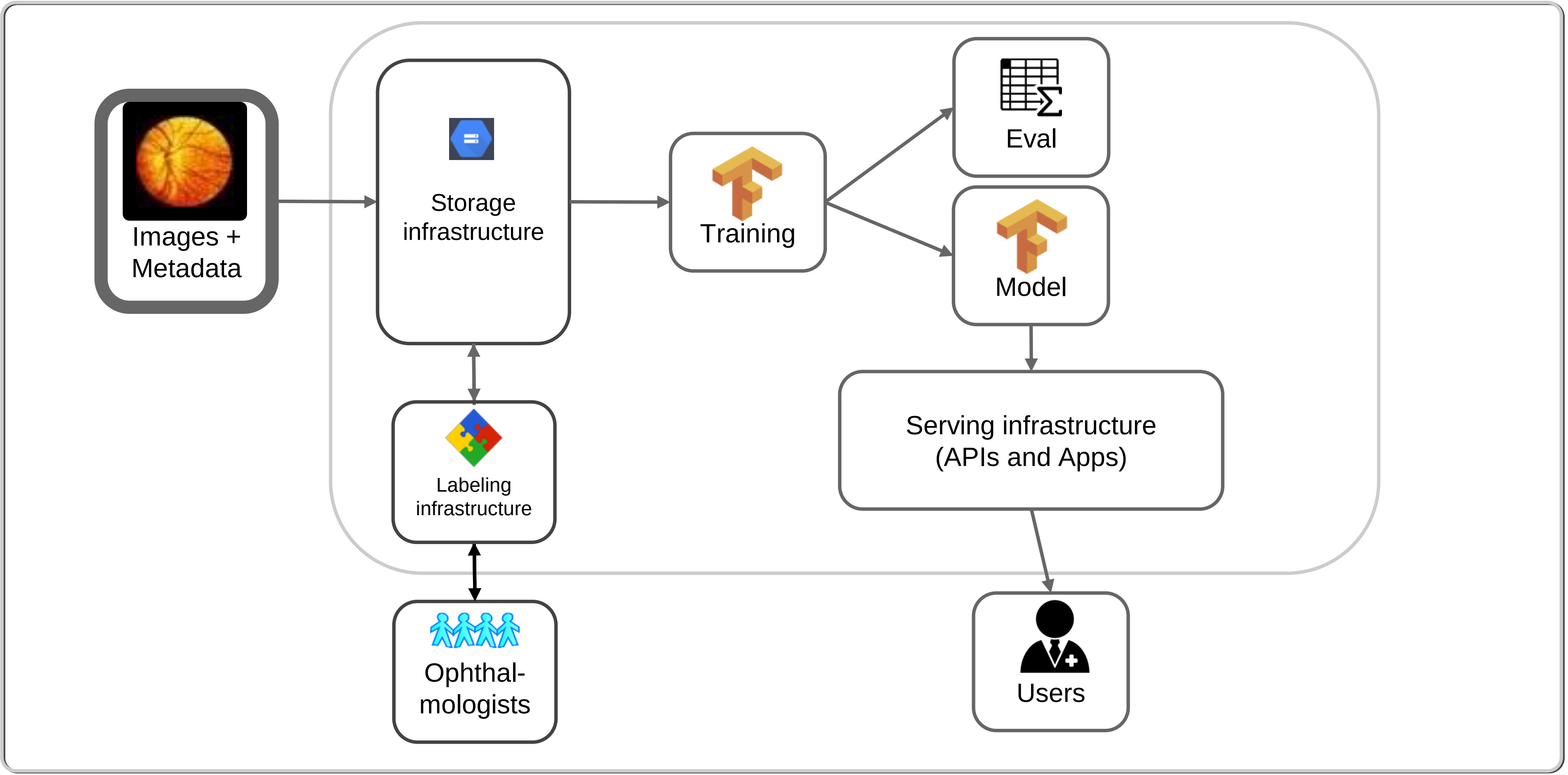
DR: Referability thresholds



Building blocks of a ML system



Building blocks of a ML system



Images and metadata

Data partners

eyePACS

SANKARA Eye Care Institutions

NARAYANA NETHRALAYA your faith shall heal you

ARAVIND EYE CARE SYSTEM

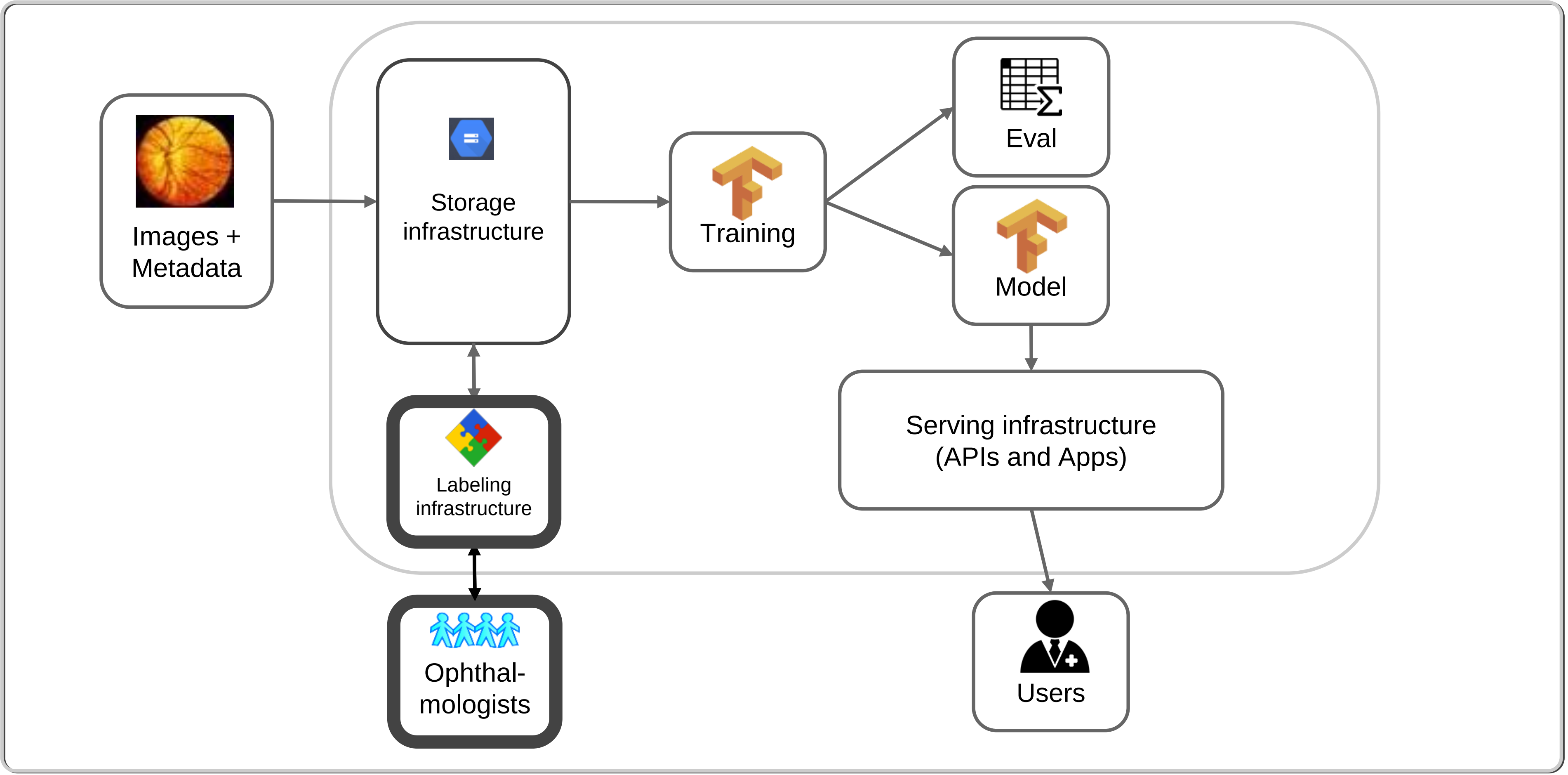
biobank uk Improving the health of future generations

AREDS

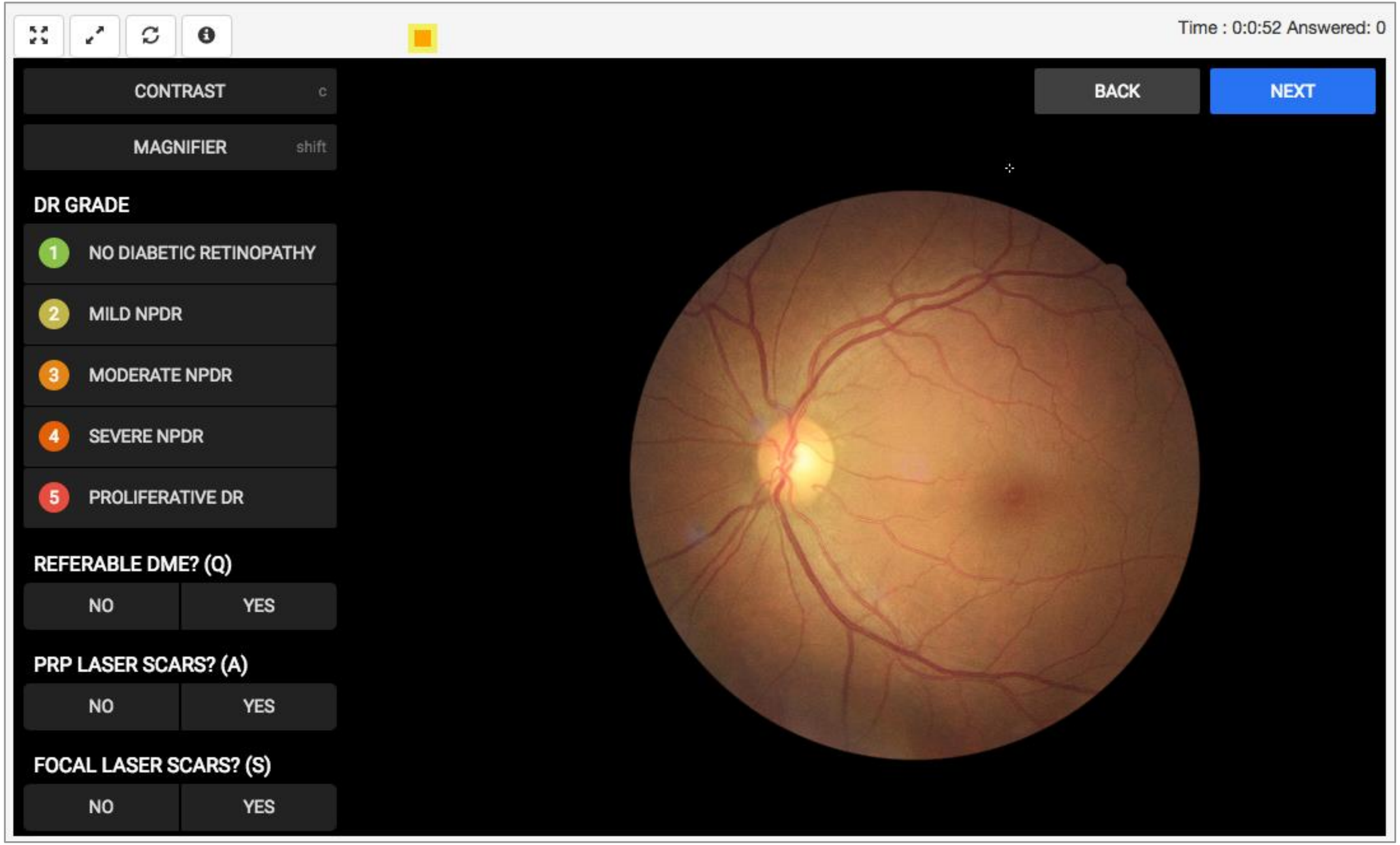
2 Million+ images across diverse ethnicities, age groups, gender, confounding diseases.

Some come with the partners DR grading as well.

