Adaptive Batchgröße zur Beschleunigung der Konvergenz Künstlicher Neuronaler Netze am Beispiel der Abbildung der Fahrzeugdynamik

Maximilian Hohn, M.Sc., Sebastian Reicherts, M.Sc., Dr.-Ing. Tobias Bruckmann, Prof. Dr.-Ing. Dr. h.c. Dieter Schramm, Universität Duisburg-Essen, Lehrstuhl für Mechatronik, 470578 Duisburg, Deutschland, {maximilian.hohn, sebastian.reicherts, tobias.bruckmann, dieter.schramm}@uni-due.de

Kurzfassung

Modellbasierte Entwicklungsmethoden werden für den technischen Fortschritt im Zuge der Digitalisierung immer relevanter. Aufgrund der stetig anwachsenden Rechenkapazitäten und der zunehmenden Verfügbarkeit von Daten, nehmen künstliche neuronale Netze auch in der Modellbildung eine bedeutende Rolle ein. Insbesondere die Abbildung komplexer Systeme durch große künstliche neuronale Netze sind sehr rechenintensiv. Zur Effizienzsteigerung solcher Entwicklungswerkzeuge tragen Verfahren zur Verringerung der Rechenzeit oder zur Steigerung der Approximationsgüte bei. Im Folgenden wird die aktuelle Forschung am Lehrstuhl für Mechatronik der Universität Duisburg-Essen hinsichtlich der Optimierung und der Auslegung einer adaptiven Batchgröße für das Training künstlicher neuronaler Netze vorgestellt. Die Untersuchungen finden am Beispiel der Abbildung der Fahrzeugquerdynamik durch ein künstliches neuronales Netz mit LSTM-Zellen statt.

Das Training künstlicher neuronale Netze erfolgt in den allermeisten Fällen durch stochastische Gradientenabstiegsverfahren (SGD). Dabei stellt die Batchgröße eine sensitiven Parameter für die Trainingsgüte und die benötigte Rechenzeit dar. Die Batchgröße definiert die Anzahl der verwendeten Beispiele für eine Optimierungsiteration. Mit zunehmender Batchgröße steigt die Güte bei der Ermittlung der Gradienten der Fehlerfunktion; im Gegenzug nimmt die Rechenzeit pro Optimierungsiteration zu. Kleine Batchgrößen liefern hohe Konvergenzraten, während größere in der Regel eine hohe Güte liefern. Durch eine adaptive Batchgröße können beide Vorteile kombiniert werden.

Verwandte Arbeiten

Devarakonda et al. [1] weisen nach, dass bei annähernd gleicher Modellgüte durch eine vordefinierte, steigende Batchgröße das Training auf einer GPU bis zu 1,49-mal schneller (6,25-mal schneller bei Verwendung mehrerer GPUs) ist als bei einer statischen Batchgröße. De et al. [2] stellten einen Algorithmus einer adaptiven Batchgröße vor, wodurch keine Konvexität des Ziels erforderlich ist. Allerdings blieb die Konvergenzrate ähnlich wie beim klassischen SGD. Neben der Batchgröße stellt die Lernrate beim Training künstlicher neuronaler Netze einen für die Modellgüte und die benötigte Rechenzeit sensitiven Hyperparameter dar. Während eine adaptive Reduktion der Lernrate beim Training künstlicher neuronaler Netze eine gängige Praxis ist, ist dies bei der adaptiven Erhöhung der Batchgröße noch nicht der Fall. Smith und Kindermans [3] konnten empirisch nachweisen, dass eine vordefinierte Erhöhung der Batchgröße, anstelle der Reduktion der Lernrate, quantitativ zu einem nahezu identischen Ergebnis führt. Dabei bietet die Erhöhung der Batchgröße anstelle der Reduktion der Lernrate eine höhere Konvergenzrate.

Vorgehen

Ziel der Untersuchungen ist die Erhöhung der Konvergenzrate des Trainings künstlicher neuronaler Netze durch eine adaptive Anpassung der Batchgröße. Dafür wird ein Algorithmus entwickelt, welcher auf Basis der beim Training entstehende Information die Batchgröße anpasst. Zur Maximierung der Konvergenzrate wird die voraussichtliche Ausführungszeit eines Iterationsschrittes für verschiedene Batchgrößen, sowie der Erwartungswert des Optimierungsfortschrittes und dessen Varianz geschätzt. Mit jeder Iteration wird eine Messung des Optimierungsfortschrittes und der Ausführungszeit vorgenommen. Weiterhin wird das Konvergenzverhalten in Abhängigkeit des Optimierungsfortschrittes und der Batchgröße untersucht. Dieser Zusammenhang kann beispielsweise in Form eines Kalmanfilters in die Optimierung einfließen. Zur Isolation der Aufgabe wird der Algorithmus zunächst an einem reinen SGD-Verfahren entwickelt. In weiteren Untersuchungen kann die Anpassung des Algorithmus auf Optimierungsverfahren mit Momentum, adaptiver Lernraten sowie der Verwendung mehrerer GPUs erfolgen.

Alle Rechte vorbehalten.

Literatur

[1] Devarakonda, Aditya; Naumov, Maxim; Garland, Michael (2017): AdaBatch: Adaptive Batch Sizes for Training Deep Neural Networks. Online verfügbar unter https://arxiv.org/pdf/1712.02029.

[2] De, Soham; Yadav, Abhay; Jacobs, David; Goldstein, Tom (2017): Automated Inference with Adaptive Batches. In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (54), S. 1504–1513. Online verfügbar unter http://proceedings.mlr.press/v54/de17a.html.

[3] Smith, Samuel L.; Kindermans, Pieter-Jan; Ying, Chris; Le V, Quoc (2017): Don't Decay the Learning Rate, Increase the Batch Size. Online verfügbar unter https://arxiv.org/pdf/1711.00489.

DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

ub universitäts bibliothek

In: Siebte IFToMM D-A-CH Konferenz 2021

Dieser Text wird über DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt. Die hier veröffentlichte Version der E-Publikation kann von einer eventuell ebenfalls veröffentlichten Verlagsversion abweichen.

DOI: 10.17185/duepublico/74045
URN: urn:nbn:de:hbz:464-20210217-072254-7

Alle Rechte vorbehalten.