Info 3950 Lecture 2
3 Sep 2019

Rise of the Machines: Deep Learning from Backgammon to Skynet

Paul Ginsparg, Physics and InfoSci
Cornell Univ

Over the past seven years, there have been significant advances in applications of artificial intelligence, machine learning, and specifically deep learning, to a variety of familiar tasks. From image and speech recognition, self-driving cars, and machine translation, to beating the Go champion, it's been difficult to stay abreast of all the breathless reports of superhuman machine performance. There has as well been a recent surge in applications of machine learning ideas to research problems in the hard sciences and medicine. I will endeavor to provide an outsider's overview of the ideas underlying these recent advances and their evolution over the past few decades, and project some prospects and pitfalls for the near future.

Aspen 16 Jan 2019

video games, poker, chess, go,
speech recognition, language translation,
medical applications (dermatology, ophthalmology),
chemical synthesis,
data analysis,
self-driving cars

Plan:
Teaser
How it all works
Historical highlights
Future

Google/Verily/Stanford [arXiv:1708.09843, Nature (2018)]

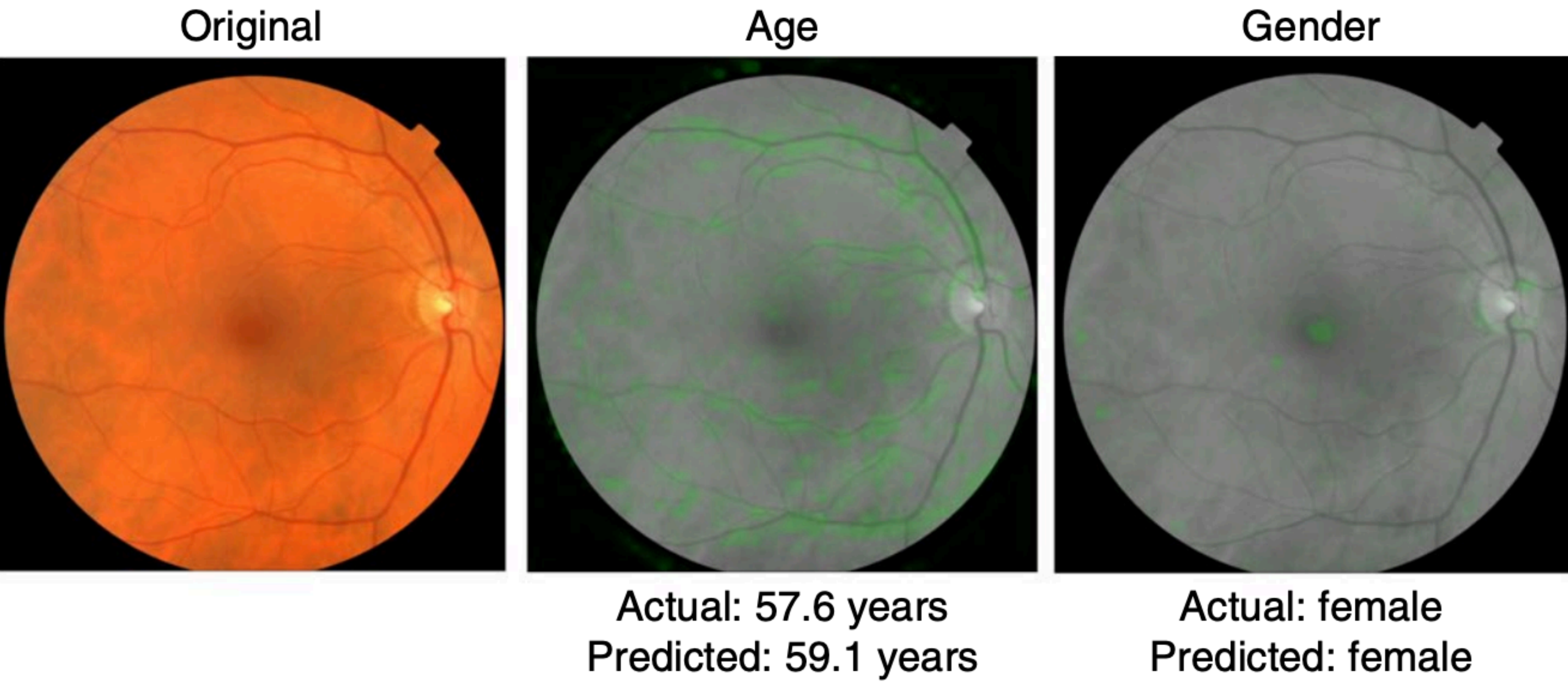
Original

"retinal fundus image":
photograph of back of eye taken through pupil
(used for over 100 years for detecting eye disease)

Now: using AI can also predict risk of heart attack or stoke.

