Garbage In, Garbage Out

How purportedly great ML models can be screwed up by bad data

Hillary Sanders
Data Scientist - operations team lead

SOPHOS

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

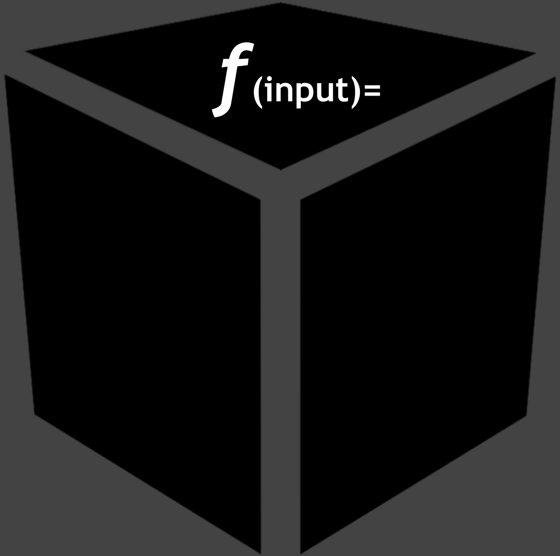
f(input) = output

f(http://www.trustus.evil.ru/paypal/login/) = .944780

f(https://www.facebook.com/) =.019367

Machine Learning

input →



→ output

http://www.trustus.evil.ru/paypal/login/ .944780

http://gsbyntwqmem.mrjz5viern.ru/start_page.exe .99981683

https://www.facebook.com/ .019367

http://imgur.com/r/cats/omgn4Zv .008448

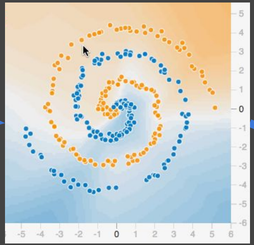
TRAINING

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

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

TRAINING

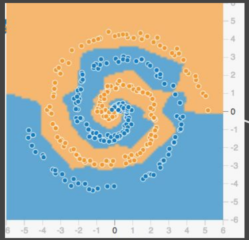
(Supervised) Machine Learning

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

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



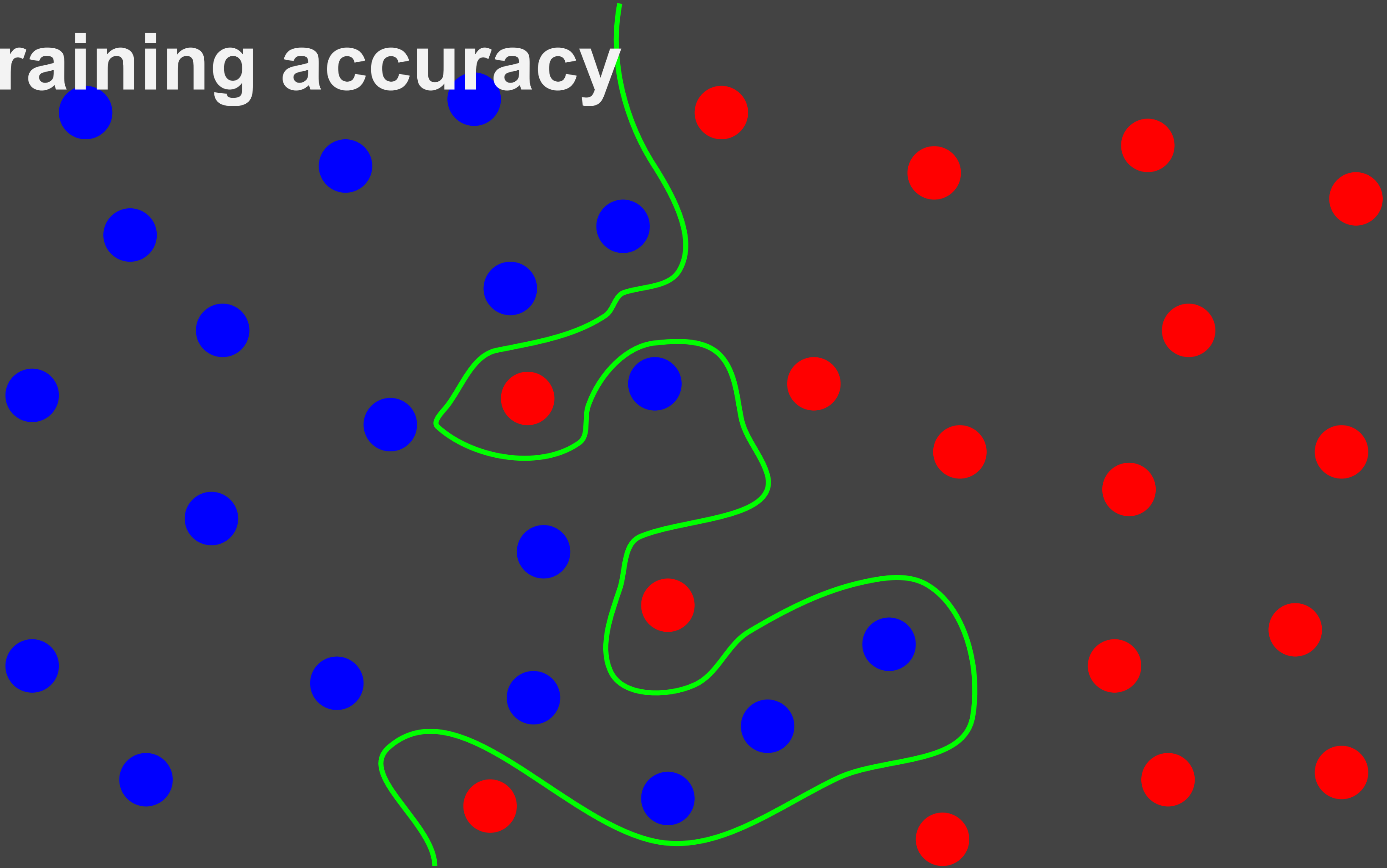
“fitted” model f

f (input)

