Mango: Fast, Parallel and Gradient-Free Challenges of Adopting Existing Neural Architecture Search

- Existing frameworks for low-end IoT devices: SpikSe, MCUNet, MicroNets, and jNAS.
- Lack of open-source tools.
- Use of coarse or inaccurate hardware metrics / proxies.
- Problematic formulation – inability to handle loss contour discontinuities and categorical variables; assumes usage of only CNN and MLP for toy applications.
- Long convergence time and requires expensive compute infrastructure.

MANGO: Fast, Parallel and Gradient-free Bayesian Optimizer

- Scalability: Outperforms current parallel searches.
- Fault-Tolerance: Detects failures at the application layer.
- Supports categorical & continuous search spaces.
- Compatible with ScPy and Scikit-learn.
- Open-source and expandable.

Neural Architecture Search Formulation

\[ f_{\text{NAS}} = \lambda_1 f_{\text{Accuracy}} + \lambda_2 f_{\text{Fault Tolerance}} + \lambda_3 f_{\text{Performance}} + \lambda_4 f_{\text{Latency}} \]

\[ f_{\text{Accuracy}} = \mathbb{E}[\text{Precision}(\theta)] \]

\[ f_{\text{Fault Tolerance}} = \sum_{\text{Faults}} \left( \text{Error} \times \text{Faults} \right) \]

\[ f_{\text{Performance}} = \frac{\text{FLOPS}}{\text{Max Target Latency}} \]

\[ f_{\text{Latency}} = \frac{\text{SRAM}}{\text{Max Latency}} \]

Qualitative Comparison Against Other Frameworks

<table>
<thead>
<tr>
<th>Method</th>
<th>Search Strategy</th>
<th>Profile</th>
<th>Tested Models</th>
<th>Optimisation Parameters</th>
<th>Open-Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>SpArSe</td>
<td>Gradient-driven Bayesian</td>
<td>Analytical</td>
<td>CNN, MLP</td>
<td>Error, SRAM, Flash, Latency</td>
<td>No</td>
</tr>
<tr>
<td>MCUNet</td>
<td>Evolutionary</td>
<td>Lookup tables, prediction models</td>
<td>CNN, MLP</td>
<td>Error, SRAM, Flash, Latency</td>
<td>No</td>
</tr>
<tr>
<td>MicroNets</td>
<td>Analysitical</td>
<td>CNN, MLP</td>
<td>Error, SRAM, Flash, Latency</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>jNAS</td>
<td>Evolutionary</td>
<td>CNN, MLP</td>
<td>Error, SRAM, Flash, Latency</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>THIN-Bayes</td>
<td>Gradient-free Bayesian</td>
<td>Platform-is-to-the-loop, analytical</td>
<td>Any model using TFLOP/operations</td>
<td>Any scalar term</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Conclusion

- Our NAS performs intelligent architectural adaptations to exploit full hardware capabilities in order to improve error.

Evaluation - Architectural Adaptation

- Our NAS performs intelligent architectural adaptations to exploit full hardware capabilities in order to improve error.

Evaluation - Analytical Proxies Are Problematic

- SRAM and Flash proxies tend to overestimate HW constraints without considering dynamic runtime SW overhead or faults.
- FLOPS is not always proportional to latency.

References