

Maschinelles Lernen im Kontext der Programmierung natürlicher Sprachen

Seminararbeit
von

Weinmann Philipp

An der Fakultät für Informatik
Institut für Programmstrukturen
und Datenorganisation (IPD)

Betreuer: Dipl. Inform. Alexander Wachtel

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Die Regeln zur Sicherung guter wissenschaftlicher Praxis im Karlsruher Institut für Technologie (KIT) habe ich befolgt.

Karlsruhe, [FILL OUT DATE HERE]

.....
(Weinmann Philipp)

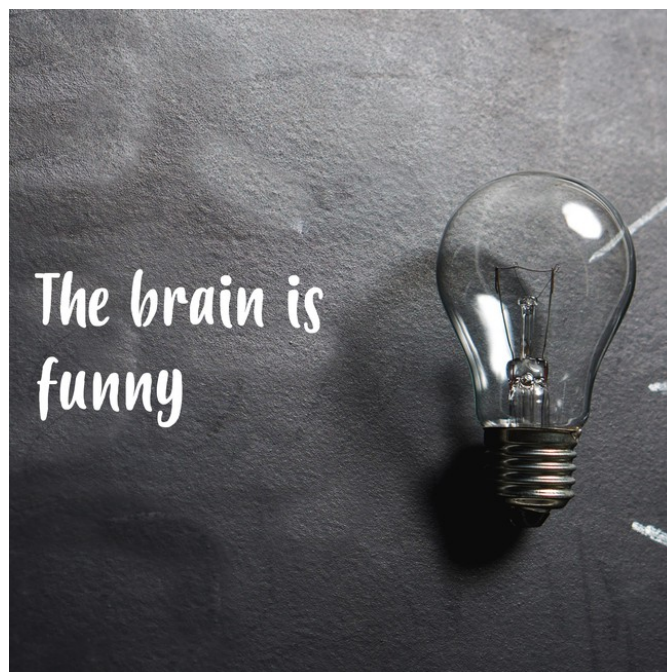


Abbildung 1: Quote die von einer KI generiert worden ist. *Die Resultate sind noch nicht zufriedenstellend* [int18]

Inhaltsverzeichnis

Abstract	ix
1 Einleitung	x
1.1 Anwendungen im Bereich der Informatik	x
1.2 Ein erstes Beispiel	xi
2 Aufkommen von Maschinellern Lernen	xii
3 Künstliche Neuronale Netze	xiii
3.1 Einführung	xiii
3.2 propagation function	xv
3.3 sigmoidfunction	xv
3.3.1 Weights and Biases	xv
3.4 Fully Connected Layers	xv
3.5 Convolution	xv
3.6 Pooling	xv
3.7 Recurrence	xv
3.8 Training of a NN	xv
3.9 Backpropagation	xv
3.9.1 Gradient descent	xv
3.10 Limitationen von Neuronalen Netzen	xv
3.10.1 Computational Power	xv
3.10.2 Debugging	xv
3.10.3 Kontrollverlust(Skynet)	xv
4 Maschine Translation	xv
4.1 Verschiedene methoden der Maschine Translation	xv
5 Neuronale Maschinenübersetzung	xvi
5.1 In der Industrie	xvi
5.2 Kuriositäten	xvi
6 Bewertung	xvi
Literaturverzeichnis	3

Abstract

[TODO] Ein Abstract ist eine prägnante Inhaltsangabe, ein Abriss ohne Interpretation und Wertung einer wissenschaftlichen Arbeit.

