

**Kompensationsarbeit im Modul Software Entwicklung**

Master:

“Data Science & Intelligent Analytics”

Betreuer :

Huber Stephan

Autor:

Jochen Hollich | 1810837475

Datum

10.12.2019

Inhalt

[Präambel 3](#_Toc26950980)

[Einleitung 4](#_Toc26950981)

[Theoretische Beschreibung 5](#_Toc26950982)

[Praktische Ausarbeitung und Beschreibung 5](#_Toc26950983)

[Ausblick 5](#_Toc26950984)

[Anhang 5](#_Toc26950985)

[Glossar 5](#_Toc26950986)

[Quellen 5](#_Toc26950987)

# Präambel

Aufgrund des begrenzeten Seitenspektrums werden in dieser Arbeit keine Begrifflichkeiten definiert.

# Einleitung

Artificial Neuronal Networks sind von der Biologie inspirierte technische Systeme, welche wiederum einen weit verbreiteten Anwendungszweck besitzen.

Die Funktionsweise eines CNN lässt sich aus menschlicher Perspektive am leichtesten anhand eines Bild-Datensatzes erklären. Dies bedeutet nicht dass die tatsächliche Anwendung auf Bild-Materialien beschränkt ist.

Technologien aus dem Bereich Machine-Learning & AI

BSP wer ist auf den Bildern

Was soll im Newsfeed angezeigt werden

Versteckte Produkte => wie wird da ein Produkt draus

Nicht nur Algorithmus- sondern alles ausenrum

Hier noch den Podcast weiter integrieren

Google Trends

Abgrenzung

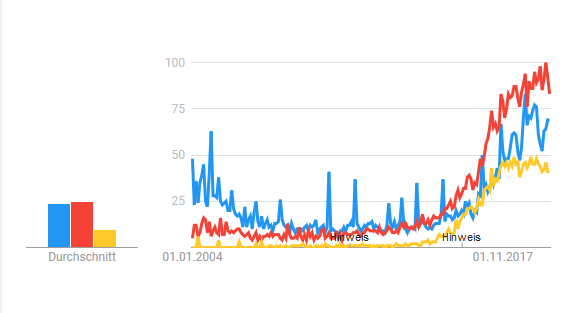
AI = Technik Maschine imitiert Menschliches Verhalten

ML = Lernen ohne Explizte Programmierung

DL = Extract Pattern using NN

ML Probleme eigentlich immer Optimierungsproblem

Ziel Vorhersagen zu Treffen, Reihe von Training dann Instanz noch nie gesehen



KI, ML, DL 2004- Heute || https://trends.google.de/trends/explore?date=all&geo=DE&q=K%C3%BCnstliche%20Intelligenz,Machine%20Learning,Deep%20Learning

Machine Learning ist ein Sammelbegriff für Technologien, welche rechenbasierte,- logische und binäre - Prozesse automatisiert. Tatsächlich sind die ersten Methoden des ML bereits vor

Konkret können wir Machine Learning nach folgenden Oberbegriffen und Ausprägungen gruppieren:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Verwendungszweck** | **Lernmethode** | **Eigenschaften** | **Learn Mgmt** | **Generalisierungsansatz** | **Single- / Multitask** |
|  |  |  |  |  |  |
| Regression | Supervised | Schnelligkeit | Online | Instancebased / lazy Leaning | SingletaskTask Learning |
| Klassifikation   1. Binäre Klassifikation 2. Multiclass Klassifikation 3. Multioutput Klassifikation | Unsupervised | Anpassbarkeit | Batch | Modelbased   1. Single Model 2. Ensemble Learning | Multitask Learning |
| Clustering | Semisupervised | Automatismus | Transfer-Learning |  |  |
| Anolmaly Detection | Reinforcement Learning | Anwendungsvielfalt | Out-Of-Core-Learning |  |  |

Diese Aufgabenfelder werden entweder durch vergleichsweise simple Algorithmen oder mittels komplexer Neuronaler Netzte ausgeführt.

Vorliegende Ausarbeitung zielt darauf ab, die Klassifikationsaufgaben mittels eines Convolutional Neuronal Networks (CNNs) zu durchleuchten. Hierzu wird zunächst im theoretischen Block die Vorgehensweise des CNNs erläutert. Anschließend wird das CNN für 3 ausgewählten Beispielen angewendet. Hierbeit wird zunächst ein binärer Bilderdatensatz MNIST, anschließend ein Farb-Bilder-Datensatz CIFAR und zuletzt Netzwerkdaten klassifiziert.

