Garbage In, Garbage Out

How purportedly great ML models can be screwed up by bad data
What I’ll show...

1. Model accuracy claimed by security ML researchers is misleading

2. It’s generally biased in an overly optimistic direction

3. → Estimating the severity of that bias is important, and will help you make sure that your model isn’t… garbage.
Machine Learning

\[ f(\text{input}) = \text{output} \]

\[ f(\text{http://www.trustus.evil.ru/paypal/login/}) = .944780 \]
\[ f(\text{https://www.facebook.com/}) = .019367 \]
Machine Learning

\[ f(\text{input}) = \]

- http://gsbyntwqmem.mrjz5viern.ru/start_page.exe
- https://www.facebook.com/
- http://imgur.com/r/cats/omgn4Zv
TRAINING
(Supervised) Machine Learning

Training Data
- http://www.evil.ru/paypal/login/, 1
- http://gsbynr.ru/start_page.exe, 1
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

Test Data
- http://www.evil.ru/paypal/login/, 1
- http://git.demo.nick.net.nz/, 0
(Supervised) Machine Learning

Training Data
- http://www.anonymous.com/1, 1
- http://socialmedia.page.com, 1
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

Test Data
- http://evil.ru/paypal/login/, 1
- http://gsbynr.ru/start_page.exe, 1
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

"fitted" model $f$
TEST accuracy
Training accuracy
Test accuracy
Training accuracy
Training accuracy

Test accuracy

← Supposed to represent “real world” accuracy
(Supervised) Machine Learning

Training Data
- http://www.evil.ru/paypal/login/, 1
- http://gsbynr.ru/start_page.exe, 1
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

Test Data
- http://www.evil.ru/paypal/login/, 1
- http://git.demo.nick.net.nz/, 0

“fitted” model $f$
(Supervised) Machine Learning

Test Data:
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

Training Data:
- http://www.evil.ru/paypal/login/, 1
- http://gsbynr.ru/start_page.exe, 1
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

"fitted" model $f$

Test Accuracy:
- .99 (error $\approx .01$)
- .01 (error $\approx .01$)
DEPLOYMENT accuracy
(Supervised) Machine Learning

Training Data
- http://www.evilsite.com/, 1
- http://truegoodsite.com, 0
- https://www.facebook.com/, 0
- http://imgur.com/r/cats/omgn4Zv, 0

Test Data
- dwprgbyfykriizylpqcltzx.biz/, 1
- http://git.demo.nick.net.nz/, 0

Deployment Data!
- ???.evil.com
- ???.good.com

“fitted” model $f$

Deployment Accuracy?
- ???, ???
<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lab</strong></td>
<td><img src="check.png" alt="Green Check" /></td>
<td><img src="check.png" alt="Green Check" /></td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td><img src="question.png" alt="Question Mark" /></td>
<td><img src="question.png" alt="Question Mark" /></td>
</tr>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
</tr>
<tr>
<td>-------</td>
<td>----------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Lab</strong></td>
<td><img src="check.png" alt="Green Check" /></td>
<td><img src="check.png" alt="Green Check" /></td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td><img src="x.png" alt="Red X" /></td>
<td><img src="x.png" alt="Red X" /></td>
</tr>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>---------</td>
</tr>
<tr>
<td>Lab</td>
<td><img src="checkmark.png" alt="Green Check" /></td>
<td><img src="checkmark.png" alt="Green Check" /></td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td><img src="x.png" alt="Red X" /></td>
<td><img src="x.png" alt="Red X" /></td>
</tr>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Lab</strong></td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
</tr>
<tr>
<td><strong>Deployment</strong></td>
<td>![X]</td>
<td>![X]</td>
</tr>
</tbody>
</table>
Train / Test “Sensitivity Analysis”: Identifying training data that leads to improved and consistent performance on new datasets

Train and test the same model across different datasets, and evaluate the results:

1. What training datasets generalize better to others?

2. How sensitive is a model’s accuracy to changes in test datasets?
Train / Test “Sensitivity Analysis”
Train / Test “Sensitivity Analysis”