1 Einleitung

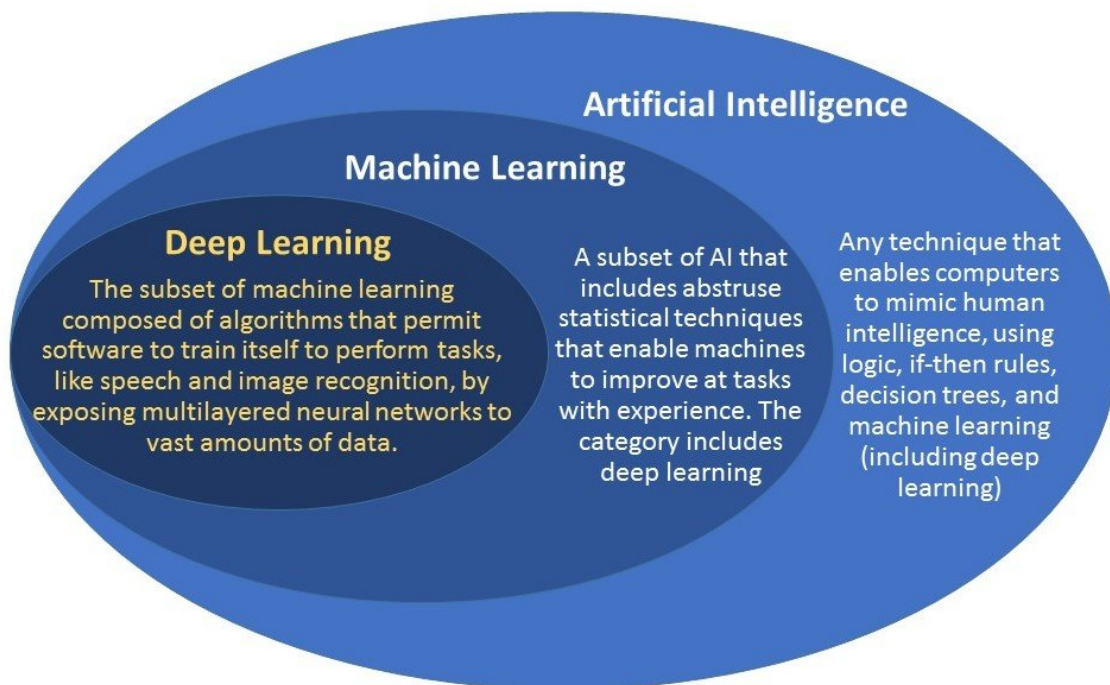


Abbildung 1: Veranschaulichung, wie Maschinelles Lernen einzuordnen ist. [Dha17]

Maschinelles Lernen (Maschinelles Lernen) ist ein Teilbereich der Künstlichen Intelligenz (KI). Es handelt sich also um eine Methode, die es Maschinen ermöglicht auf ihre Umwelt zu reagieren. Es ist jedoch oft schwierig wenn nicht unmöglich von Hand zu erkennen welche Reaktion das beste Ergebnis liefert, hier kommt Maschinelles Lernen ins Spiel. Dank statistischer Auswertungen können Algorithmen entstehen, die mit einer gewissen Wahrscheinlichkeit korrekte Ergebnisse liefern, ohne dass der Programmierer sich Gedanken machen muss, wie der Algorithmus letztendlich aufgebaut ist.

1.1 Anwendungen im Bereich der Informatik

In jedem Gebiet in dem große Mengen an Daten zur Verfügung stehen bzw. generiert werden können, ist Maschinelles Lernen theoretisch anwendbar. Weil dies auf so ziemlich jeden Bereich der Informatik zutrifft, erhofft man sich große Fortschritte von dieser neuen Technologie.

Obwohl die Theorie hinter dieser Art der Datenauswertung seit langem bekannt ist, so finden mächtigere Algorithmen die meist auf Neuronale Netze basieren erst seit kurzem verbreitete Anwendung dank verbesserter Rechenleistung. [?]

Anwendungsbereiche sind zum Beispiel:

- Gesichtserkennung
- Spamerkennung (Email)
- Spracherkennung
- Handschrifterkennung
- automatische Medikamentenentwicklung

- Maschinelle $\tilde{\text{Ä}}$ bersetzungen

In dieser Ausarbeitung werden wir uns insbesondere f $\tilde{\text{Ä}}$ jr Maschinelle $\tilde{\text{Ä}}$ bersetzungen interessieren.