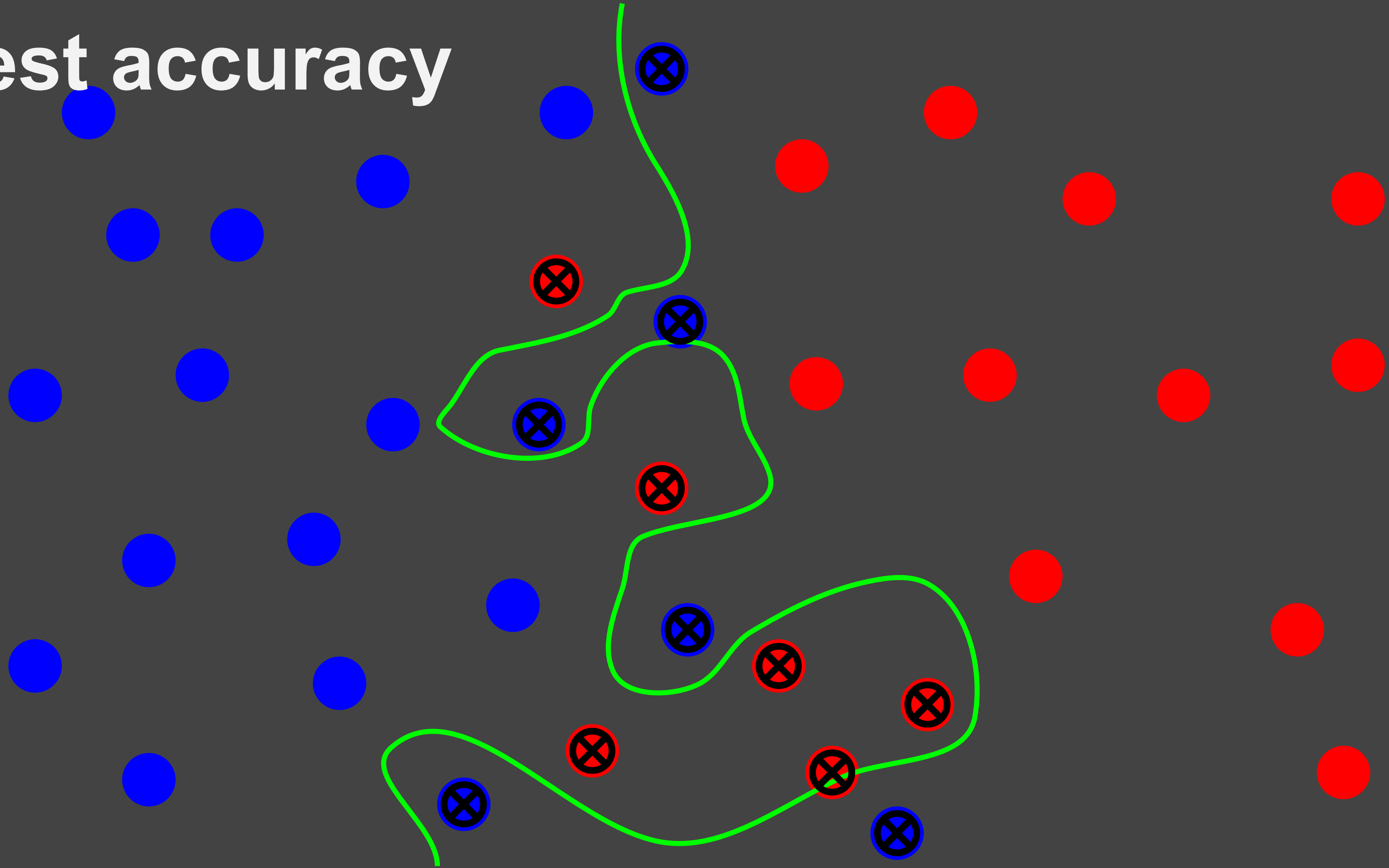
TEST

accuracy

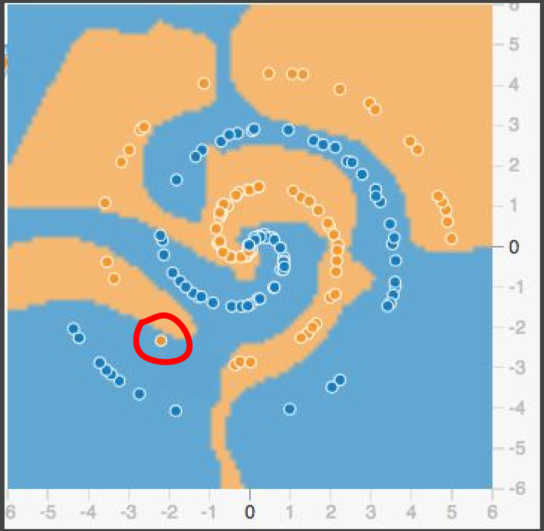
Training accuracy



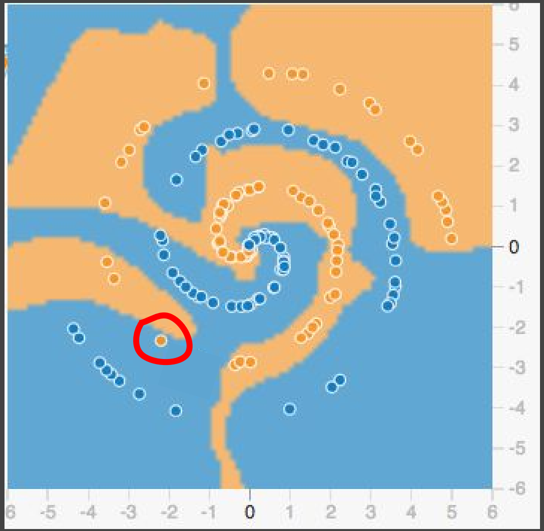
Test accuracy



Training accuracy

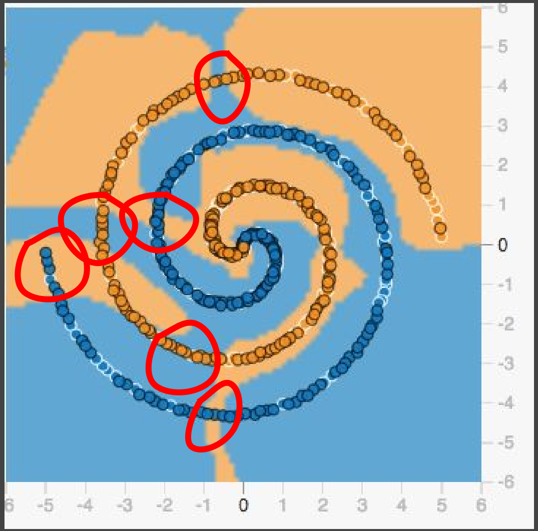


Training accuracy



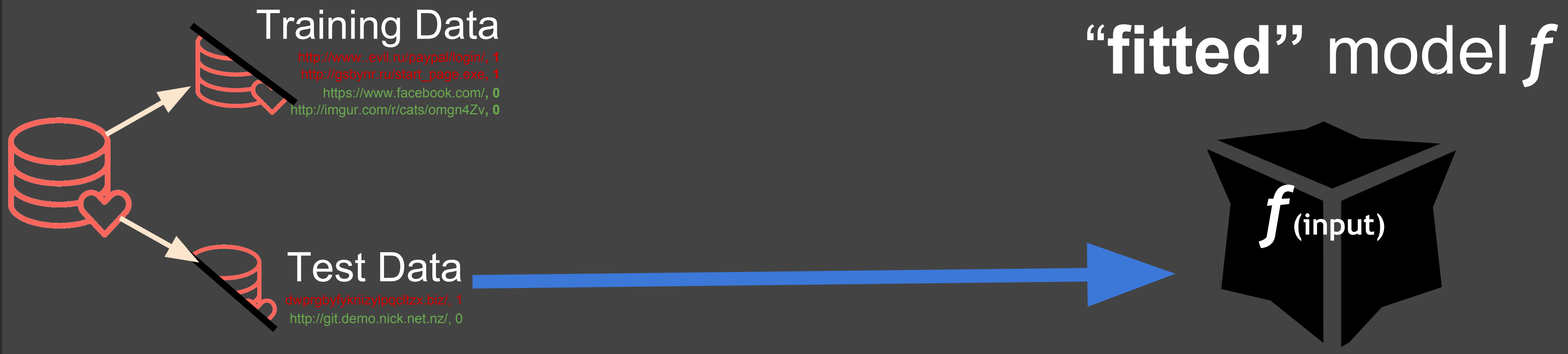
Test **accuracy**

← Supposed to represent "real world" accuracy

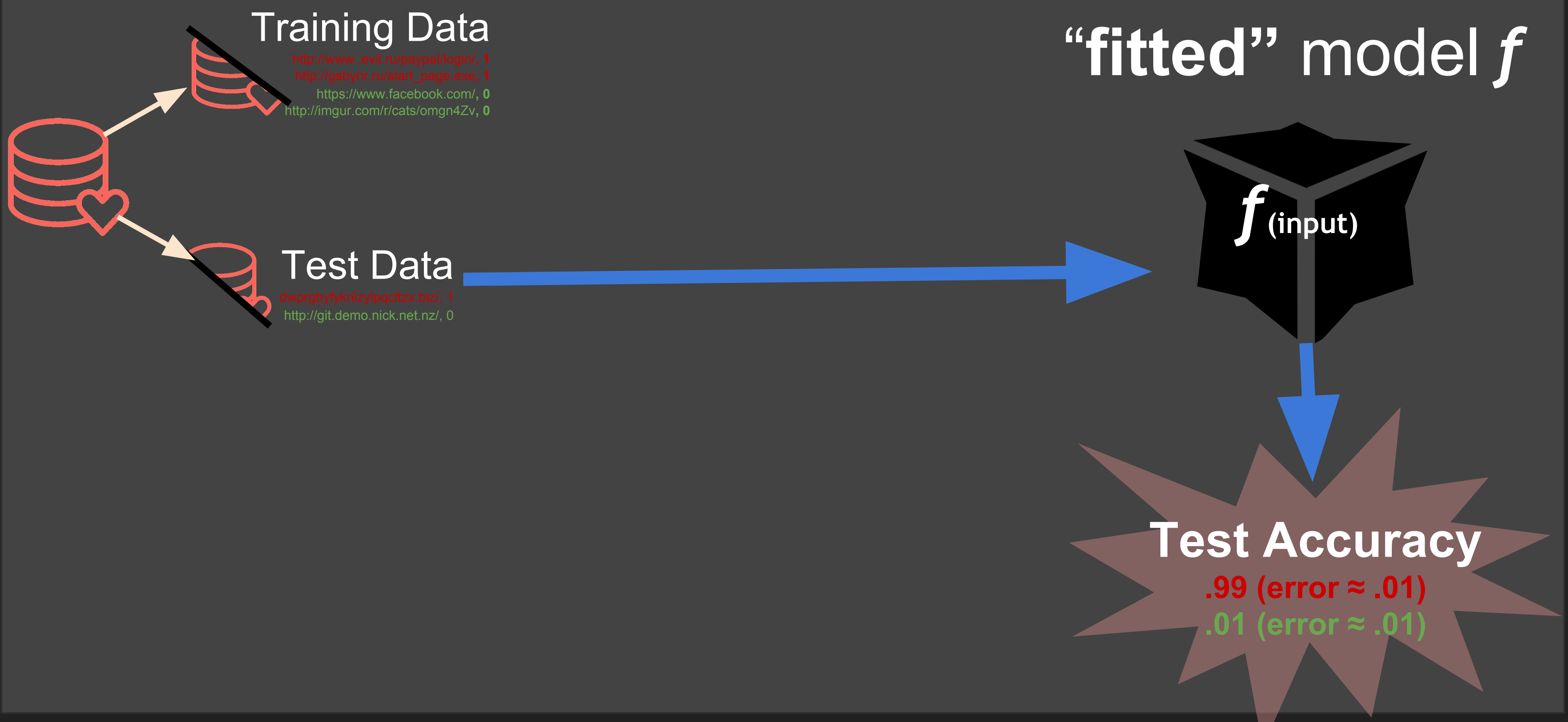


(Supervised) Machine Learning

TESTING



(Supervised) Machine Learning

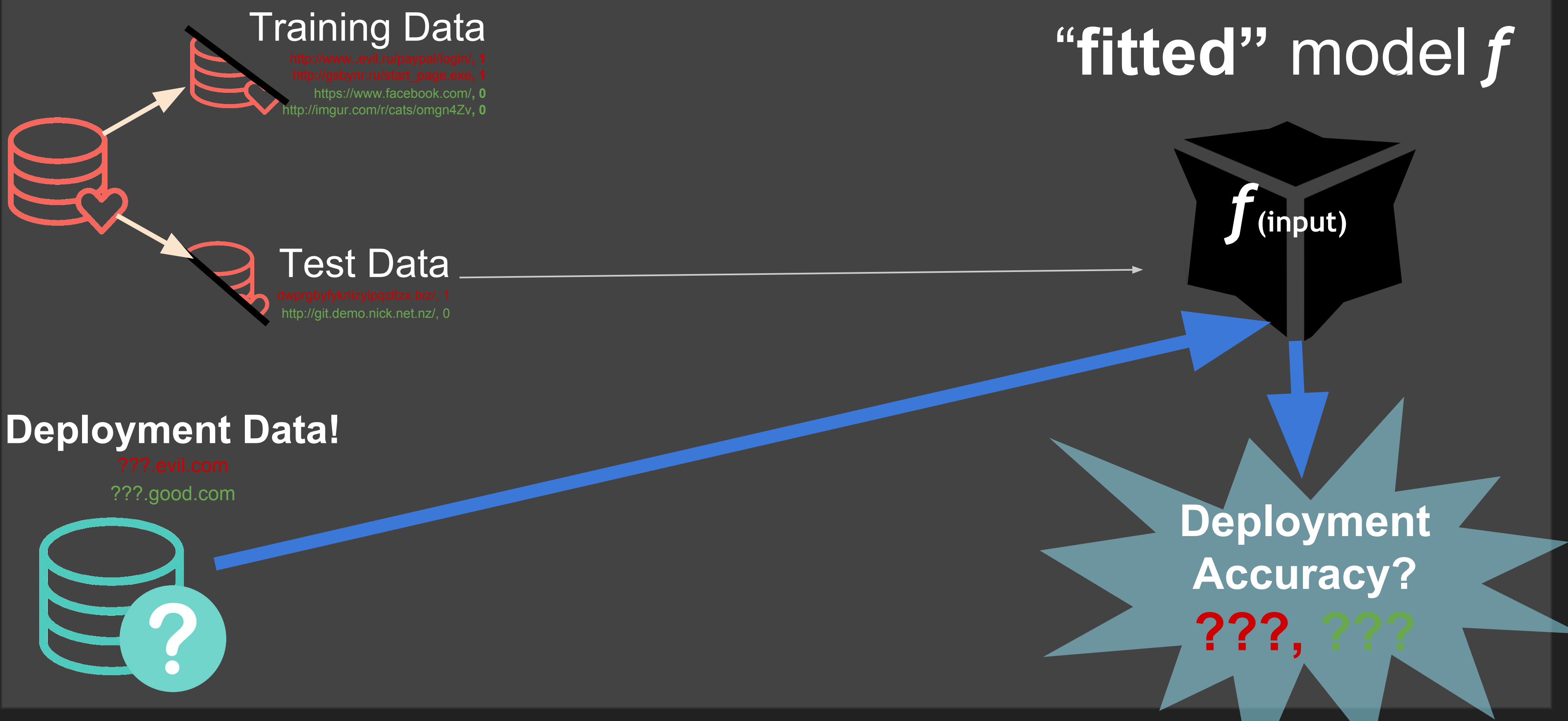


DEPLOYMENT

accuracy

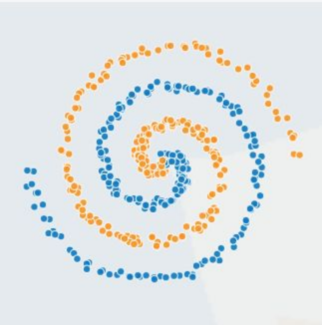
(Supervised) Machine Learning

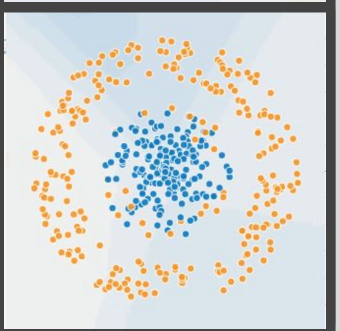
TRAINING



