

Abgabe

Digitale Bildverarbeitung

Luisa Teresa Theilmann

Matrikelnummer: 6096076

Kurs: TFE21-2

03. Januar 2024

Aufbau der Modell Architektur

Erweitert und verbessert wurde das im Jupyter Notebook verwendete neuronale Netz. Das ursprüngliche Netz besteht insgesamt aus vier Layern: Zuerst wird ein Flatten-Layer verwendet, gefolgt von einem Dense-Layer. Danach wird ein Dropout-Layer hinzugefügt, um schließlich das neuronale Netz mit einem weiteren Dense-Layer zu vervollständigen. Das Netz enthält 1.013.002 Parameter und bei einer Lernrate von 0.0001 wird eine Testgenauigkeit von 95,89 % erreicht. Das Netz wird für 60 Epochen mit einer Batch-Größe von 1024 trainiert und mit einer Batch Size von 128 evaluiert.

Um die Testgenauigkeit zu erhöhen, werden weitere Layer in das vorhandene Netzwerk eingefügt. Zunächst werden zwei Layer am Anfang des Netzes hinzugefügt. Der erste Layer ist ein Convolutional 2D Layer (Conv2D), der lokale Muster und Merkmale der Eingabedaten erkennt. Direkt hinter dem Conv2D-Layer befindet sich ein Pooling-Layer (Max Pooling) als zweite Schicht. Diese Schicht trägt zur Ermittlung eines maximal repräsentativen Wertes für einen lokalen Bereich bei und dient somit der Dimensionalitätsreduktion. Durch die Aggregation lokaler Bereiche kann die räumliche Dimension der Daten verringert werden. Dadurch wird auch die Anzahl der Parameter sowie die Rechenlast in den folgenden Schichten reduziert. Das Neuronale Netz besteht aus insgesamt sechs Layern und 592.074 Parametern. Unter denselben Trainings- und Evaluationsbedingungen wie oben beschrieben, weist diese Architektur eine Testgenauigkeit von 98,95 % auf.

Durch das Hinzufügen eines weiteren Conv2D-Layers als zweite Schicht wird die Parameteranzahl auf 437.482 verringert und die Test Accuracy erneut auf 99,16 % angehoben. Diese Schicht trägt dazu bei, weitere räumliche Hierarchien zu lernen.

Indem an dritter Stelle eine dritte Conv2D-Schicht hinzugefügt wird, wird die räumliche Abstraktion erhöht, wodurch die Parameteranzahl wieder auf 315.658 reduziert und die Testgenauigkeit auf 99,21 % erhöht wird.

Um das Training zu stabilisieren, wird eine Batch-Normalisierung verwendet. Dadurch erhöht sich die Anzahl der Parameter leicht auf 315.786. Außerdem wird die Testgenauigkeit auf 99,4 % erhöht.

Die Anzahl der Neuronen in den Dense-Schichten wird festgelegt. In der ersten Dense-Schicht, die insgesamt die siebte Schicht bildet, wurde bisher ein Dense-Layer mit 128 Neuronen erstellt. Wenn die Anzahl der Neuronen in diesem Layer erhöht wird, werden mehr Parameter generiert, die sich auf die Fähigkeit auswirken, komplexe Abbildungen zwischen Eingang und Ausgang zu erfassen. Wenn die Anzahl der Neuronen verdoppelt wird und somit auf 256 gesetzt wird, verdoppelt sich fast die Anzahl der Parameter auf 579.338. Allerdings kann nun eine Testgenauigkeit von 99,43 % erreicht werden. Die Anzahl der Neuronen im Dense Layer kann beliebig gewählt werden. Wenn jedoch die Anzahl der Neuronen zu hoch angesetzt wird, kann es zu Overfitting kommen. Außerdem steigt die Anzahl der Parameter stark an, während sich die Testgenauigkeit nur leicht erhöht.

Das vorgestellte Neuronale Netz mit neun Schichten bildet die Grundlage für die weiteren Versuche. Die Trainingsbedingungen des Netzes bleiben durchweg gleich: Es wird immer mit 60 Epochen, einer Batch Size von 1024 und einer Lernrate von 0.0001 trainiert. Die Conv2D-Layer werden standardmäßig mit der ReLU-Aktivierungsfunktion aktiviert und greifen auf ein 5x5 Filter zu.

Auswirkungen von Aktivierungsfunktionen auf Test Accuracy

Die Genauigkeit eines neuronalen Netzes hängt von einer Vielzahl von Parametern ab. Auch verschiedene Aktivierungsfunktionen haben unterschiedliche Auswirkungen auf die Testgenauigkeit. Im wissenschaftlichen Paper „Improvement of MNIST Image Recognition Based on CNN“¹ wird beschrieben, dass die drei verschiedenen Aktivierungsfunktionen - Sigmoid, ReLU (Rectified Linear Unit) und ELU (Exponential Linear Unit) - sich unterschiedlich auf die Genauigkeit eines Convolutional Neuronalen Netzes (CNN) auswirken. Laut der Studie weist die Sigmoid-Aktivierungsfunktion generell eine niedrigere Genauigkeit auf als ELU und ReLU. ELU und ReLU ähneln sich stark in ihrer Auswirkung auf die Genauigkeit. Allerdings weist ELU bei wenigen Epochen eine etwas höhere Genauigkeit auf. Bei mehr Epochen haben ELU und ReLU jedoch die gleichen Auswirkungen auf die Genauigkeit.

