Deep Learning for Future Army Systems

Michael S. Lee, Ph.D., Simulation Sciences Branch, Computational Sciences Division

Collaborators: John Hyatt, Sam Edwards, Peng Wang, Eric Mark

US Army Research Laboratory
• Why are we looking at Deep Learning?
• What is deep learning?

• We are applying it to the study of a diverse set of future Army systems:

1. Detecting crack damage from ultrasound for Sustainment and Future Vertical Lift
2. Intrusion detection / malware analysis (Network / C3I)
3. Classification of radio modulation (Network / C3I)
5. Monitoring of additive manufacturing for sustainment (NGCV & FVL)
There has been rapid advances in machine learning...

• Game-playing AI – DeepGo can beat top humans

• Semantic segmentation: Towards self-driving cars

• Image classification with ~95% accuracy

• Language translation: “Error reduction by 55 to 85%”
CAN DEEP LEARNING HELP US?

- What Army problems can be solved with DL?
- Can we trust these black box methods?
- Can DL fit within our power/size constraints?
- Can DL be easily fooled?
- DL usually needs lots of data, can we overcome this challenge?

At ARL, we are looking at all of these questions.

DEEP LEARNING

• Optimize the **parameters** of a complicated function that **transforms** some **input** (e.g., picture of a cat) into some **output** (e.g., ‘label: cat’)

• A neural network with multiple “hidden” layers

• Uses a mix of convolutional and pooling (downsampling) layers
PROJECT #1: DETECTING CRACK DAMAGE FROM ULTRASOUND

Sustainment Goal:
1) Detect the damage before it even becomes visible.
2) Only replace parts when there is damage.
DETECTING CRACK DAMAGE FROM ULTRASOUND

The signals are complex, but ML can simplify (and automate) the readout.
In the same way that the Post Office automatically reads zip codes, convert probe signals to crack damage indicator.
DETECTING CRACK DAMAGE FROM ULTRASOUND


Learn more:
Traditionally, threat vectors to a computer system are detected by matching strings from a known threat database (e.g., antivirus software).

New threats, however, are either encrypted or are uniquely developed for targeted attacks on a particular asset.

Therefore, we have to look beyond a threat’s DNA (i.e., executable code) to detect and understand it.
**ML FOR INTRUSION DETECTION**

**Strategy:**

Encrypted/obfuscated executable \(\rightarrow\) CPU \(\rightarrow\) CPU stream in the clear and unrolled \(\rightarrow\) CPU

**Questions:**

- Are CPU instruction streams sufficient to distinguish good and bad activities?
- Does all data need to be fed in? (i.e., could the processing of data be **intermittent** – realistic scenario)
- Can **novel threat activities** be detected?
USING CONVOLUTIONAL NN TO CLASSIFY PROGRAM FUNCTIONS

Input: CPU stream fragments (1000 opcodes each)
Embedding layer (16-dim vector encodes 100+ unique opcodes)
Convolution layer #1 ($N = 30$, size = 11)
MaxPool (size = 10)
Dropout (0.2)
Convolution layer #2 ($N = 30$, size = 5)
MaxPool (size = 2)
Dropout (0.2)
Convolution layer #3 ($N = 30$, size = 5)
MaxPool (size = 2)
Dropout (0.2)
Flatten
Dense layer
Softmax classifier

<table>
<thead>
<tr>
<th>Class</th>
<th>Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>GNU cryptographic hashes</td>
<td>md5sum, sha256sum, sha384sum, sha512sum</td>
</tr>
<tr>
<td>OpenSSL cryptographic hashes</td>
<td>-sha128, -sha256, -sha384, -sha512</td>
</tr>
<tr>
<td>OpenSSL decryption algorithms</td>
<td>-camellia-256-cbc, -rc2-64-cbc, -aes-256-cbc, -blowfish</td>
</tr>
<tr>
<td>GLIBC memory operation tests</td>
<td>test-memcpy, test-memchr, test-memmem, test-memcmp</td>
</tr>
<tr>
<td>Compression tools</td>
<td>gzip, xz, bzip2, zip</td>
</tr>
</tbody>
</table>

#3: CLASSIFICATION OF DIGITAL RADIO MODULATION

- **Goals:**
  - Detect adversary RF modulation, use smart jamming
  - Detect adversarial interference, choose best mitigation strategy

**Goals:**

- Detect adversary RF modulation, use smart jamming
- Detect adversarial interference, choose best mitigation strategy

**SNR = 10 dB**

“Clean signal”

**SNR = -10 dB**

“Noisy signal”

**SNR = 10 dB**

“Clean signal”

**SNR = -10 dB**

“Noisy signal”

**Army Rapid Capabilities Office Challenge:**

24 modulation classes, 6 signal-to-noise ratios
We developed a neural network that extracts and detects shift(phase)-invariant features from a single data sample.
#4: HEALTH MONITORING OF GROUND VEHICLES

- **Identify useful indicators** in data collected from Army Multi-Purpose Vehicle testing.

- **Detect anomalous** events.

- Devise **automated** strategies to detect these anomalies.
#4: ANOMALY DETECTION IN AMPV DATA

Labelled engine speed data

Learned features
Additive manufacturing holds promise for making parts in the field.

However, we need assurances that these parts are up to our standards.

Deep learning will enable

- Closed-loop monitoring (repair issues on-the-fly)
- Certification by real-time assessment
#5: MONITORING FOR ADDITIVE MANUFACTURING

Layer #60

Layer #42

Image

Autoencoder

Simple edge detection
# FUTURE WORK

<table>
<thead>
<tr>
<th>Project</th>
<th>Future efforts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Malware detection</td>
<td>More programs (incl. malware); Real-time monitoring; autonomous cyber agents</td>
</tr>
<tr>
<td>Radio classification</td>
<td>ML-based demodulation in the presence of interference</td>
</tr>
<tr>
<td>AMPV</td>
<td>Data mining with various ML algorithms</td>
</tr>
<tr>
<td>Additive manufacturing</td>
<td>Complex builds, intended and unintended defects</td>
</tr>
<tr>
<td>&lt;Your Idea Here&gt;</td>
<td>???</td>
</tr>
</tbody>
</table>