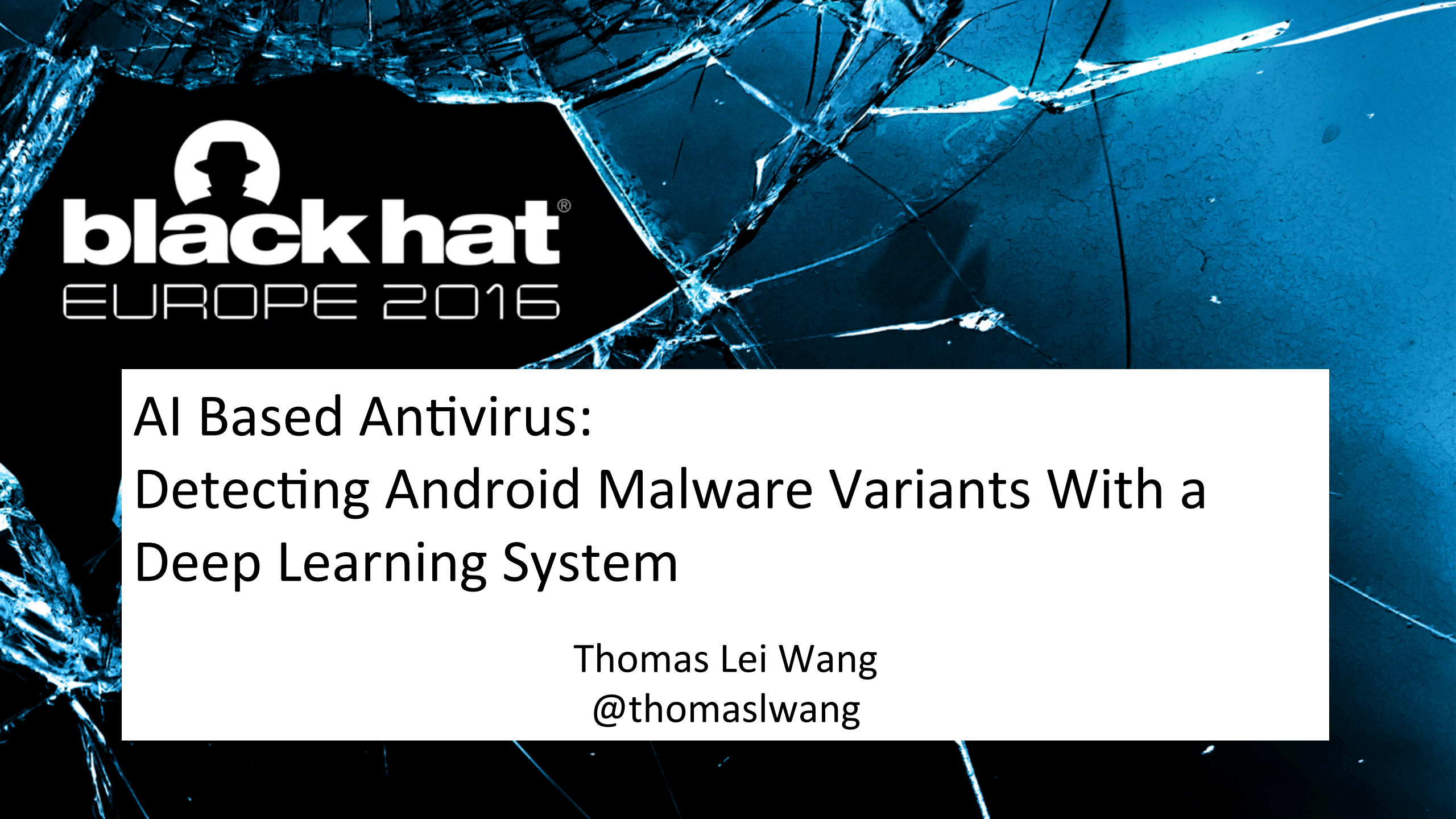


AI Based Antivirus: Detecting Android Malware Variants With a Deep Learning System