Um dieses Verhalten nachzuweisen, werden in dem oben beschriebenen Netzwerk die Aktivierungsfunktionen abgeändert und die Testgenauigkeit im Anschluss verglichen. Es werden lediglich die Aktivierungsfunktionen aller Conv2D-Layer und des ersten Dense-Layers verändert. Die Aktivierung der letzten Dense-Layer und somit auch der letzten Schicht des Modells wird nicht verändert. Dies ist notwendig, da die Softmax-Aktivierungsfunktion benötigt wird, um für jede der zehn Ziffern eine Wahrscheinlichkeit anzugeben. Wenn man die beiden genannten Layer mit der Sigmoid-Aktivierungsfunktion initialisiert, erreicht man eine Testgenauigkeit von 98,86 %. Mit der ELU-Aktivierungsfunktion erzielt man ein Ergebnis von 99,03 % und mit der ReLU-Aktivierungsfunktion 99,45 % Testgenauigkeit. Dieses Verhalten steht im Widerspruch zur in dem Paper erörterten Theorie. Da die beiden zugrunde liegenden Netze unterschiedlich aufgebaut sind, können auch die Reaktionen auf die verschiedenen Aktivierungsfunktionen unterschiedlich ausfallen.

Auswirkungen von Filtergrößen auf Test Accuracy

Die Filtergröße innerhalb eines Conv2D-Layers beeinflusst die Test Accuracy. Das Verhalten der Test Accuracy bei unterschiedlichen Filtergrößen wird in dem wissenschaftlichen Artikel „Effect of filter sizes on image classification in CNN: a case study on CFIR10 and Fashion-MNIST datasets“² beschrieben. In diesem Artikel wird beschrieben, dass ein 5x5-Filter im Vergleich zu einem 7x7-Filter oder einem 3x3-Filter die beste Testgenauigkeit liefern kann. Mit dem vorliegenden Netz werden alle drei Filter verwendet und das jeweilige Netz trainiert. Mit der Filtergröße ändert sich auch die Anzahl der Parameter. So erreicht der 5x5 Filter mit 579.338 Parametern die beste Testgenauigkeit von 99,45 %. Der 7x7 Filter hat mit 309.770 Parametern eine Test Accuracy von 99,29 %, während der 3x3 Filter mit 1.013.002 Parametern die meisten Parameter und die geringste Test Accuracy von 99,28 % hat. In diesem Fall stimmen die hier berechneten Ergebnisse mit den Ergebnissen aus dem Artikel überein.

¹<https://ijeast.com/papers/310-316,Tesma412,IJEAST.pdf>

²https://www.researchgate.net/profile/Samad-Dadvandipour-2/publication/356686393_Effect_of_filter_sizes_on_image_classification_in_CNN_a_case_study_on_CFIR10_and_Fashion-MNIST_datasets/links/61b335c5590a0b7ed6352cfc/Effect-of-filter-sizes-on-image-classification-in-CNN-a-case-study-on-CFIR10-and-Fashion-MNIST-datasets.pdf

Data Augmentation

Mit Hilfe der implementierten Augmentor Pipeline können die verschiedenen Transformationen parametrisiert werden. So kann z.B. für die Rotation der maximale Drehwinkel und eine Wahrscheinlichkeit angegeben werden, mit welcher ein Bild verändert wird oder nicht. Um die Auswirkungen der Transformationen visuell darzustellen, werden für jede Ziffer zehn durch die Augmentor Pipeline veränderte Bilder angezeigt. Je nachdem, wie stark die Parameter der einzelnen Transformationen sind, desto größer oder kleiner sind die Unterschiede zwischen den einzelnen Bildern.

Einer der größten Vorteile der Data Augmentation ist die Möglichkeit, die Größe der Eingangsdaten enorm zu reduzieren. Ursprünglich werden etwa 60.000 handgeschriebene Ziffern dem oben definierten Modell übergeben. Durch Anpassung der Trainingsdaten, der Anzahl der Epochen und der verwendeten Lernrate, kann das Netz mit Hilfe der Augmentor Pipeline auch mit einem Bruchteil der Eingangsdaten ähnliche Testgenauigkeiten erzielen. So kann mit den ersten 3000 Bildern, einer Lernrate von 0,001, einer Batch Size von 1024 und mit 300 Epochen eine Testgenauigkeit von 94,31 % erzielt werden. Sowohl die Testgenauigkeit als auch die Anzahl der Eingabedaten können unter Umständen durch die Verwendung veränderter Trainingsparameter, wie z.B. einer größeren Anzahl von Epochen, verbessert werden.