

Bachelorarbeit

Thema: Reversible Residual Networks

Beschreibung:

Reversible Residual Networks (RevNets) [4] sind so entworfen, dass Berechnungen nicht nur vom Input Layer zum Output Layer erfolgen können, sondern auch in die umgekehrte Richtung. Die Berechnung der Eingabe einer Schicht aus deren Ausgabe ermöglicht das Training tiefer Neuronaler Netzwerke ohne dass für den Backpropagation-Trainingsschritt alle Ergebnisse des Forward Pass gespeichert werden müssen. Daher werden für das Training dieser tiefen Netzwerke keine großen Rechencluster mehr benötigt, stattdessen kann das Training auf einer einzigen Grafikkarte erfolgen. Durch die Einsparungen des GPU-Speichers können auch die Inputs größer gewählt werden. Ein aktuelles Beispiel ist die Reformer-Architektur [1] für die Textübersetzung, bei der nicht mehr einzelne Sätze als Input dienen, sondern bis zu einer Millionen Wörter als Kontext verarbeitet werden können.

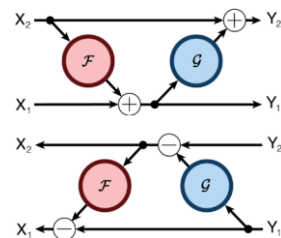


Bild: Grundprinzip RevNets.
Quelle: [4]

Aufgabenstellung:

Im Rahmen dieser Bachelorarbeit sollen Neuronale Netzwerk-Architekturen aufgearbeitet werden, die Berechnungen vom Output in Richtung Input ermöglichen. Anhand einer exemplarisch ausgewählten Architektur soll untersucht werden, wie viel Speicher beim Training eingespart werden kann und ob sich dadurch neue Möglichkeiten für eine größere Batch Size bei der semantischen Segmentierung sowie für das Training auf einer NVIDIA Jetson Low-Power GPU ergeben.

Arbeitspunkte:

- Aufarbeitung und Systematisierung des State of the Art zu Reversiblen Neuronalen Netzwerken, u.a. [2–11], aufzeigen der aktuellen Anwendungsgebiete (u.a. [1])
- Speichereffiziente Umsetzung eines ausgewählten Ansatzes (z.B. [4] oder [5]) in PyTorch oder Trax angelehnt an frei verfügbare Implementierungen (siehe Literaturhinweise zu GitHub)
- Untersuchung zum Speicherverbrauch auf der GPU bei Verwendung eines tiefen Netzwerks
- Experimente zu oben beschriebenen neuen Möglichkeiten (Segmentierung, Jetson)
- Ausarbeitung von Präsentationen für den Eröffnungs- und Abschlussvortrag inkl. Vorträgen und Diskussion entsprechend den Vorgaben des FG NI+KR
- Verfassen der Bachelorarbeit entsprechend der Vorgaben

Literatur:

- [1] Kitaev et al.: Reformer: The Efficient Transformer. ICLR, 2020. ([Link](#))
 - [2] Dinh et al.: NICE: Non-Linear Independent Components Estimation. ICLR, 2015. ([Link](#))
 - [3] Dinh et al.: Density Estimation Using Real NVP. ICLR, 2017. ([Link](#))
 - [4] Gomez et al.: The Reversible Residual Network: Backpropagation Without Storing Activations. NIPS, 2017. ([Link](#))
 - [5] van de Leemput et al.: MemCNN: A Framework for Developing Memory Efficient Deep Invertible Networks. ICLR, 2018. ([Link](#))
 - [6] Jacobsen et al.: i-RevNet: Deep Invertible Networks. ICLR, 2018. ([Link](#))
 - [7] Grathwohl, et al.: FFJORD: Scalable Reversible Generative Models with Free-Form Continuous Dynamics. SAABI, 2018. ([Link](#))
 - [8] MacKay et al.: Reversible Recurrent Neural Networks. NeurIPS, 2018. ([Link](#))
 - [9] Kingma, et al.: Glow: Generative Flow with Invertible 1x1 Convolutions. NeurIPS, 2018. ([Link](#))
 - [10] Chen, et al.: Neural Ordinary Differential Equations. NeurIPS, 2018. ([Link](#))
 - [11] Behrmann et al.: Invertible Residual Networks. ICML, 2019. ([Link](#))
- Elektronische Literaturdatenbank des FG NI&KR mit Recherchemöglichkeiten
 - Elektronische Konferenzproceedings Datenbank des FG NI&KR
 - IEEE Recherchesystem www.ieeeexplore.ieee.org (nur aus dem Uni-Netz bzw. via VPN)
 - Google Scholar scholar.google.com
 - Microsoft Academic Search academic.research.microsoft.com
 - GitHub: github.com/google/trax (RevNet [4] in Trax), github.com/silvandleemput/memcnn (MemCNN [5] in PyTorch)
 - Proceedings der relevanten Konferenzen (NIPS, ICLR, ICML, ICANN, IJCNN, WCCI, CVPR, ICCV, ECCV, BMVC, AVSS, ICPR, ICIP, ...)

Betreuer:

Dr. Markus Eisenbach (Markus.Eisenbach@tu-ilmenau.de)
Dustin Aganian, M.Sc. (Dustin.Aganian@tu-ilmenau.de)

Betr. Hochschullehrer: Prof. Dr. H.M. Groß

Bearbeiter: offen