tinyML® Research Symposium

Enabling Ultra-low Power Machine Learning at the Edge

March 27, 2023

www.tinyML.org
TinyRCE: Forward Learning Under Tiny Constraints

Danilo Pau, Prem Kumar Ambrose
System Research and Applications
STMicroelectronics, Agrate Brianza, Italy
Backpropagation
Backpropagation
Backpropagation
Reduce activations, not trainable parameters for efficient on-device learning\textsuperscript{1}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{chart}
\caption{Memory Footprint (MB) for inference and training with batch sizes.}
\end{figure}

Can incremental learning of multiple categories happen, totally on-line w/out catastrophic forgetting, w/out K-fold back-prop and be deployable on memory constrained devices?
## Case studies

<table>
<thead>
<tr>
<th></th>
<th>SHL</th>
<th>PAMAP2</th>
<th>CWRU</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sampling freq [Hz]</strong></td>
<td>100</td>
<td>100</td>
<td>12000</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>HAR</td>
<td>HAR</td>
<td>BBAC</td>
</tr>
<tr>
<td><strong>Sensor data</strong></td>
<td>Accelerometer (3 axis) + gyroscope (3 axis)</td>
<td>Accelerometer (3 axis) + gyroscope (3 axis)</td>
<td>Accelerometer (3 axis)</td>
</tr>
<tr>
<td><strong>Number of classes</strong></td>
<td>8</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td><strong>Number of users</strong></td>
<td>3</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td><strong>Frame size</strong></td>
<td>1 sec (6 * 100)</td>
<td>1 sec (6 * 100)</td>
<td>16.6ms (3*200)</td>
</tr>
</tbody>
</table>
Interleaved incremental learning and testing

Class 1
20% train
80% test

Class 2
20% train
80% test

Class 3
20% train
80% test

Time

0.5 sec

100 Frames

400 Frames
Proposed framework

Sensor data streams

OR

BBAC

Acc(3)
200*(3) data

OR

HAR

Acc(3) +
Gyroscope(3)
100*(6) data

Randomly
initialized weights
CNN-FE_ELM

Offline trained
CNN-FE_BP

CNN-SM_BP

TinyRCE

Feature extractor

K-fold validation = 5
RCE while learning

**Overlapped feature space with input copies stored (like K-NN)**
- Radius keeps changing
- Ambiguous feature assignments
- Fully parallel distance measurements
- No default neural allocation rate control


TinyRCE was meant to detect unknown input patterns, thus triggering the user to provide the corresponding label.
TinyRCE was meant to detect unknown input patterns, thus triggering the user to provide the corresponding label.
TinyRCE vs baseline RCE

Learn within memory caps
The maximum number of hidden neurons the MCU shall store is depending on the budgeted memory.

Aging
Increased/decreased depending on \( \frac{\text{dist}(h_j,x_t)}{R_j} \)

Pruning
Prevent uncontrolled instantiation of hidden neurons (causing overflowing available RAM) during the learning phase

Unambiguous assignment
When input falls into multiple hidden neurons regions of influence, the output of the one to which the input falls closer (Hamming or Euclidian distance) is considered
Performances on single user

Datasets

- SHL (8 classes)
  - CNN-SM_BP (K-fold=5): 99.68%
  - CNN-FE_ELM + TinyRCE: 99.61%

- PAMAP2 (12 classes)
  - CNN-SM_BP (K-fold=5): 92.54%
  - CNN-FE_ELM + TinyRCE: 91.84%

- CWRU (10 classes)
  - CNN-SM_BP (K-fold=5): 97.09%
  - CNN-FE_ELM + TinyRCE: 95.21%
<table>
<thead>
<tr>
<th>Metrics</th>
<th>CNN-SM_BP (single inference) FP32</th>
<th>CNN-SM_BP (learning) FP32 (K fold=1)</th>
<th>CNN-FE_ELM + TinyRCE (single inference) FP32</th>
<th>CNN-FE_ELM + TinyRCE (learning) FP32</th>
</tr>
</thead>
<tbody>
<tr>
<td>MACC</td>
<td>1.213 M</td>
<td>8 G</td>
<td>1.251 M</td>
<td>23.4 M</td>
</tr>
<tr>
<td>FLASH (KiB)</td>
<td>76.45</td>
<td>-</td>
<td>76.45 (*)</td>
<td></td>
</tr>
<tr>
<td>RAM (KiB)</td>
<td>13.59</td>
<td>954.57</td>
<td>40.2</td>
<td></td>
</tr>
<tr>
<td>Latency STM32L4 (ms) @80MHz</td>
<td>123.4</td>
<td>813.25 sec</td>
<td>127.2</td>
<td>2.38 sec</td>
</tr>
<tr>
<td>Latency STM32H7 (ms) @480 MHz</td>
<td>11.21</td>
<td>73.86 sec</td>
<td>11.55</td>
<td>216</td>
</tr>
</tbody>
</table>

\[
MACC_{Inference} = h \times [(n \times 5) + 10]
\[
MACC_{Learning} = (h \times [(n \times 5) + 10]) \times (F \times E)
\]

* random weights
TinyRCE

- Capable of personalized, incremental learning
- w.r.t. backpropagation
  -0.01% to -1.88% single user
  -2% to -3.52% avg 8 users, 8 commons classes
- Deployable on MCUs, w.r.t. backpropagation
  learning 342x less complex
  inference same complexity
- Memory
- same FLASH with random weights
- 24x less RAM

Tiny learning

- Open new path to applications
- Save dramatically complexity on the cloud
- Scale faster!
- More reactive Federated Learning

Future works

- Extend to MLCommons/Tiny tasks
- Deep quantization of the CNN_FE to save further memory
- Add concept drift detection to trigger learning phase autonomously
Thanks for listening.

Q&A

danilo.pau@st.com
Copyright Notice

This presentation in this publication was presented at the tinyML® Research Symposium (March 27, 2023). The content reflects the opinion of the author(s) and their respective companies. The inclusion of presentations in this publication does not constitute an endorsement by tinyML Foundation or the sponsors.

There is no copyright protection claimed by this publication. However, each presentation is the work of the authors and their respective companies and may contain copyrighted material. As such, it is strongly encouraged that any use reflect proper acknowledgement to the appropriate source. Any questions regarding the use of any materials presented should be directed to the author(s) or their companies.

tinyML is a registered trademark of the tinyML Foundation.

www.tinyml.org