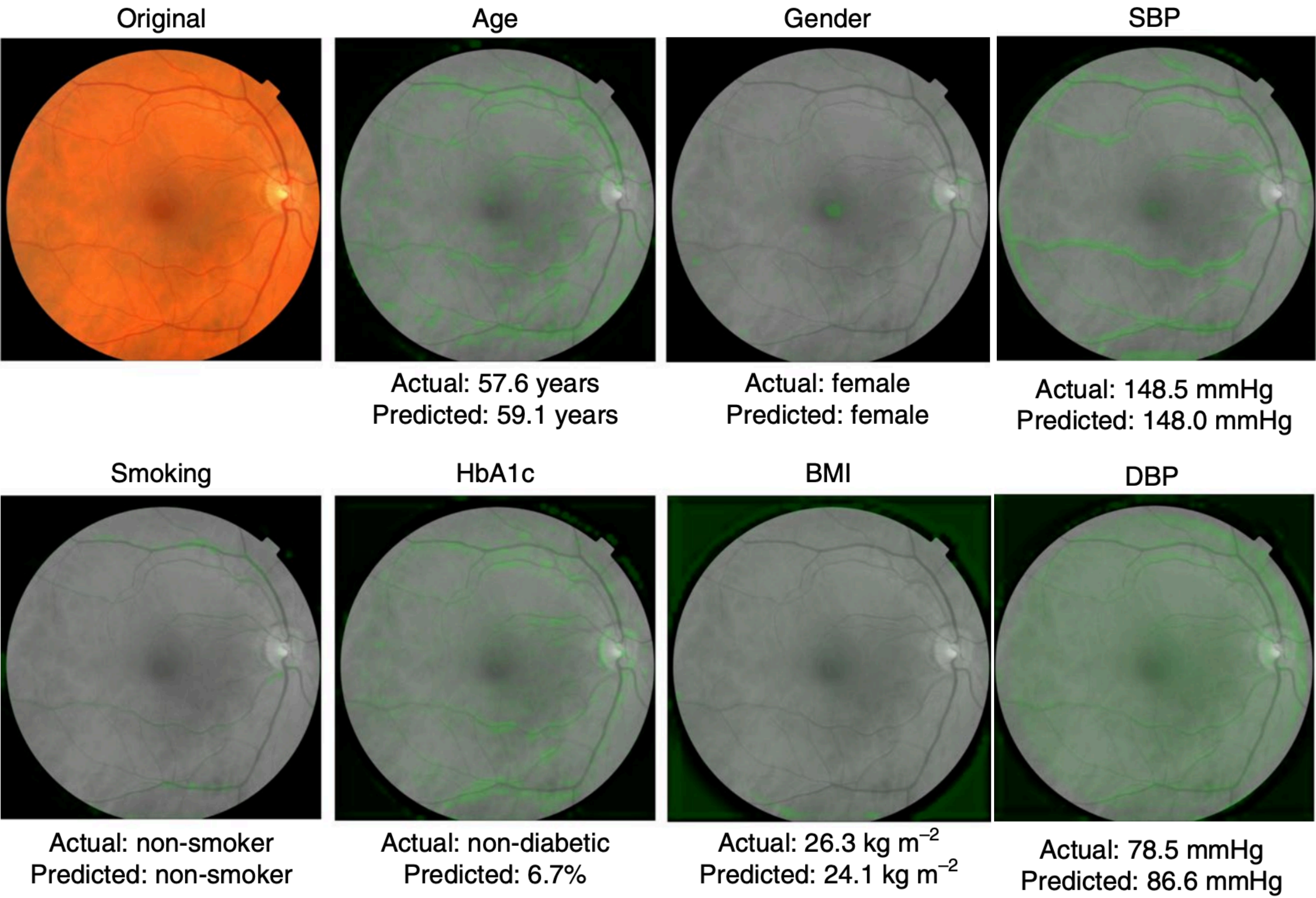
and more …

Google/Verily/Stanford [arXiv:1708.09843, Nature (2018)]

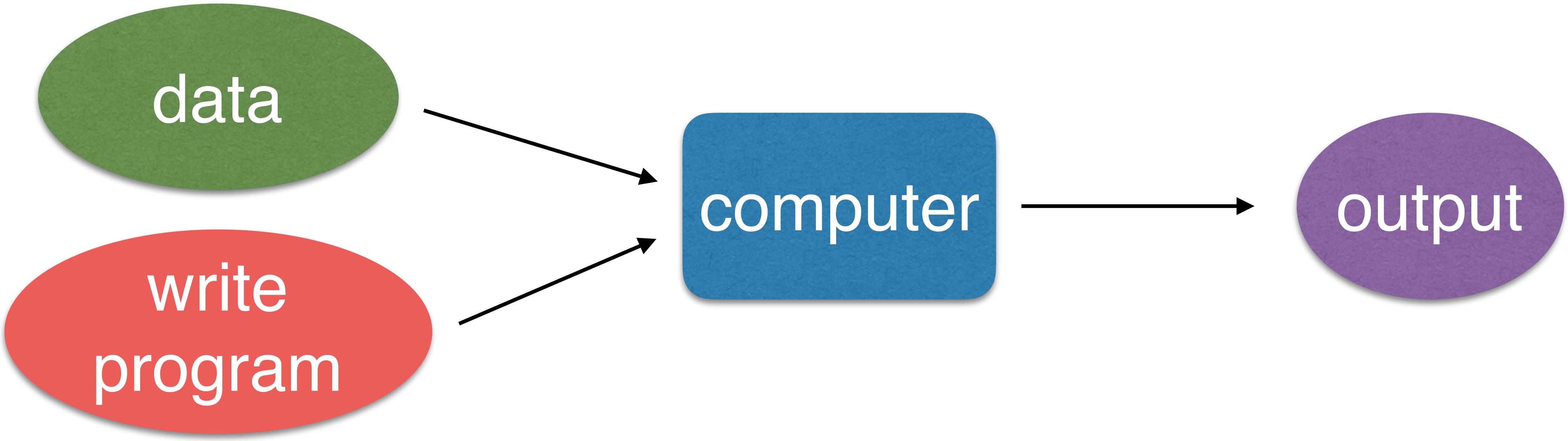


Deep learning models trained on data from 284,335 patients and validated on two independent datasets of 12,026 and 999 patients