# Theoretische Beschreibung

### Ziel der KlassifikationKlassifikation

Performance-Measure

ROC / Confusion-Matrix / Accuracy

### Beschreibung NN

Ziel: Gewichte findeen, welche den geringsten Loss aufweisen

Multivariate, vektorbasierte Funktion

Bestandteile

Knoten, Gewicht, Epoche, Featurevector = Bild = reihenfolge der Pixel, Bei Text Auftreten von Worten, Layer, Eingang, Ausgang, Netzwerkarchitektur , Rechenleistung, High-Level-Feature-Extraction, Komplexe Aufgaben, Dsign(CNN, (un-)superviesed Networkt, Recurrent Neuronal Networks, Recursive NN), Axon from Neuron, Perceptron, Output Axon, Aktivierungsfunktion, Singlelayer NN, Multilayer NN, HiddenLayer, Forward, Backward Propagation, Lernrate, Input-Output Beziehung

Finetuning

SGD, ADAM, MLE

Auslagern eines Working Examples zu einem NN

### Umsetzug eines CNNs

Vergleich menschlicher Kortext

Vorgehen simple am BSP MNIST:

Bild in Vector Transformieren

Abgeflachter Vektor = > sprich von 24X24 runter und runter usw

CNN vergleicht mittels filter(Kernel) Teile eines Bildes(Patch)

Padding = gleiten des Filters über das Bild hinweg

Stride = Die Schrittweite des Bildes beim Padding

Hauptkomponeten CNN

Concolutional Layer + RELU

Pooling Layer

Fully Connected Layer

Softmax Layer

Outpur

Convlutional = Aufteilen eines Bildes in mehrere X externe Bilder

Pooling = Verkleinern des Bildstabpels

RELU = alle negative werte auf 0 setzen

Deep Learning = wiederholung des Vorgehens um N times

Fully Connected Layer = Convolutional Matirzen zu einem Vector

Parameter bei CNN

Convolutional

Anzahl der Feature

Größe der Feature

Pooling

Fenstergröße bei Sliding Windows

Stride

Fully-Connected

Anzahl der Neuonen

Architecture

Wieviel Layer

Reihenfolge der Layer

Rule of thumbs = CNN nur wenn die Feature Reihenfolge wichtig(bsp bild wenn ich Spalten des Bildes austausche, dann kommt was gänzlich anderes raus), ansons

Bsp Auslagern

BSP Translation Scaling rotation weight, vergrößern und verkleinern

Zu Konfiguierende Hyperparameter

Mögliche Archtekturen

Mögliche Anwendungen

Computer Vision

Image Analysis

Feature Extraction

Klassifiaction

Objekterekkenung

3-D rekonstruktion

Image Retrieval

Image Registation

Image Synthesis

Outdoor Analyses

Aerial & Satellite Images

Ground Truth

### Einfache Klassifikationsalgorithmen

### Komplexe Klassifikationsalgorithmen

Gradient Descent = Parameter schrittweise optimiert, Ziel Costfunction über Trainigssatz hin runter … Varianten Batch GD, Stochastic GD, Mini Batch SGD

Anwendung im NN = Kompromiss zwischen

### Evaluierungen

# Praktische Ausarbeitung und Beschreibung

MNIST

CIFAR

Netzwerkdaten

# Ausblick

Deep-Learning

Ensemble Learning

# Anhang

## Glossar

Generalisierung = Durch das Training erworbene Wissen auf neue Beispiele anwenden

Inferrenz/Interpretierbarkeit = Menschlich Nachvollziehbar

+ Lineare Regressino

* NN / SVM / RF

Prediction | Vorhersagegenauigkeit

Paramter Gewichte in Neuronalen Netzen / Support Vector

Hyperparameter = der wert der geändert wird

Objective Function = Die Funktion die optimiert werden soll || error / LossCostfunction runter

# Quellen

<https://t3n.de/news/alles-ueber-ki-wissen-t3n-1100259/>

<https://www.htwsaar.de//wiwi/fakultaet/personen/profile/selle-stefan/Selle2018e_Kuenstliche_Neuronale_Netzwerke.pdf>

https://datascience.stackexchange.com/questions/12851/how-do-you-visualize-neural-network-architectures

https://www.cbcity.de/tutorial-neuronale-netze-einfach-erklaert

Grafiken

https://trends.google.de/trends/explore?date=all&geo=DE&q=K%C3%BCnstliche%20Intelligenz,Machine%20Learning,Deep%20Learning