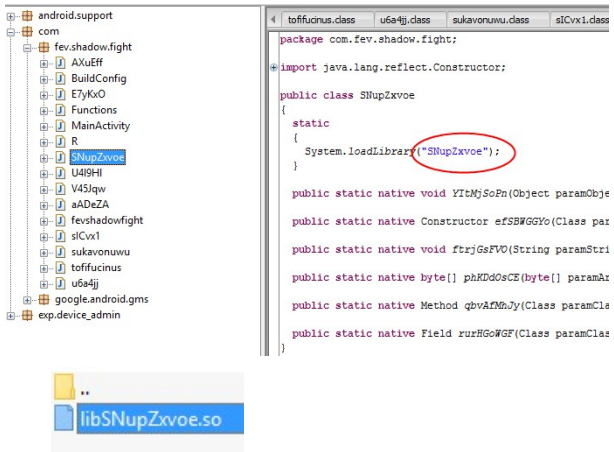
Thomas Lei Wang
@thomaslwang

About me

- My first (boring) job was a virus analyst in 2004.
- I had a dream…

Virus Analysis VS Image Recognition

```
<service android:name="lv.xpwhma.lqda"></service>
<activity android:theme="@android:0103000F" android:name="nckrf.das.umtjq"></ac
<activity android:theme="@android:0103000F" android:name="ntxrwg.jenu.gb" andrc
<service android:name="ntxrwg.jenu.kzmwu"></service>
<receiver android:name="fkc.vxfkc.vx.kjc"></receiver>
```



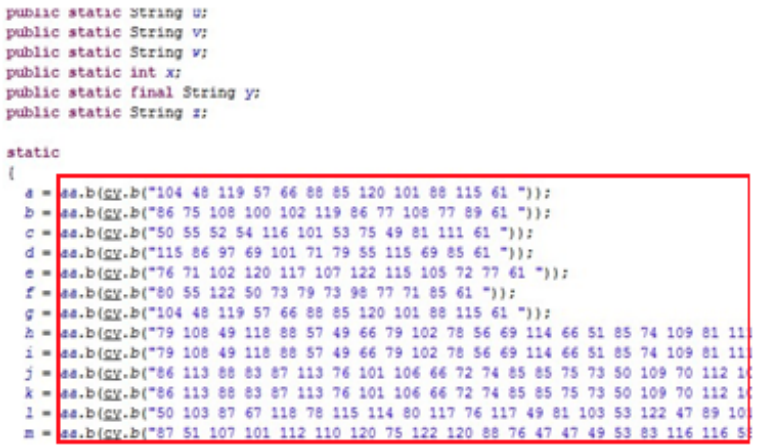
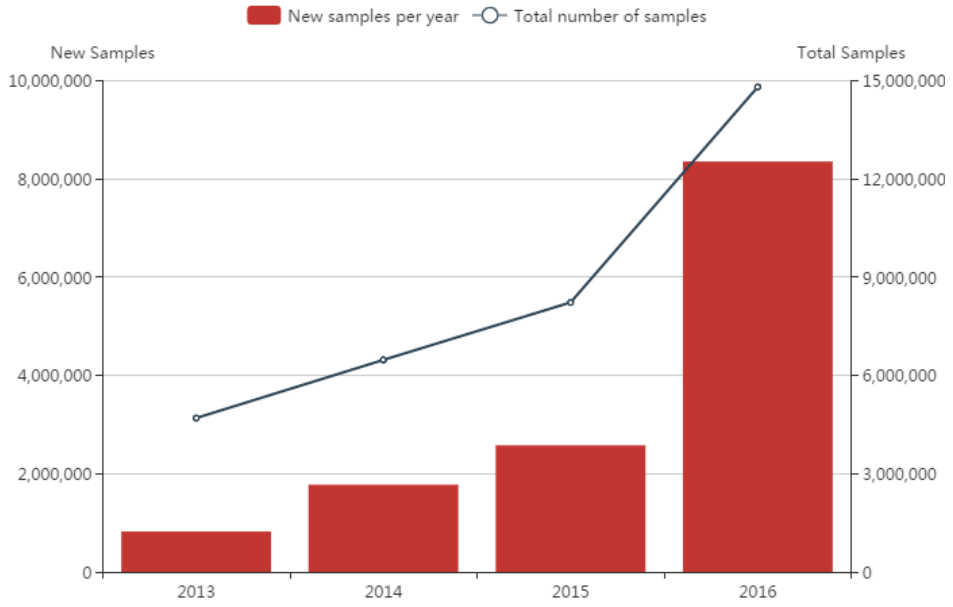