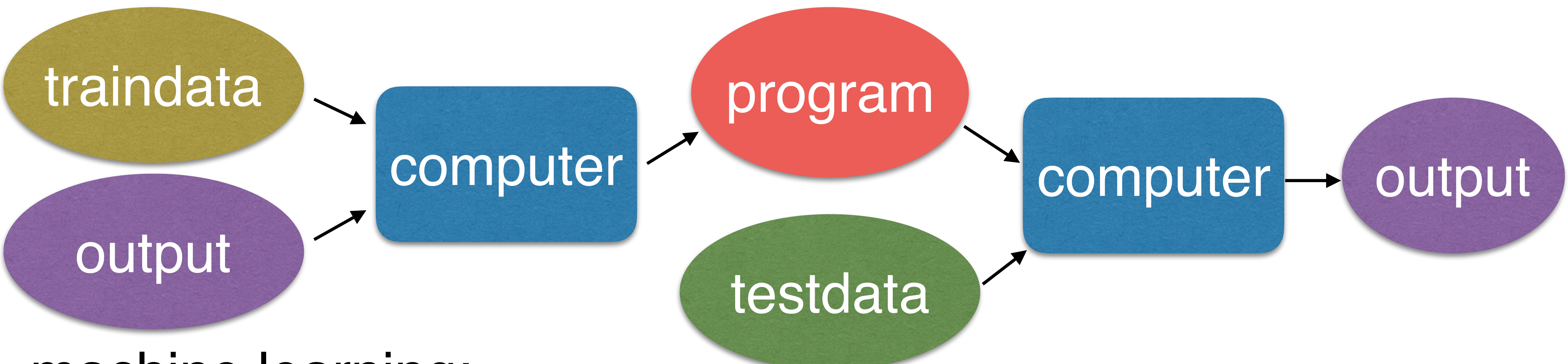
Google/Verily/Stanford [arXiv:1708.09843, Nature (2018)]



traditional cs:



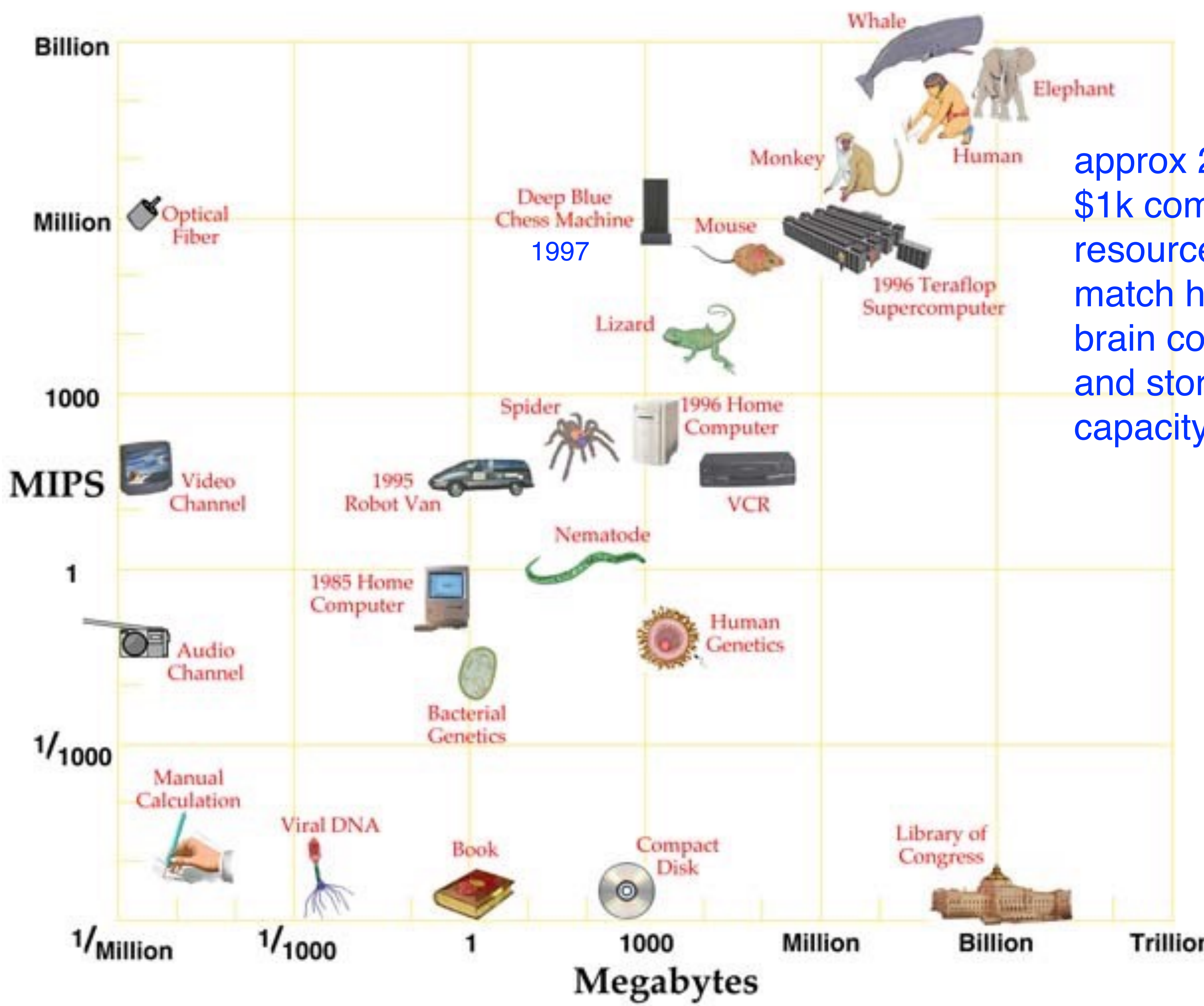
but now: data is fmri scan, task is to determine probability of Alzheimers
We don't know how to write the program …



machine learning:

use training data and output to generate program,
which then generates output for test data.