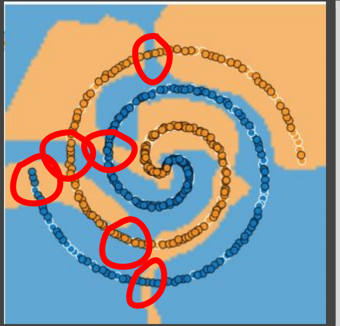
	Training	Testing
Lab	✓	✓
Deployment	?	?

	Training	Testing
Lab	✓	✓
Deployment	✗	✗

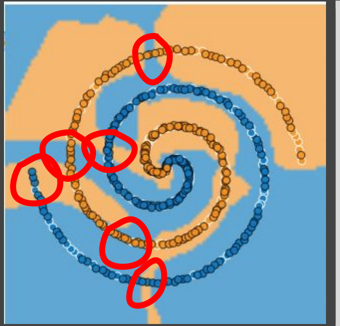
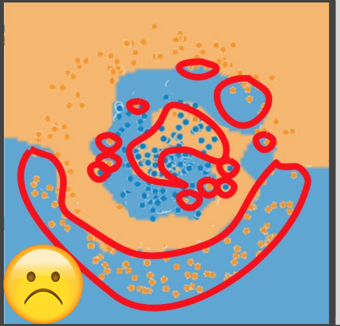




	Training	Testing
Lab	✓	✓
Deployment	✗	✗



	Training	Testing
Lab	✓	✓
Deployment	✗	✗

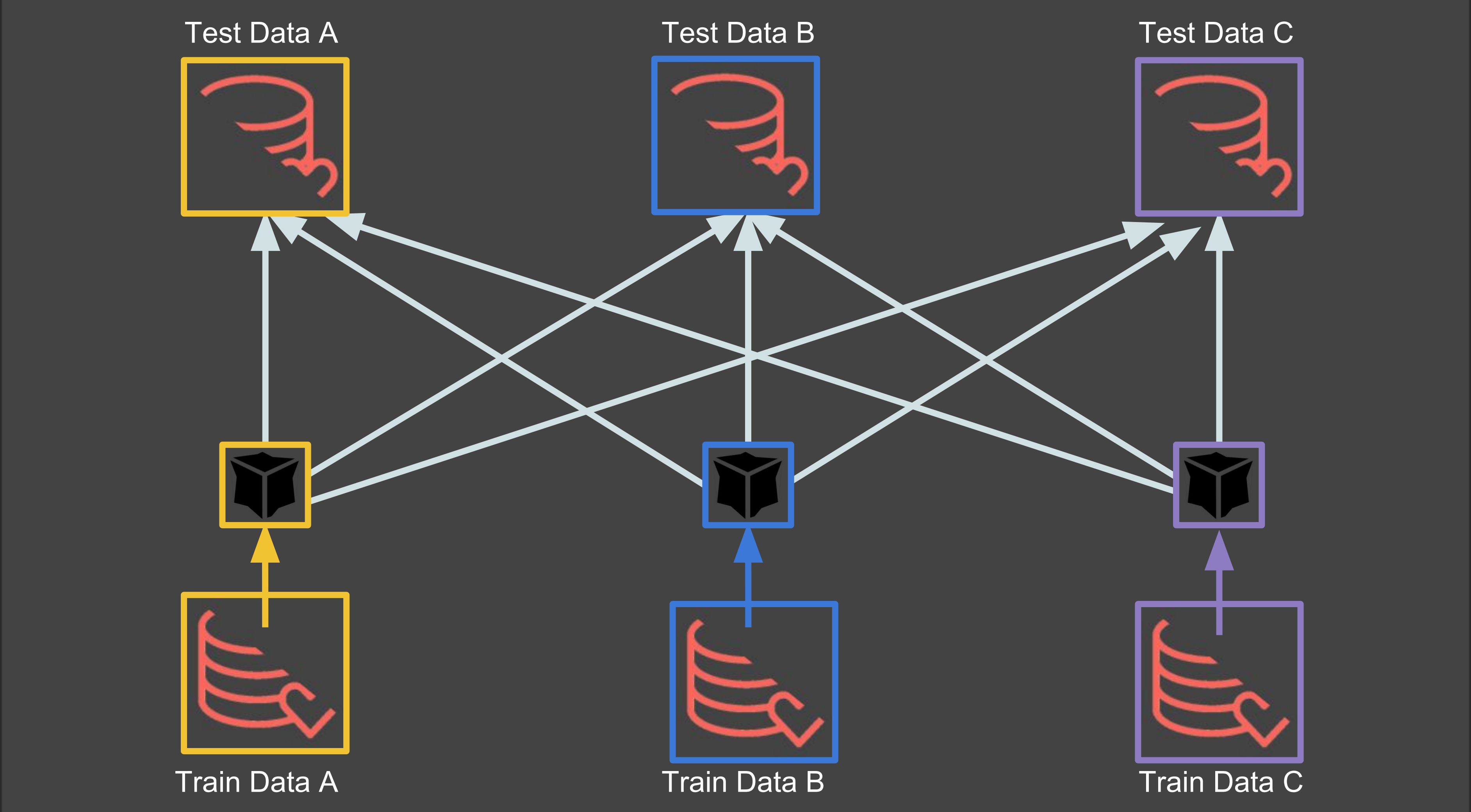
		Training	Testing
	Lab	✓	✓
	*Deployment*	✗	✗

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?


Test Data A
Test Data B
Test Data C
Train Data A
Train Data B
Train Data C

Train / Test "Sensitivity Analysis"

IRL!

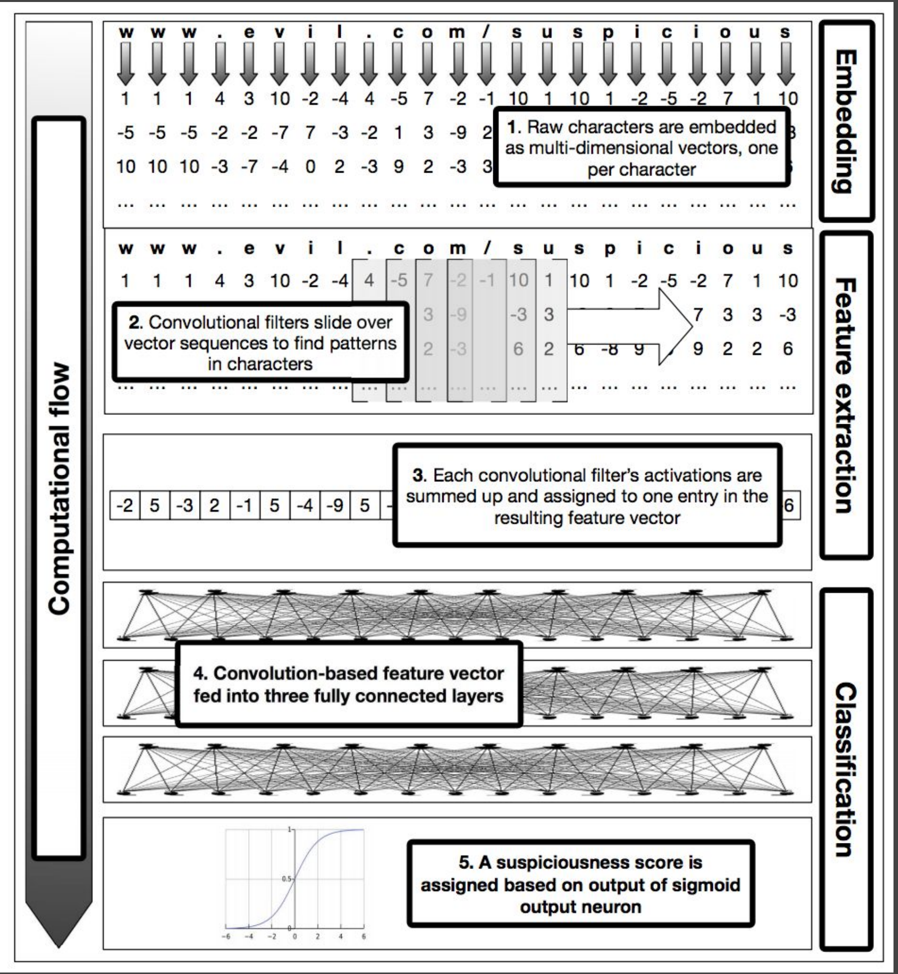
Train / Test “Sensitivity Analysis”

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!