1. Model Used
2. Accuracy Metric Used: AUC
3. Datasets Used
4. Results!
Train / Test “Sensitivity Analysis”

1. Model Used

2. Accuracy Metric Used: AUC

3. Datasets Used

4. Results!
A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys

Joshua Saxe, Konstantin Berlin
A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys

Joshua Saxe, Konstantin Berlin

URL Model

\[
f(\text{input}) =
\]

\[
\text{http://www.trustus.evil.ru/paypal/login/ } \quad .999583
\]

\[
\text{https://www.facebook.com/ } \quad .001491
\]
Train / Test “Sensitivity Analysis”

1. Model Used

2. Accuracy Metric Used: AUC

3. Datasets Used

4. Results!
AUC = “Area Under the [ROC] Curve”

ROC Curve - Example

- AUC=.999
- AUC=.98
- AUC=.92
- AUC=.77
- random (coin flip) - AUC=.5
Train / Test “Sensitivity Analysis”

1. Model Used

2. Accuracy Metric Used: AUC

3. Datasets Used

4. Results!
CommonCrawl & PhishTank

10 million URLs from January 2017*
≈ 20k malware samples

* plus pre-Jan ‘17 phishtank malicious URLs, due to lack of data

Sophos

10 million internal URLs from January 2017
≈ 4% malware

VirusTotal

10 million URLs from January 2017
≈ 4% malware
CommonCrawl & Phishtank

10 million URLs from January 2017*

≈ 20k malware samples

* plus pre-Jan ‘17 phishtank malicious URLs, due to lack of data

Sophos

10 million internal URLs from January 2017

≈ 4% malware

VirusTotal

10 million URLs from January 2017

≈ 4% malware
<table>
<thead>
<tr>
<th>Source</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonCrawl &amp; Phishtank</td>
<td>10 million URLs from January 2017* ≈ 20k malware samples</td>
</tr>
<tr>
<td>Sophos</td>
<td>10 million internal URLs from January 2017 ≈ 4% malware</td>
</tr>
<tr>
<td>VirusTotal</td>
<td>10 million URLs from January 2017 ≈ 4% malware</td>
</tr>
</tbody>
</table>

* plus pre-Jan ‘17 phishtank malicious URLs, due to lack of data
VirusTotal
10 million URLs from January 2017
≈ 4% malware

CommonCrawl & Phishtank
10 million URLs from January 2017*
≈ 20k malware samples

Sophos
10 million internal URLs from January 2017
≈ 4% malware

* plus pre-Jan ‘17 phishtank malicious URLs, due to lack of data
CommonCrawl & PT model

Sophos model

VirusTotal model

(January ‘17 data)
Train / Test “Sensitivity Analysis”

1. Model Used

2. Accuracy Metric Used: AUC

3. Datasets Used

4. Results!
AUC

Test Data (Feb & March)

VirusTotal

0.7172

0.8204

0.9052

Sophos

0.5879

0.9993

CommonCrawl & PhishTank

0.985

0.7951

Training Data (January)

CommonCrawl & PhishTank

Sophos

VirusTotal
Sophos Model Tested On Sophos Test Data

![Graph showing AUC over time with dates from January to May 2017. The AUC values range from 0.9885 to 1.0000. The graph is labeled with 'Sophos URLs (trained on 10m, January, Sophos labels).']
What did we learn?

- Model accuracy is extremely dependent on the training and test datasets used
- Which datasets generalize better
- Expected variance in accuracy on new, inherently different data
What did we learn?

- Model accuracy is *extremely* dependent on the training and test datasets used

- Which datasets generalize better

- Expected variance in accuracy on new, inherently different data
What did we learn?

- Model accuracy is *extremely* dependent on the training and test datasets used

- Which datasets generalize better

- Expected variance in accuracy on new, inherently different data
How minimize the probability... of failing spectacularly

Models are liable to fail on different, future data.

 Especially when we lack deployment test data, we need to map the limitations of our models using train / test dataset sensitivity analyses.

 This technique can help us choose better training datasets and gain a better understanding of how sensitive model accuracy is to new test data distributions. This allows us to develop models that work in the real world, not just in idealized laboratory settings.
Thanks!