1.2 Ein erstes Beispiel

H $\tilde{\text{Ä}}$ ren sie sich dieses Beispiel an: audiofile

Sie erkennen sofort, das es sich hier um spielende Kinder handelt. Unser Gehirn schafft es mit extremer Genauigkeit Komplexe ger $\tilde{\text{Ä}}$ d'usche zu erkennen und zu analysieren. Auch wenn wir nicht erkennen was jedes einzelne Kind ruft, so wissen wir instinktiv das es sich um Kinder handelt. Schauen wir uns einmal die Wellenfunktion eines Abschnittes dieses Audiofile an:

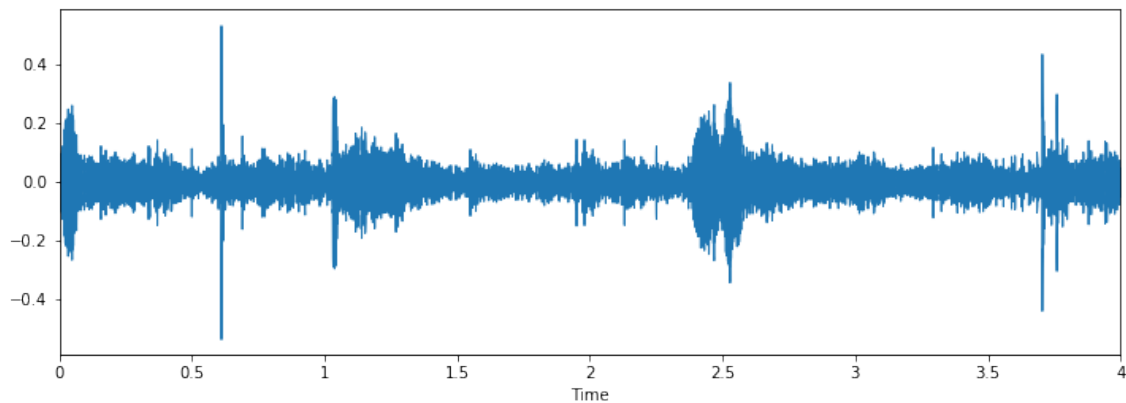


Abbildung 2: Audiofile Spielende Kinder. Luftdruck/Umgebungsdruck in Funktion der Zeit. [Shaa]

Vergleichen wir dieses mit der Wellenfunktion eines PressluftHammers:

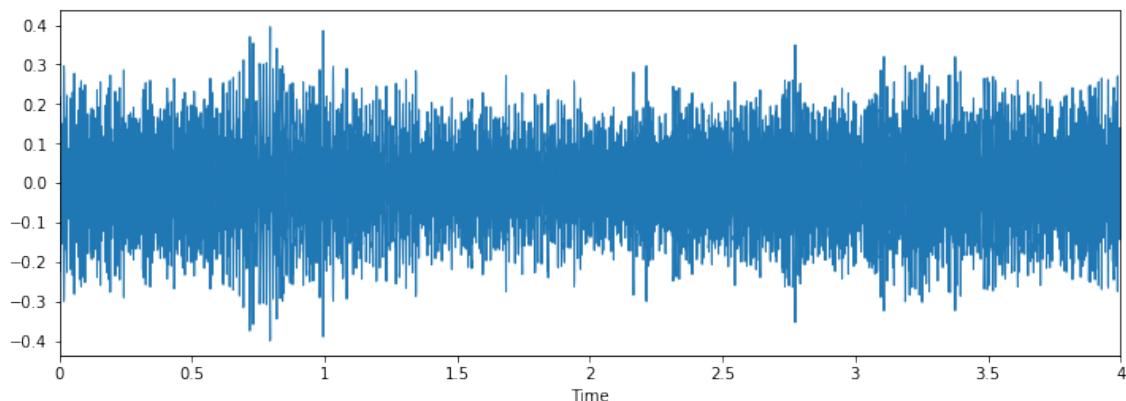


Abbildung 3: Audiofile eines pressluftHammers. Luftdruck/Umgebungsdruck in Funktion der Zeit. [Shab]