All Thinks, Great and Small (H. Moravec, CMU, 1998)



approx 2030 $1k compute resources will match human brain compute and storage capacity

1943 1950 1958

1958 New York Times…

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Without Human Controls

The Navy said the perceptron would be the first non-living mechanism "capable of receiving, recognizing and identifying its surroundings without any human training or control."

The "brain" is designed to remember images and information it has perceived itself. Ordinary computers remember only what is fed into them on punch cards or magnetic tape.

Later Perceptrons will be able to recognize people and call out their names and instantly translate speech in one language to speech or writing in another language, it was predicted.

Mr. Rosenblatt said in principle it would be possible to build brains that could reproduce themselves on an assembly line and which would be conscious of their existence.

In today's demonstration, the "704" was fed two cards, one with squares marked on the left side and the other with squares on the right side.

Learns by Doing

In the first fifty trials, the machine made no distinction between them. It then started registering a "Q" for the left squares and "O" for the right squares.

Dr. Rosenblatt said he could explain why the machine learned only in highly technical terms. But he said the computer had undergone a "self-induced change in the wiring diagram."

The first Perceptron will have about 1,000 electronic "association cells" receiving electrical impulses from an eye-like scanning device with 400 photo-cells. The human brain has 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

1969 Minsky-Papert "Perceptrons"

"AI winter #1"

1980s

HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | U.S. Edition

The New York Times

Science

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION

ENVIRONMENT | SPACE & COSMOS

COMPUTER SCIENTISTS STYMIED IN THEIR QUEST TO MATCH HUMAN VISION

By WILLIAM J. BROAD
Published: September 25, 1984

EXPERTS pursuing one of man's most audacious dreams - to create machines that think - have stumbled while taking what seemed to be an elementary first step. They have failed to master vision.

After two decades of research, they have yet to teach machines the seemingly simple act of being able to recognize everyday objects and to distinguish one from another.

Instead, they have developed a profound new respect for the sophistication of human sight and have scoured such fields as mathematics, physics, biology and psychology for clues to help them achieve the goal of machine vision.

FACEBOOK
TWITTER
GOOGLE+
EMAIL
SHARE
PRINT
REPRINTS

BIG Data!

2010

GPU 70x faster
to train (week->hrs)

.35% on MNIST

Hinton students ->
Google, Microsoft
(e.g., Android speech
recognition)

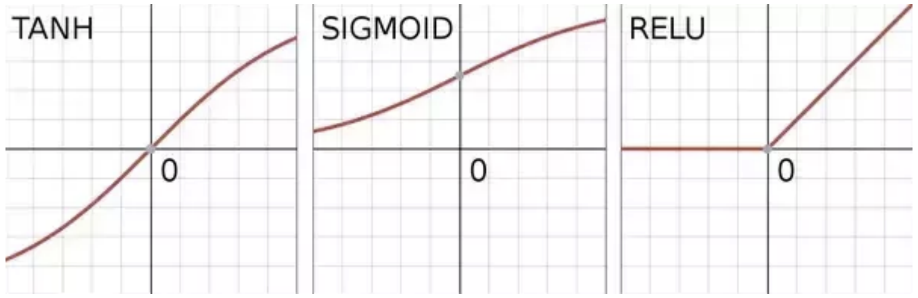
2011

IBM Watson
wins
Jeopardy

2012

Multiple groups
speech recognition
Imagenet (Ng, Dean, et al)
70% improvement
Dropout, 16k CPUs
1B weights
(1M for MNIST)

Choice of
Activation
matters



BIG Data!

2010 2011 2012

Computer Science > Learning

Building high-level features using large scale unsupervised learning

Quoc V. Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Jeff Dean, Andrew Y. Ng

(Submitted on 29 Dec 2011 (v1), last revised 12 Jul 2012 (this version, v5))

We consider the problem of building high-level, class-specific feature detectors from only unlabeled data. For example, is it possible to learn a face detector using only unlabeled images? To answer this, we train a 9-layered locally connected sparse autoencoder with pooling and local contrast normalization on a large dataset of images (the model has 1 billion connections, the dataset has 10 million 200x200 pixel images downloaded from the Internet). We train this network using model parallelism and asynchronous SGD on a cluster with 1,000 machines (16,000 cores) for three days. Contrary to what appears to be a widely-held intuition, our experimental results reveal that it is possible to train a face detector without having to label images as containing a face or not. Control experiments show that this feature detector is robust not only to translation but also to scaling and out-of-plane rotation. We also find that the same network is sensitive to other high-level concepts such as cat faces and human bodies. Starting with these learned features, we trained our network to obtain 15.8% accuracy in recognizing 20,000 object categories from ImageNet, a leap of 70% relative improvement over the previous state-of-the-art.

9 layer sparse autoencoder
1B parameters
10M 200x200 images
down to 18x18
16000 cores
visual cortex is 1M x larger

Subjects: **Learning (cs.LG)**
Cite as: arXiv:1112.6209 [cs.LG]
(or arXiv:1112.6209v5 [cs.LG] for this version)

BIG Data!

2010 2011 2012

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

BIG Data!