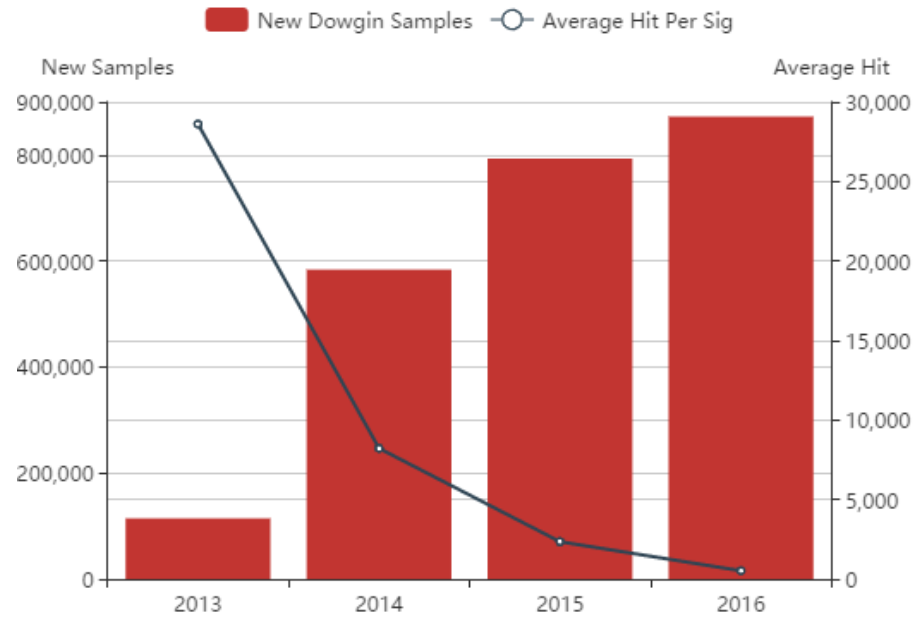
Image Provided by the MNIST handwritten database

Experienced virus analyst sometimes is doing image recognition!