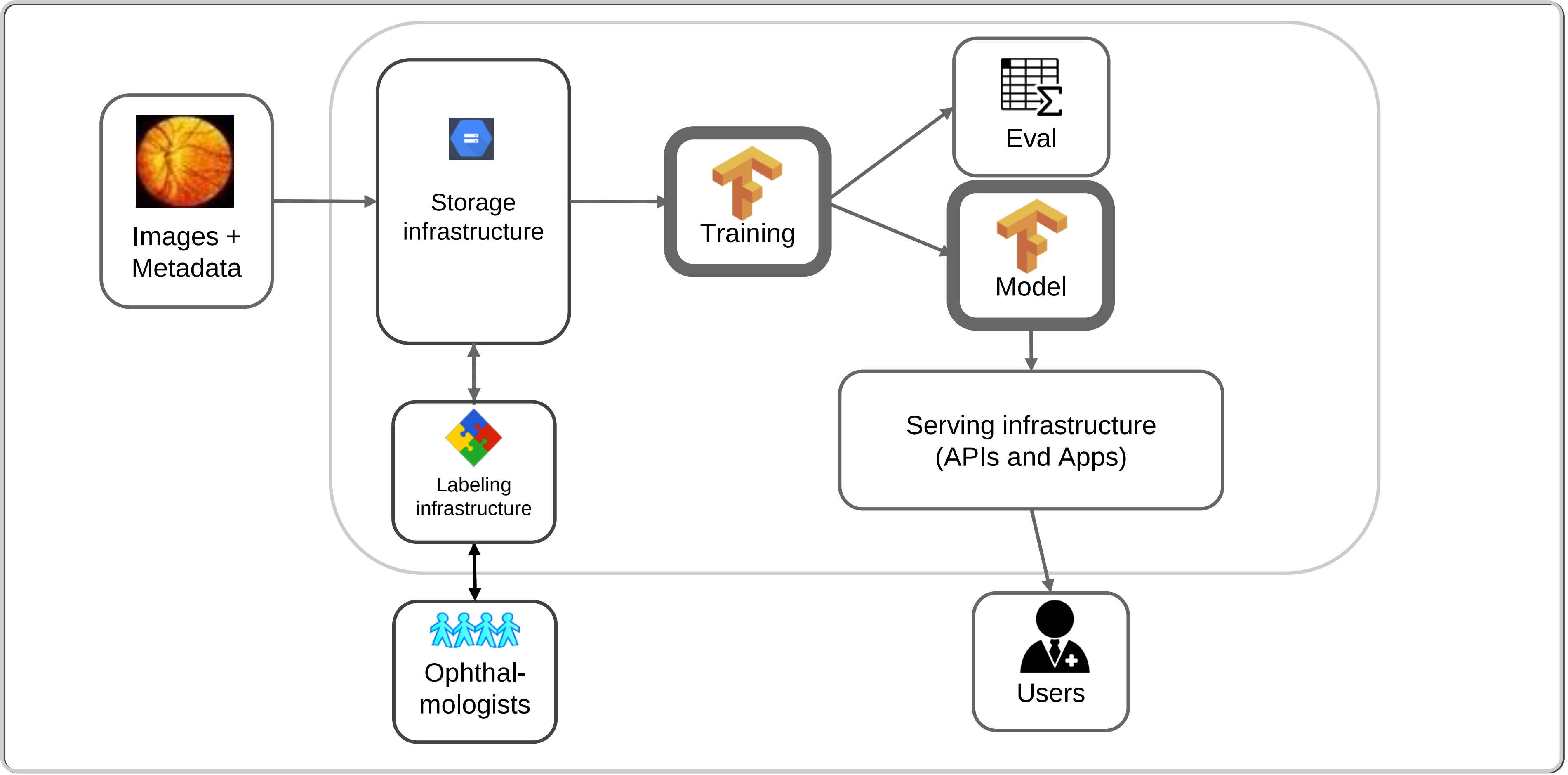
Building blocks of a ML system



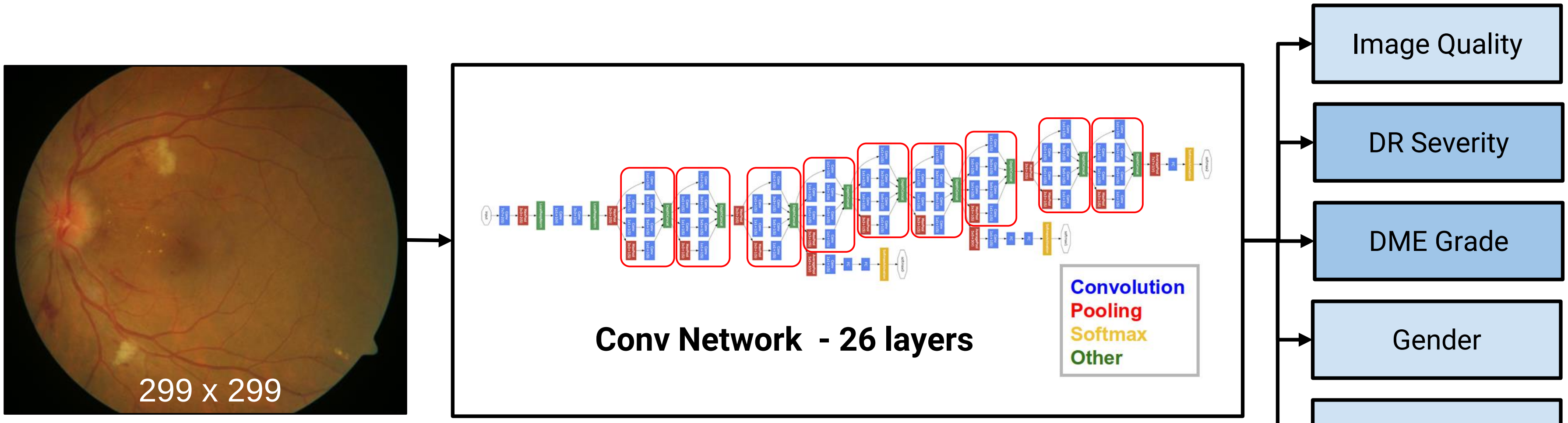
Labeling tool for Ophthalmologists



Building blocks of a ML system

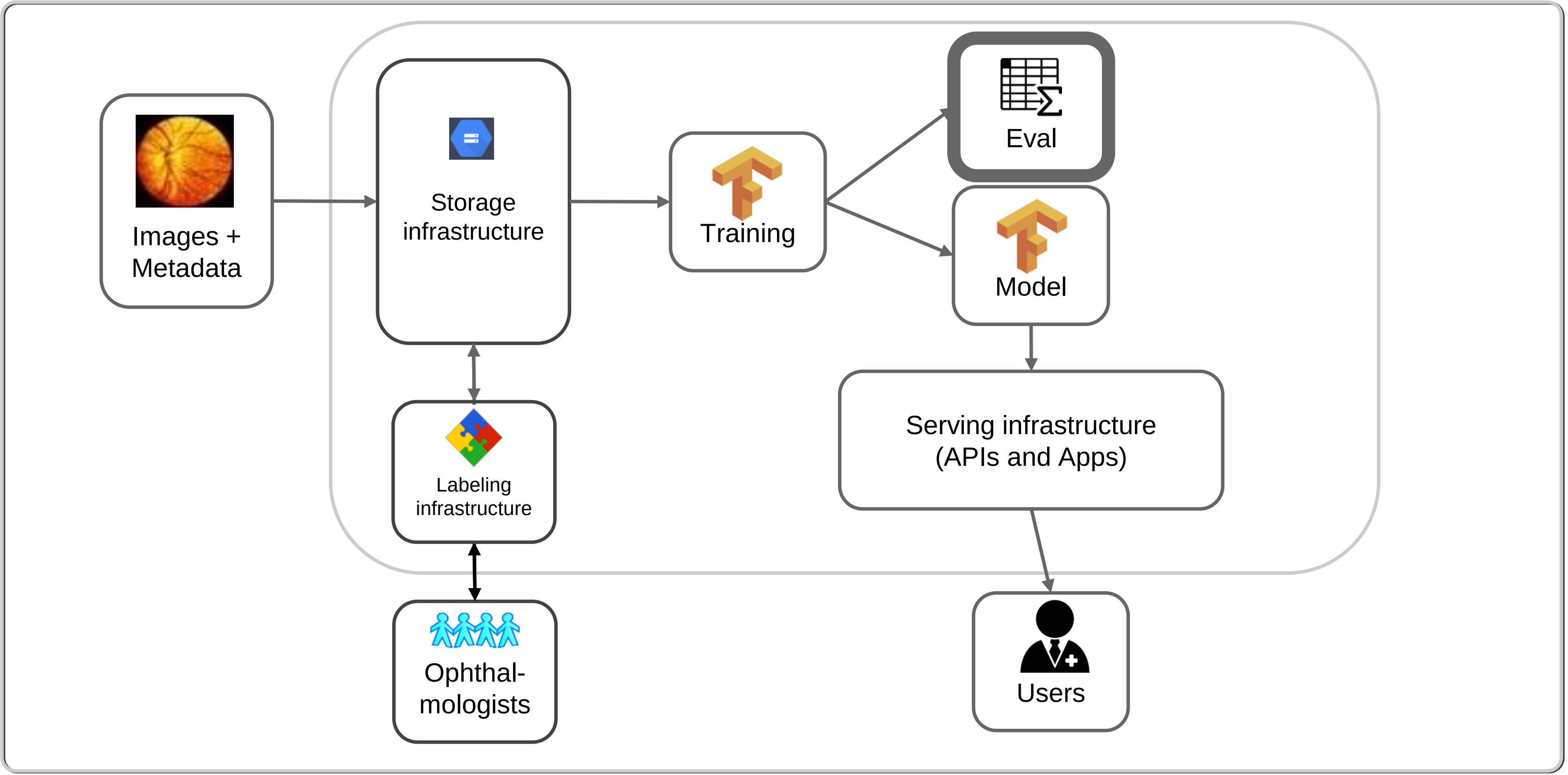


Model: Deep neural networks

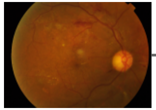


GoogLeNet

Building blocks of a ML system



Measuring performance: Sensitivity and specificity



DR grader

Binary decision

Ok

Visit ophtha

Sensitivity

100 referable patients

DR grader

3 Ok

97 Visit ophtha

$$\text{Sensitivity} = \frac{\text{True positives}}{\text{Total positives}} = 97\%$$

Specificity

1000 healthy patients

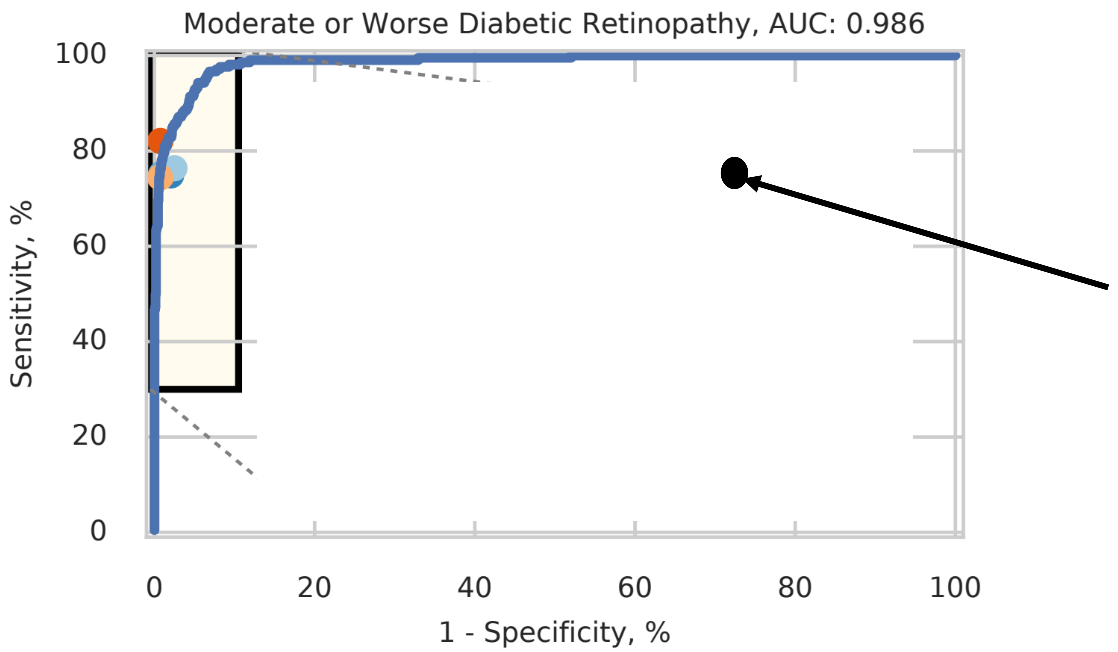
DR grader

900 Ok

100 Visit ophtha

$$\text{Specificity} = \frac{\text{True negatives}}{\text{Total negatives}} = 90\%$$

ROC curves



	DR	
	Sensitivity	Specificity
Algorithm	97.1%	92.3%

Full paper at: https://arxiv.org/pdf/1710.01711.pdf [Published in Ophthalmology]

Improvement of our original work published in JAMA: https://research.google.com/pubs/pub45732.html

Project ARDA

SIGN OUT

Drag another image to analzye, or **CHOOSE IMAGE**

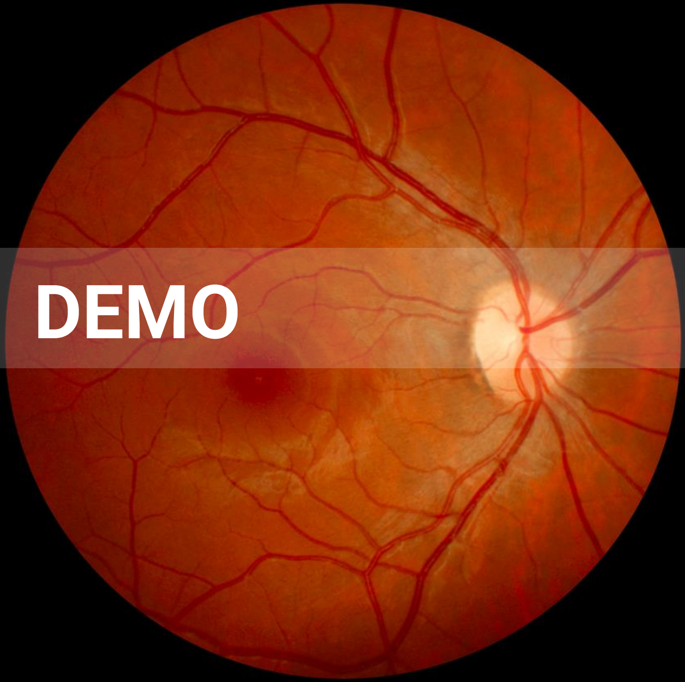
FILENAME (SIZE)
07.jpg (276 KB)

DIABETIC RETINOPATHY GRADE
1.7 (Mild)

DIAGNOSIS ID
028371031

Recent Images

FILENAME / DR GRADE
07.jpg / 2.6
06.jpg / 3.2
05.jpg / 2.3
04.jpg / 2.6
03.jpg / 1.6
02.jpg / 0.9



DEMO

Outline

- Deep dive into a specific Medical imaging problem (Diabetic retinopathy screening)

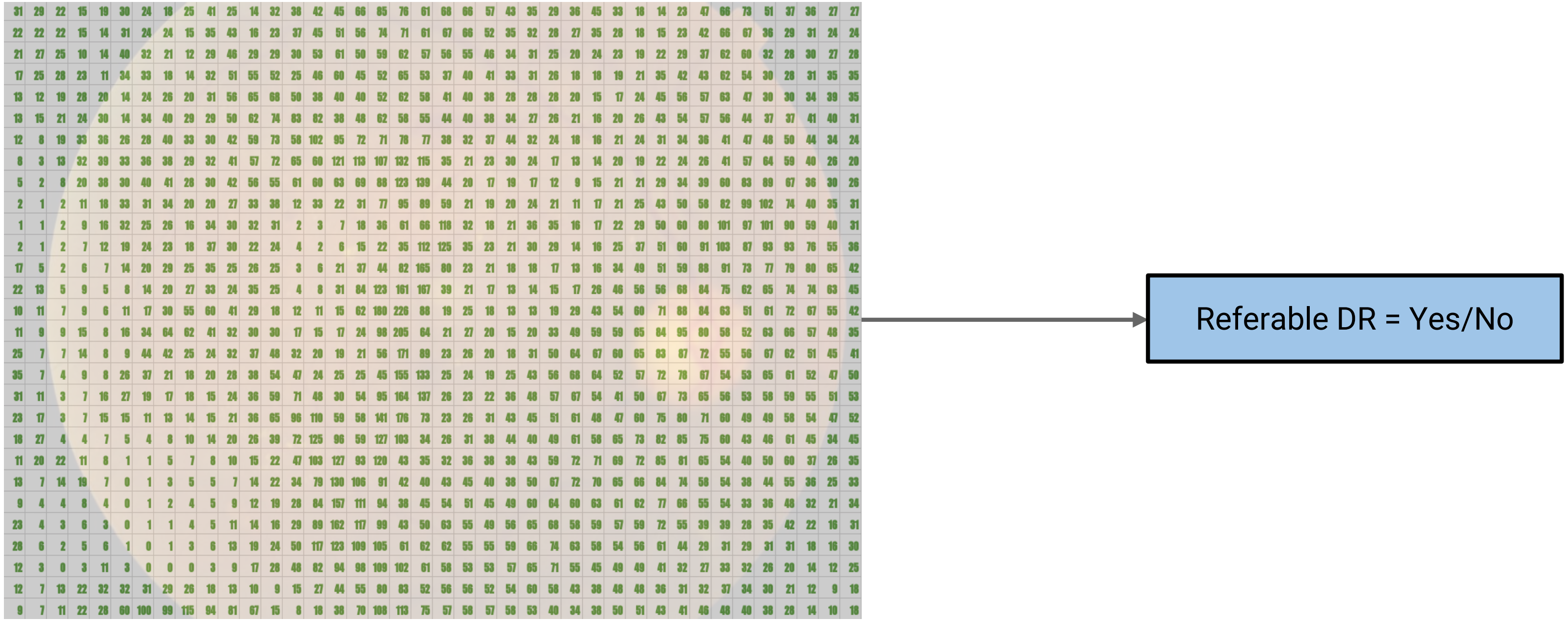
- **Overview of Neural networks (specifically Convolutional neural network) and optimization.**

