Datasets, bias, and adaption

Bill Freeman, Antonio Torralba, Phillip Isola
6.819 / 6.869
A machine learning algorithm will do whatever the training data tells it to do.

If the data is bad or biased, the learned algorithm will be too.
Microsoft’s Tay chatbot

Chatbot released on twitter.

Learned from interactions with users (?)

Started mimicking offensive language, was shut down.
<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>what is the yellow thing?</td>
<td>frisbee</td>
<td>79.844%</td>
</tr>
<tr>
<td>Predicted top-5 answers with confidence:</td>
<td>surfboard</td>
<td>7.319%</td>
</tr>
<tr>
<td></td>
<td>banana</td>
<td>2.044%</td>
</tr>
<tr>
<td></td>
<td>lemon</td>
<td>2.438%</td>
</tr>
<tr>
<td></td>
<td>surfboards</td>
<td>2.52%</td>
</tr>
</tbody>
</table>
how many trains are in the picture?

Predicted top-5 answers with confidence:

3
30.23%

5
18.27%

4
17.00%

2
11.34%

6
7.80%
Of number questions (e.g. “how many…”), 26.04% of the time, the answer is 2.

Of yes/no questions, 58.83% of the time, the answer is yes.

[“Colorful image colorization”, Zhang et al., ECCV 2016]
[“Colorful image colorization”, Zhang et al., ECCV 2016]
[“Colorful image colorization”, Zhang et al., ECCV 2016]
Training data

\(\{ x, y \} \),

\(\{ x, y \} \),

\(\{ x, y \} \),

\(\ldots\)

Test data

\(x'\)

\(\text{Test data} \)

\(x'\)
Training data

What Google thinks are student bedrooms

Search

About 66,700,000 results (0.15 seconds)
Training data

Driving simulator (GTA)

Test data

Driving in the real world
Let’s revisit the problem of generalization
Training data

\[ \{x_i, y_i\}_{i=1}^N \]

Test data

\[ \{x, y\} \sim p_{\text{data}} \]
Training data

Test data

True data-generating process

$p_{\text{data}}$

$\{x^{(\text{train})}_i, y^{(\text{train})}_i\}_{i=1}^N \sim p_{\text{data}}$

$\{x^{(\text{test})}_i, y^{(\text{test})}_i\}_{i=1}^M \sim p_{\text{data}}$
This is a huge assumption! Almost never true in practice!
Training data

Test data

Much more commonly, we have

\[ p_{\text{train}} \neq p_{\text{test}} \]
Our training data did cover the part of the distribution that was tested (biased data)
Domain gap between $p_{train}$ and $p_{test}$ will cause us to fail to generalize.

Space of natural images

Training data

Test data
Social consequences

Color Matters in Computer Vision
Facial recognition algorithms made by Microsoft, IBM and Face++ were more likely to misidentify the gender of black women than white men.

Gender was misidentified in **up to 1 percent of lighter-skinned males** in a set of 385 photos.

Gender was misidentified in **35 percent of darker-skinned females** in a set of 271 photos.

How can we collect good data?

+ Correctly labeled
+ Unbiased (good coverage of all relevant kinds of data)
The value of data

The Large Hadron Collider
$10^{10}$

Amazon Mechanical Turk
$10^2 - 10^4$
But can humans collect good data?
Getting more humans in the annotation loop

Labeling to get a Ph.D.

Labeling for fun
Luis Von Ahn and Laura Dabbish 2004

Labeling for money
(Sorokin, Forsyth, 2008)

Labeling because it gives you added value

Visipedia
(Endogie, Perona, et al)

Just for labeling

(amazon)
Beware of the human in your loop

• What do you know about them?
• Will they do the work you pay for?

Let’s check a few simple experiments
People have biases...

Turkers were offered 1 cent to pick a number from 1 to 10.

~850 turkers

Experiment by Greg Little
From http://groups.csail.mit.edu/uid/deneme/
Do humans have consistent biases?

Results form 100 HITS:

Experiment by Greg Little
From http://groups.csail.mit.edu/uid/deneme/
Do humans do what you ask for?

