

Generic and Static Detection of Mobile Malware Using Machine Learning

Minh Tran

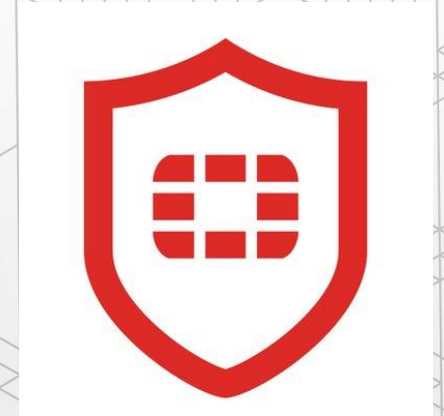
<https://www.linkedin.com/in/minhtq/>

Agenda

- Introduction
- Background
- Architecture
- Results
- Conclusions

Introduction

- Sr. Security Researcher @ Fortinet (FortiGuard)
 - Sr. Malware Research Engineer @ Palo Alto Networks
- 13+ years of experience
- PhD Candidate @ North Carolina State University
 - Master of Science - 2011
- #56 of Microsoft's Top 100 Security Researchers
 - <https://blogs.technet.microsoft.com/msrc/2018/08/08/microsofts-top-100-security-researchers-black-hat-2018-edition/>
- *Opinions are my own



Motivating Example

- Marcher!
- Social engineering attacks
- Corrupted

The screenshot displays three app listings from the 'org.slempto.service' developer:

- Adobe Flash Player (org.slempto.service)**: Corrupted. Rating: -1. Date: Apr 27, 2017 12:04:00 AM.
- Volksbank Verify (org.slempto.service)**: Rating: -4. Date: Apr 21, 2017 7:02:09 PM - Android.
- Postbank Finanzassistent (org.slempto.service)**: Corrupted. Rating: -2. Date: Apr 19, 2017 9:21:19 AM.

Why Signature-based and Behavior-based Malware Detection Are Still not Sufficient?

- Not resilient against variations.
- Malware samples can be corrupted
- Rooms for improvement!