URL Model



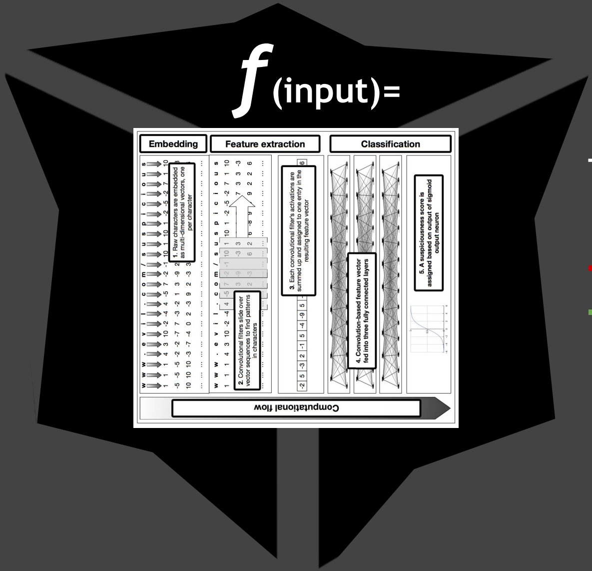
A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys

Joshua Saxe, Konstantin Berlin

URL Model

input →

http://www.trustus.evil.ru/paypal/login/
https://www.facebook.com/



→ output

.999583
.001491

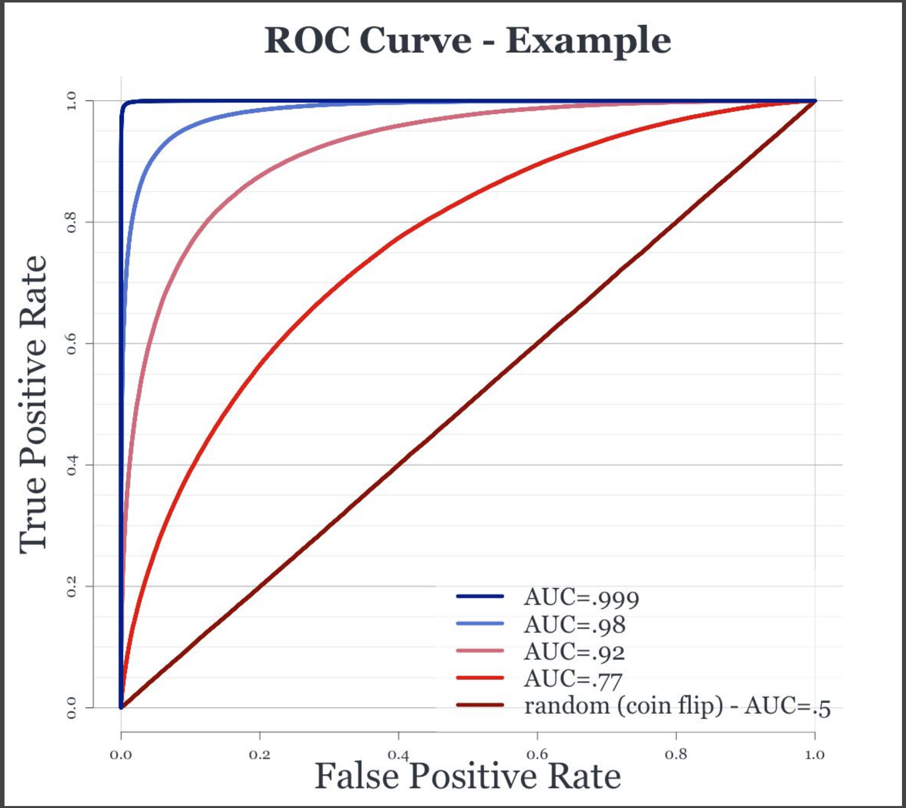
A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys

Joshua Saxe, Konstantin Berlin

Train / Test “Sensitivity Analysis”

1. Model Used
2. Accuracy Metric Used: AUC
3. Datasets Used
4. Results!

AUC = “Area Under the [ROC] Curve”



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

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

CommonCrawl & PT model

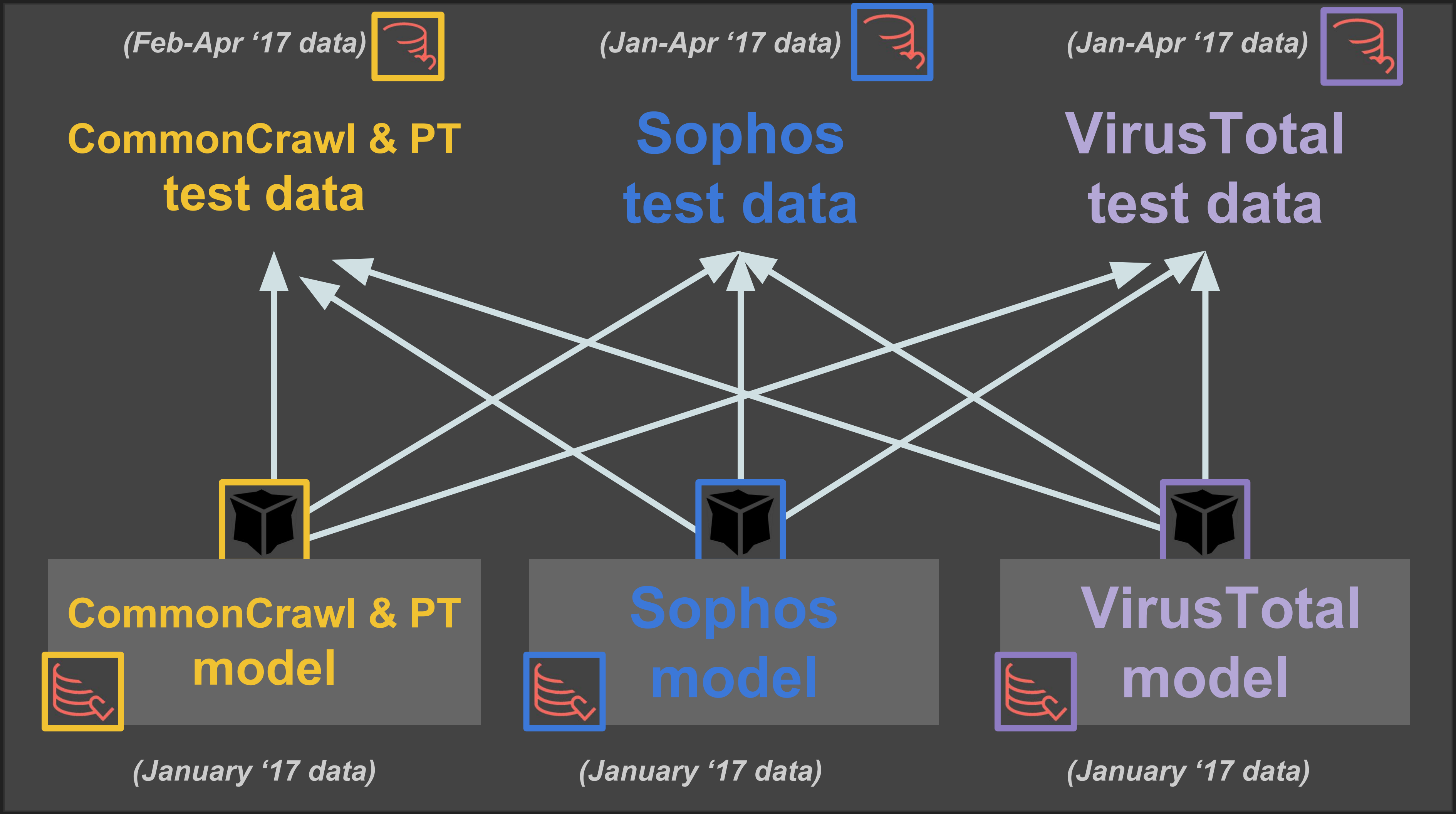
(January '17 data)

Sophos model

(January '17 data)