- The zoo of ML models and applications

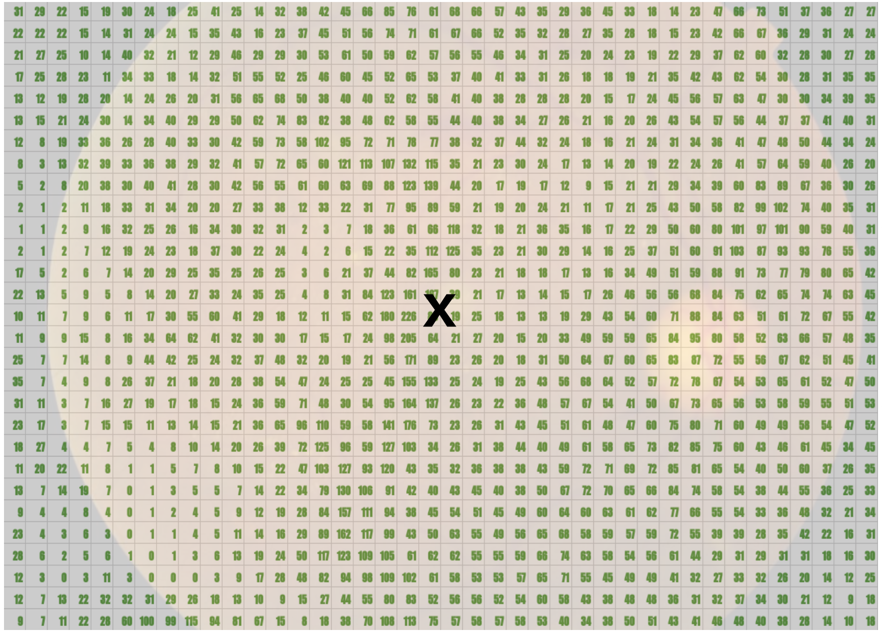
The function we want to learn

Referable DR = Yes/No

Input representation

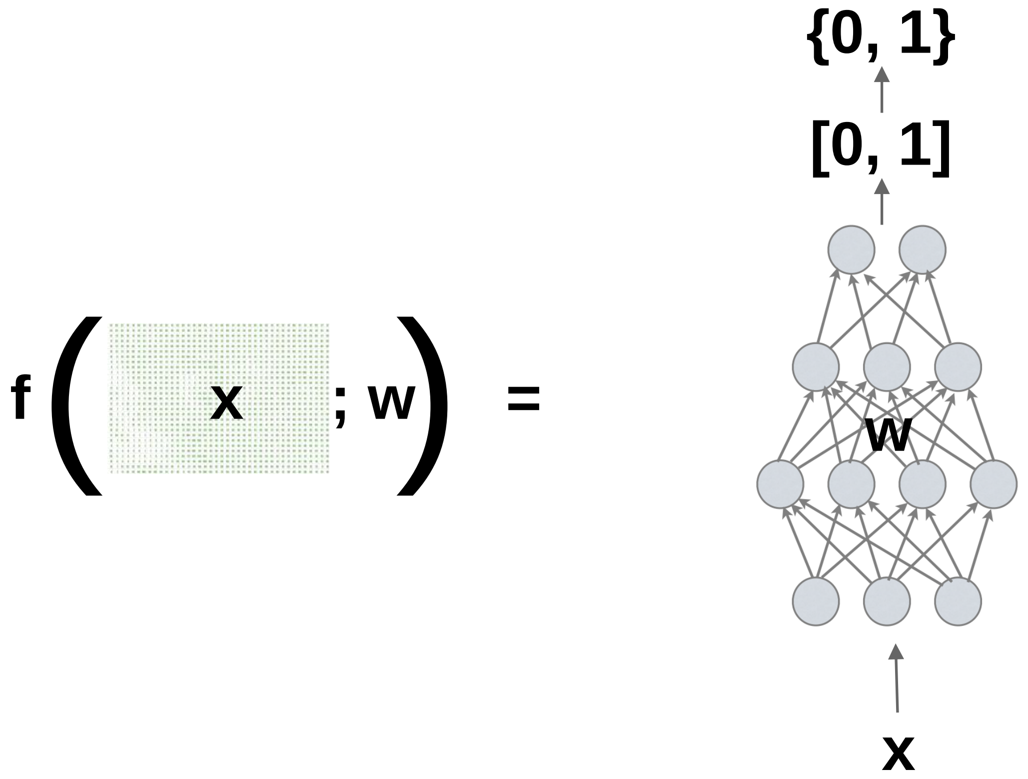


Neural networks learn this complex function



$$f(x) = \begin{cases} 0 & \text{If no referable DR} \\ 1 & \text{if referable DR} \end{cases}$$

Neural networks learn this complex function

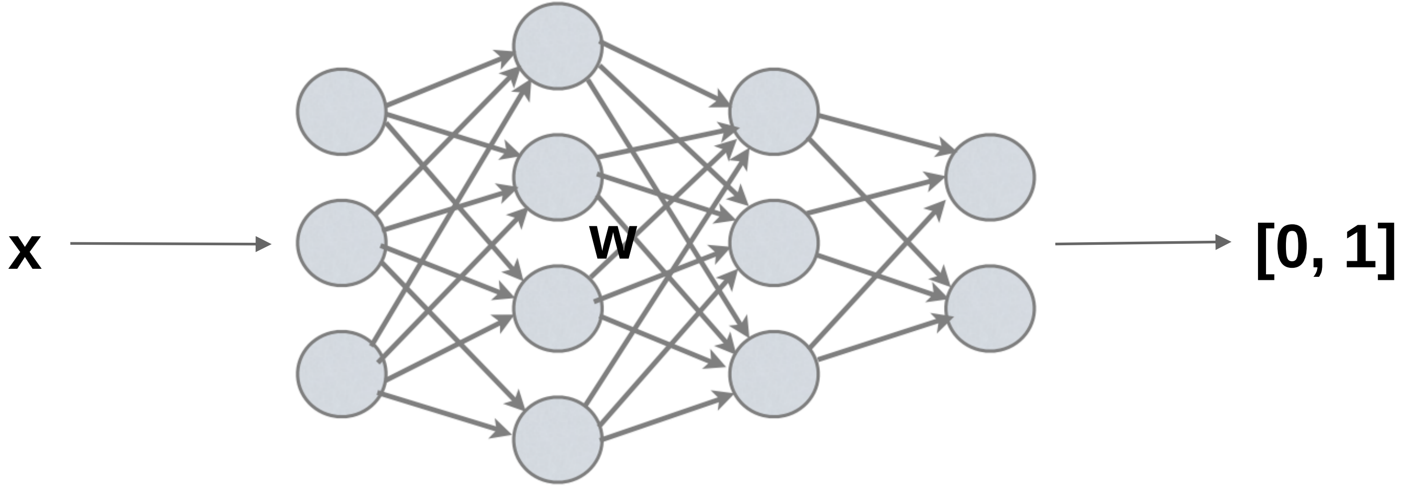


Feed raw pixels **x** into model

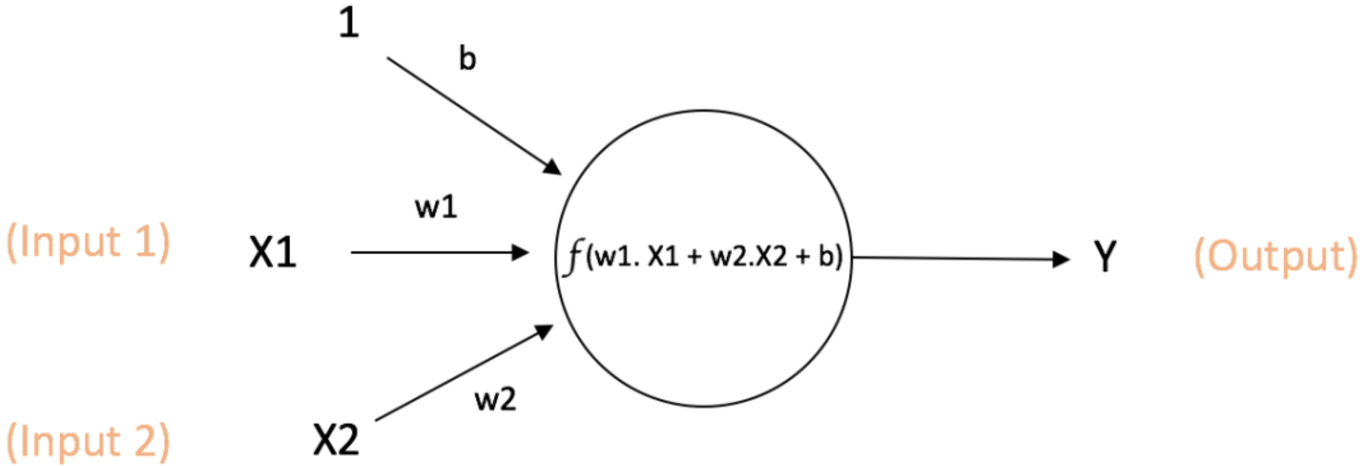
Learn good values for **w** over training data

Have to choose good network architectures

What is inside the neural network?



A single neuron



Non linear transformation
(called activation function)

Linear transformation

$$\text{Output of neuron} = Y = \underbrace{f}_{\text{Non linear transformation}}\underbrace{(w1.\,X1 + w2.X2 + b)}_{\text{Linear transformation}}$$