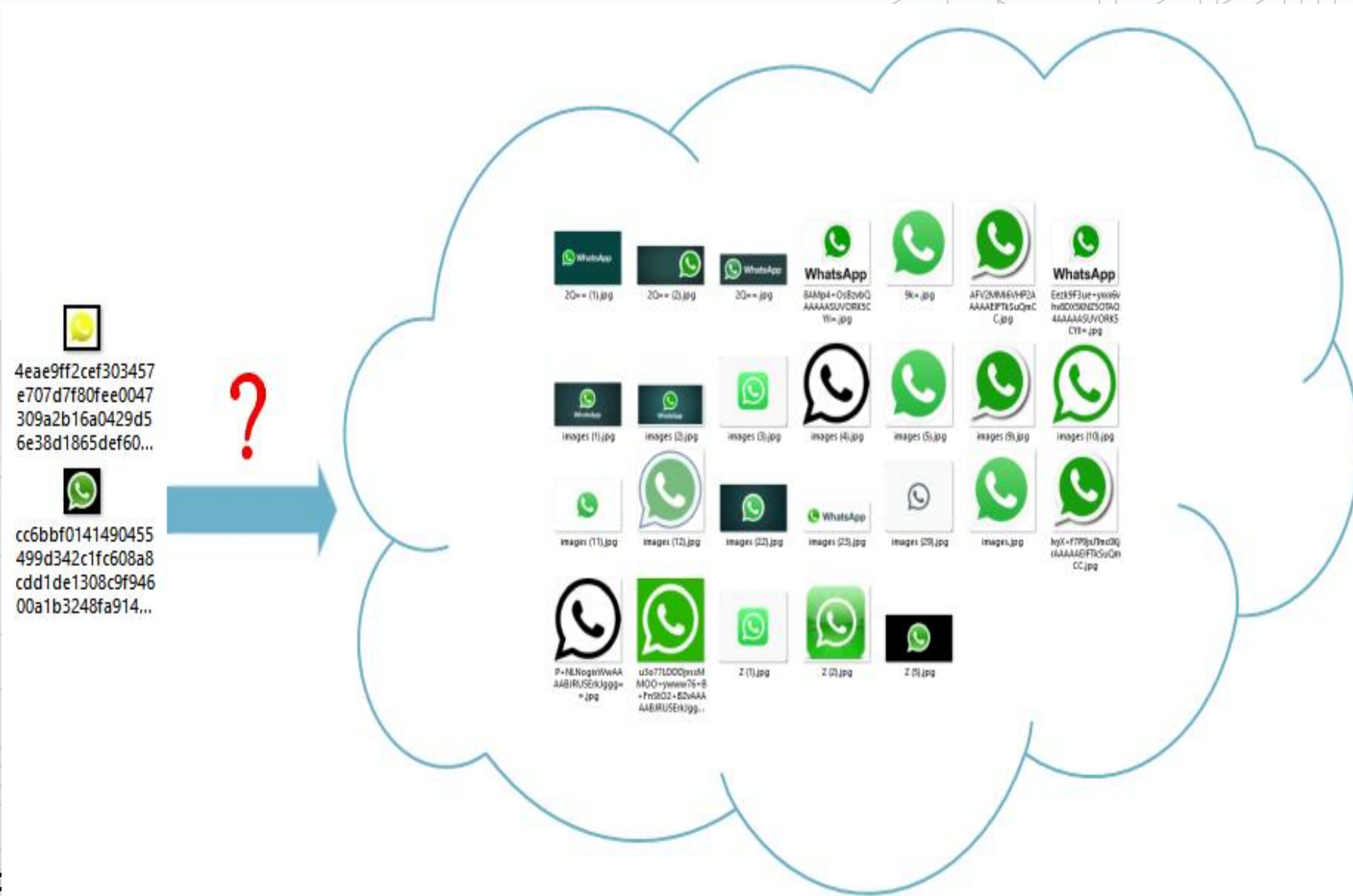
The screenshot displays a list of malware samples with their respective icons, names, hashes, and dates. The first entry is Adobe Flash Player (org.slempto.service) marked as Corrupted with a red '-1' icon. The second is Volksbank Verify (org.slempto.service) with a green Android icon and a red '-4' icon. The third is Postbank Finanzassistent (org.slempto.service) marked as Corrupted, Slempto, and banker with a blue icon and a red '-2' icon.

Icon	Name	Hash	Date	Tags
-1	Adobe Flash Player (org.slempto.service)	f3182d0ed107930df64f2b7e9170fceb5a0ca19589c0260becc814f029296943	Apr 27, 2017 12:04:00 AM -	Corrupted
-4	Volksbank Verify (org.slempto.service)	6274f62c805c75412b049b1da417fba0762e0e6429b90252e51152577647124d	Apr 21, 2017 7:02:09 PM - Android	
-2	Postbank Finanzassistent (org.slempto.service)	149cce8574dd5f74c152cc88eb4d4a61db6667a97e636b3668e480cb5a58d2b6	Apr 19, 2017 9:21:19 AM -	Corrupted, Slempto, banker

Key Insights

- *Legit: **com.symantec.mobilesecurity***
- vs
- Marcher: **etcqlnzwauf.hflivryhdnjb**
- Key Insight 1: obfuscation.
 - Use your enemy's strength against them!