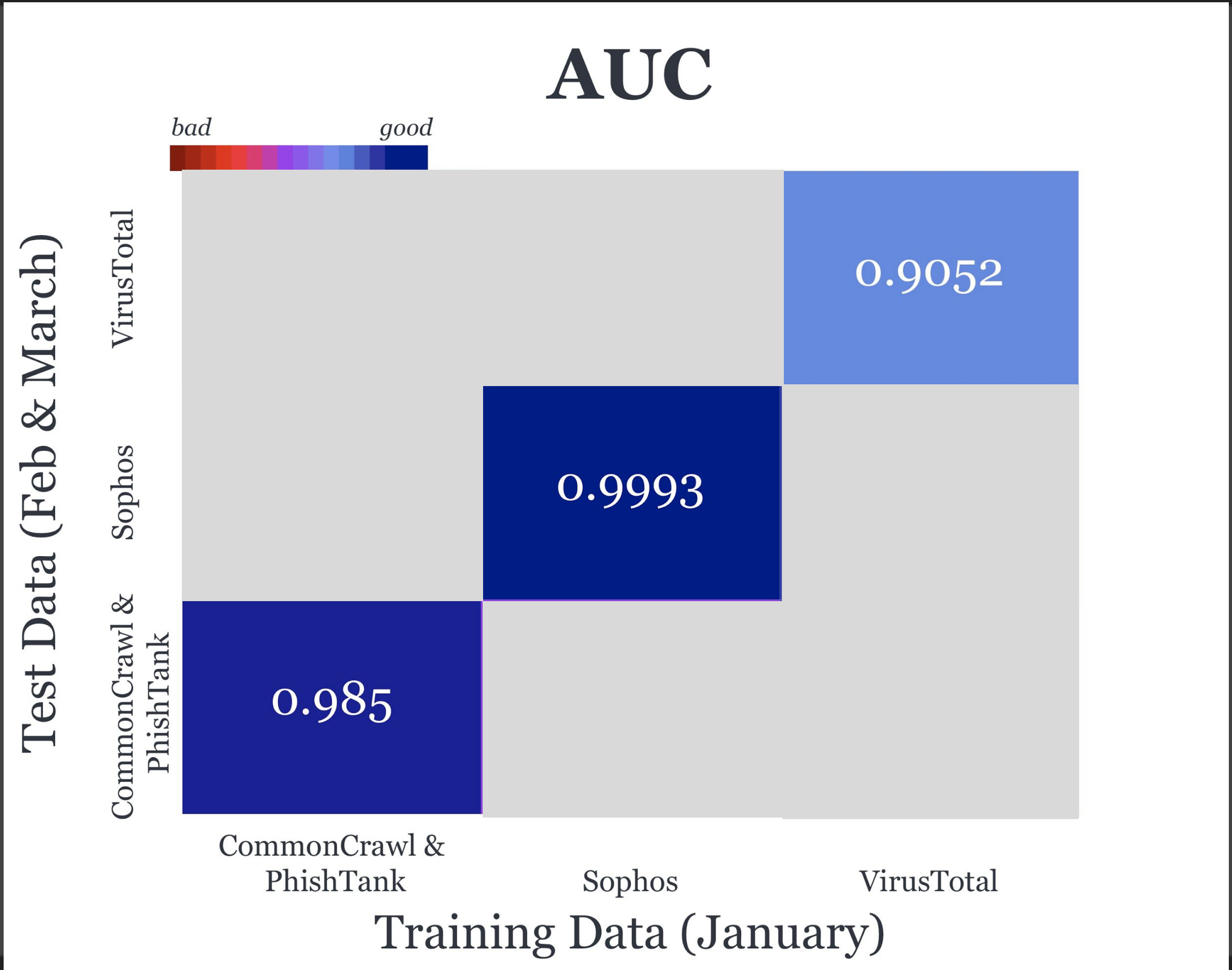
VirusTotal model

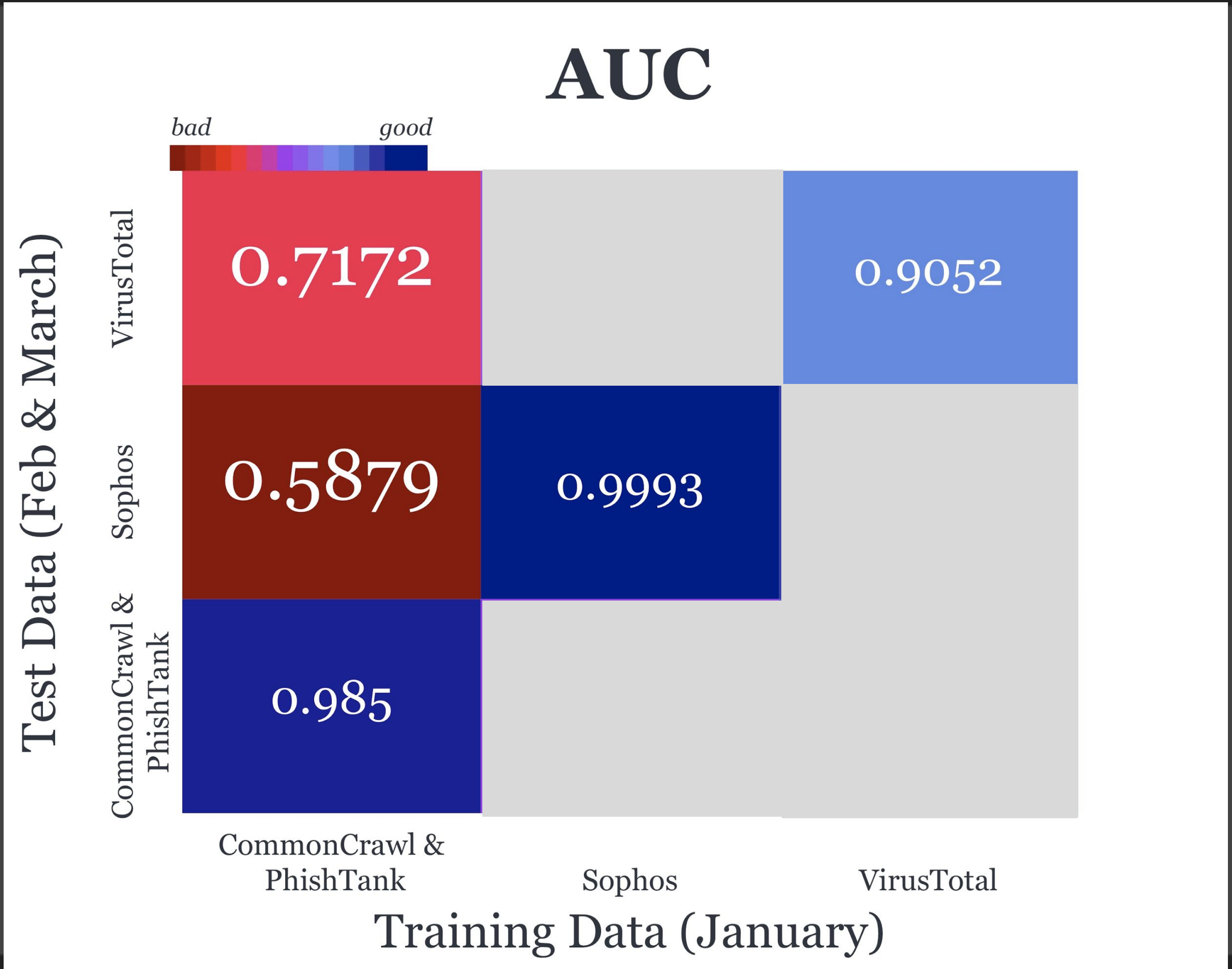
(January '17 data)


(Feb-Apr '17 data)
(Jan-Apr '17 data)
(Jan-Apr '17 data)
CommonCrawl & PT test data
Sophos test data
VirusTotal test data
CommonCrawl & PT model
Sophos model
VirusTotal model
(January '17 data)
(January '17 data)
(January '17 data)

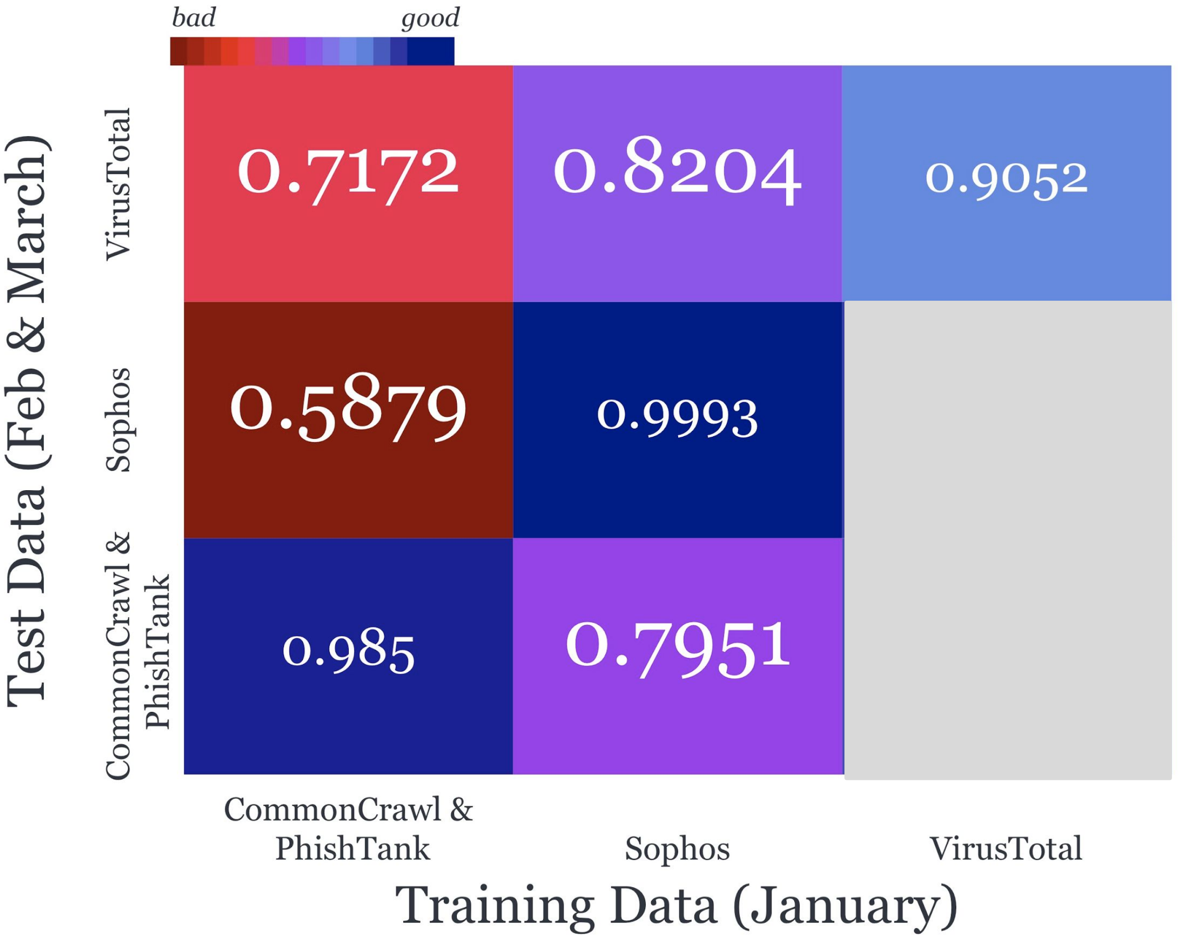
Train / Test “Sensitivity Analysis”