2010 2011 2012

ImageNet Challenge

- IMAGENET Large Scale Visual Recognition Challenge (ILSVRC)
  - **1.2M** training images with **1K** categories
  - Measure top-5 classification error

Image classification

Easiest classes

Output
Scale
T-shirt
Steel drum
Drumstick
Mud turtle

✓

Output
Scale
T-shirt
Giant panda
Drumstick
Mud turtle

✗

J. Deng, Wei Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei , "**ImageNet: A Large-Scale Hierarchical Image Database**", CVPR 2009.
O. Russakovsky et al., "**ImageNet Large Scale Visual Recognition Challenge**", Int. J. Comput. Vis.,, Vol. 115, Issue 3, pp 211-252, 2015.

BIG Data!

2010 2011 2012

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

BIG Data!

2010

GPU 70x faster to train (week->hrs)

.35% on MNIST

Hinton students -> Google, Microsoft (e.g., Android speech recognition)

2011

IBM Watson wins Jeopardy

2012

Multiple groups speech recognition

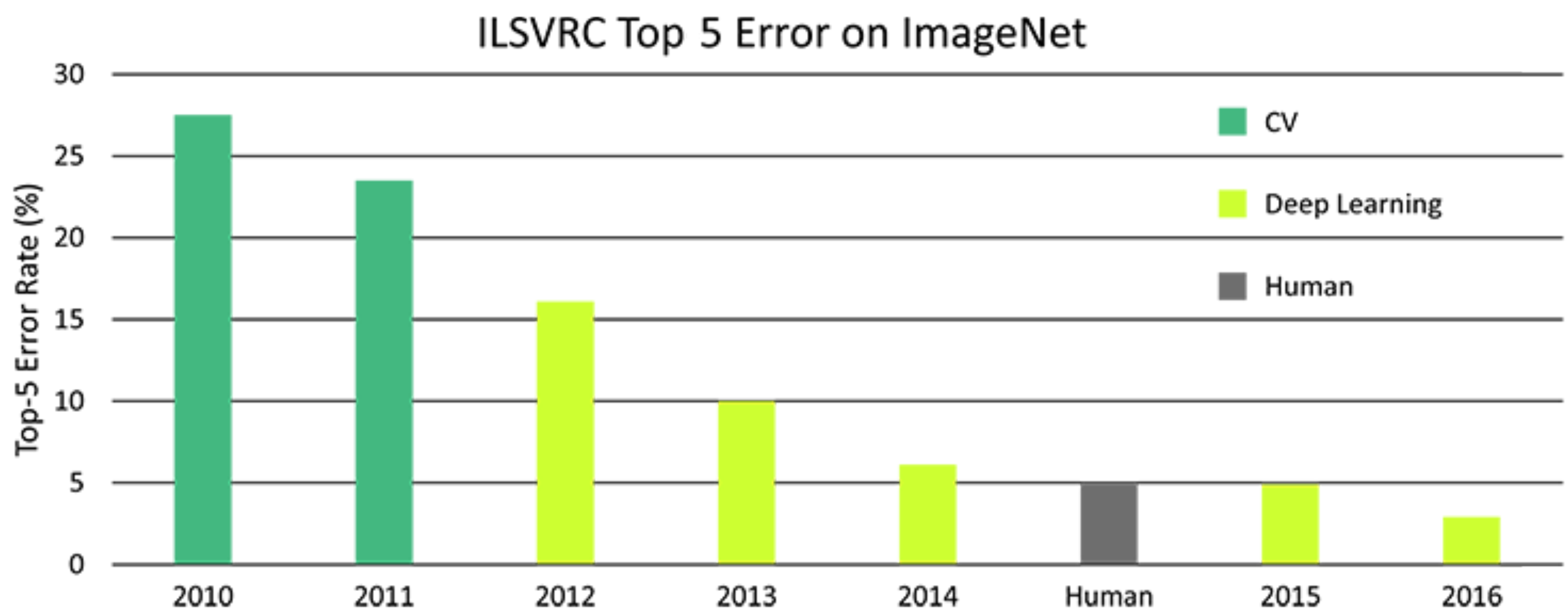
Imagenet (Ng, Dean, et al)

70% improvement

Dropout, 16k CPUs

1B weights

(1M for MNIST)



BIG Data!

2010

GPU 70x faster to train (week->hrs)

2011

IBM Watson wins Jeopardy

2012

Multiple groups speech recognition

Imagenet (Ng, Dean, et al)

SCIENCE

Researchers Announce Advance in Image-Recognition Software

By JOHN MARKOFF NOV. 17, 2014

Email

Share

Tweet

Save

More

MOUNTAIN VIEW, Calif. — Two groups of scientists, working independently, have created artificial intelligence software capable of recognizing and describing the content of photographs and videos with far greater accuracy than ever before, sometimes even mimicking human levels of understanding.

Until now, so-called computer vision has largely been limited to recognizing individual objects. The new software, described on Monday by researchers at Google and at Stanford University, teaches itself to identify entire scenes: a group of young men playing Frisbee, for example, or a herd of elephants marching on a grassy plain.

The software then writes a caption in English describing the picture. Compared with human observations, the researchers found, the computer-written descriptions are surprisingly accurate.

2013	2014	2015	2016	2017
Deep RL beats human expert at Atari games	GAN	microsoft real-time translation (speech to speech) NIPS Dec	Mar: google alphaGO beats Lee Sedol (just 19 yrs after chess, not 30-40 years)	Jan: no-limit texas hold'em CMU program beats top humans (not another 10 yrs) Mar: AlphaGo Master beats Ke Jie (world #1)

self-driving vehicles,
superhuman
performance in
image recog,
… ->

Why us? Why now?

