

Grundlagen KI Projekt persönlicher Report

Malte Elvers

Technische Universität Clausthal, Clausthal-Zellerfeld 38678, Deutschland

1 Individueller Beitrag

Im Rahmen dieses Projekts habe ich mich mit der Erkennung von Verkehrsschildern mithilfe eines neuronalen Netzwerks beschäftigt. Mein Ziel war es, ein Modell zu trainieren, das automatisch die Position von Verkehrsschildern in Bildern erkennt und entsprechende Bounding Boxes sowie den ausgeschnittenen Bildbereich ausgibt. Dafür habe ich ein Faster R-CNN-Modell mit PyTorch implementiert, das auf einem ResNet-50-Backbone basiert. Neben der Modellerstellung habe ich einen Datensatzloader entwickelt, um die Bilder und die zugehörigen Bounding-Box-Annotationen aus einzulesen und zu verarbeiten. Zusätzlich habe ich die Daten transformiert, damit die Bilder auf eine feste Größe skaliert werden und gleichzeitig die Bounding Boxes entsprechend angepasst werden. Anschließend habe ich das Modell trainiert und getestet.

2 Eigene Arbeitsweise und Reflexion

Zu Beginn habe ich versucht, das Problem mit einem einfachen CNN und einer direkten Bounding-Box-Regression zu lösen. Dabei habe ich jedoch festgestellt, dass das Modell keine zuverlässigen Positionen vorhergesagt hat, sondern stets eine Box in der Bildmitte platziert hat. Dies lag daran, dass das Modell keine gezielte Objekterkennung durchführen konnte und stattdessen versuchte, den durchschnittlichen Fehler zu minimieren, was dazu führte, dass es die Box immer zentral platzierte.

Um dieses Problem zu lösen, habe ich mich für ein Faster R-CNN entschieden, das nicht direkt eine Bounding Box ausgibt, sondern mit einem Region Proposal Network (RPN) arbeitet, das potenzielle Objektregionen vorschlägt. Dadurch konnte das Modell gezielt nach Schildern suchen, anstatt die Boxen immer an den durchschnittlichen "besten Positionen" zu platzieren, unabhängig von den tatsächlich vorhandenen Schildern. Durch den Einsatz eines ResNet-50-Backbones wurden zudem automatisch wichtige Bildmerkmale extrahiert, wodurch das Modell präziser lernen konnte.

Ein weiteres Problem war die lange Trainingszeit auf der CPU. Zu Beginn habe ich das Modell auf der CPU trainiert, was für jede Iteration extrem viel Zeit in Anspruch genommen hat. Um das Training zu beschleunigen, habe ich das Modell auf CUDA umgestellt, sodass es auf einer GPU ausgeführt werden konnte. Dies führte zu einer erheblichen Reduzierung der Rechenzeit, wodurch

ich schneller verschiedene Varianten testen konnte. Das umstellen des Modells war sehr schnell möglich, da nur einige Python-Zeilen angepasst werden mussten, die meiste Zeit bei dieser Teilaufgabe hat die eigentliche Installation des CUDA-Toolkits in Anspruch genommen.

3 Persönliche Entwicklung

Durch dieses Projekt habe ich meine Kenntnisse im Bereich der Objekterkennung vertieft, insbesondere mit Faster R-CNN. Ich habe gelernt, warum eine einfache Bounding-Box-Regression nicht ausreicht und warum ein Backbone zur Feature-Extraktion notwendig ist, um präzisere Ergebnisse zu erzielen, also dass ein neuronales Netz nicht einfach durch viele Trainingsbilder magisch Objekte erkennen kann, sondern dass es eine spezielle Struktur benötigt, um gezielt nach Objekten zu suchen.

Ein wichtiger Lerneffekt war auch die Nutzung der GPU für das Training, da ich gesehen habe, wie stark sich die Rechenzeit durch die Verwendung von CUDA verkürzen lässt. Das Projekt hat mir gezeigt, wie wichtig es ist, verschiedene Ansätze zu testen und sich nicht von Anfang an auf nur einen Lösungsweg festzusetzen, da dieser möglicherweise nicht zum gewünschten Ergebnis führt.