Sample increase VS signature efficiency decrease



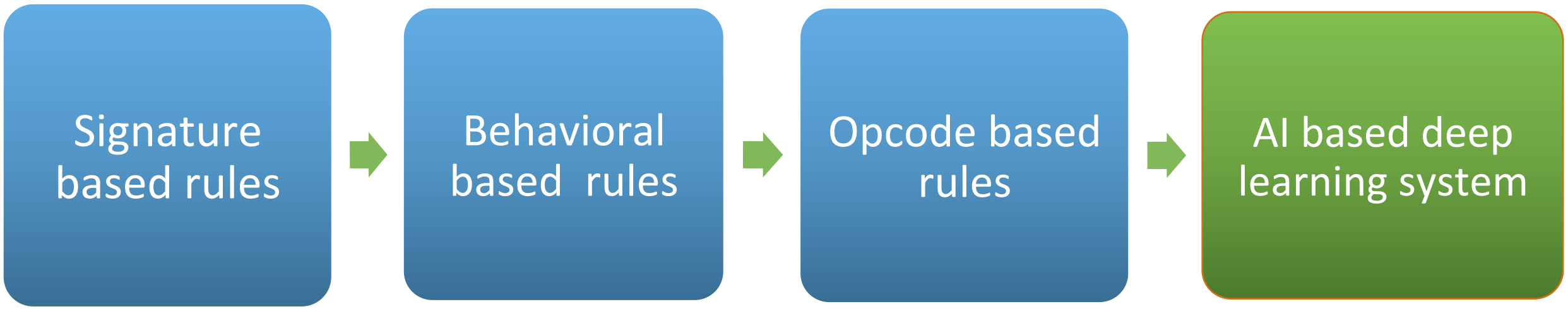
Number of Malicious Android Apps



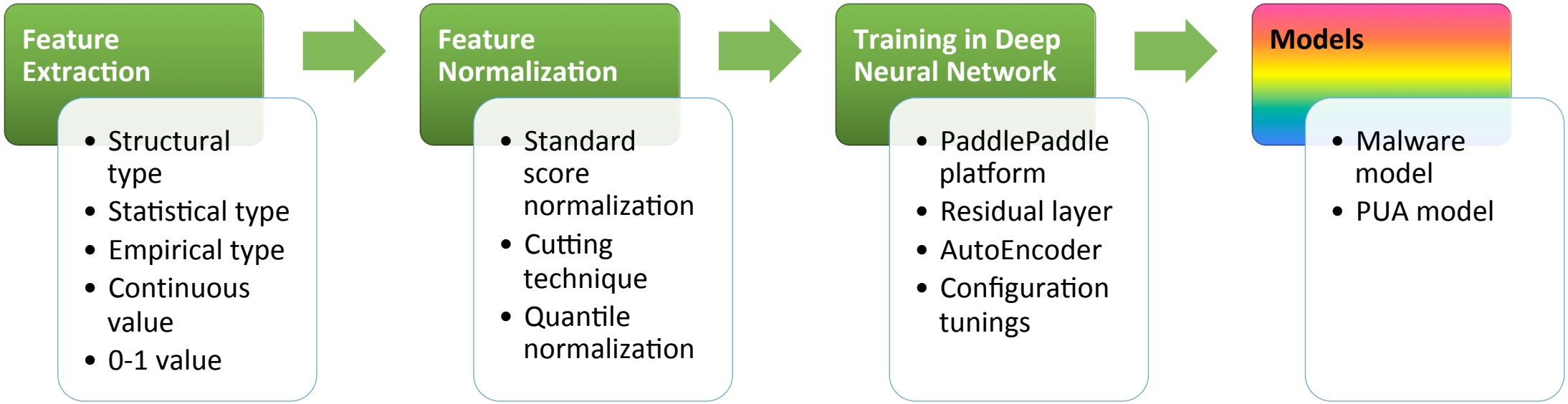
Dowgin: A Rich Variants Android Adware Family
New Dowgin Samples VS Average Dowgin Samples Hit Per Signature

Malicious apps, Dowgin samples and Dowgin signatures are counted from our database.

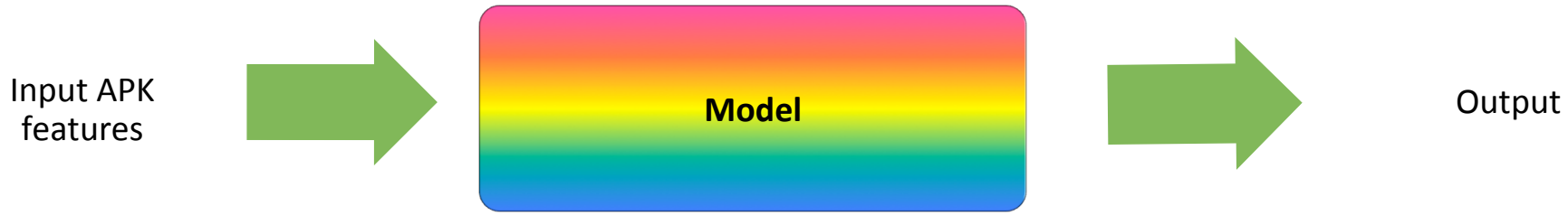
Our evolution



Training



Prediction



Feature extraction

APK

Structural features

- Num of uses-permissons in AndroidManifest
- Number of picture files in /res
- Size of /res
- Number of classes starts with Lcom/
- Num of classes starts with Ljava/
- Num of fields type boolean
- Num of methods which has parameters > 20