Wir bemerken einige klare Unterschiede. Falls wir nun ein Programm schreiben wollen, das erkennt welches erkennen kann ob ein audiofile eher spielende Kinder oder das Ger $\tilde{\text{Ä}}$ d'usch eines Presslufthammers enth $\tilde{\text{Ä}}$ lt, so k $\tilde{\text{Ä}}$ nnen wir Anhand der Wellenfunktion dieser Ger $\tilde{\text{Ä}}$ d'usche einige Ans $\tilde{\text{Ä}}$ tze herausfolgern. Wir bemerken zum Beispiel, das es bei kreischenden Kindern deutlich mehr ausrei $\tilde{\text{Ä}}$ ser gibt als bei dem Ger $\tilde{\text{Ä}}$ d'usch eines Presslufthammers. Wir k $\tilde{\text{Ä}}$ nnnten also ein Programm schreiben, das ein Audiofile nach der Standartabweichung der Varianz berechnet.

```
String classify(int[] audiofile) {
    int varianz = getVarianz(audiofile);
    if (varianz > (spielkinderDurchschnittsVarianz
        + pressluftHammerDurchschnittsVarianz) / 2)
    ){
        return "SpielendeKinder";
    } else {
        return "PressluftHammer";
    }
}
```

Man bemerkt jedoch, dass man hierfür schon wissen muss, welchen Wert die Varianzen von spielenden Kindern bzw von Presslufthammern im Durchschnitt annehmen. Dazu braucht man einen großen Datensatz an AudioDaten, bei denen man weiß um welche Geräusche es sich handelt. Falls diese Vorhanden sind, kann man folgendes Programm schreiben, welches den Durchschnittswert der Varianz für die jeweilige Kategorie ermittelt:

```
int mittelwertVarianz = (spielkinderDurchschnittVarianz +
    pressluftHammerDurchschnittVarianz) / 2);
String classify(int[] audiofile, String category) {
    int varianz = getVarianz(audiofile);
    if (varianz > mittelwertVarianz){
        updateVarianzen(varianz, category);
        return "SpielendeKinder";
    } else {
        updateVarianzen(varianz, category);
        return "PressluftHammer";
    }
}

void updateVarianzen(int varianz, String category) {
    if (category.equals(spielendeKinder)) {
        spielendeKindervarianzen.add(varianz);
    } else {
        pressluftHammerarianzen.add(varianz);
    }
}
```

Man bemerkt, dass dieses Programm noch immer ermittelt, welcher Kategorie die Audiofiles angehören. Daher kann es, falls bekannt ist um welches der beiden Kategorien es sich handelt weiterverwendet werden. Die Durchschnittsvarianzen werden damit mit jedem neuen Audiofile präziser, das Programm lernt mit der Zeit. Dies ist ein Beispiel sehr rudimentären Maschinellen Lernen.

Während dieses Programm sehr einfach ist, so gibt es weitaus mächtigere Programme, in denen der Algorithmus nicht nur den Wert einer Variablen erkennt, sondern auch die Kriterien zur Unterscheidung zwischen Kategorien oder sogar Kategorien selber ermittelt.

2 Aufkommen von Maschinellen Lernen

Maschinelles Lernen generell und insbesondere Neuronale Netze erfreuen sich seit einigen Jahren (Stand 2018) großer Aufmerksamkeit. Der Grundstein für diese Verfahren wurde jedoch schon Ende des 18. Jahrhunderts von Thomas Bayes gelegt[BPC63].