Experiment by Rob Miller
From http://groups.csail.mit.edu/uid/deneme/
Are humans reliable even in simple tasks?

Experiment by Greg Little
From http://groups.csail.mit.edu/uid/deneme/
Tool went online July 1st, 2005
Labelme.csail.mit.edu

B. Russell, A. Torralba, K. Murphy, W.T. Freeman. IJCV 2008
Please label as many objects as you want in this image. Scroll down to see the entire image.
Task: Label one object in this image
LabelMe iterations

1) Label as many objects as you can

2) Delete any wrong polygon

3) Go to 1
Label some objects
Delete any wrong polygons.
Label some objects
Delete any wrong polygons
Label some objects
Delete any wrong polygons
Label some objects
Who does the work?

Let's hire that one!

From http://groups.csail.mit.edu/uid/deneme/
### COCO vs ADE20K

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Images</th>
<th>Obj. inst.</th>
<th>Obj. classes</th>
<th>Part inst.</th>
<th>Part classes</th>
<th>Obj. classes per image</th>
</tr>
</thead>
<tbody>
<tr>
<td>COCO</td>
<td>123,287</td>
<td>886,284</td>
<td>91</td>
<td>0</td>
<td>0</td>
<td>3.5</td>
</tr>
<tr>
<td>ImageNet*</td>
<td>476,688</td>
<td>534,309</td>
<td>200</td>
<td>0</td>
<td>0</td>
<td>1.7</td>
</tr>
<tr>
<td>NYU Depth V2</td>
<td>1,449</td>
<td>34,064</td>
<td>894</td>
<td>0</td>
<td>0</td>
<td>14.1</td>
</tr>
<tr>
<td>Cityscapes</td>
<td>25,000</td>
<td>N/A</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>SUN</td>
<td>16,873</td>
<td>313,884</td>
<td>4,479</td>
<td>0</td>
<td>0</td>
<td>9.8</td>
</tr>
<tr>
<td>OpenSurfaces</td>
<td>22,214</td>
<td>71,460</td>
<td>160</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>PascalContext</td>
<td>10,103</td>
<td>(\sim104,398)**</td>
<td>540</td>
<td>181,770</td>
<td>40</td>
<td>5.1</td>
</tr>
<tr>
<td>ADE20K</td>
<td>22,000</td>
<td>415,099</td>
<td>2,944</td>
<td>171,148</td>
<td>354</td>
<td>10.5</td>
</tr>
</tbody>
</table>

* has only bounding boxes (no pixel-level segmentation). Sparse annotations.
** PascalContext dataset does not have instance segmentation. In order to estimate the number of instances, we find connected components (having at least 150 pixels) for each class label.
Sorokin, Forsyth, 2008

Carl Vondrick, Deva Ramanan, Don Patterson

N. Kumar, A. C. Berg,
P. N. Belhumeur, and S. K. Nayar, ICCV 2009

Farhadi Endres Hoiem Forsyth CVPR 2008

And many more...
So we can sometimes collect good training data.

But suppose we messed up. Our test setting doesn’t look like the training data!

How can we bridge the domain gap?
Domain gap between $p_{\text{train}}$ and $p_{\text{test}}$ will cause us to fail to generalize.
Domain gap between \( p_{source} \) and \( p_{target} \) will cause us to fail to generalize.
Idea #1: transform the target domain to look like the source domain

(Or vice versa) This is called **domain adaptation**
Domain adaptation

- We have source domain pairs \( \{x_{\text{source}}, y_{\text{source}}\} \)
- Learn a mapping \( F: x_{\text{source}} \rightarrow y_{\text{source}} \)
- We want to apply \( F \) to target domain data \( x_{\text{target}} \)
- Find transformation \( T: x_{\text{target}} \rightarrow x_{\text{source}} \)
- Now apply \( F(T(x_{\text{target}})) \) to predict \( y_{\text{target}} \)
It's a just another distribution mapping problem!
GANs