1) Bigger Data
2) Faster CPU (+GPU)
3) Better Initialization
4) Right non-linearity

2013	2014	2015	2016	2017
Deep RL beats human expert at Atari games	GAN	microsoft real-time translation (speech to speech) NIPS Dec	Mar: google alphaGO beats Lee Sedol (just 19 yrs after chess, not 30-40 years)	Jan: no-limit texas hold'em CMU program beats top humans (not another 10 yrs) Mar: AlphaGo Master beats Ke Jie (world #1)

self-driving vehicles,
superhuman
performance in
image recog,
… ->

2013 | 2014 | 2015 | 2016 | 2017

2013
Deep RL beats human expert at Atari games

2014
GAN

2015
microsoft real-time translation (speech to speech)

NIPS Dec

2016
Mar: google alphaGO beats Lee Sedol (just 19 yrs after chess, not 30-40 years)

2017
Jan: no-limit texas hold'em CMU program beats top humans (not another 10 yrs)

Mar: AlphaGo Master beats Ke Jie (world #1)

self-driving vehicles, superhuman performance in image recog, … ->

Why us? Why now?

1) Bigger Data [WWW -> social media -> text/data sharing]
2) Faster CPU (+GPU)
3) Better Initialization
4) Right non-linearity

software (TensorFlow, torch … caffe2, decaffeine, matconvnet, microsoft cognitive toolkit, pytorch)

2017

Oct: AlphaGo Zero
(3 days to beat AlphaGo Lee,
21 Days to beat AlphaGo Master)

Dec: AlphaZero
(24 hours to superhuman
chess, shogi, go)

“tabula rasa”

Kasparov: “the truth”

2018

Medical Image Analysis
(CT scans for stroke,
Image Generation;
entire MRI processing chain,
acquisition to image retrieval,
segmentation to disease prediction)

Dec: AlphaZero
(24 hours to superhuman
chess, shogi, go
— discovered the principles on its own
and quickly became best player)

2019?

AI-enabled chips
IoT + AI at the edge
interoperability (ONNX)
auto-ML
AI+DevOps= AIOps

some science problems (protein folding) like Go: well-known rules and a well-described goal. similar algorithms might be applied to similar tasks in quantum chemistry, materials design and robotics

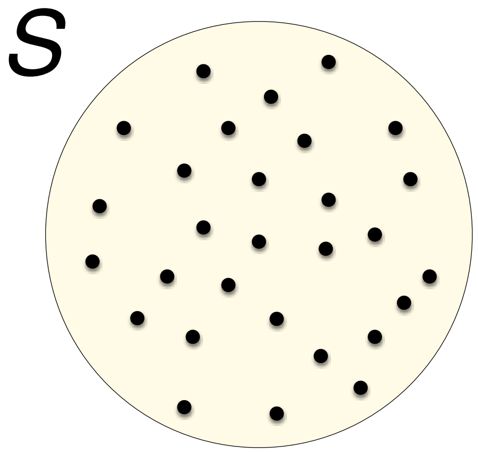
All pervasive:

e.g., google: search, image search, driverless cars, voice recog, youtube recommender, street labels
facebook: images through two nn's, tag friends, understand image, (e.g., no food), major companies hiring like crazy. ibm watson, siri, yelp (also fraud), tesla, netflix, skype live translation,

Discrete Probability and Counting

A *finite probability space* is a set S and a real function $p(s)$ on S such that:

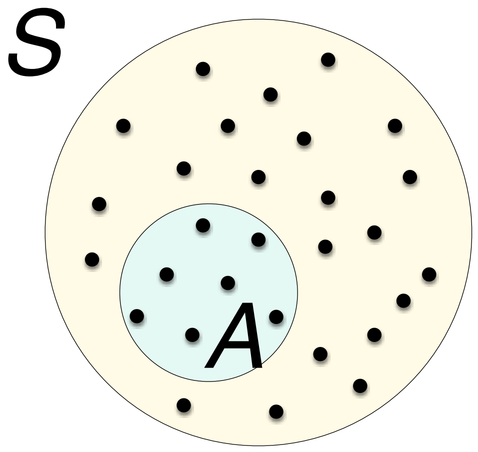
- $p(s) \geq 0$, $\forall s \in S$, and
- $\sum_{s \in S} p(s) = 1$.



We refer to S as the *sample space*, subsets of S as *events*, and p as the *probability distribution*.

The probability of an event $A \subseteq S$ is $p(A) = \sum_{a \in A} p(a)$.

(Note that $p(\emptyset) = 0$.)



Conditional Probability

Suppose we know that one event has happened and we wish to ask about another.