Image credit: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

Some commonly used activation functions

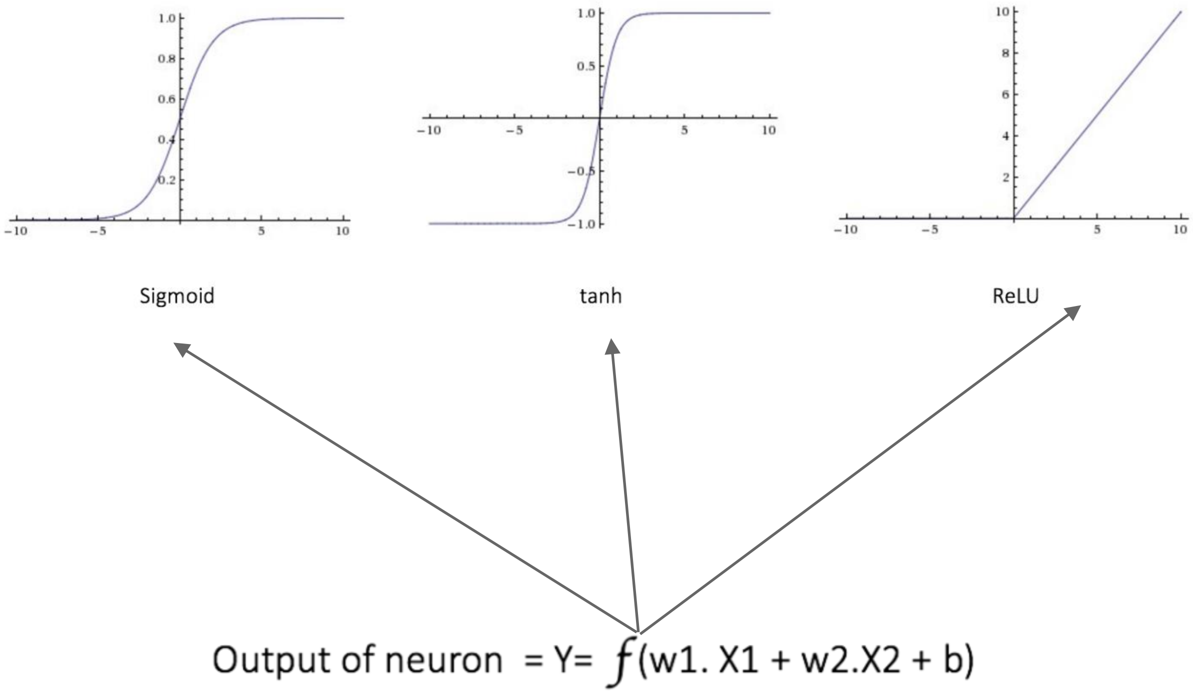


Image credit: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

A single neuron

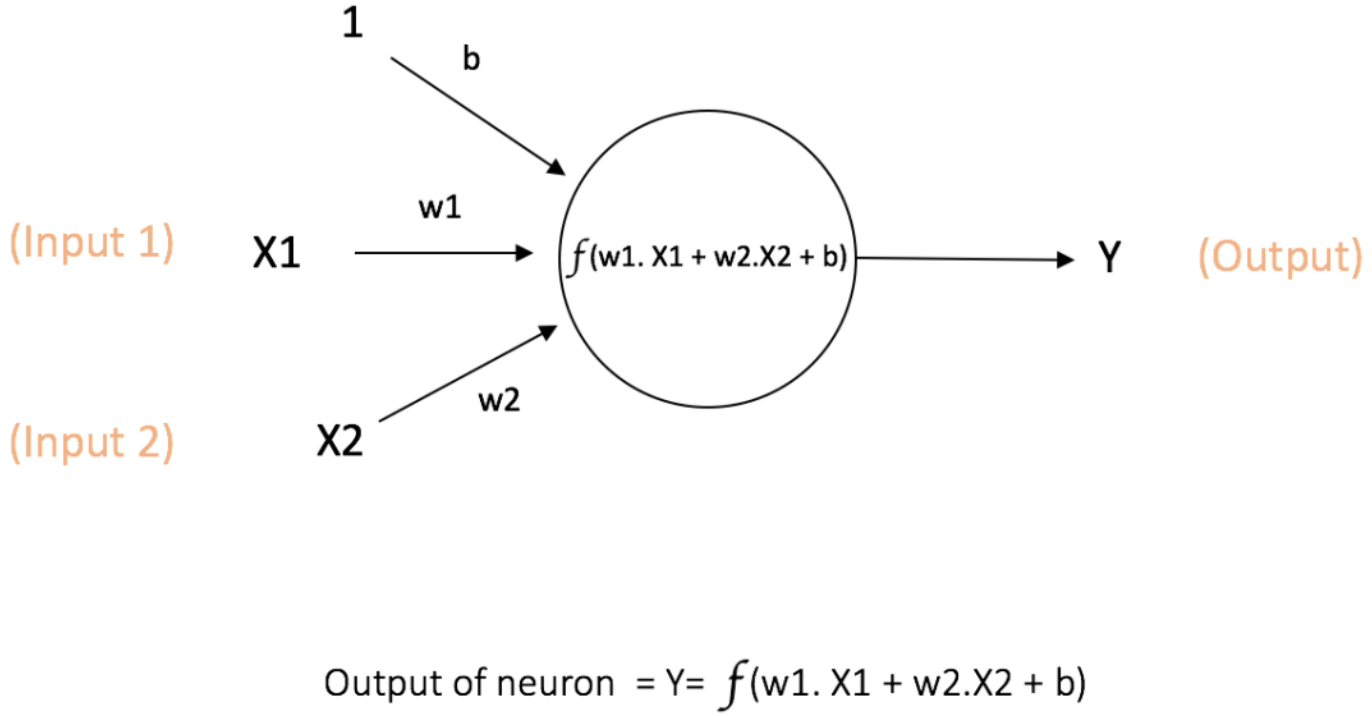


Image credit: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

Composing neurons to represent more complex functions

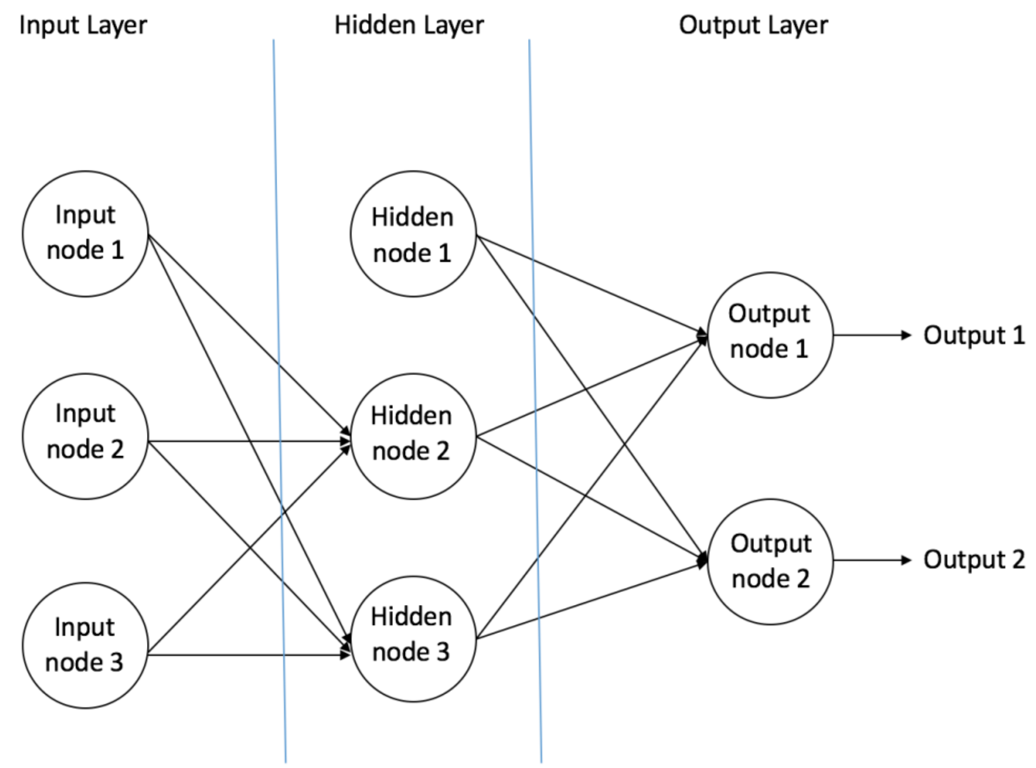
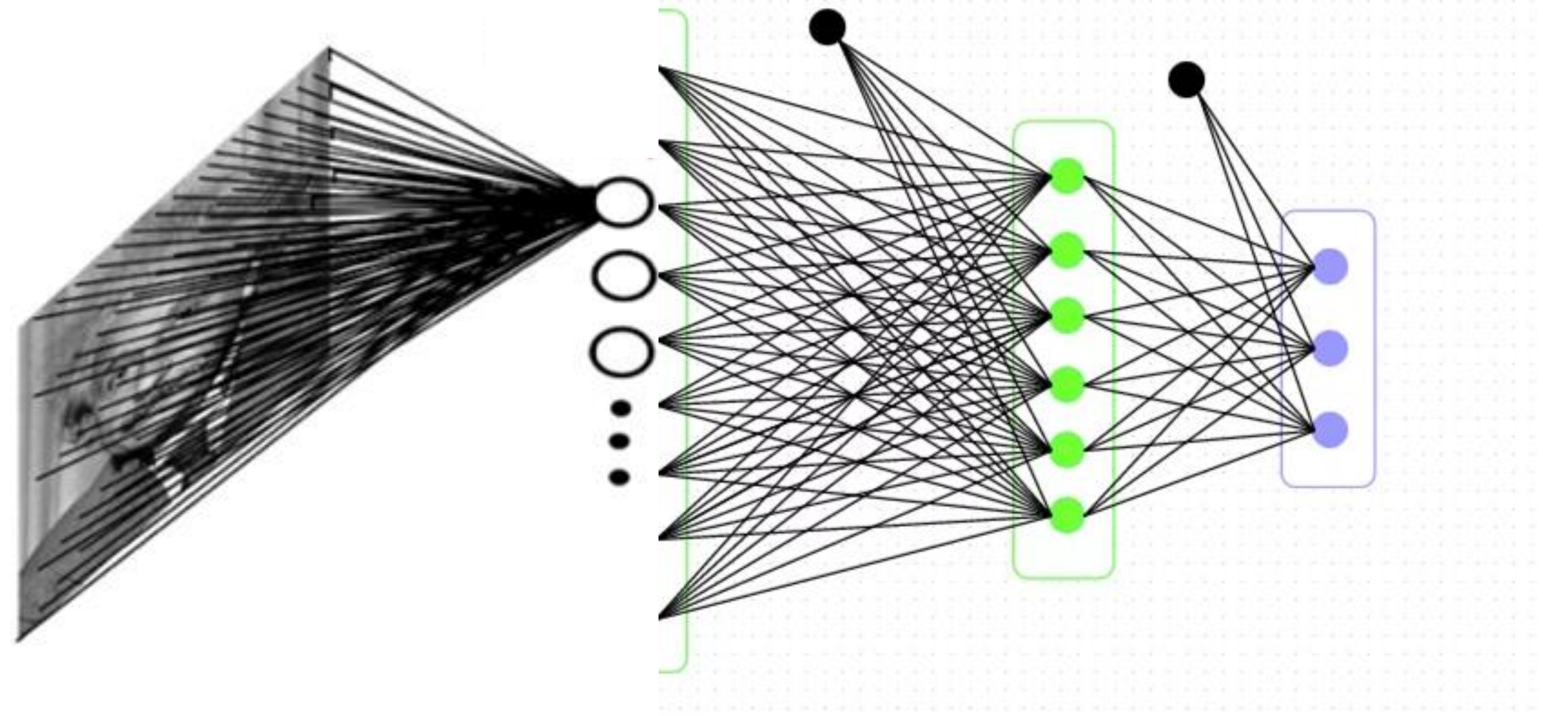


Image credit: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

Fully connected network on the image



A fully connected layer

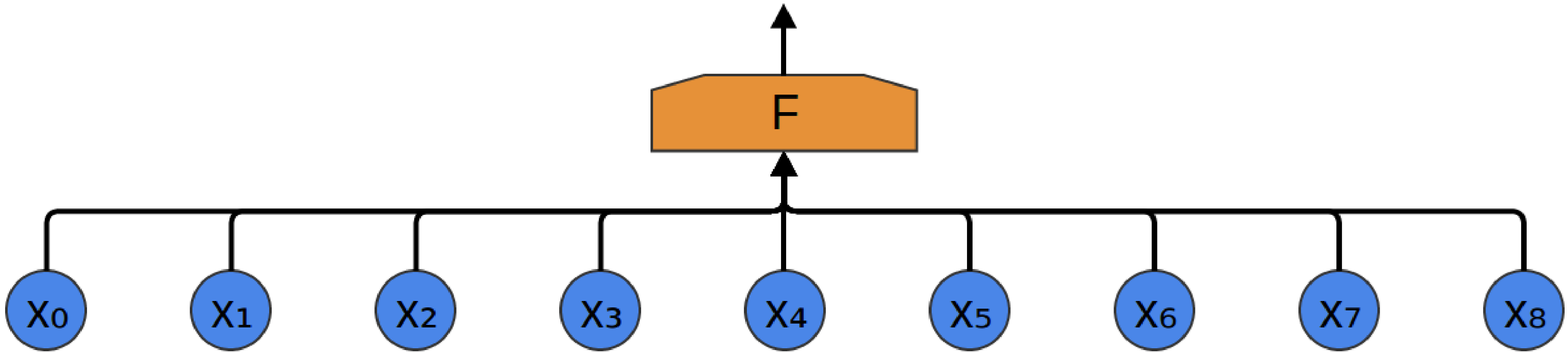


Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

A convolutional layer

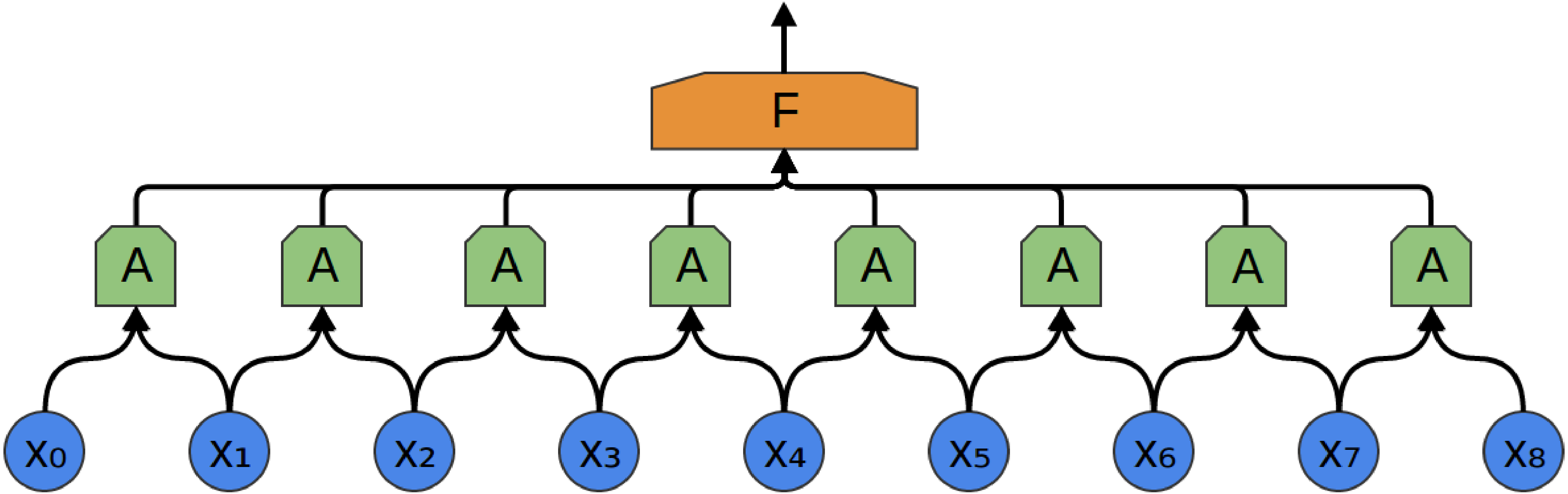


Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

Repeat this across layers

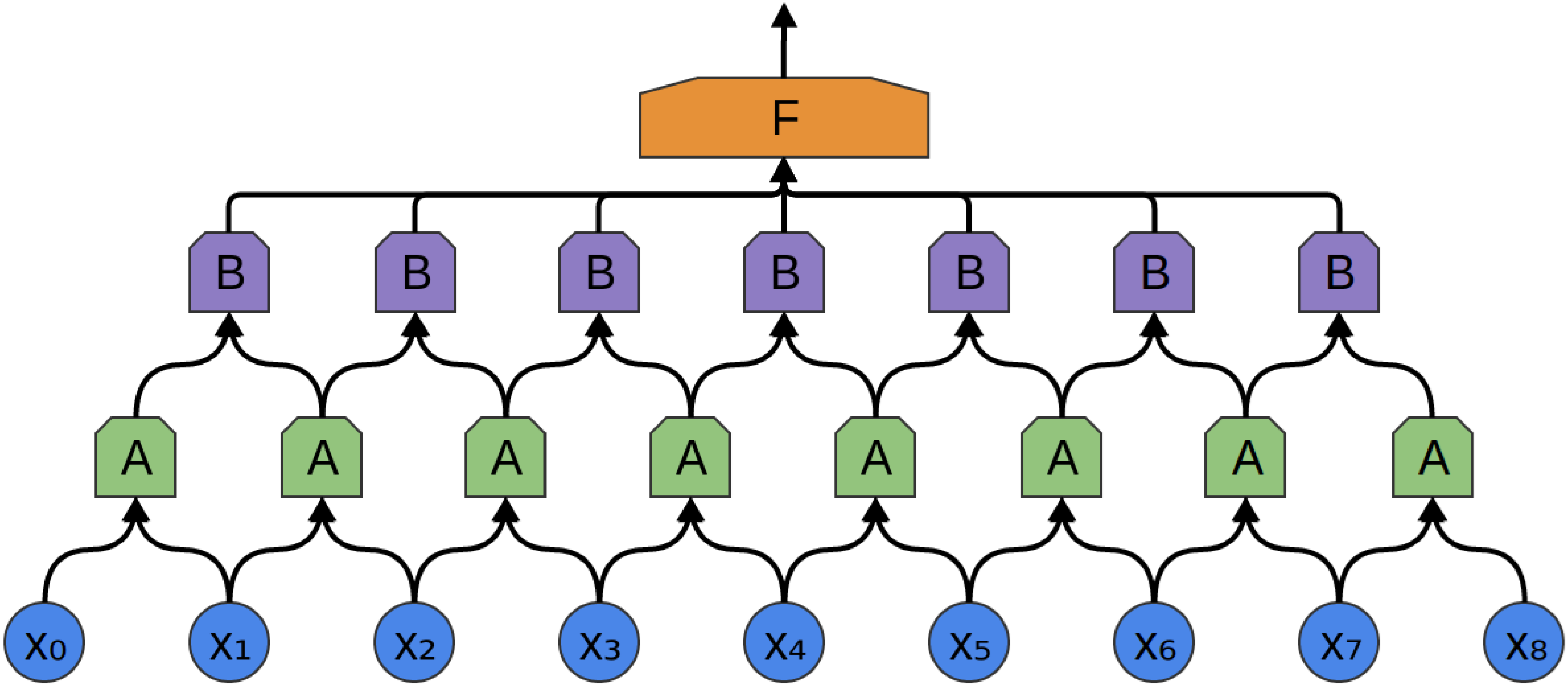


Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

Reduce dimensions by pooling (like a zoom out)