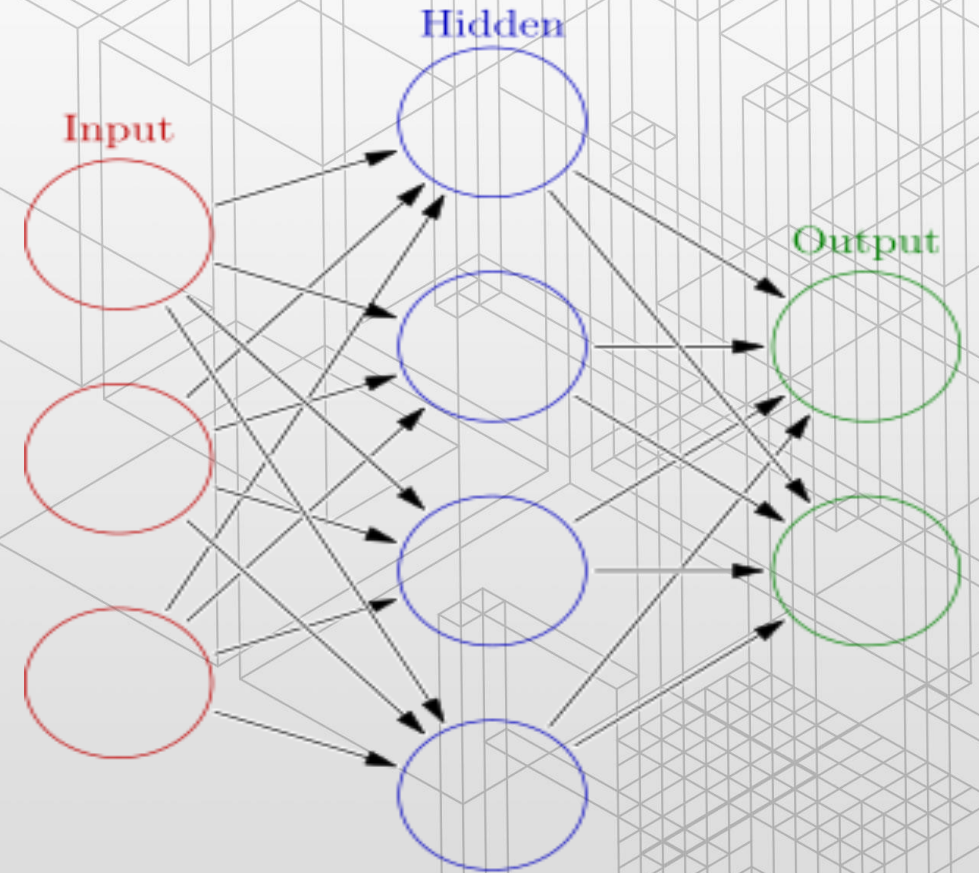
- Social engineering attacks



- A benign app should NOT do both at the same time!
 - Popular apps

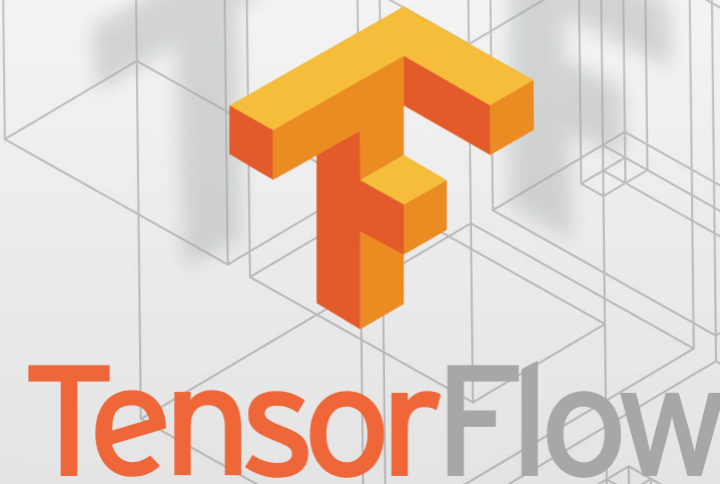


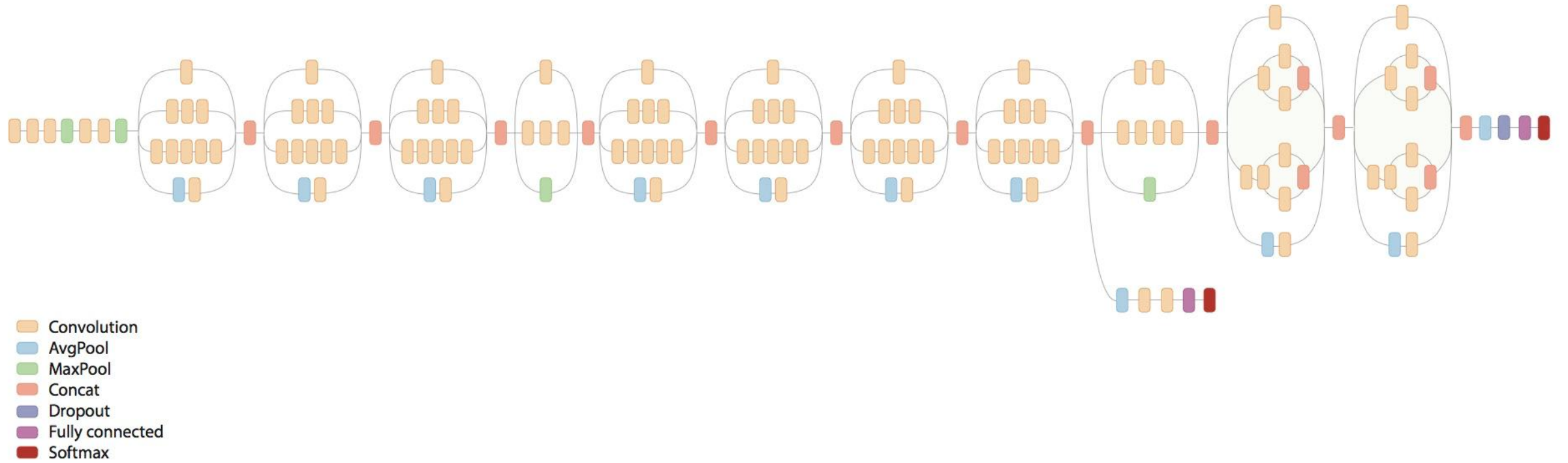
Machine Learning To The Rescue