For two events A and B, the *joint probability* of A and B is defined as

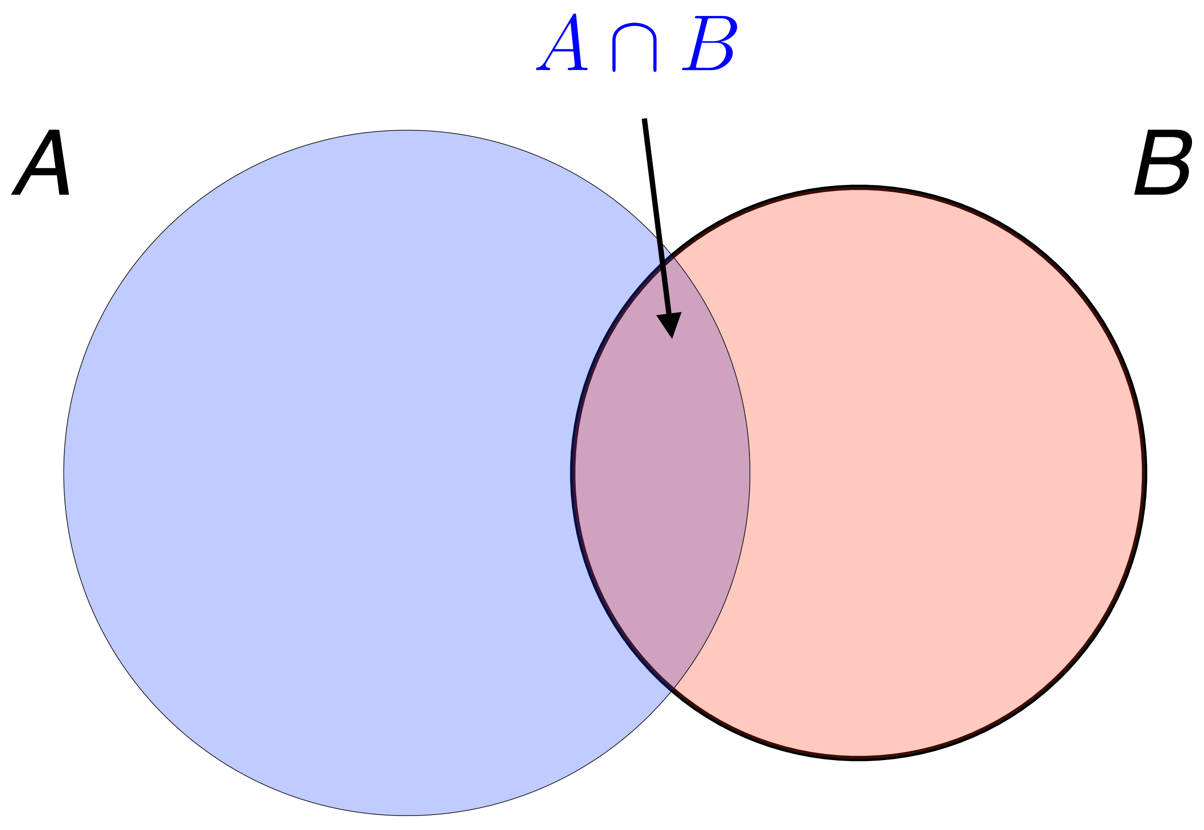
$$p(A, B) = p(A \cap B)$$

the probability of the intersection of events A and B in the sample space,

equivalently the probability that events A and B both occur

The *conditional probability* of A relative to B is

$$p(A|B) = p(A \cap B)/p(B)$$ "the probability of A given B"



$$p(A|B) = p(A \cap B)/p(B)$$

"the probability of A given B"

$$= p(A,B) \ / \ p(B)$$

Bayes' Rule

A simple formula follows from the above definitions and symmetry of the joint probability:

$p(A|B)p(B) = p(A,B) = p(B,A) = p(B|A)p(A)$:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$

Called "Bayes' theorem" or "Bayes' rule" — connects inductive and deductive inference

(Rev. Thomas Bayes (1763), Pierre-Simon Laplace (1812), Sir Harold Jeffreys (1939))

For mutually disjoint sets A_i with $\bigcup_{i=1}^{n} A_i = S$, Bayes' rule takes the form

$$p(A_i|B) = \frac{p(B|A_i)p(A_i)}{p(B|A_1)p(A_1) + \ldots + p(B|A_n)p(A_n)} .$$

Example 1: Consider a casino with loaded and unloaded dice.

For a loaded die (L), probability of rolling a 6 is 50%:

$p(6|L) = 1/2$, and $p(i|L) = 1/10$ $(i = 1, \ldots, 5)$

For a fair die ($\overline{L}$), the probabilities are $p(i|\overline{L}) = 1/6$ $(i = 1, \ldots, 6)$.

Suppose there's a 1% probability of choosing a loaded die:

$p(L) = 1/100$.

If we select a die at random and roll three consecutive 6's with it,

what is the posterior probability, $P(L|6, 6, 6)$, that it was loaded?

The probability of the die being loaded, given 3 consecutive 6's, is

$$
\begin{aligned}
p(L|6,6,6) &= \frac{p(6,6,6|L)p(L)}{p(6,6,6)} = \frac{p(6|L)^3 p(L)}{p(6|L)^3 p(L) + p(6|\overline{L})^3 p(\overline{L})} \\
&= \frac{(1/2)^3 \cdot (1/100)}{(1/2)^3 \cdot (1/100) + (1/6)^3 \cdot (99/100)} \\
&= \frac{1}{1 + (1/3)^3 \cdot 99} = \frac{1}{1 + 11/3} = \frac{3}{14} \approx .21 \ ,
\end{aligned}
$$

so only a roughly 21% chance that it was loaded.

(Note that the Bayesian "prior" in the above is $p(L) = 1/100$, giving the expected probability before collecting the data from actual rolls, and significantly affects the inferred posterior probability.)

Binary Classifiers:

Use a set of features to determine whether objects have binary (yes or no) properties.

Examples: whether or not a text is classified as medicine,
or whether an email is classified as spam.

In those cases, the features of interest might be the words the text or email contains.