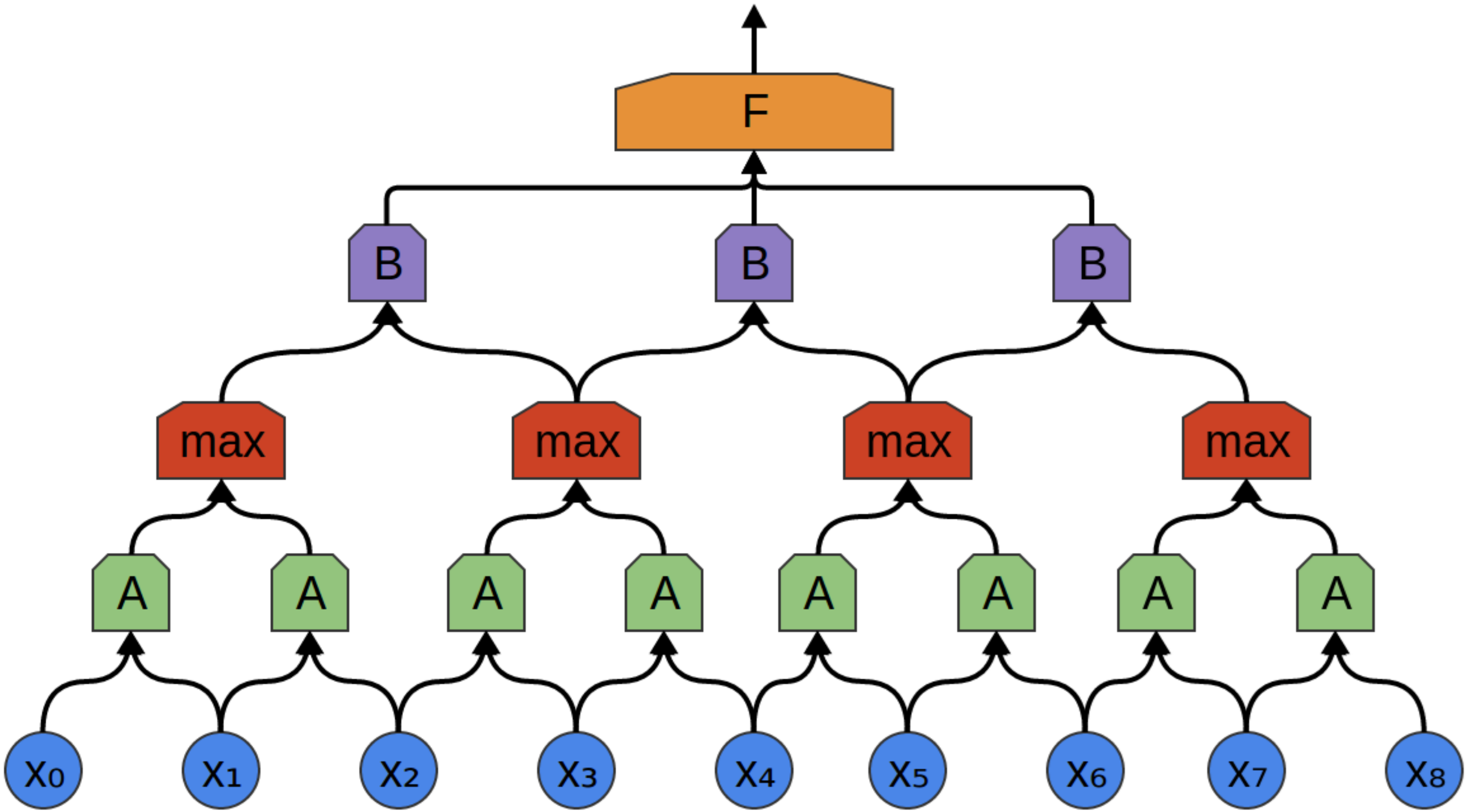


Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

2D version of the visualization

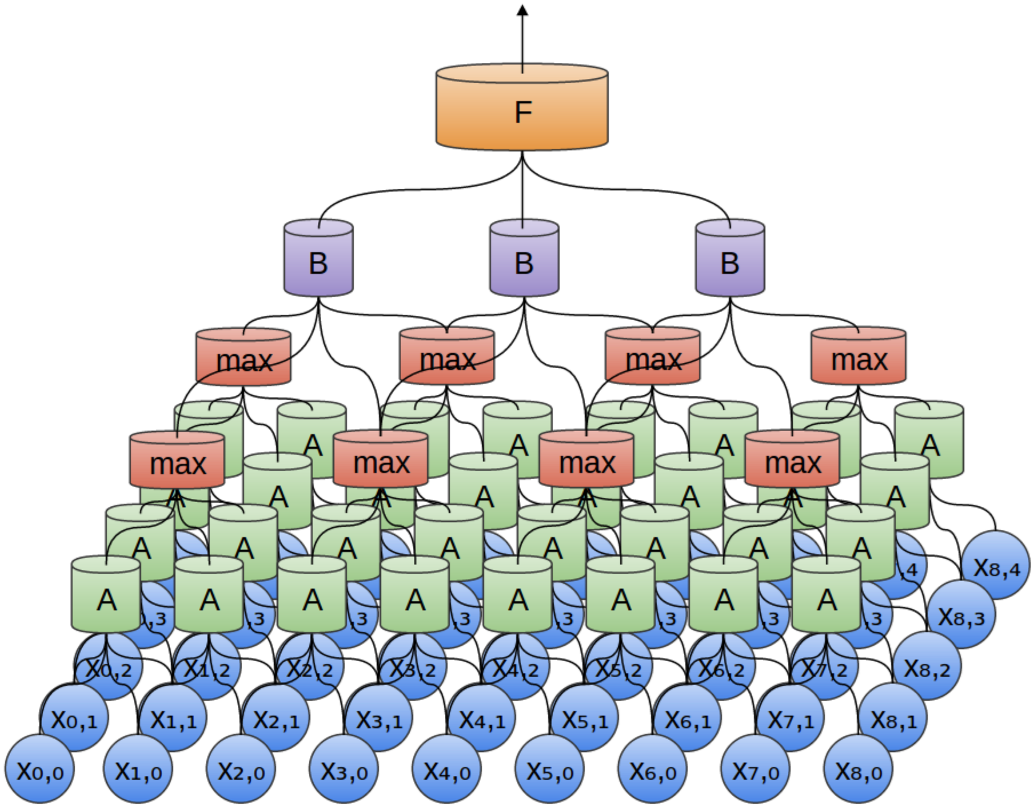
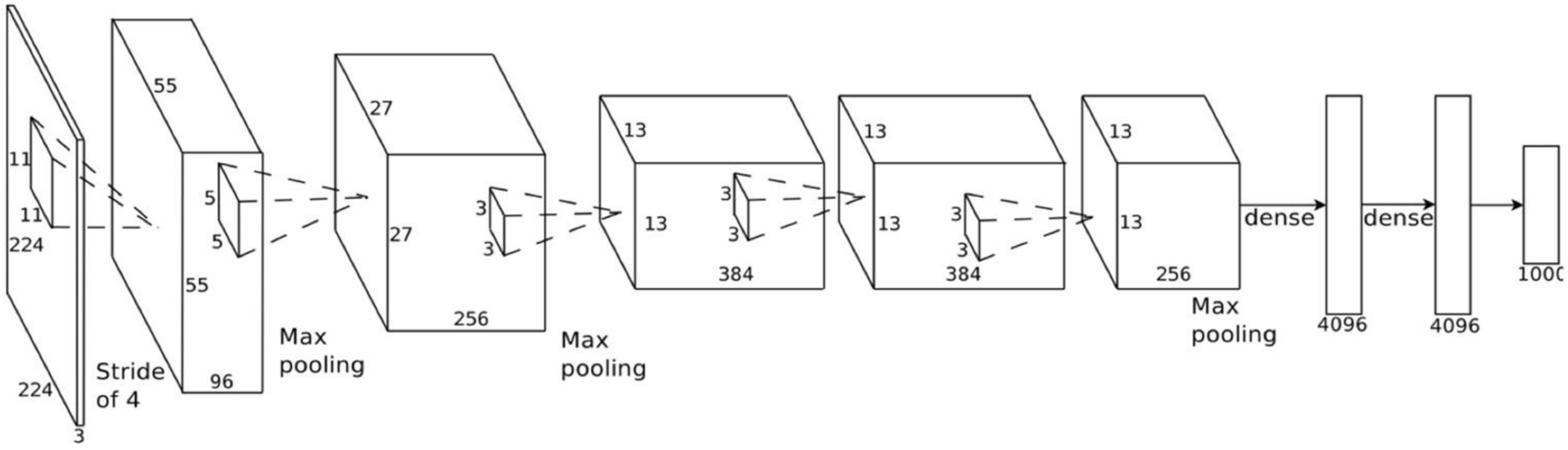
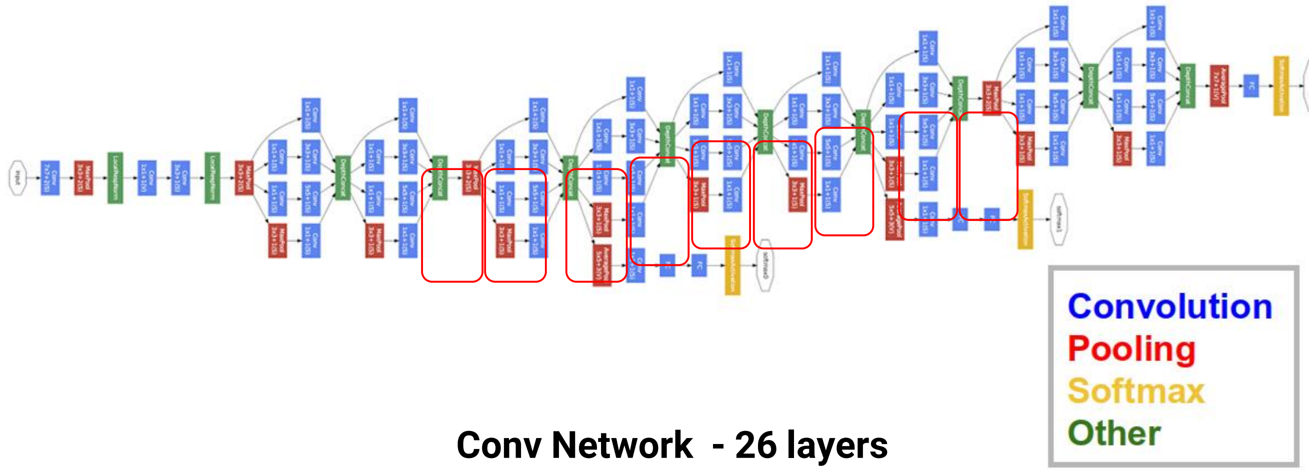


Image credit: http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

A more abstract version of CNN

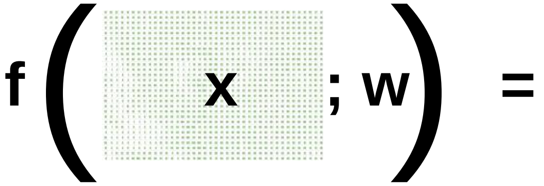


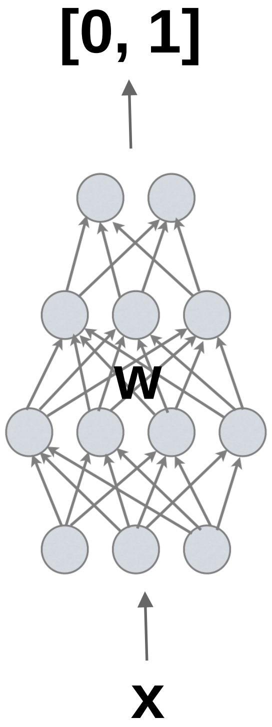
Even more abstract version of a CNN



GoogLeNet

Back to the original abstraction





Loss functions for training the network

The parameters w are optimized to minimize a loss:

$$w_{\text{opt}} = \arg\min_{w} \sum_{i=1}^{N} L(f(x_i, w), y_i)$$

Index over training data set

Input representation of i^{th} training data point

Desired output for i^{th} training point

Losses commonly used

Cross entropy loss (used in classification)

$$L(f(x_i, w), y_i) = -(y_i \log(f(x_i, w)) + (1 - y_i) \log(1 - f(x_i, w))$$

These are either 0 or 1 (indicator variables denoting presence/absence of membership to a particular class)

L2 Loss (used in regression, when the predicted value is a continuous variable)

$$L(f(x_i, w), y_i) = \|y_i - f(x_i, w)\|^2$$

The optimization problem

$$L(w) = \min_w \sum_{i=1}^{N} L(f(x_i, w), y_i)$$

Use your favorite optimizer (Nesterov gradient, LBFGS, so on) to find the best w.

In practice N is very large, and x_i has very high dimensionality: These computational constraints restrict us to using Stochastic Gradient descent, which can be run in a distributed manner across several machines.

See: https://research.google.com/pubs/pub40565.html

Tensorflow abstracts out this complexity for you

TensorFlow

http://tensorflow.org/

and

https://github.com/tensorflow/tensorflow

Open, standard software for general machine learning

Great for Deep Learning in particular

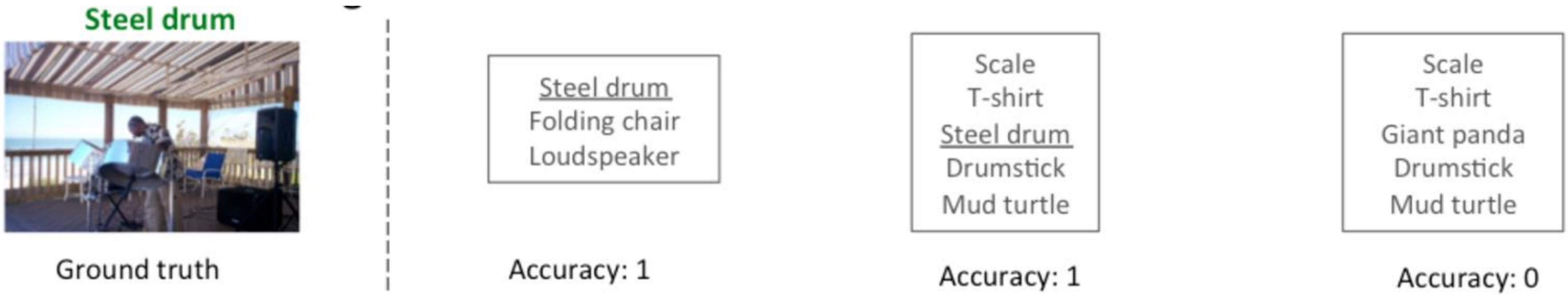
First released Nov 2015

Apache 2.0 license

Outline

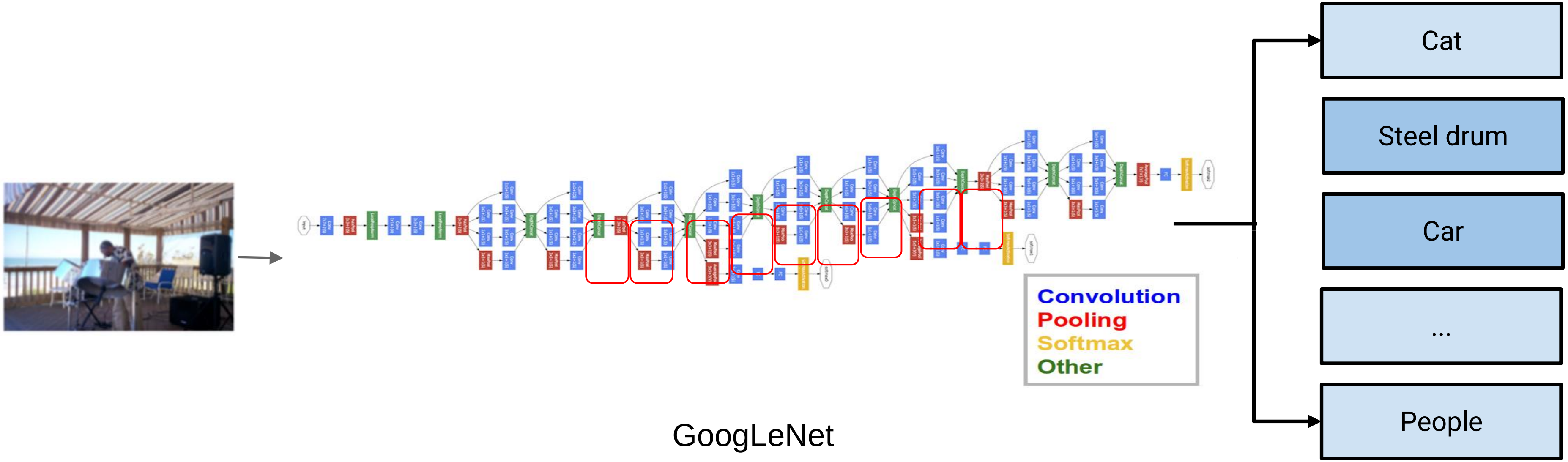
- Deep dive into a specific Medical imaging problem (Diabetic retinopathy screening)
- Overview of Neural networks (specifically Convolutional neural network) and optimization.
- **The zoo of ML models and applications**