Machine Learning To The Rescue

- Classify package names:
 - N-gram
- Classify images:
 - Neural Networks



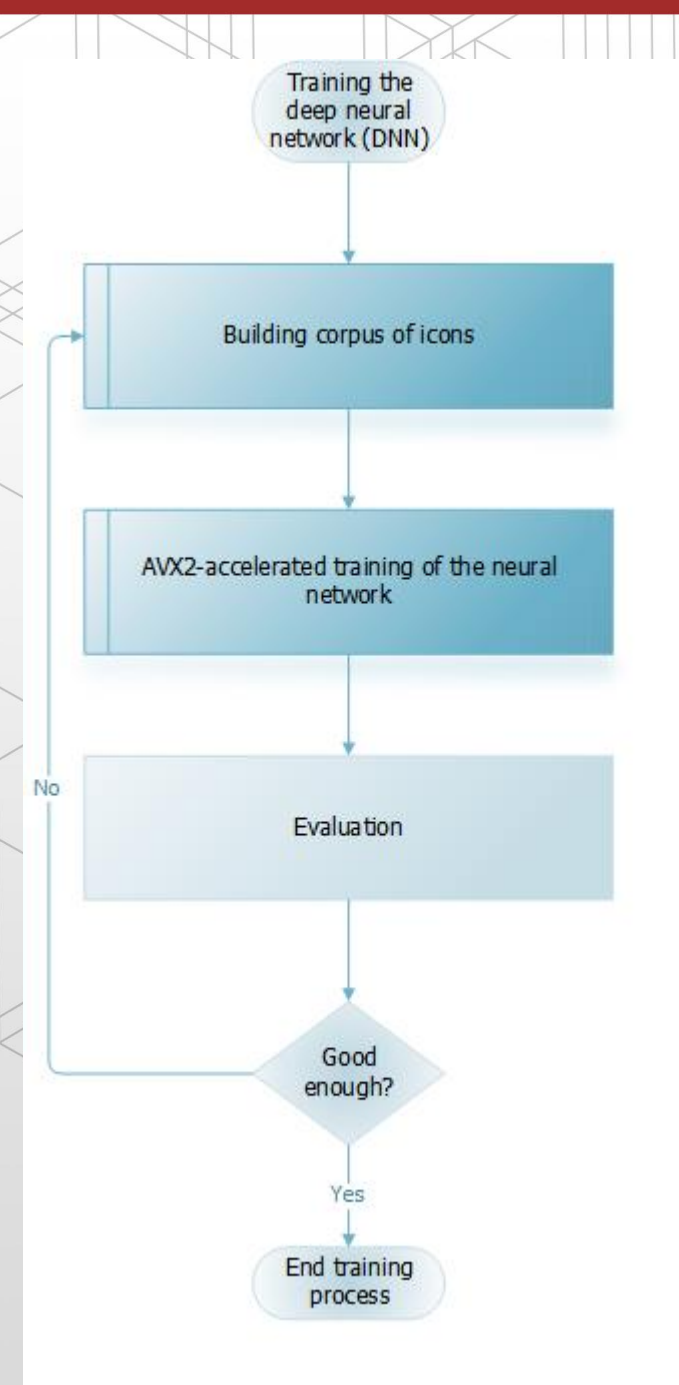


- PoC: Inception-v4:
 - 43 layers (deep learning!)
 - Lower computational cost (vs e.g. VGGNet)