Gaussian

Target distribution

$Z$ $\rightarrow$ $Y$
CycleGAN

Horses

Zebras

X

→

Y
Domain adaptation

$p_{\text{source}}$ $\rightarrow$ $p_{\text{target}}$
Sim2real

Simulated data → Real data

[Richter*, Vineet* et al. 2016] [Krähenbühl et al. 2018]
Synthetic Data as Supervision

GTA5 images

Segmentation labels

[Richter*, Vineet* et al. 2016] [Krähenbühl et al. 2018]
CycleGAN

Training data

[Hoffman, Tzeng, Park, Zhu, Isola, Saenko, Darrell, Efros, 2018]
CycleGAN

[Image of truck on road]

Training data

[Image of truck with highlights]

[Hoffman, Tzeng, Park, Zhu, Isola, Saenko, Darrell, Efros, 2018]
CycleGAN  

Semantic segmentation

Training data

[Hoffman, Tzeng, Park, Zhu, Isola, Saenko, Darrell, Efros, 2018]
### Domain Adaptation with CycleGAN

**Train on GTA5 data**

**Test on real images**

<table>
<thead>
<tr>
<th></th>
<th>meanIOU</th>
<th>Per-pixel accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle (Train and test on Real)</td>
<td>60.3</td>
<td>93.1</td>
</tr>
<tr>
<td>Train on CG, test on Real</td>
<td>17.9</td>
<td>54.0</td>
</tr>
</tbody>
</table>
Domain Adaptation with CycleGAN

Train on CycleGAN data

Test on real images

<table>
<thead>
<tr>
<th>Training Setting</th>
<th>meanIOU</th>
<th>Per-pixel accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle (Train and test on Real)</td>
<td>60.3</td>
<td>93.1</td>
</tr>
<tr>
<td>Train on CG, test on Real</td>
<td>17.9</td>
<td>54.0</td>
</tr>
<tr>
<td>Train on CycleGAN, test on Real</td>
<td>34.8</td>
<td>82.8</td>
</tr>
</tbody>
</table>
Idea #2: train on randomly perturbed data, so that test set just looks like another random perturbation.

This is called domain randomization or data augmentation.
Data augmentation

Training data

\[ \{ x \} \rightarrow \{ y \} \]

- Fish
- Chameleon
- Grizzly

\[ \{ \} \rightarrow \{ \} \]

- Mirror
- Crop
- Darken
Domain randomization

Training data

Test data

[Sadeghi & Levine 2016]
Above example is from [Tobin et al. 2017]
What if we go waaaaay outside of the training distribution?
Our training data did cover the part of the distribution that was tested (biased data)
Out here, model response is highly unpredictable.
“Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images”
[Nguyen, Yosinski, and Clune, CVPR 2015]
“Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images”  
[Nguyen, Yosinski, and Clune, CVPR 2015]
Weirdness of high-dimensional space:

Usually, there are *blind spots* where the model has not fit the distribution well, and behaves unpredictably.
Adversarial noise

\[ \begin{align*}
\text{arg max } r & \; p(y = \text{ostrich}|x + r) \; \text{ subject to } \; \|r\| < \epsilon \\
\end{align*} \]

Anything to worry about?

“NO Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles”, Lu et al. 2017

(Early) 2017’s attacks fail on physical objects, since they are optimized to attack a single view!
Anything to worry about?

Later in 2017…

“Synthesizing Robust Adversarial Examples”, Athalye, Engstrom, Ilyas, Kwok, 2017

3D-printed turtle model classified as rifle from most viewpoints
Anything to worry about?

- Current deep models have bad **worst-case performance**
- Can be exploited by an adversary
- Few guarantees, can’t fully trust what the models output
Towards ML You Can Rely On

Prof. Aleksander Madry
madry-lab.ml
6.883

Today: Symposium on Robust, Interpretable Deep Learning Systems
2:30-6:30pm, 46-3002
Anything else to worry about?

- Our datasets are often poorly labeled
- And usually biased (overrepresent certain categories)
- ML method perform beautifully on laboratory data, but often generalize poorly to real-world data
- Can have negative social consequences