Statistical features

- Count certificate fields in samples to get 100 strings with discriminative info. E.g. emailAddress=MGame@mobile.com malicious/benign = 52

Empirical features

- Has executable file in /res
- Has apk file in /assets
- Register DEVICE_ADMIN_ENABLED broadcast and has sendSMSMessage permission

Numeralization(N = 1235)

205	3	34.5	143234
285	68	296	7
13850	157	11218	847
1.23e+9	422	1004	177
0	398	13.333	125

Continuous value (N = 571)

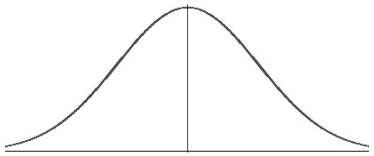
0	0	0	0
0	0	0	1
0	1	0	0
0	1	1	0
0	1	0	0

0-1 value (N = 664)

Feature normalization

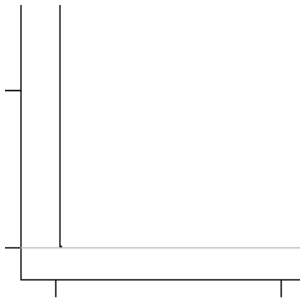
To make features more discriminative
Precision increased by 9%

Continuous value



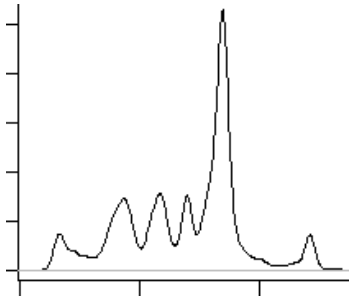
Gaussian distribution

Standard score normalization $z = \frac{x - \mu}{\sigma}$



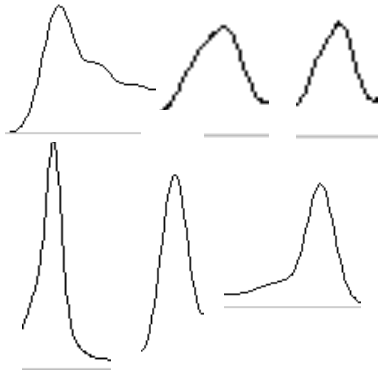
Noise problem

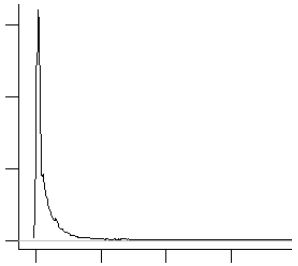
Cutting technique



Multimodal distribution

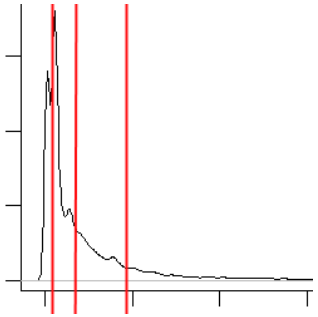
Cutting technique





Long-tailed distribution

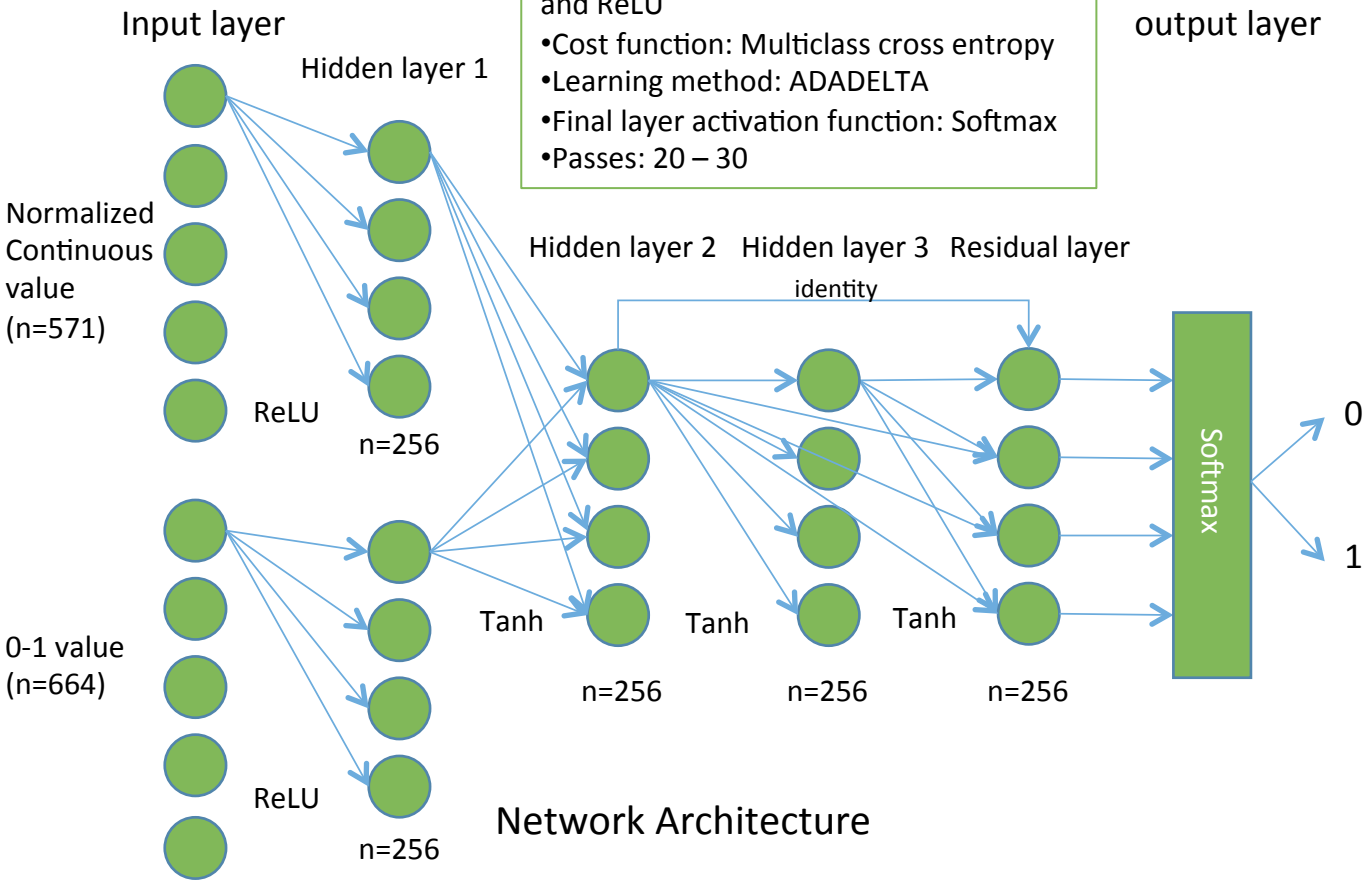
Quantile normalization



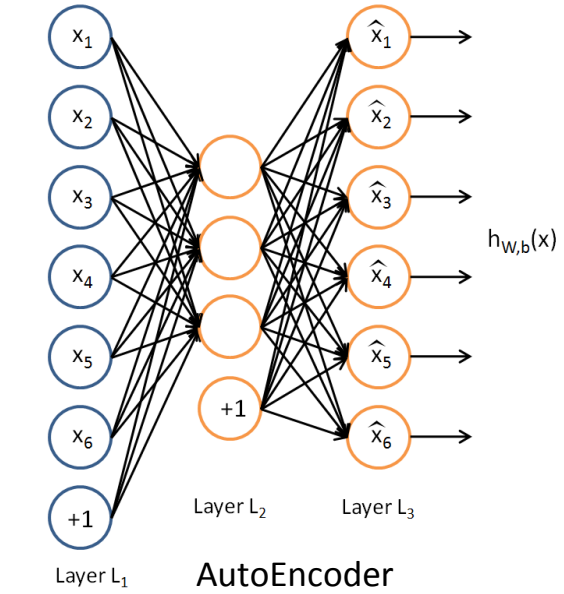
[-1, 1]