**Credit belongs to the respective owners

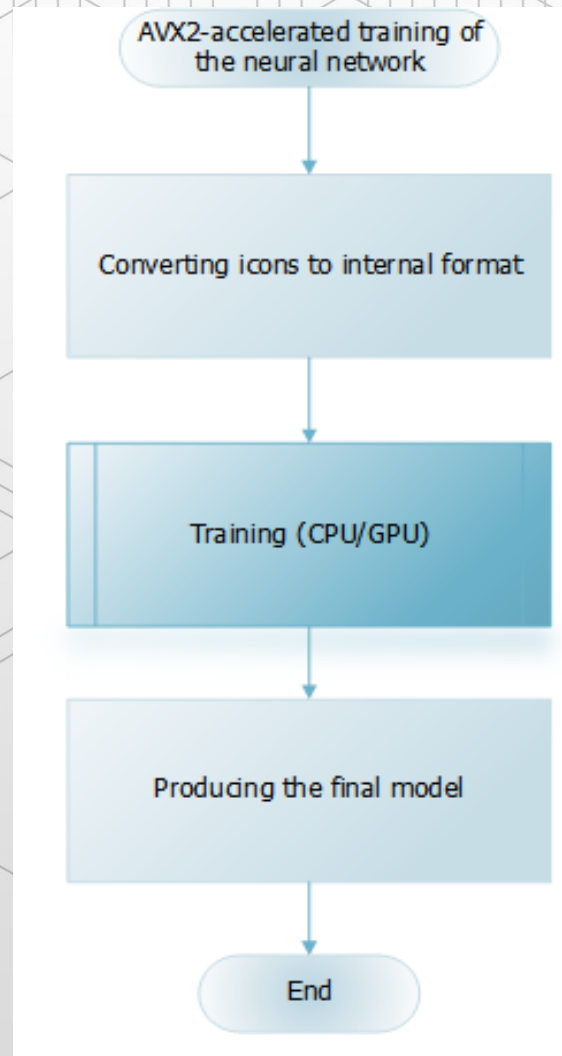
Workflow to Train the DNN

- Building corpus of icons
- Training of the neural network
 - Produce model files
- Evaluation using the test corpus