Während die ersten Anwendungen zum Großteil (meist abstrakter) mathematischer Natur waren[Leg05], so befasst sich schon 1913 das erste Paper mit der Analyse von Gedichten[Mar06]. Dort analysiert Markov ein Gedicht und bemerkt, dass man Wahrscheinlichkeiten formulieren kann, welche Eigenschaften weitere Teile vom Text haben ohne dass man die Gesamtheit des Gedichtes in Betracht ziehen muss.

Ab der zweiten Hälfte des 20. Jahrhunderts wurden Entdeckungen im Bereich des maschinellen Lernens gemacht, die heute jedem der mit dem Fach zu tun hat bekannt sind. 1950 formuliert Alan Turing die Turing Learning Machine [Mac50], ein Jahr später entwickeln und bauen zwei Wissenschaftler das erste Neuronale Netz [MM51]. 1957 entwickelt Frank Rosenblatt den "perceptron" [Ros58], ein erstes Modell eines fully connected neural network.

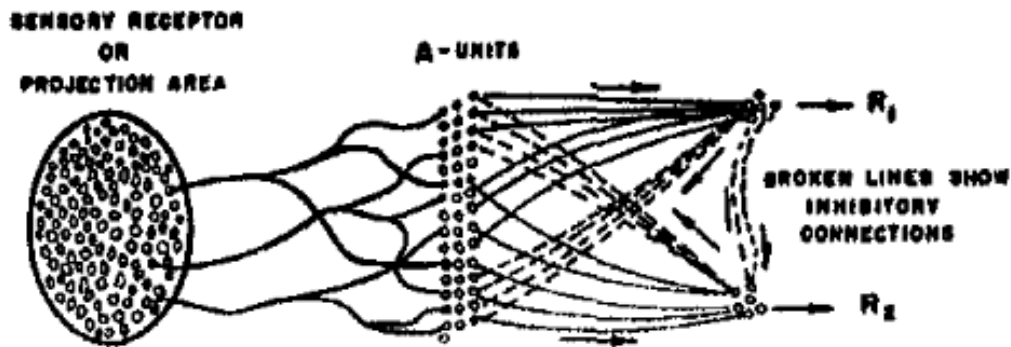


FIG. 2A. Schematic representation of connections in a simple perceptron.

Abbildung 4: original draft of a perceptron in the original paper

Nachdem [CONTINUE HERE]

3 Künstliche Neuronale Netze

3.1 Einführung

[UNINTERESSANT? NIMMT VIEL ZU VIEL PLATZ EIN; JEDER DER DAS LIEST WEIß WIE EIN NN IM PRINZIP AUFGEBAUT IST] Ein technischer Durchbruch ist oft "nur" die gelungene Nachahmung eines in der Natur vorkommenden Phänomens. Neuronale Netze kann man Vergleichen mit dem Versuch das menschliche Gehirn nachzubauen. [CITATION?] Während diese Unterfangen nicht oder nur teilweise gelungen sind, haben sich einige Nebenprodukte dieser Forschung als sehr nützlich erwiesen. Ein Neuronales Netz besteht aus Knoten die "künstliche Neuronen" genannt. Diese können auf verschiedene Weise miteinander verbunden sein. Diese Verbindungen werden oft Synapsen genannt. Jedes Neuron kann Signale empfangen und weiterleiten. Der Graph der gebildet wird nennt man die Topologie des Neuronalen Netzes.