Training in deep neural network

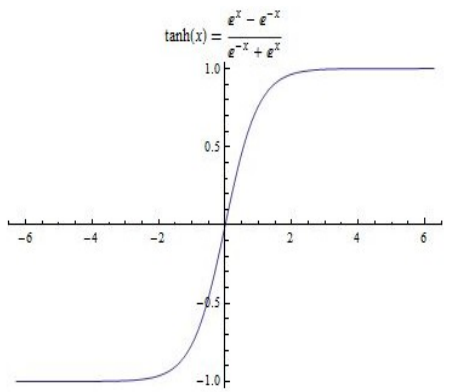
Configurations:
- Hidden layer activation function: Tanh and ReLU
- Cost function: Multiclass cross entropy
- Learning method: ADADELTA
- Final layer activation function: Softmax
- Passes: 20 – 30



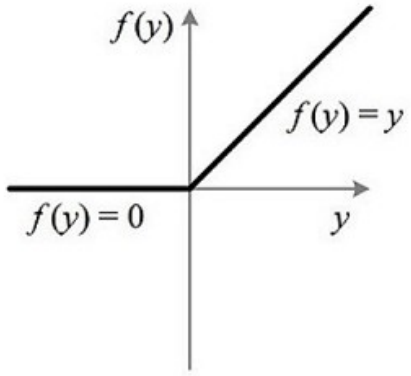
Network Architecture



AutoEncoder



Tanh

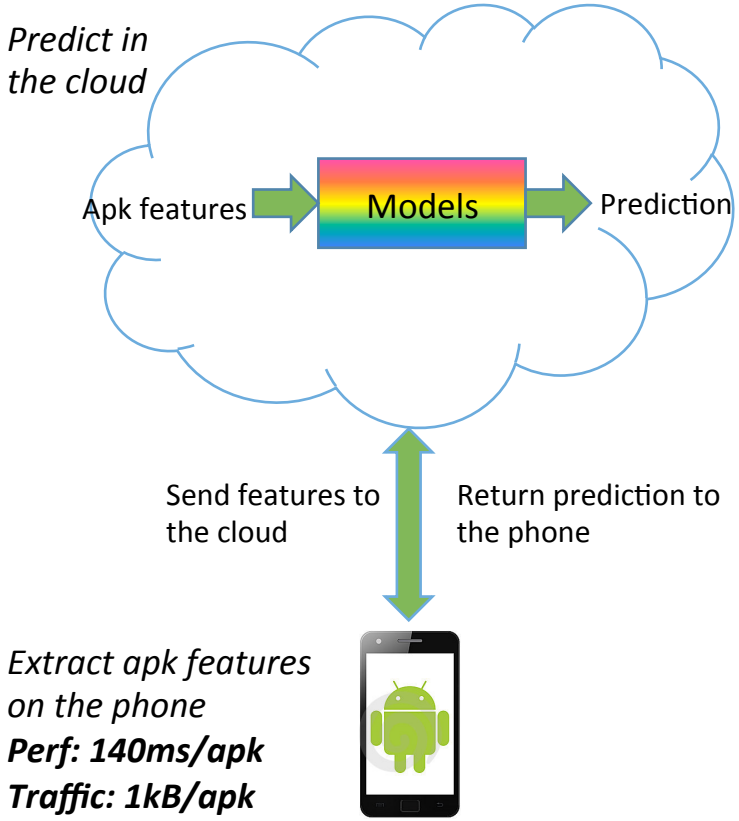


ReLU

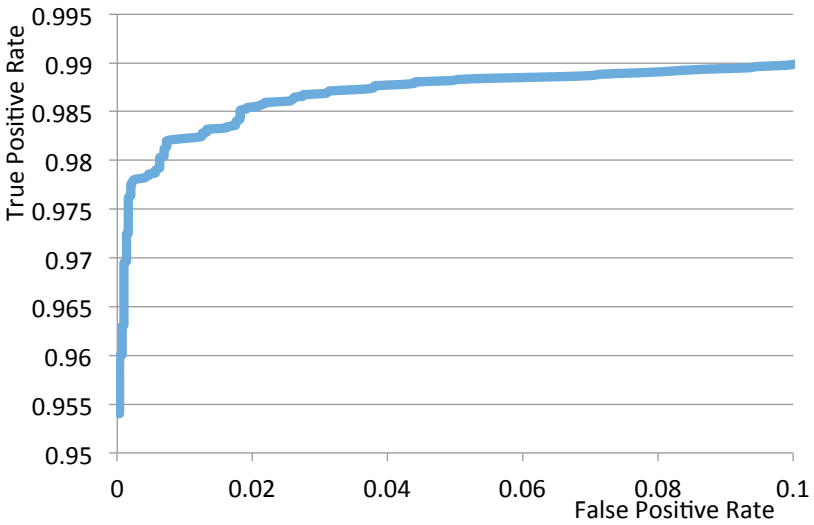
Trained on PaddlePaddle platform with 15M+ samples

Prediction & Evaluation

Production deployment



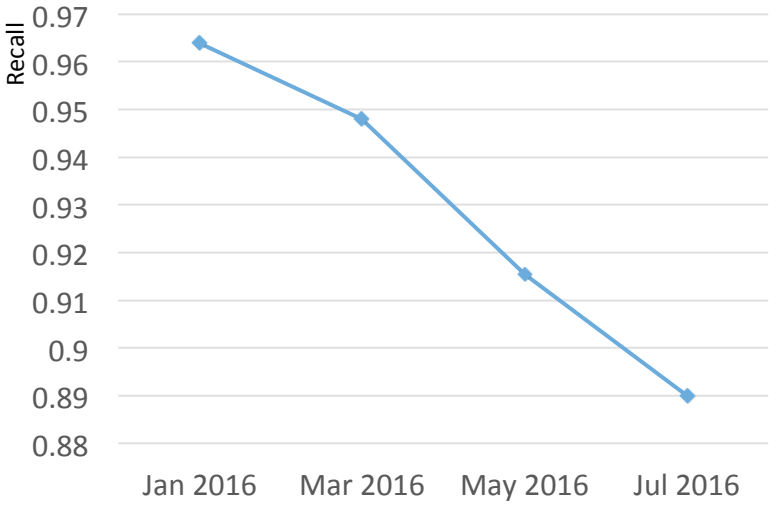
Detection performance



Detection performance as ROC curve

ROC curve is test against AV-TEST July's samples: 7613 Android malware, 3020 legitimate Android apps, total 10633.

Model lifetime



The lifetime of model trained on Jan 2016

The model is trained on Jan 2016 and tested against AV-TEST Jan, Mar, May and July's samples. Recall rate dropped by 7.6% in 6 months.

Limitations

- Can’t provide explanations for its detection results
- Can’t understand code meaning.
- Build on static analysis and lack of dynamic inspection.
- Can’t self learning, need continuous training with labeled data.

Advantages

- More difficult to evade
- Fixed-size

Conclusion

- Feature extraction is the key step
  - Virus analyst experience can help to find valuable features.
  - AutoEncoder neural network can be used to extract the most valuable features from a large number of features.
- This system is designed to detect Android malware, but these methods can also be used in detecting malware in other platforms.
- Our system learns in image recognition way. It’s effective only in detecting malware variants.

Thank you

- Welcome contact me
  - Twitter: @thomaslwang
  - Email: thomas.l.wang@gmail.com
- Welcome cooperation and partnership with us
- Acknowledgement
  - Baidu IDL: Lyv Qin, Xiao Zhou, Jie Zhou, Errui Ding, Yuanqing Lin, Andrew Ng
  - Partner: Liuping Hou, Jinke Liu, Zhijun Jia, Yanyan Ji
  - PaddlePaddle platform http://paddlepaddle.org