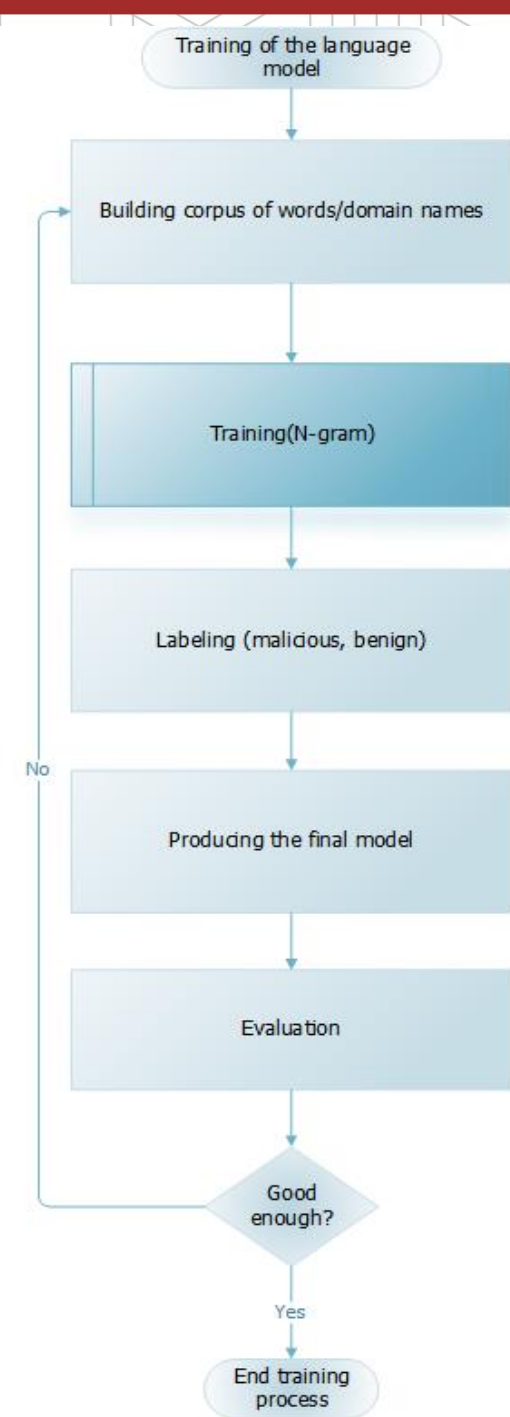
Training of the Neural Network Model

- Converting icons to internal format
- Training the neural network for n steps (e.g. n = 3000)
- Producing the final model (i.e. model files with the optimal weights & biases for neurons)
- Evaluation based on testing corpus