1. Model Used
2. Accuracy Metric Used: AUC
3. Datasets Used
4. Results!

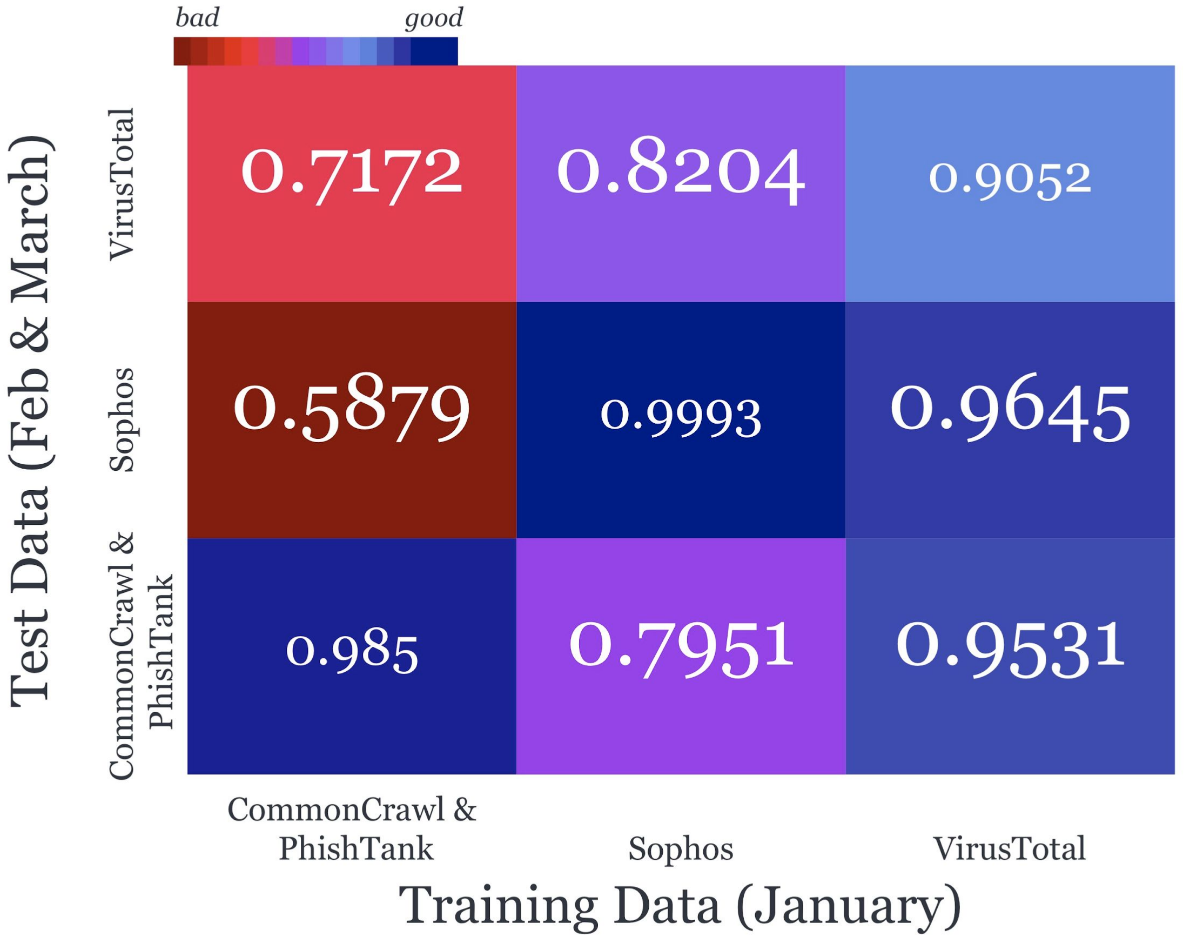

AUC
bad
good
Test Data (Feb & March)
VirusTotal
Sophos
CommonCrawl & PhishTank
0.9052
0.9993
0.985
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

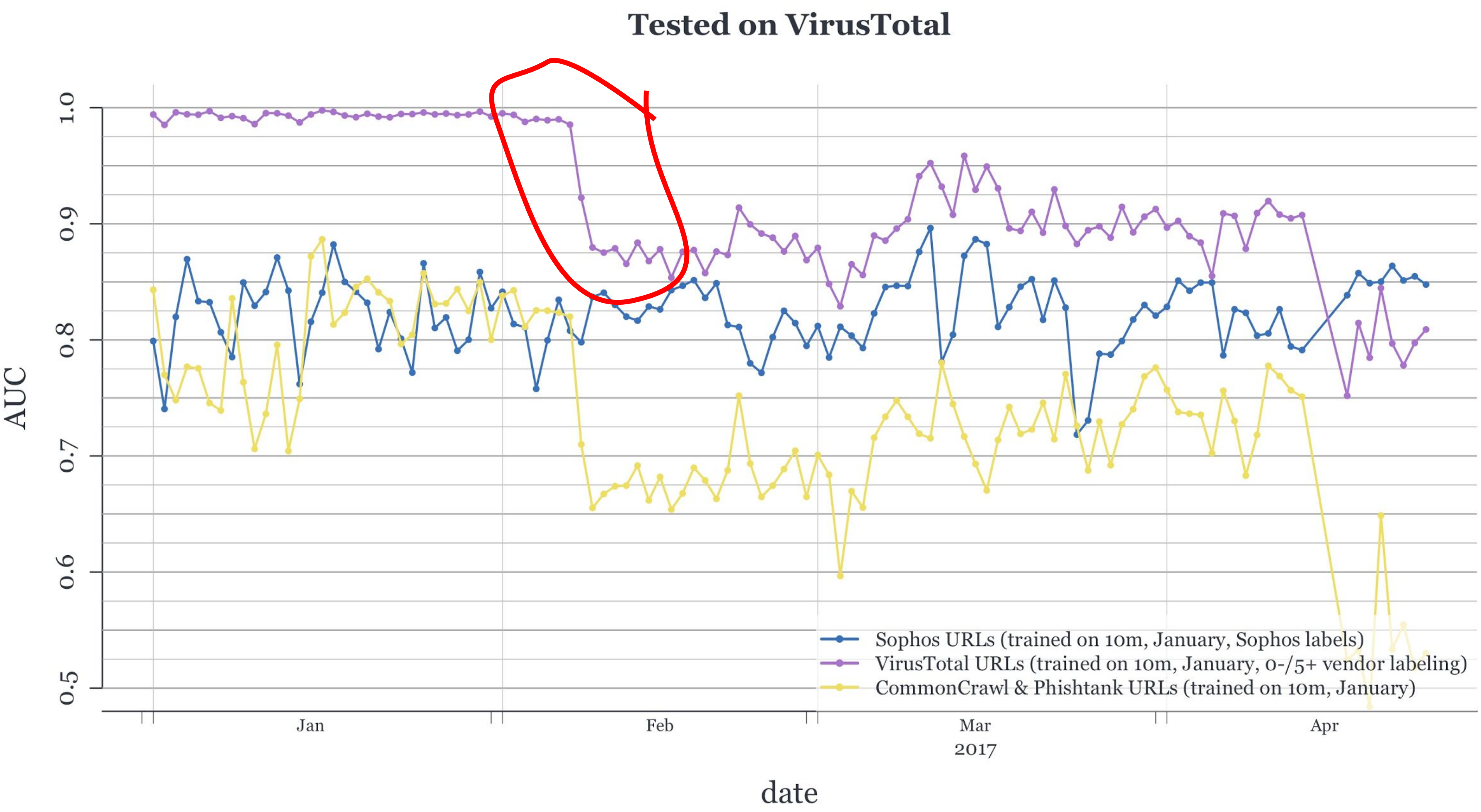

AUC
bad
good
Test Data (Feb & March)
VirusTotal
Sophos
CommonCrawl & PhishTank
0.7172
0.9052
0.5879
0.9993
0.985
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