ML models and applications: Image classification



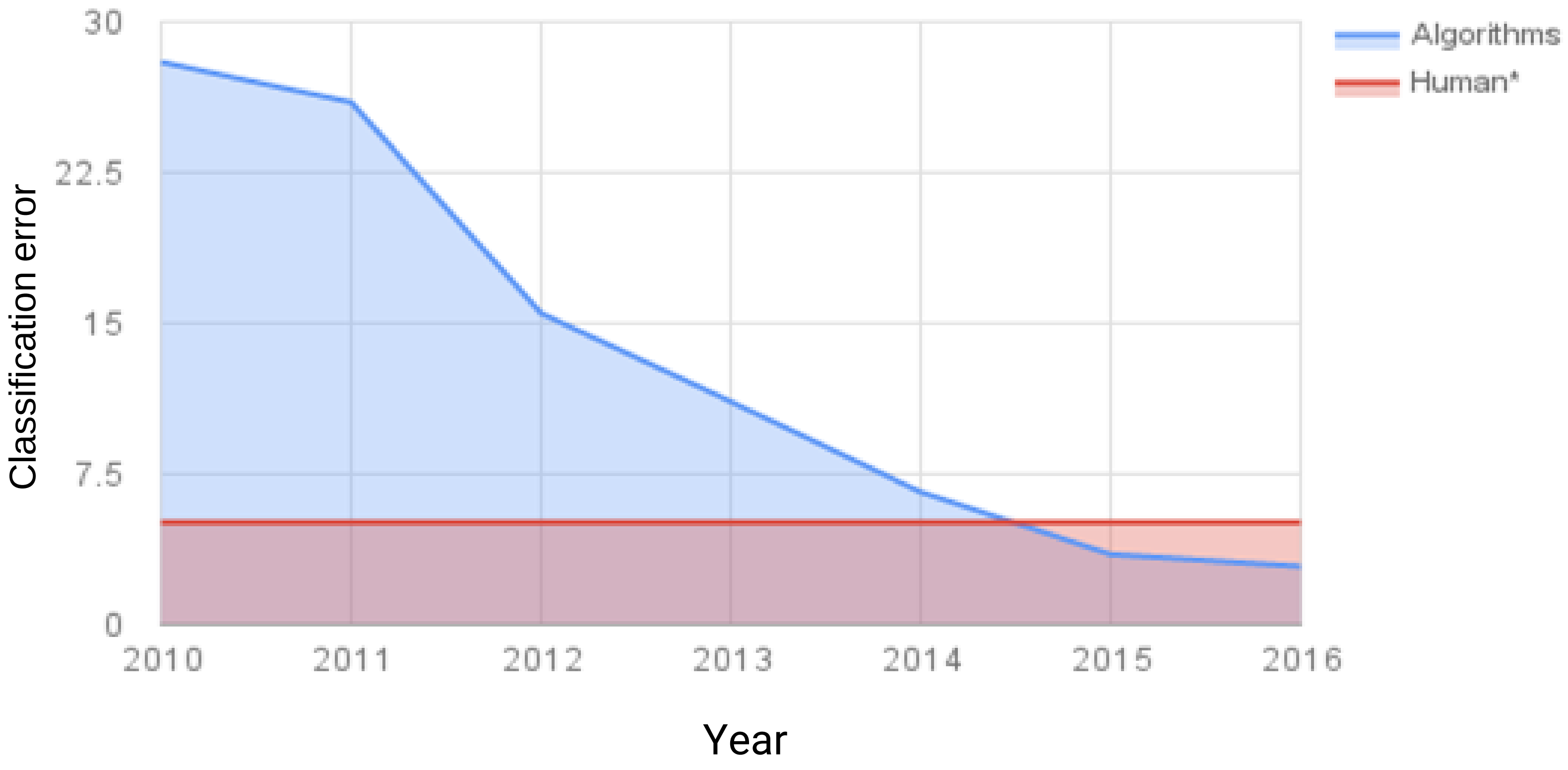
ImageNet challenge: https://arxiv.org/pdf/1409.0575.pdf

Image classification models



GoogLeNet

Image classification performance



* Human Performance based on analysis done by Andrej Karpathy. More details here.

ML models and applications: Object detection

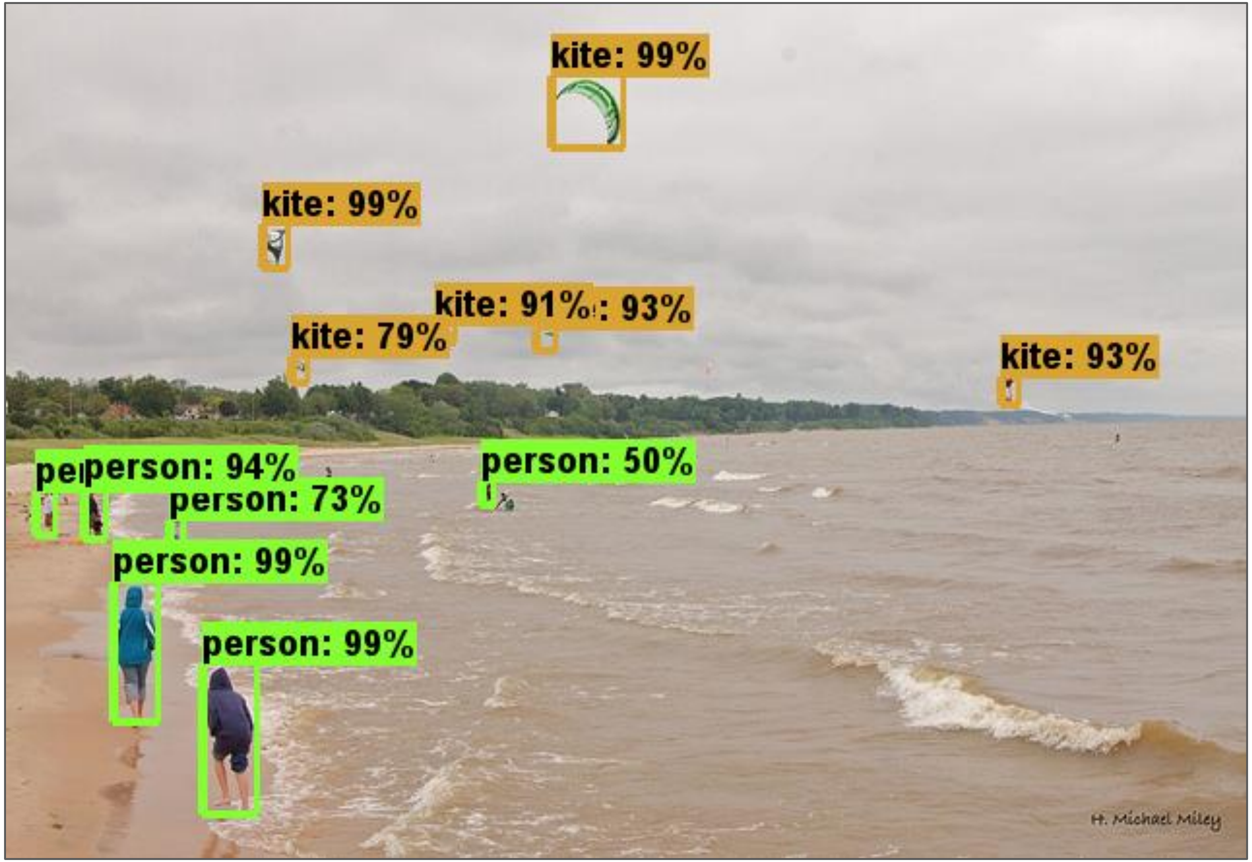


Image credit: https://arxiv.org/pdf/1611.10012.pdf

Object detection: Model architecture

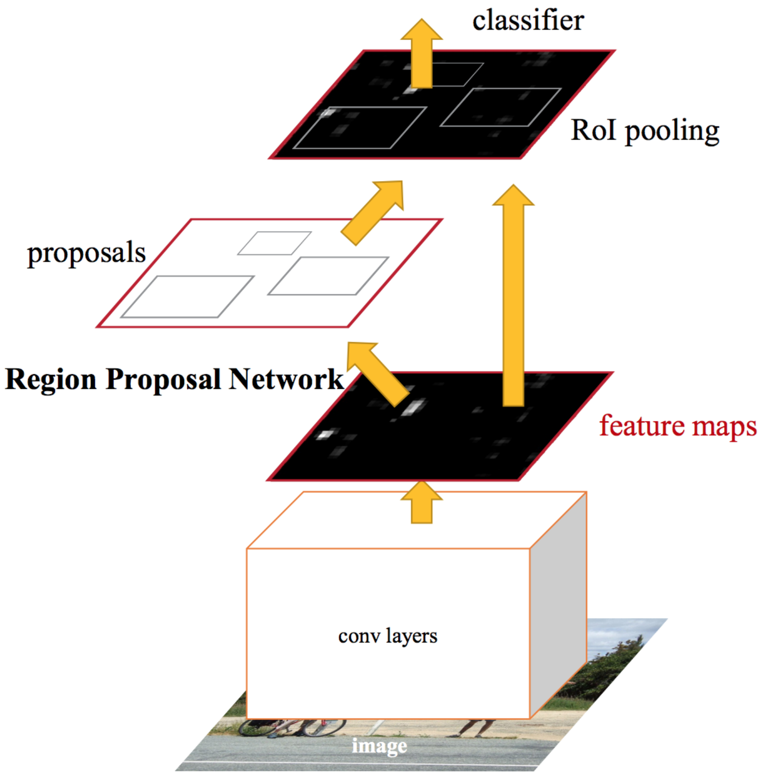


Image credit:https://arxiv.org/pdf/1506.01497.pdf

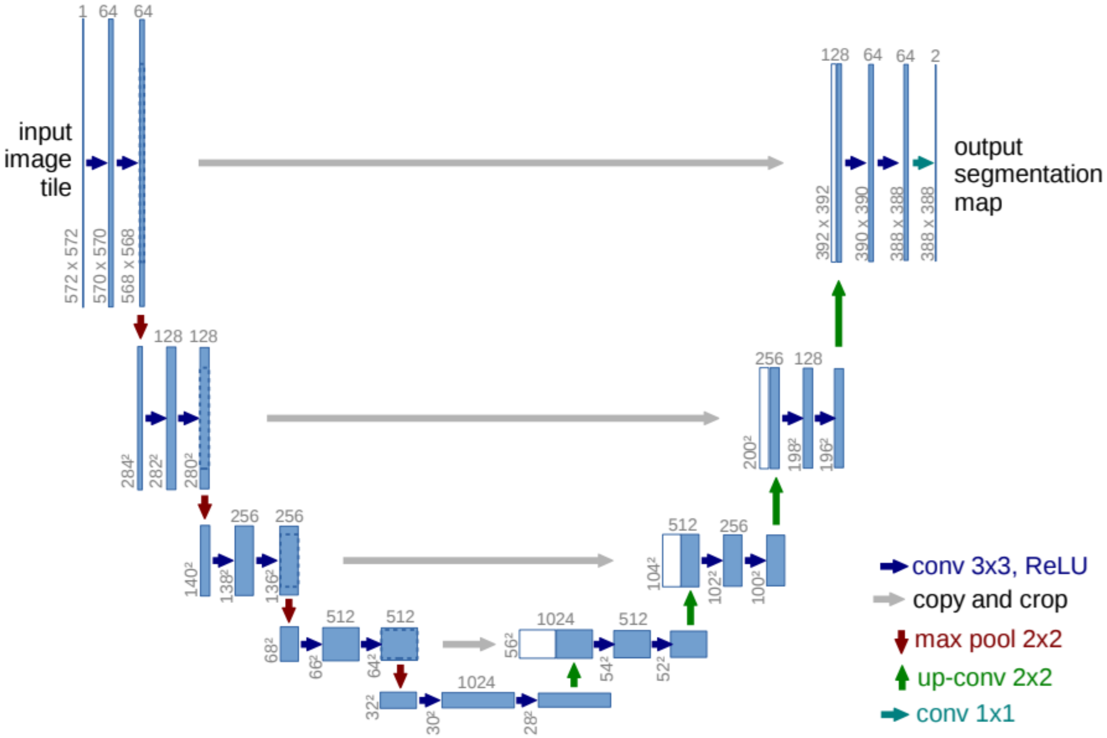
Object detection: Model performance

TODO: Get PR Curve and plot operating point

ML models and applications: Image segmentation

Image from: http://cocodataset.org/#detections-challenge2017

ML models and applications: Image segmentation



https://arxiv.org/pdf/1505.04597.pdf

Image segmentation performance

Copy to Clipboard | Export to CSV

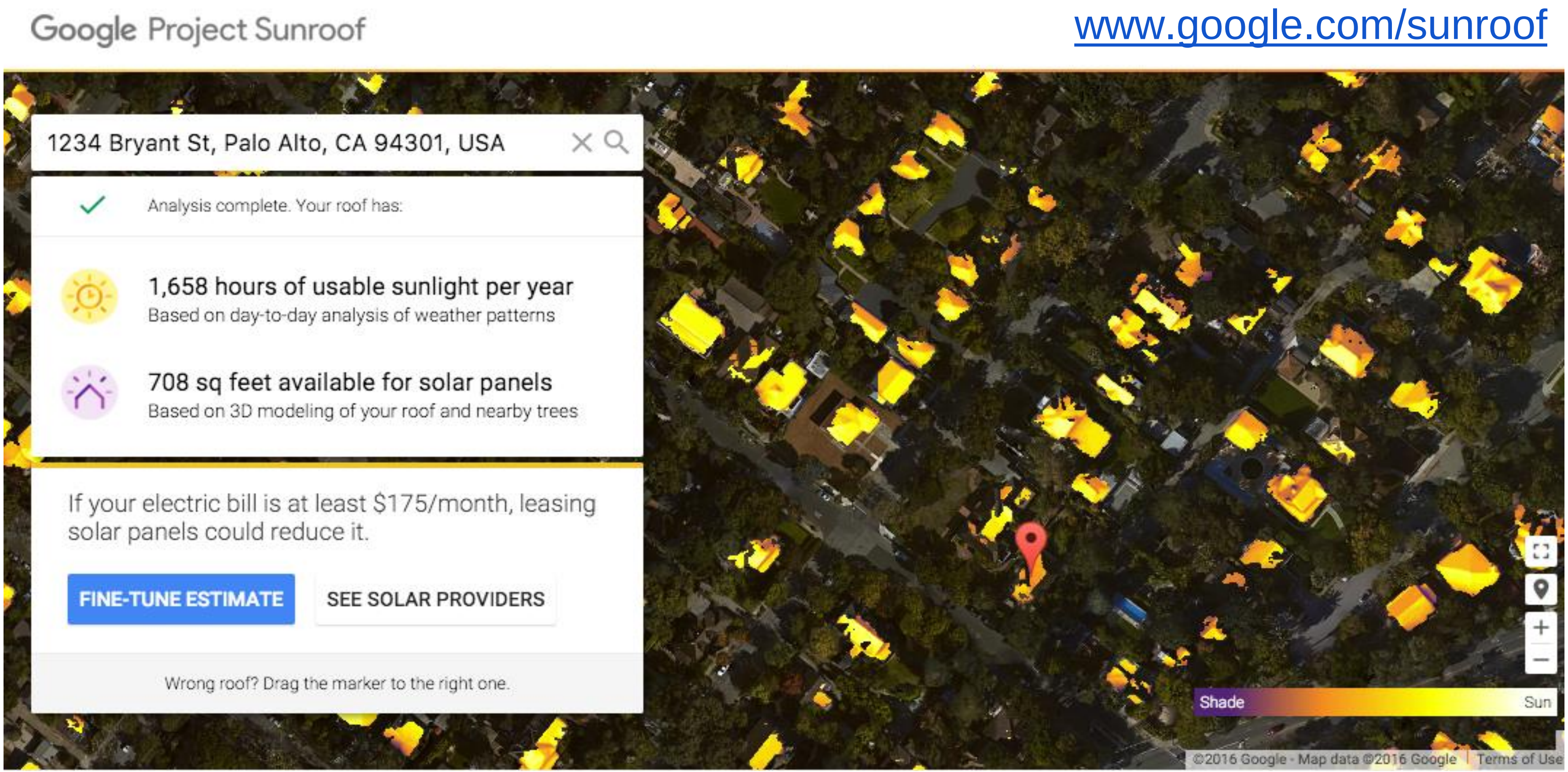
	mIoU	fIoU	mAcc	pAcc
ResNeXt-FPN	0.287	0.560	0.423	0.694
G-RMI	0.264	0.522	0.403	0.656
Oxford Active Vision Lab	0.232	0.505	0.339	0.653
Deeplab VGG-16	0.200	0.479	0.280	0.649
Vllab	0.124	0.394	0.176	0.581