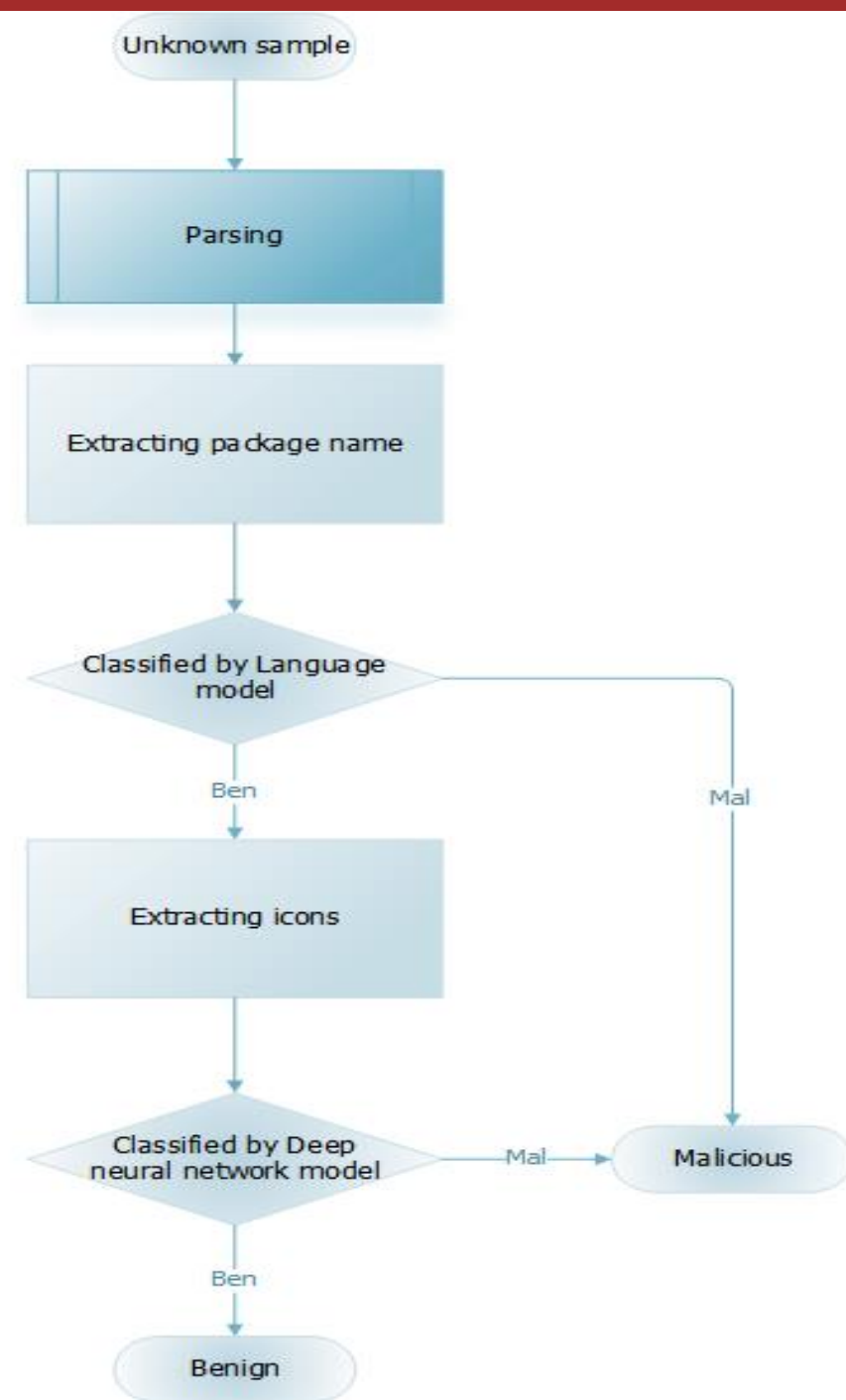
Training of the Language Model

- Building corpus of words/domain names/package names (e.g. Alexa, Majestic Million)
- Training (N-gram with n is a customizable length parameter e.g. n = 2)
- Labeling (malicious, benign) based on ground truths (from existing malware collections)
- Producing the final model
- Evaluation based on testing corpus



Workflow to Classify Samples

- Parsing packages
- Extracting package name and feed into the Language Model
- Extracting icons and feed into the Neural Network Model



Results

- Test set: 306847 samples
- 2gram total detection: right 271133 vs wrong 35714 = 11.64%
FN 88.13% FP 11.87%
- **3gram** total detection: right 277024 vs wrong 29823 = 9.72%
FN 78.88% FP 21.12%
- 4gram total detection: right 274412 vs wrong 32435 = 10.57%
FN 84.69% FP 15.31%

Results

- Our system classifies all Android malware, but especially effective against social engineering malware who masquerade as legitimate apps
- Our system has better coverage: many samples can be corrupted and our system still works because fundamentally speaking it is static analysis whereas solutions based on dynamic analysis fail.
- Our system has better performance: it is faster than dynamic analysis because no execution in sandbox is required
- Effectively speaking, detection rate is 99.928%.

Conclusions

- ML is valuable to malware detection
- Future research
 - Increasing the quality and the quantity of the data set: different languages etc
 - Improving training performance: distributed training etc

Questions