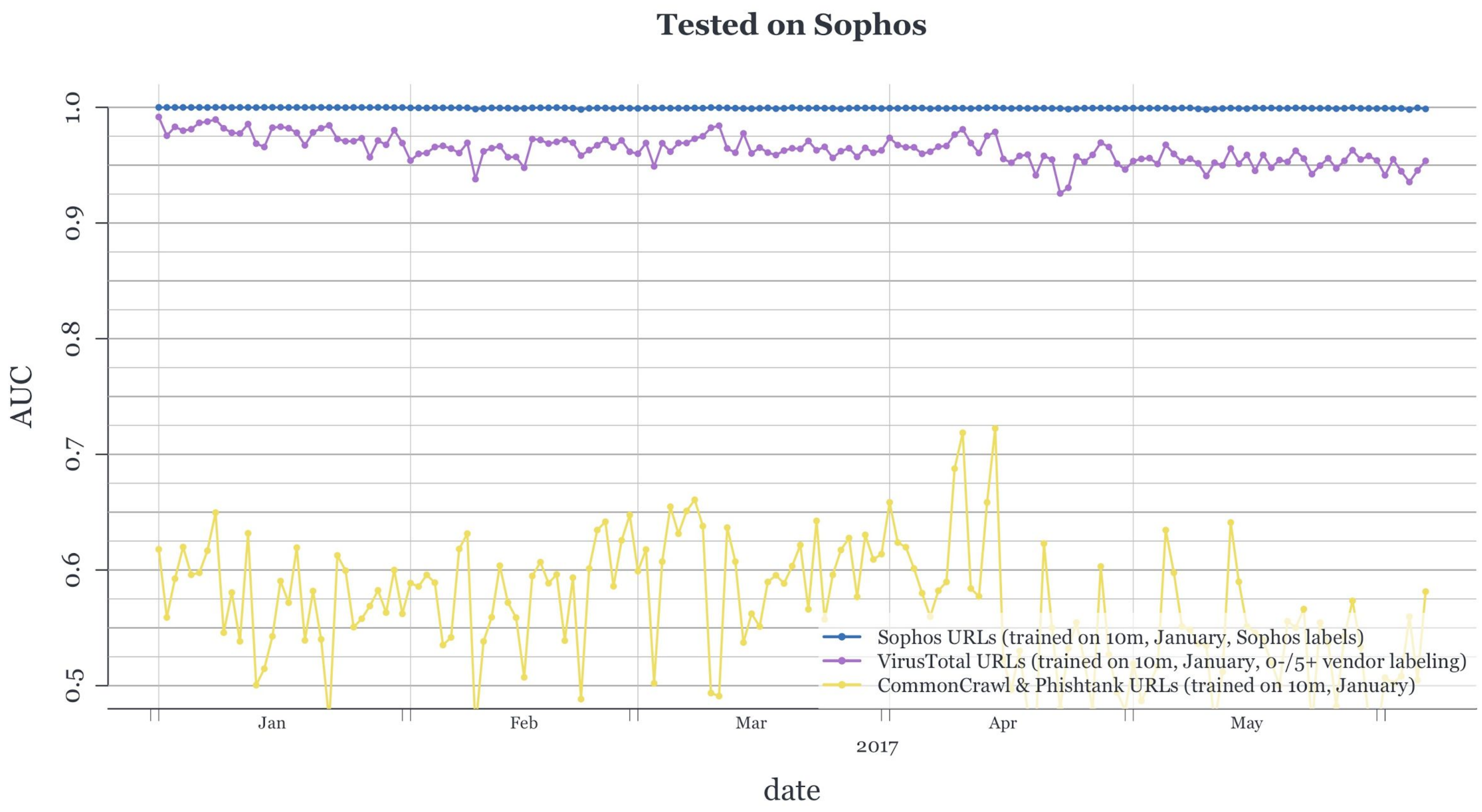
AUC

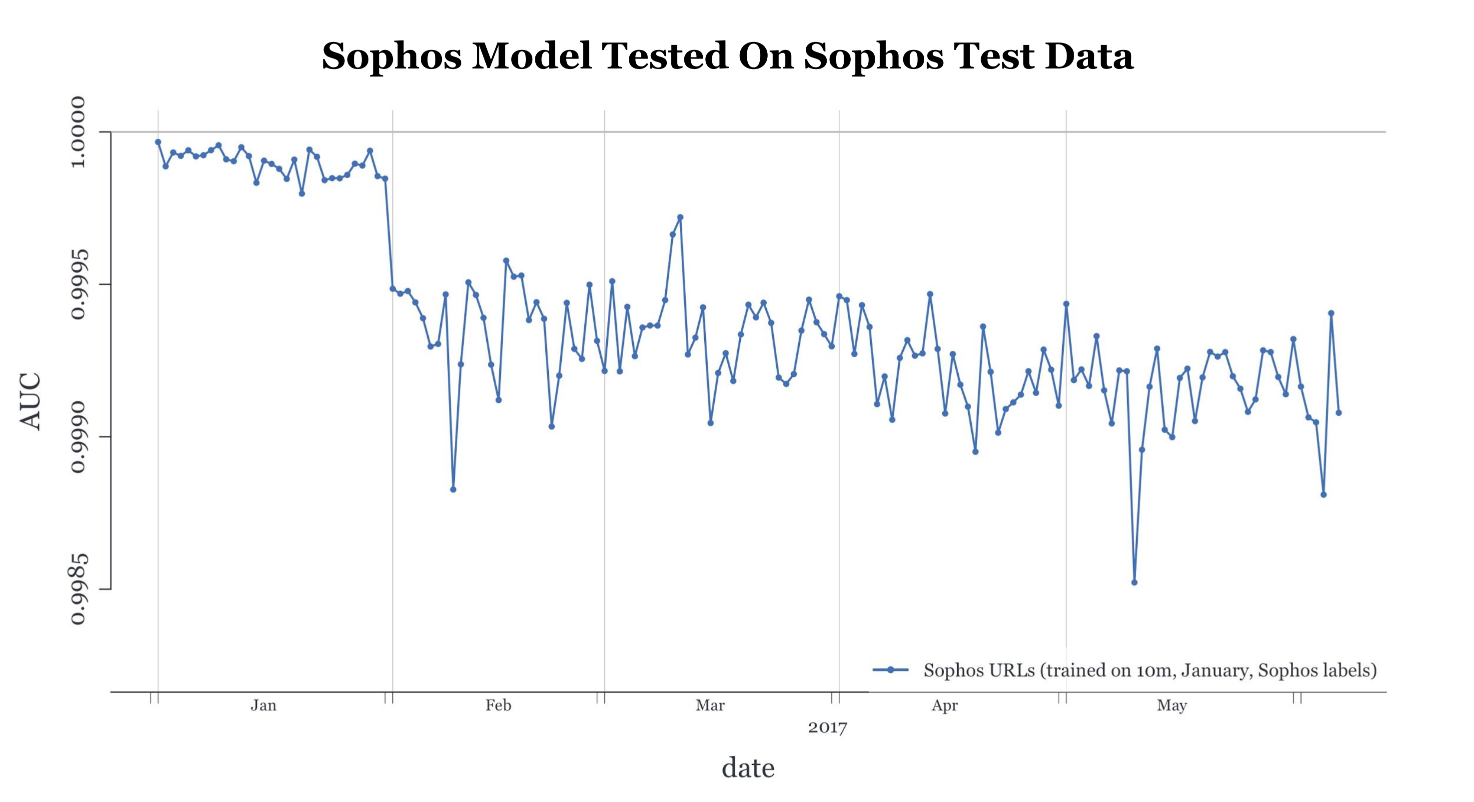

bad
good
0.7172
0.8204
0.9052
0.5879
0.9993
0.985
0.7951
VirusTotal
Sophos
CommonCrawl & PhishTank
Test Data (Feb & March)
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

AUC


bad
good
VirusTotal
Sophos
CommonCrawl & PhishTank
0.7172
0.8204
0.9052
0.5879
0.9993
0.9645
0.985
0.7951
0.9531
Test Data (Feb & March)
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)


Tested on VirusTotal
AUC
1.0
0.9
0.8
0.7
0.6
0.5
Jan
Feb
Mar
Apr
2017
date
Sophos URLs (trained on 10m, January, Sophos labels)
VirusTotal URLs (trained on 10m, January, 0-/5+ vendor labeling)
CommonCrawl & Phishtank URLs (trained on 10m, January)


Tested on Sophos
AUC
1.0
0.9
0.8
0.7
0.6
0.5
Jan
Feb
Mar
Apr
May
2017
date
Sophos URLs (trained on 10m, January, Sophos labels)
VirusTotal URLs (trained on 10m, January, 0-/5+ vendor labeling)
CommonCrawl & Phishtank URLs (trained on 10m, January)

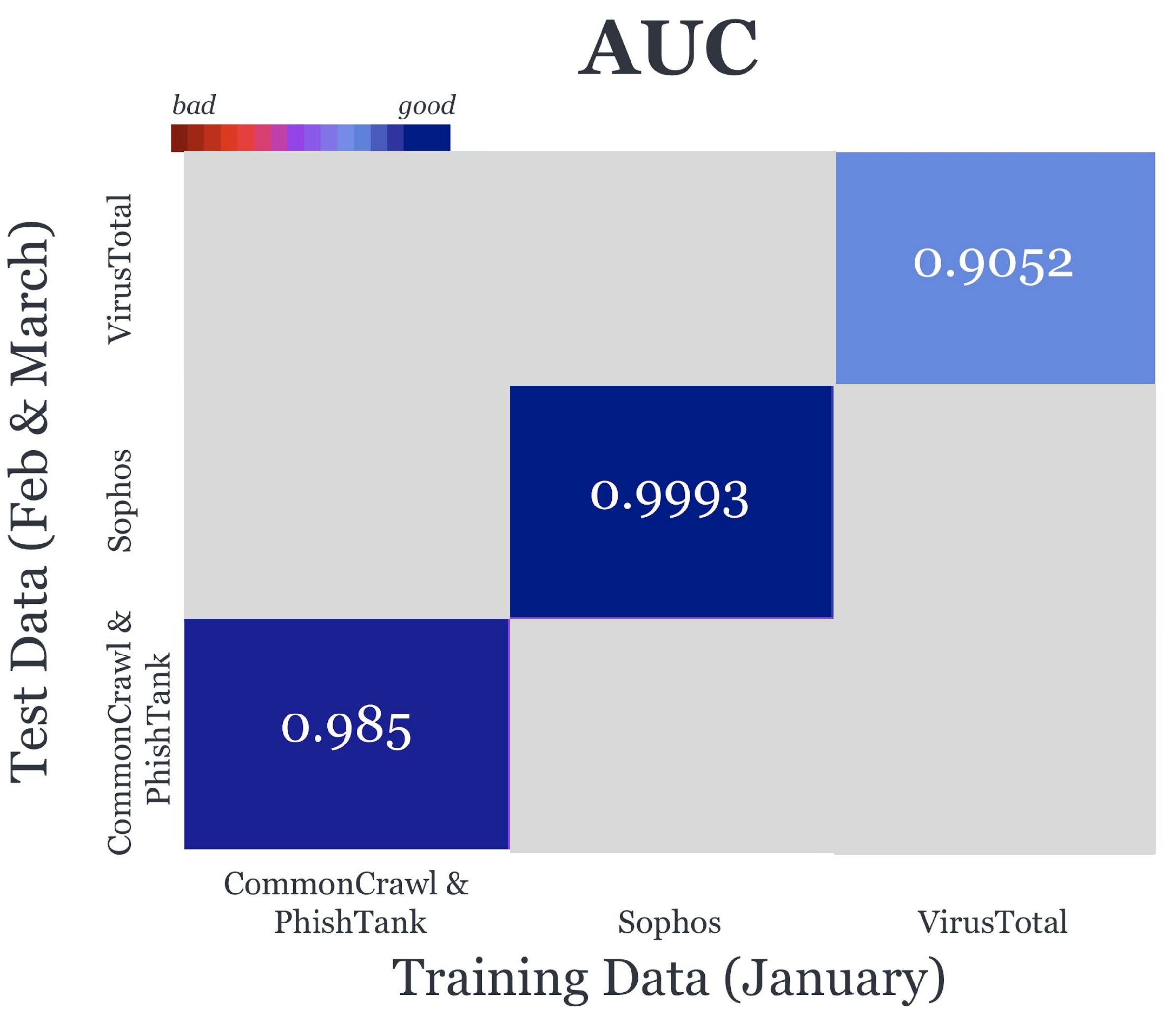

Sophos Model Tested On Sophos Test Data
AUC
1.0000
0.9995
0.9990
0.9985
Sophos URLs (trained on 10m, January, Sophos labels)
Jan
Feb
Mar
Apr
May
2017
date

What did we learn?