Show 25 entries

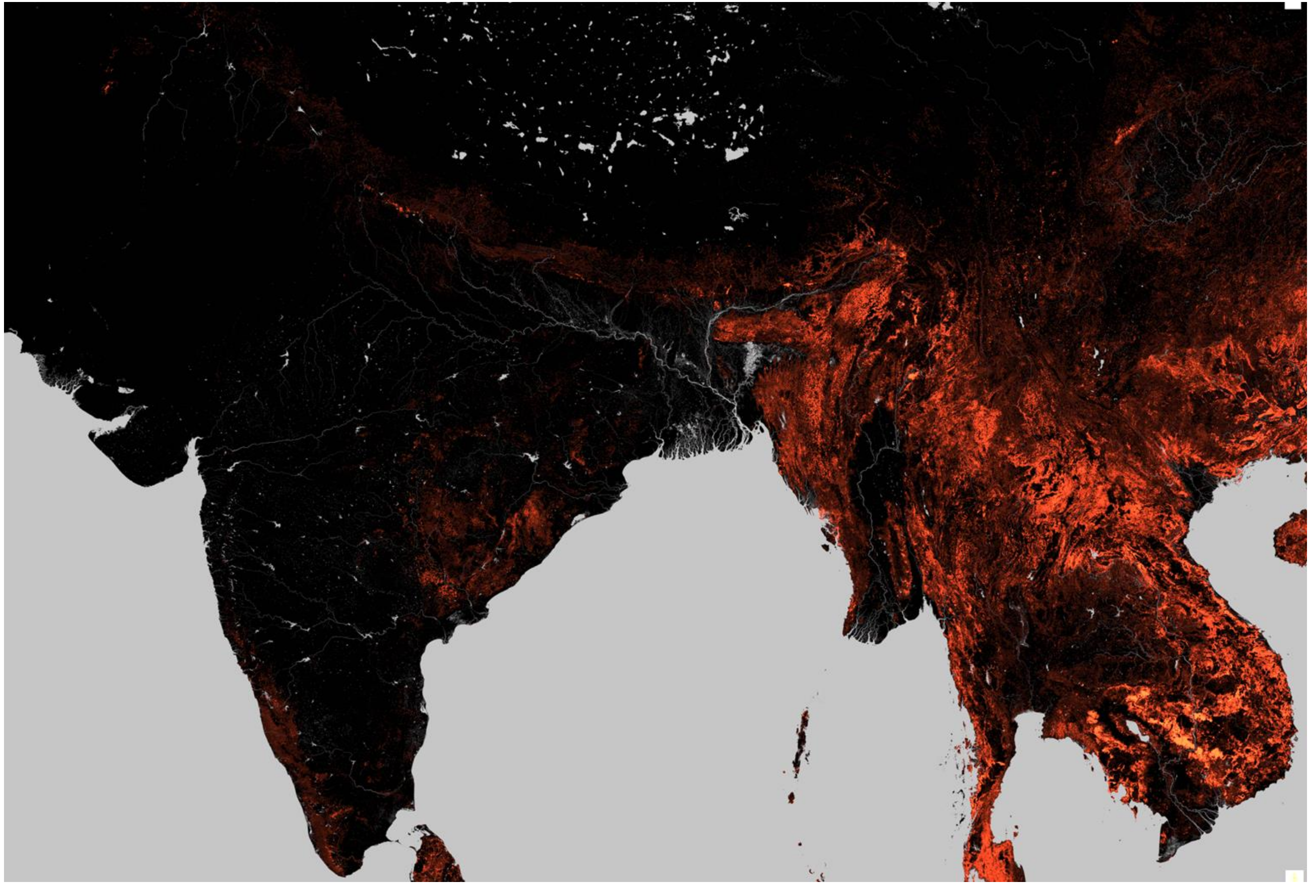
http://cocodataset.org/#stuff-leaderboard

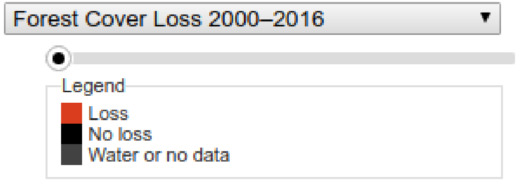
ML models and applications: Aerial image analysis

www.google.com/sunroof



ML models and applications: Satellite image analysis





Screenshot from live tool at: http://earthenginepartners.appspot.com/science-2013-global-forest

ML models and applications: Translation

“Hello, how are you?” → “Bonjour, comment allez-vous?”

Translation model: LSTMs

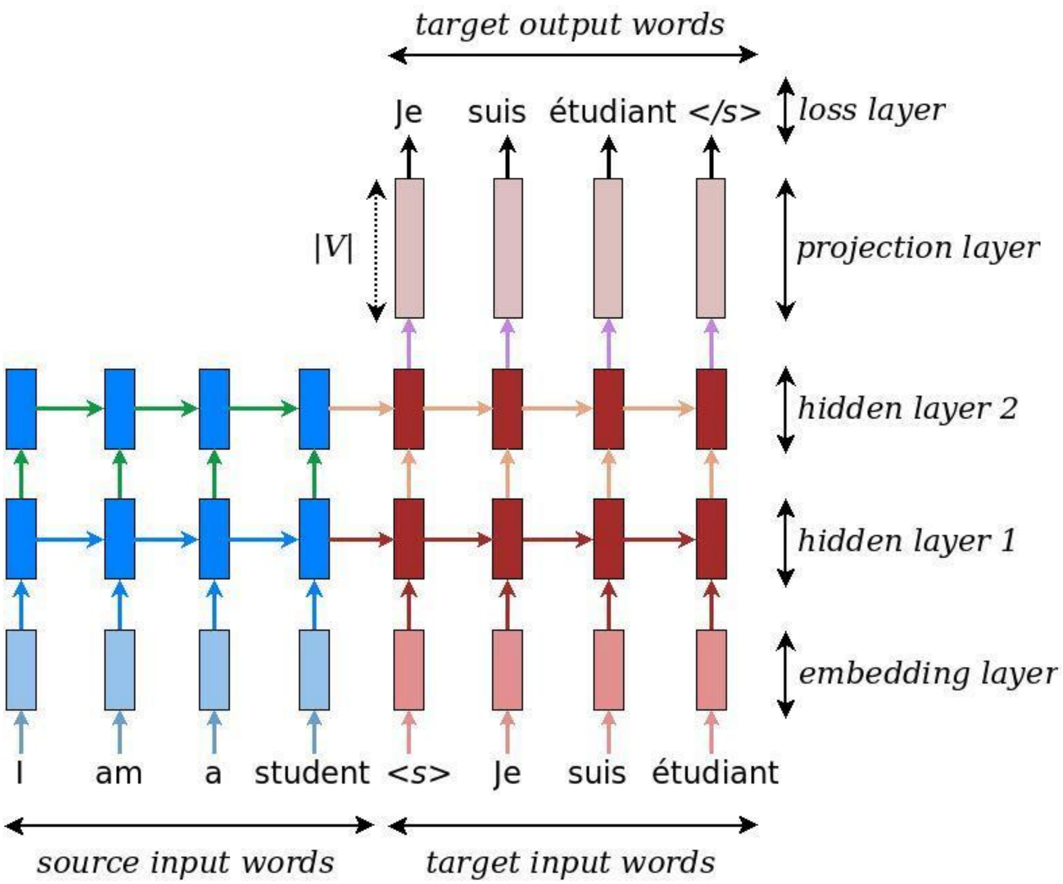
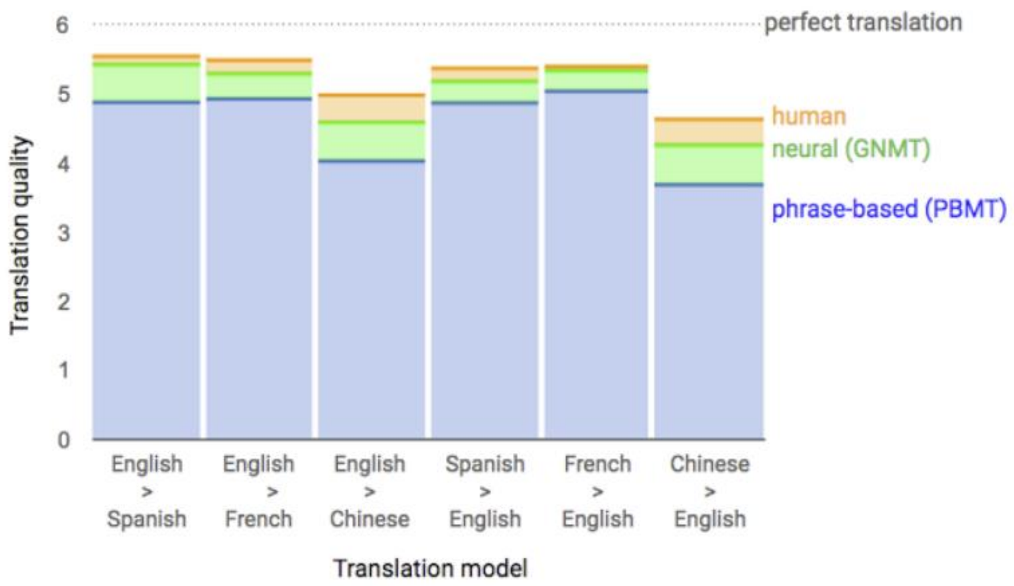


Image source:
https://www.tensorflow.org/tutorials/seq2seq

Other Useful resources:
http://colah.github.io/posts/2014-07-NLP-RNNs-Representations/
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Translation model performance



Data from side-by-side evaluations, where human raters compare the quality of translations for a given source sentence. Scores range from 0 to 6, with 0 meaning "completely nonsense translation", and 6 meaning "perfect translation."

Source: https://research.googleblog.com/2016/09/a-neural-network-for-machine.html

ML models and applications: Predicting molecule properties

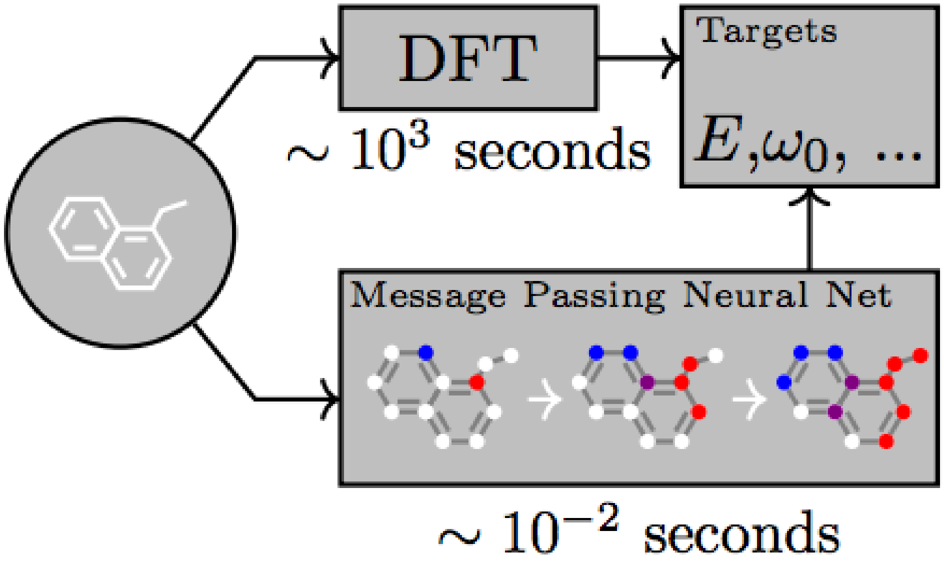


Figure 1. A Message Passing Neural Network predicts quantum properties of an organic molecule by modeling a computationally expensive DFT calculation.

https://research.googleblog.com/2017/04/predicting-properties-of-molecules-with.html