"Naive Bayes" methodology:

statistical method (making use of the word probability distribution)

as contrasted with a "rule-based" method
(where a set of heuristic rules is constructed and then has to be maintained over time)

Spam Filters

Spam filter = binary classifier where property is whether message is spam (S) or non-spam ($\overline{S}$).

Features = words of the message.

Assume we have a **training set** of messages tagged as spam or non-spam

and use the document frequency of words in the two partitions as evidence regarding whether new messages are spam.

(baby machine learning)

Example 1 (Rosen p. 422):

Suppose the word "Rolex" appears in 250 messages of a set of 2000 spam messages, and in 5 of 1000 non spam messages.

Then we estimate $p(\text{“Rolex”}|S) = 250/2000 = .125$
and $p(\text{“Rolex”}|\overline{S}) = 5/1000 = .005$.

Assuming a "flat prior" $(p(S) = p(\overline{S}) = 1/2)$ in Bayes' law gives

$$p(S|\text{“Rolex”}) = \frac{p(\text{“Rolex”}|S)p(S)}{p(\text{“Rolex”}|S)p(S) + p(\text{“Rolex”}|\overline{S})p(\overline{S})} = \frac{.125}{.125 + .005} = \frac{.125}{.130} = .962\ .$$

With a rejection threshold of .9, this would be rejected.

Example 2 (two words, "stock" and "undervalued"):

Now suppose in a training set of 2000 spam messages and 1000 non-spam messages,

the word "stock" appears in 400 spam messages and 60 non-spam,
and the word "undervalued" appears in 200 spam and 25 non-spam messages.

Then we estimate

$$p(\text{“stock”}|S) = 400/2000 = .2$$
$$p(\text{“stock”}|\overline{S}) = 60/1000 = .06$$
$$p(\text{“undervalued”}|S) = 200/2000 = .1$$
$$p(\text{“undervalued”}|\overline{S}) = 25/1000 = .025\ .$$

Key assumption: assume statistical independence to estimate as

$$p(w_1, w_2|S) = p(w_1|S) \cdot p(w_2|S)$$

$$p(w_1, w_2|\overline{S}) = p(w_1|\overline{S}) \cdot p(w_2|\overline{S})$$

(This assumption is **not** true in practice: words are **not** statistically independent. But we're only interested in determining whether above or below some threshold, not trying to calculate an accurate $p(S|\{w_1, w_2, \ldots, w_n\})$)

Write $w_1 =$ “stock” and $w_2 =$ “undervalued”, and recall:

$$p(w_1|S) = 400/2000 = .2 \qquad p(w_1|\overline{S}) = 60/1000 = .06,$$
$$p(w_2|S) = 200/2000 = .1 \qquad p(w_2|\overline{S}) = 25/1000 = .025$$

So assuming a flat prior ($p(S) = p(\overline{S}) = 1/2$), and independence of the features gives

$$p(S|w_1, w_2) = \frac{p(w_1, w_2|S)p(S)}{p(w_1, w_2|S)p(S) + p(w_1, w_2|\overline{S})p(\overline{S})}$$

“naive”

$$= \frac{p(w_1|S)p(w_2|S)p(S)}{p(w_1|S)p(w_2|S)p(S) + p(w_1|\overline{S})p(w_2|\overline{S})p(\overline{S})} = \frac{.2 \cdot .1}{.2 \cdot .1 + .06 \cdot .025} = .930$$

at a .9 probability threshold a message containing those two words would be rejected as spam.

More generally, for n features (words)

$$p(S|\{w_1, w_2, \ldots, w_n\}) = \frac{p(\{w_1, w_2, \ldots, w_n\}|S)p(S)}{p(\{w_1, w_2, \ldots, w_n\})}$$

$$= \frac{p(\{w_1, w_2, \ldots, w_n\}|S)p(S)}{p(\{w_1, w_2, \ldots, w_n\}|S)p(S) + p(\{w_1, w_2, \ldots, w_n\}|\overline{S})p(\overline{S})}$$

"naive"

$$= \frac{p(w_1|S)p(w_2|S)\cdots p(w_n|S)p(S)}{p(w_1|S)p(w_2|S)\cdots p(w_n|S)p(S) + p(w_1|\overline{S})p(w_2|\overline{S})\cdots p(w_n|\overline{S})p(\overline{S})}$$

$$= \frac{p(S)\prod_{i=1}^{n} p(w_i|S)}{p(S)\prod_{i=1}^{n} p(w_i|S) + p(\overline{S})\prod_{i=1}^{n} p(w_i|\overline{S})}$$

naive bayes

Bayes: $p(C|w) = p(w|C)p(C)/p(w)$

Naive: $p(\{w_i\}|C) = \prod_i p(w_i|C)$

- **spam filter (**$p(S|\{w_i\})/p(\overline{S}|\{w_i\})$**)**
- **text classification (on arXiv** $> 95\%$ **now)**
- **spell correction**
- **voice recognition**
- $\ldots$

simplest algorithm works better with more data.

for arXiv use multigram vocab: genetic_algorithm, black_hole

"The Unreasonable Effectiveness of Naive Bayes in the Data Sciences"

$$p(c \mid w) \propto \overbrace{p(w \mid c)}^{\text{Error model}}\ \overbrace{p(c)}^{\text{Language model}}$$

(spell correction)

$$p(e \mid f) \propto \overbrace{p(f \mid e)}^{\text{Translation model}}\ \overbrace{p(e)}^{\text{Language model}}$$

(machine translation)

$$p(c \mid s) \propto \overbrace{p(s \mid c)}^{\text{Acoustic model}}\ \overbrace{p(c)}^{\text{Language model}}$$

(speech recognition)