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

- Which datasets generalize better

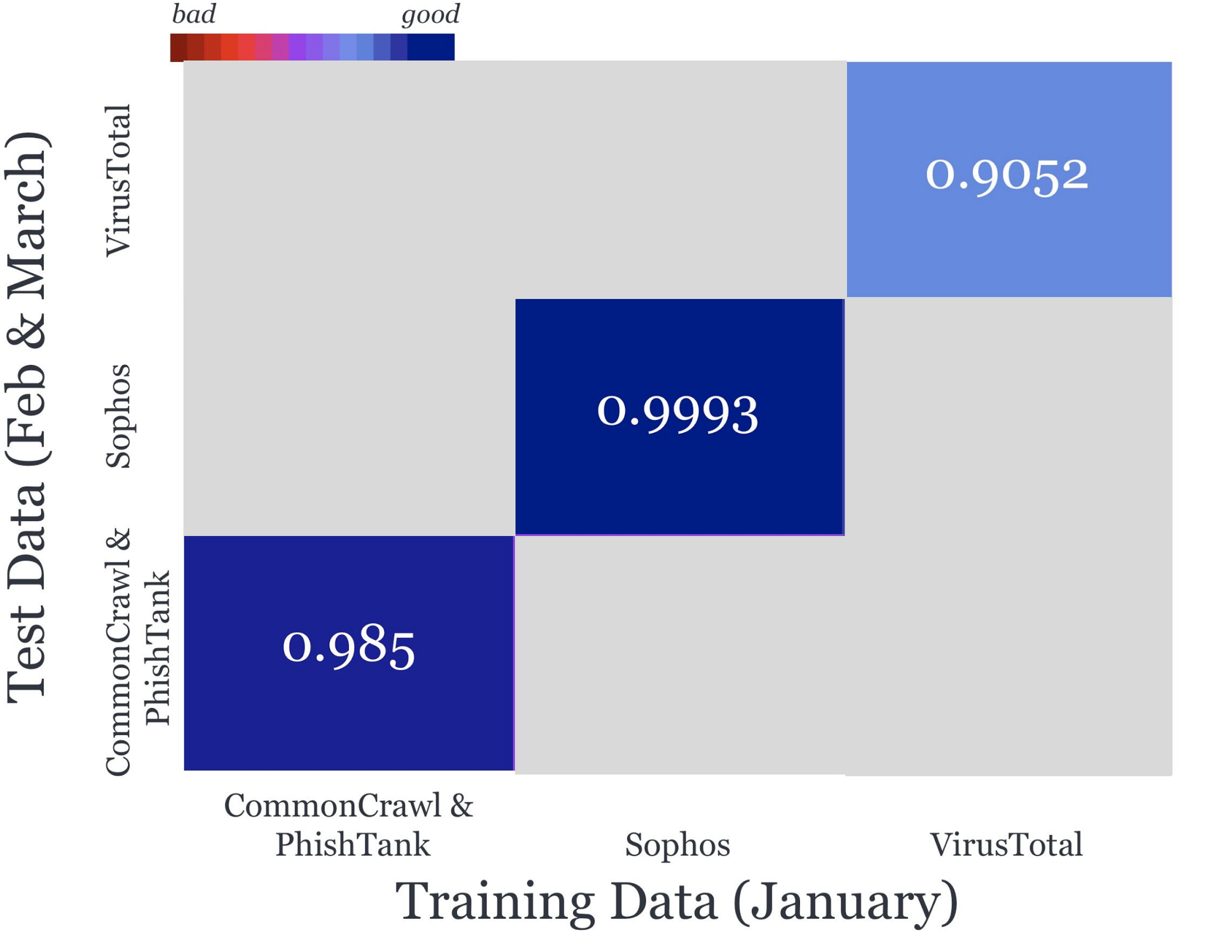
- Expected variance in accuracy on new, inherently different data


AUC
bad
good
Test Data (Feb & March)
VirusTotal
Sophos
CommonCrawl & PhishTank
0.9052
0.9993
0.985
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

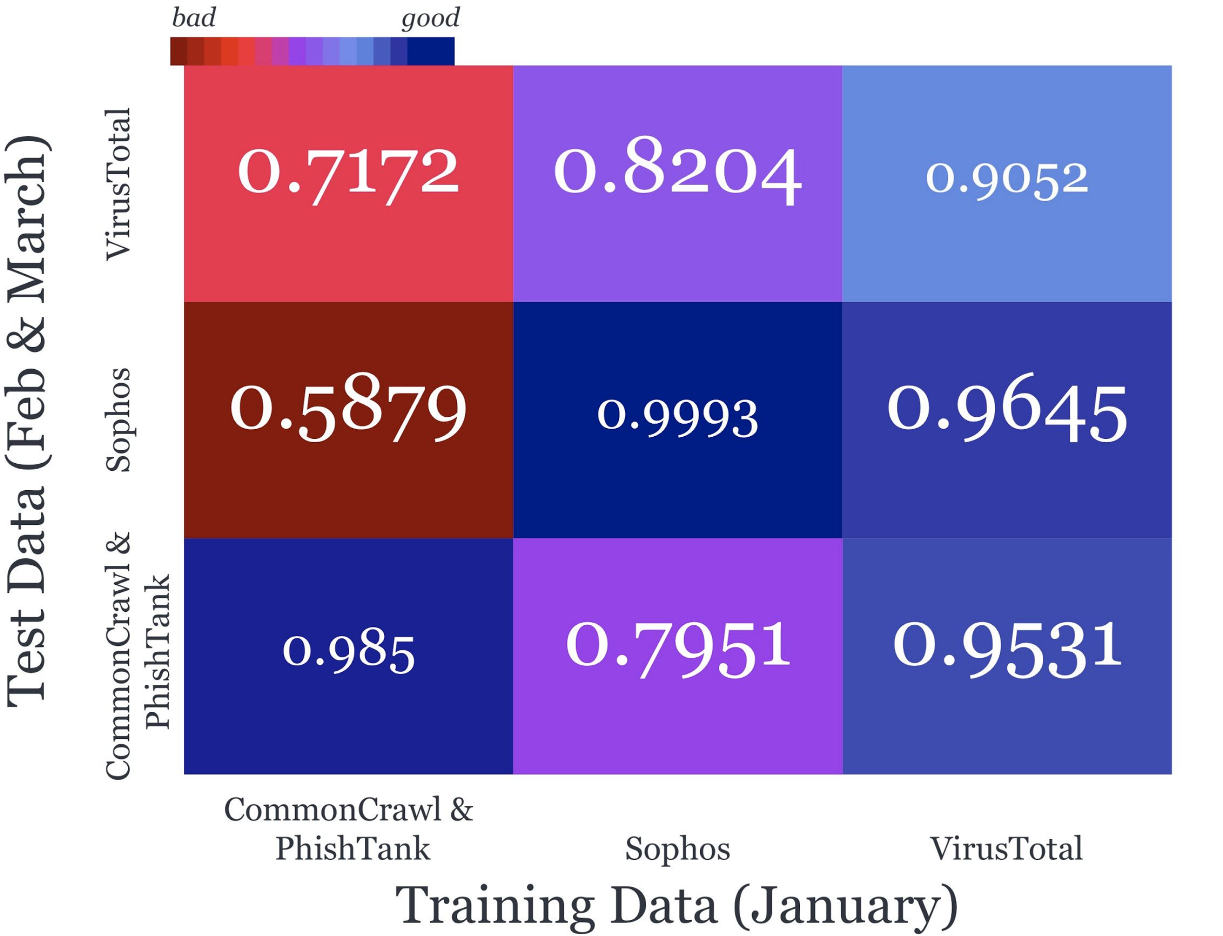
What did we learn?

- 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

AUC


bad
good
VirusTotal
Sophos
CommonCrawl & PhishTank
0.9052
0.9993
0.985
CommonCrawl & PhishTank
Sophos
VirusTotal
Test Data (Feb & March)
Training Data (January)

AUC


bad
good
Test Data (Feb & March)
VirusTotal
Sophos
CommonCrawl & PhishTank
0.7172
0.8204
0.9052
0.5879
0.9993
0.9645
0.985
0.7951
0.9531
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

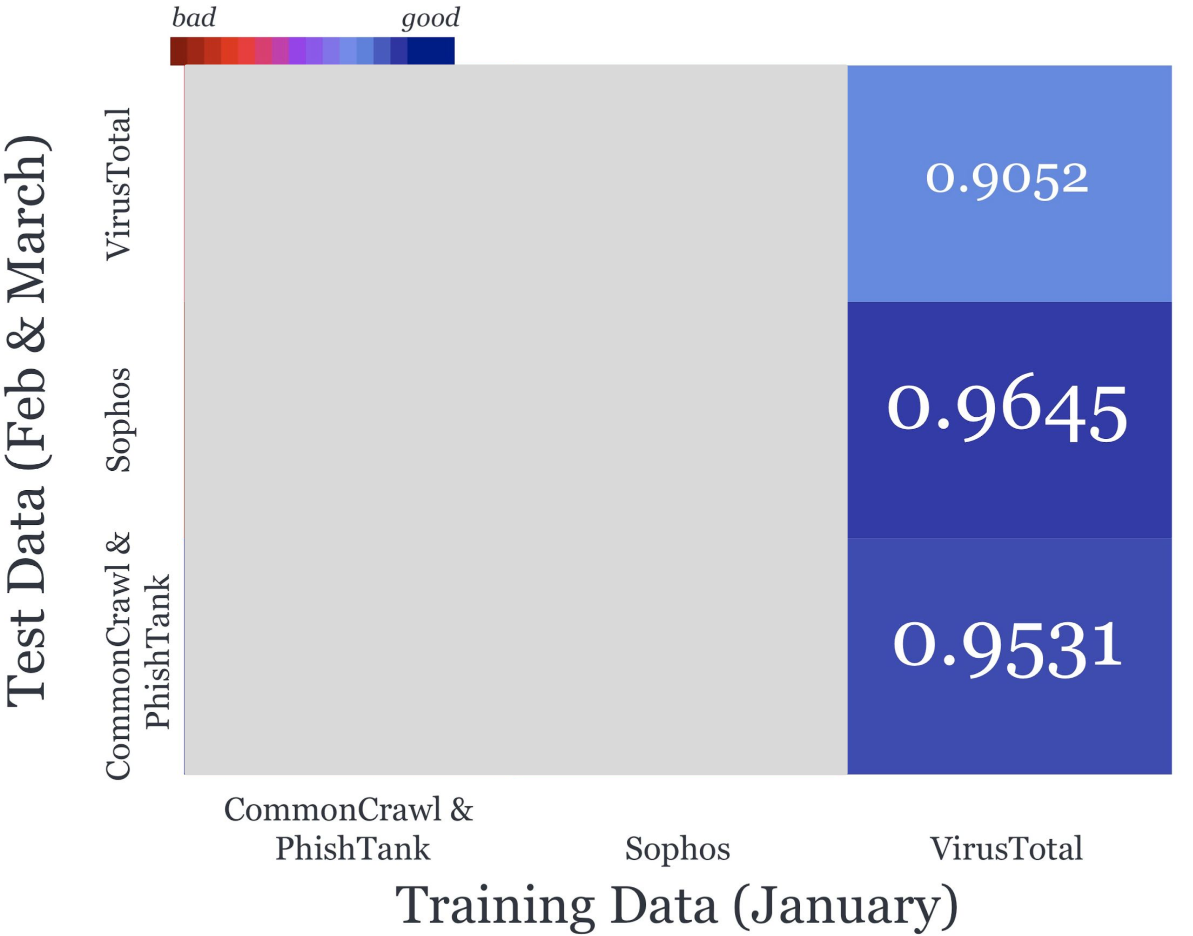
What did we learn?

- 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**

AUC


bad
good
VirusTotal
Sophos
CommonCrawl & PhishTank
Test Data (Feb & March)
0.9052
0.9645
0.9531
CommonCrawl & PhishTank
Sophos
VirusTotal
Training Data (January)

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!

SOPHOS