Predicting molecule properties: Performance

		μ	α	$\varepsilon_{\mathrm{HOMO}}$	$\varepsilon_{\mathrm{LUMO}}$	$\Delta\varepsilon$	$\langle R^2 \rangle$	ZPVE	U_0	C_v	ω_1	NMMAE
		Debye	Bohr^3	eV	eV	eV	Bohr^2	eV	eV	cal/molK	cm^{-1}	arb. u.
Mean		2.67	75.3	-6.54	0.322	6.86	1190	4.06	-76.6	31.6	3500	
MAD		1.17	6.29	0.439	1.05	1.07	203	0.717	8.19	3.21	238	
Target[a]		0.10	0.10	0.043	0.043	0.043	1.2	0.0012	0.043	0.050	10	
DFT[b]		0.09	0.74	2.0	2.6	1.19	-	0.0097	0.10	0.34	28	
EN	CM	0.844	1.33	0.338	0.631	0.722	55.5	0.0265	0.911	0.906	131	0.423
	BOB	0.763	1.20	0.283	0.521	0.614	55.3	0.0232	0.602	0.700	81.4	0.35
	BAML	0.686	0.793	0.186	0.275	0.339	32.6	0.0129	0.212	0.439	60.4	0.231
	ECFP4	0.737	3.45	0.224	0.344	0.383	118	0.270	3.68	1.51	86.6	0.462
	HDAD	0.563	0.437	0.139	0.238	0.278	6.19	0.00647	0.0983	0.0876	94.2	0.183
	HD	0.705	0.638	0.203	0.299	0.360	6.70	0.00949	0.192	0.195	104	0.236
	MARAD	0.707	0.698	0.222	0.305	0.391	27.4	0.00808	0.183	0.206	108	0.256
	Mean	0.715	1.22	0.228	0.373	0.441	43.1	0.0509	0.840	0.578	95.1	
BR	CM	0.844	1.33	0.338	0.632	0.723	55.5	0.0265	0.911	0.907	131	0.424
	BOB	0.761	1.14	0.279	0.521	0.614	48.0	0.0222	0.586	0.684	80.9	0.343
	BAML	0.685	0.785	0.183	0.275	0.339	30.4	0.0129	0.202	0.444	60.4	0.229
	ECFP4	0.737	3.45	0.224	0.344	0.383	118.0	0.270	3.69	1.51	86.7	0.462
	HDAD	0.565	0.43	0.14	0.238	0.278	5.94	0.00318	0.0614	0.0787	94.8	0.182
	HD	0.705	0.633	0.203	0.298	0.359	6.8	0.00693	0.171	0.19	104	0.235
	MARAD	0.647	0.533	0.18	0.257	0.315	26.8	0.00854	0.171	0.201	103	0.226
	Mean	0.706	1.19	0.221	0.367	0.430	41.7	0.0500	0.828	0.574	94.5	
RF	CM	0.608	1.04	0.208	0.302	0.373	45.0	0.0199	0.431	0.777	13.2	0.239
	BOB	0.450	0.623	0.120	0.137	0.164	39.0	0.0111	0.202	0.443	3.55	0.142
	BAML	0.434	0.638	0.107	0.118	0.141	51.1	0.0132	0.200	0.451	**2.71**	0.141
	ECFP4	0.483	3.70	0.143	0.145	0.166	109	0.242	3.66	1.57	14.7	0.349
	HDAD	0.454	1.71	0.116	0.136	0.156	48.3	0.0525	1.44	0.895	3.45	0.198
	HD	0.457	1.66	0.126	0.139	0.150	46.8	0.0497	1.39	0.879	4.18	0.197
	MARAD	0.607	0.676	0.178	0.243	0.311	45.3	0.0102	0.21	0.311	19.4	0.199
	Mean	0.499	1.43	0.142	0.174	0.209	54.9	0.0569	1.08	0.761	8.74	
KRR	CM	0.449	0.433	0.133	0.183	0.229	3.39	0.0048	0.128	0.118	33.5	0.136
	BOB	0.423	0.298	0.0948	0.122	0.148	**0.978**	0.00364	0.0667	0.0917	13.2	0.0981
	BAML	0.460	0.301	0.0946	0.121	0.152	3.9	0.00331	0.0519	0.082	19.9	0.105
	ECFP4	0.490	4.17	0.124	0.133	0.174	128	0.248	4.25	1.84	26.7	0.383
	HDAD	0.334	0.175	0.0662	0.0842	0.107	1.62	**0.00191**	**0.0251**	**0.0441**	23.1	0.0768
	HD	0.364	0.299	0.0874	0.113	0.143	1.72	0.00316	0.0644	0.0844	21.3	0.0935
	MARAD	0.468	0.343	0.103	0.124	0.163	7.58	0.00301	0.0529	0.0758	21.3	0.112
	Mean	0.427	0.859	0.101	0.126	0.159	21.1	0.0383	0.662	0.333	22.7	
GG	MG	0.238	**0.151**	0.0587	0.0564	**0.0835**	5.98	0.00291	0.0317	0.0724	6.32	0.058
GC	MG	**0.0696**	0.227	**0.0509**	**0.0471**	**0.0766**	5.68	0.00975	0.13	0.0892	3.15	0.0427

https://arxiv.org/abs/1702.05532

Predicting molecule properties: Performance

		μ	α	ε_{HOMO}	ε_{LUMO}	$\Delta\varepsilon$	$\langle R^2 \rangle$	ZPVE	U_0	C_v	ω_1	NMMAE
		Debye	Bohr^3	eV	eV	eV	Bohr^2	eV	eV	cal/molK	cm^{-1}	arb. u.
Mean		2.67	75.3	-6.54	0.322	6.86	1190	4.06	-76.6	31.6	3500	
MAD		1.17	6.29	0.439	1.05	1.07	203	0.717	8.19	3.21	238	
Target[a]		0.10	0.10	0.043	0.043	0.043	1.2	0.0012	0.043	0.050	10	
DFT[b]		0.09	0.74	2.0	2.6	1.19	-	0.0097	0.10	0.34	28	
EN	CM	0.844	1.33	0.338	0.631	0.722	55.5	0.0265	0.911	0.906	131	0.423
	BOB	0.763	1.20	0.283	0.521	0.614	55.3	0.0232	0.602	0.700	81.4	0.35
	BAML	0.686	0.793	0.186	0.275	0.339	32.6	0.0129	0.212	0.439	60.4	0.231
	ECFP4	0.737	3.45	0.224	0.344	0.383	118	0.270	3.68	1.51	86.6	0.462
	HDAD	0.563	0.437	0.139	0.238	0.278	6.19	0.00647	0.0983	0.0876	94.2	0.183
	HD	0.705	0.638	0.203	0.299	0.360	6.70	0.00949	0.192	0.195	104	0.236
	MARAD	0.707	0.698	0.222	0.305	0.391	27.4	0.00808	0.183	0.206	108	0.256
	Mean	0.715	1.22	0.228	0.373	0.441	43.1	0.0509	0.840	0.578	95.1	
BR	CM	0.844	1.33	0.338	0.632	0.723	55.5	0.0265	0.911	0.907	131	0.424
	BOB	0.761	1.14	0.279	0.521	0.614	48.0	0.0222	0.586	0.684	80.9	0.343
	BAML	0.685	0.785	0.183	0.275	0.339	30.4	0.0129	0.202	0.444	60.4	0.229
	ECFP4	0.737	3.45	0.224	0.344	0.383	118.0	0.270	3.69	1.51	86.7	0.462
	HDAD	0.565	0.43	0.14	0.238	0.278	5.94	0.00318	0.0614	0.0787	94.8	0.182
	HD	0.705	0.633	0.203	0.298	0.359	6.8	0.00693	0.171	0.19	104	0.235
	MARAD	0.647	0.533	0.18	0.257	0.315	26.8	0.00854	0.171	0.201	103	0.226
	Mean	0.706	1.19	0.221	0.367	0.430	41.7	0.0500	0.828	0.574	94.5	
RF	CM	0.608	1.04	0.208	0.302	0.373	45.0	0.0199	0.431	0.777	13.2	0.239
	BOB	0.450	0.623	0.120	0.137	0.164	39.0	0.0111	0.202	0.443	3.55	0.142
	BAML	0.434	0.638	0.107	0.118	0.141	51.1	0.0132	0.200	0.451	**2.71**	0.141
	ECFP4	0.483	3.70	0.143	0.145	0.166	109	0.242	3.66	1.57	14.7	0.349
	HDAD	0.454	1.71	0.116	0.136	0.156	48.3	0.0525	1.44	0.895	3.45	0.198
	HD	0.457	1.66	0.126	0.139	0.150	46.8	0.0497	1.39	0.879	4.18	0.197
	MARAD	0.607	0.676	0.178	0.243	0.311	45.3	0.0102	0.21	0.311	19.4	0.199
	Mean	0.499	1.43	0.142	0.174	0.209	54.9	0.0569	1.08	0.761	8.74	
KRR	CM	0.449	0.433	0.133	0.183	0.229	3.39	0.0048	0.128	0.118	33.5	0.136
	BOB	0.423	0.298	0.0948	0.122	0.148	**0.978**	0.00364	0.0667	0.0917	13.2	0.0981
	BAML	0.460	0.301	0.0946	0.121	0.152	3.9	0.00331	0.0519	0.082	19.9	0.105
	ECFP4	0.490	4.17	0.124	0.133	0.174	128	0.248	4.25	1.84	26.7	0.383
	HDAD	0.334	0.175	0.0662	0.0842	0.107	1.62	**0.00191**	**0.0251**	**0.0441**	23.1	0.0768
	HD	0.364	0.299	0.0874	0.113	0.143	1.72	0.00316	0.0644	0.0844	21.3	0.0935
	MARAD	0.468	0.343	0.103	0.124	0.163	7.58	0.00301	0.0529	0.0758	21.3	0.112
	Mean	0.427	0.859	0.101	0.126	0.159	21.1	0.0383	0.662	0.333	22.7	
GG	MG	0.238	**0.151**	0.0587	0.0564	**0.0835**	5.98	0.00291	0.0317	0.0724	6.32	0.058
GC	MG	**0.0696**	0.227	**0.0509**	**0.0471**	**0.0766**	5.68	0.00975	0.13	0.0892	3.15	0.0427

Various molecular properties

Different ML methods

https://arxiv.org/abs/1702.05532

Predicting molecule properties: Performance

		μ	α	ε_{HOMO}	ε_{LUMO}	$\Delta\varepsilon$	$\langle R^2 \rangle$	ZPVE	U_0	C_v	ω_1	NMMAE
		Debye	$Bohr^3$	eV	eV	eV	$Bohr^2$	eV	eV	cal/molK	cm^{-1}	arb. u.
Mean		2.67	75.3	-6.54	0.322	6.86	1190	4.06	-76.6	31.6	3500	
MAD		1.17	6.29	0.439	1.05	1.07	203	0.717	8.19	3.21	238	
Target[a]		0.10	0.10	0.043	0.043	0.043	1.2	0.0012	0.043	0.050	10	
DFT[b]		0.09	0.74	2.0	2.6	1.19	-	0.0097	0.10	0.34	28	
EN	CM	0.844	1.33	0.338	0.631	0.722	55.5	0.0265	0.911	0.906	131	0.423
	BOB	0.763	1.20	0.283	0.521	0.614	55.3	0.0232	0.602	0.700	81.4	0.35
	BAML	0.686	0.793	0.186	0.275	0.339	32.6	0.0129	0.212	0.439	60.4	0.231
	ECFP4	0.737	3.45	0.224	0.344	0.383	118	0.270	3.68	1.51	86.6	0.462
	HDAD	0.563	0.437	0.139	0.238	0.278	6.19	0.00647	0.0983	0.0876	94.2	0.183
	HD	0.705	0.638	0.203	0.299	0.360	6.70	0.00949	0.192	0.195	104	0.236
	MARAD	0.707	0.698	0.222	0.305	0.391	27.4	0.00808	0.183	0.206	108	0.256
	Mean	0.715	1.22	0.228	0.373	0.441	43.1	0.0509	0.840	0.578	95.1	
BR	CM	0.844	1.33	0.338	0.632	0.723	55.5	0.0265	0.911	0.907	131	0.424
	BOB	0.761	1.14	0.279	0.521	0.614	48.0	0.0222	0.586	0.684	80.9	0.343
	BAML	0.685	0.785	0.183	0.275	0.339	30.4	0.0129	0.202	0.444	60.4	0.229
	ECFP4	0.737	3.45	0.224	0.344	0.383	118.0	0.270	3.69	1.51	86.7	0.462
	HDAD	0.565	0.43	0.14	0.238	0.278	5.94	0.00318	0.0614	0.0787	94.8	0.182
	HD	0.705	0.633	0.203	0.298	0.359	6.8	0.00693	0.171	0.19	104	0.235
	MARAD	0.647	0.533	0.18	0.257	0.315	26.8	0.00854	0.171	0.201	103	0.226
	Mean	0.706	1.19	0.221	0.367	0.430	41.7	0.0500	0.828	0.574	94.5	
RF	CM	0.608	1.04	0.208	0.302	0.373	45.0	0.0199	0.431	0.777	13.2	0.239
	BOB	0.450	0.623	0.120	0.137	0.164	39.0	0.0111	0.202	0.443	3.55	0.142
	BAML	0.434	0.638	0.107	0.118	0.141	51.1	0.0132	0.200	0.451	**2.71**	0.141
	ECFP4	0.483	3.70	0.143	0.145	0.166	109	0.242	3.66	1.57	14.7	0.349
	HDAD	0.454	1.71	0.116	0.136	0.156	48.3	0.0525	1.44	0.895	3.45	0.198
	HD	0.457	1.66	0.126	0.139	0.150	46.8	0.0497	1.39	0.879	4.18	0.197
	MARAD	0.607	0.676	0.178	0.243	0.311	45.3	0.0102	0.21	0.311	19.4	0.199
	Mean	0.499	1.43	0.142	0.174	0.209	54.9	0.0569	1.08	0.761	8.74	
KRR	CM	0.449	0.433	0.133	0.183	0.229	3.39	0.0048	0.128	0.118	33.5	0.136
	BOB	0.423	0.298	0.0948	0.122	0.148	**0.978**	0.00364	0.0667	0.0917	13.2	0.0981
	BAML	0.460	0.301	0.0946	0.121	0.152	3.9	0.00331	0.0519	0.082	19.9	0.105
	ECFP4	0.490	4.17	0.124	0.133	0.174	128	0.248	4.25	1.84	26.7	0.383
	HDAD	0.334	0.175	0.0662	0.0842	0.107	1.62	**0.00191**	**0.0251**	**0.0441**	23.1	0.0768
	HD	0.364	0.299	0.0874	0.113	0.143	1.72	0.00316	0.0644	0.0844	21.3	0.0935
	MARAD	0.468	0.343	0.103	0.124	0.163	7.58	0.00301	0.0529	0.0758	21.3	0.112
	Mean	0.427	0.859	0.101	0.126	0.150	21.1	0.0383	0.662	0.333	22.7	
GG	MG	0.238	**0.151**	0.0587	0.0564	**0.0835**	5.98	0.00291	0.0317	0.0724	6.32	0.058
GC	MG	**0.0696**	0.227	**0.0509**	**0.0471**	**0.0766**	5.68	0.00975	0.13	0.0892	3.15	0.0427

Various molecular properties

Different ML methods

https://arxiv.org/abs/1702.05532

Neural network based ML methods

A Monsoon prediction paper!

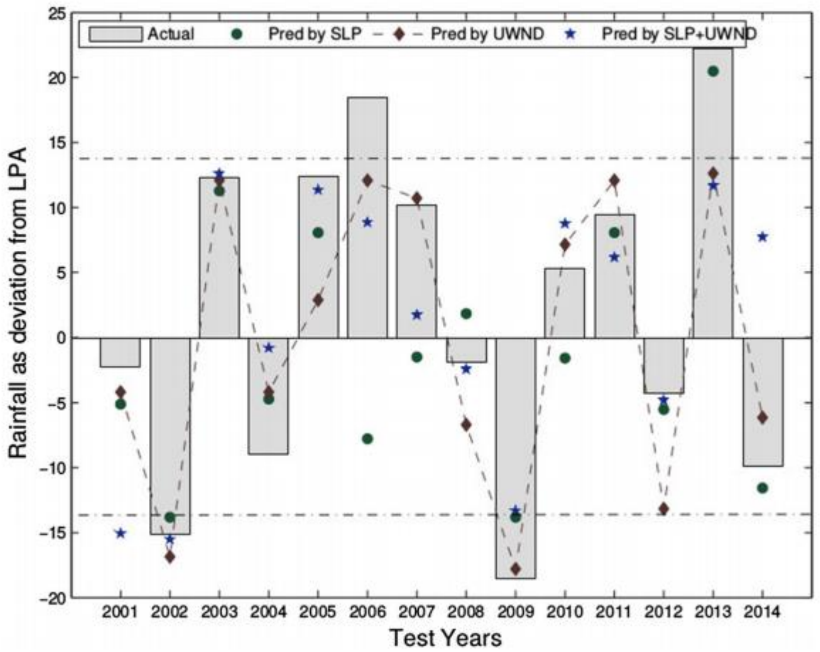


Figure 8. Forecast of the central India summer monsoon (June–September) by SLP, UWND and SLP+UWND during 2001–2014.

http://cse.iitkgp.ac.in/~pabitra/paper/jess17.pdf