

Full Stack IT Nerd

[Home](#) [Kontakt](#) [PGP](#)

Lessons learned: Machine Learning

Nachdem ich in vielen Blogs und Magazinen auf das Thema “Machine Learning” gestoßen bin, habe ich mich auch mal wieder mit diesem Thema beschäftigt. Den letzten Kontakt zu Machine Learning hatte ich in meinem Studium, das schon einige Jahre her ist. Mittlerweile habe ich mich nicht nur wieder in die Grundlagen eingearbeitet, sondern habe ML auch schon in einigen Projekten einsetzen können. Einige Dinge, die ich in der letzten Zeit gelernt habe, möchte ich euch hier mitgeben:

Machine Learning ist gar nicht so schwer

Wie jede Technik ist auch Machine Learning auf den ersten Blick sehr kompliziert. Es tauchen unbekannte Fachbegriffe auf, die man nicht einordnen kann. Aber wenn man erstmal die Grundlagen verstanden hat, stellt man fest, dass das Ganze doch recht verständlich ist. Allerdings sollte man sich im Klaren sein, dass man für Machine Learning den ein oder anderen Mathematik-Skill benötigt.

Man sollte sich das Lernen nicht zu einfach machen

In vielen Blogs findet man recht einfache Tutorials, in denen die bekannten Bibliotheken Keras [↗](#), Tensorflow [↗](#) und PyTorch [↗](#) verwendet werden. Gerne werden dabei dann die MNIST-Daten [↗](#) analysiert und ein Modell zur Erkennung von handgeschriebenen Zahlen entwickelt. Das Ganze kann man dann sogar mit weniger als 100 Zeilen Python-Code bauen. Wie bei jedem Tutorial ist es aber auch hier so, dass man auf die Probleme stößt, wenn man das Tutorial verläßt und eigene Ideen umsetzt. Daher ist es unglaublich wichtig, zu verstehen, was hinter den Bibliotheken steckt. Daher habe ich auch mein erstes Modell ohne Keras, Tensorflow und Co. aufgebaut.

Man sollte sich das Lernen nicht zu schwer machen

Es gibt eine Vielzahl von Möglichkeiten sich in die Thematik einzuarbeiten. Das ist auch notwendig, da jeder Mensch ein wenig anders lernt. Während eine Person eher aus Büchern lernt, lernt eine andere Person eher aus Videokursen. Auch die Sprache ist dabei nicht unwichtig. Obwohl die aktuellsten Bücher sicherlich auf Englisch zu finden sind, sollte man eher die Sprache zum Lernen nutzen, in der man denkt. Da ich mittlerweile rausgefunden habe, wie ich am besten lerne, habe ich mehrere Videokurse auf Deutsch genutzt, um mir das Thema am Anfang zu erschließen.

Achtet auf die Daten

In den erwähnten Tutorials werden meistens sehr gut aufbereitete Daten genutzt. Verläßt man die Ebene der Tutorials und macht eigene Projekte, muss man allerdings darauf selbst achten. Das ist allerdings nicht immer so einfach. In meinem ersten eigenen Modell habe ich eine Bilderkennung für Parkplätze gebaut, um zu erkennen, welche Parkplätze belegt sind und welche nicht. Das hat sogar recht gut funktioniert. Bei genauer Betrachtung war aber ein Parkplatz in den Trainingsdaten nie belegt. Daher hat die Vorhersage für diesen Platz auch nie wirklich funktioniert. Achtet daher auf einen guten Mix in euren Daten.

Ebenfalls muss man auf die Vorhersagbarkeit der Daten achten. Man findet tatsächlich in einigen Tutorials den Code, um Aktienkurse mithilfe von Machine Learning vorherzusagen. Dabei werden nur die vorherigen Daten genutzt. Da der Aktienkurs aber nicht anhand der alten Kurse berechenbar ist, sollte man so ein Modell nicht wirklich produktiv einsetzen.

Manchmal braucht man Hardware

Für das Training reicht manchmal die eigene Hardware nicht mehr aus. Gerade, wenn man sich mit Deep Learning beschäftigt, hilft es auch andere Hardware in Betracht zu ziehen. Das Gute ist, dass es mittlerweile auch Angebote aus der Cloud gibt. Hier sind auf jeden Fall Amazon AWS SageMaker [☞](#) und Google Colab [☞](#) zu erwähnen, bei denen man seine

Modelle auf Cloud Ressourcen trainieren lässt. Zur Nutzung der Modelle sind die Hardwareanforderungen meist deutlich geringer, sodass man manche Modelle sogar auf einem Raspberry Pi betreiben kann.

Wenn du eine Frage hast oder einen Kommentar loswerden willst, kannst du mir gerne eine Nachricht schicken.

LESSONS LEARNED

- DynDNS selbst bauen
- E-Mail Migration
- Machine Learning
- TOR Onion Service

KLEINE TOOLS

- Datei-Prüfsummen ermitteln
- Eigene QR-Code generieren
- Kurz-URLs auflösen
- Serveradressen auflösen
- SSL/TLS Serverzertifikat prüfen
- Welche Header schickt der Browser?
- Zufällige Passwörter erstellen

DIVERSES

- Game of Life
- Pairs