Loss-driven sampling for online neural network training with large scale simulations

Sofya Dymchenko, Bruno Raffin

To cite this version:
Sofya Dymchenko, Bruno Raffin. Loss-driven sampling for online neural network training with large scale simulations. LIG PhD day 2023, Sep 2023, Grenoble, France. pp.1-1, 2023. hal-04305159

HAL Id: hal-04305159
https://hal.science/hal-04305159
Submitted on 24 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License
Context

We want to advance applications of DL to sciences by improving online training techniques. DataMove team contributed with DeepMelissa framework for large-scale deep surrogate training on supercomputers.

Motivation

* Avoidance of I/O bottleneck: with file-free processing high-dimensional data is given "on-the-fly".
* Infinite data stream: no limitations for generalization.
* Ability to control data stream: choice of data to sample can be explicit rather than random. Training can be more data-efficient. But how to choose?

Problem statement

Simulation: program that returns outputs by given inputs, e.g. initial state of water flow in tube and t-state.

Neural network: any NN that uses data from simulation, e.g. a surrogate that imitates simulation.

Sampler: takes inputs from defined prior distribution, e.g. uniform distribution. Can we do better?

Method

Hypothesis

High loss \(\implies\) difficult sample to learn
\(\implies\) region needs to be fed to neural network

Adaptive sampling

First batch is sampled uniformly. Loop-process:
1) NN provides loss values statistic per batch-point
2) Select points w.r.t. distribution calculated as normalised loss
3) Next batch is sampled from their Gaussian neighbourhoods

R-value

The value R is a ratio (percentage) of points to be sampled for next epoch in a loss-driven manner. It is a trade-off between "exploring hard points" and "remembering/learning easy points".

Experiments

The simulation gives closest minima by running gradient descent on the fixed surface. Current fixed surface is two imbalanced pits.

Results

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Uniform</th>
<th>Linear + Constant (0.15, 0.7, 0.4)</th>
<th>Linear + Constant (0.15, 0.7, 0.1)</th>
</tr>
</thead>
</table>

Conclusion

* Overall speed-up in convergence
* Significantly increased prediction accuracy for "hard area"