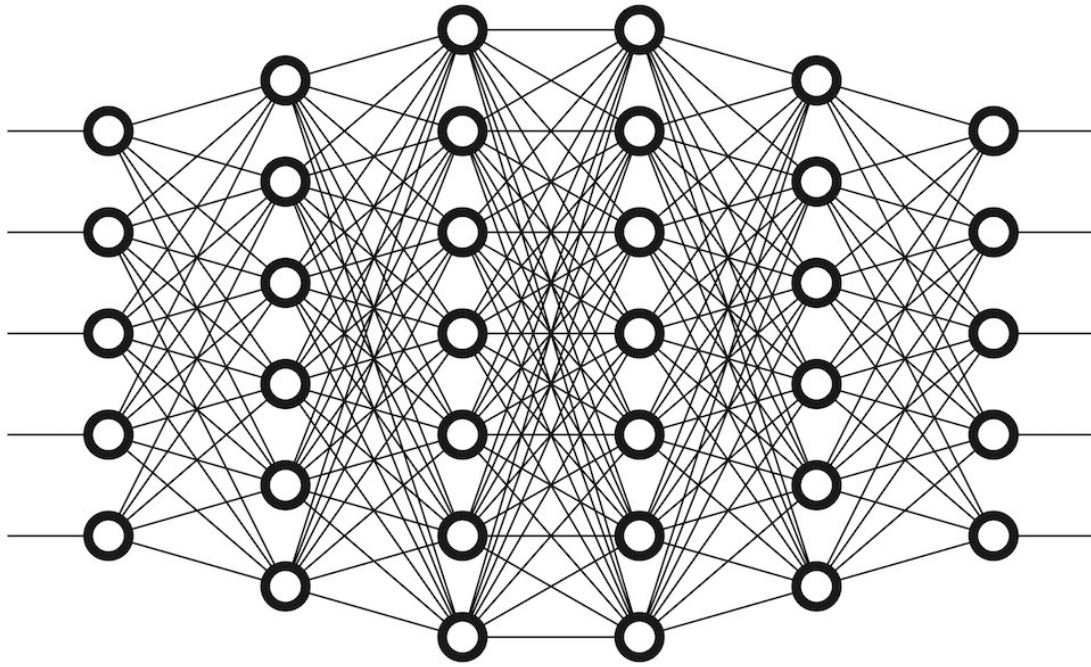


Abbildung 5: Topologie eines Fully Connected Feed Forward Neuronalen Netzes. [Neu]

Neuronen werden meist in Schichten zusammengefasst, die Stufen des Neuronalen Netzes darstellt, die mithilfe der Breitensuche bestimmt werden können.

[AB HIER WIEDER?] Wie in unserem Beispiel erwähnt, ist das Ziel von Neuronalen Netzen Eigenschaften zu erkennen, die gewisse Objekte gemeinsam haben. Neuronale Netze sollen:

- Klassifizierung
- Zusammenhänge erkennen

3.2 propagation function

3.3 sigmoidfunction

3.3.1 Weights and Biases

3.4 Fully Connected Layers

3.5 Convolution

3.6 Pooling

3.7 Reccurence

3.8 Training of a NN

3.9 Backpropagation

3.9.1 Gradient descent

3.10 Limitationen von Neuronalen Netzen

3.10.1 Computational Power

3.10.2 Debugging

3.10.3 Kontrollverlust(Skynet)

Diese Eigenschaften machen Maschine Learning zu einem Vielversprechenden Werkzeug für maschinelle AIJbersetzungen.

4 Maschine Translation

Die Anwendung von Software um Text von einer Sprache zur anderen zu AIJbersetzen.

4.1 Verschiedene methoden der Maschine Translation

- Regelbasiert
- Statistische AIJbersetzung
- Neuronale MaschinenAIJbersetzung

5 Neuronale MaschinenÜbersetzung

- Recurrent Neural Networks [Warum gerade diese?] [Elman Network, Jordan Netzwerk, Hopfield Network,...]
-

5.1 In der Industrie

- Google
- Microsoft
- Yahoo

5.2 Kuriositäten

- Facebook chatbots haben eine eigene Sprache erfinden um zu kommunizieren.
- Google Translate hat eine Sprache erfunden die als Zwischensprache dient.

Fazit: Es ist schwer bzw unmöglich die Funktionsweise von Programmen die anhand von NN netzen entstanden sind zu verstehen oder zu kontrollieren.

6 Bewertung

Test der Arbeit IA [Goo17]

Test02 [int18]

Literaturverzeichnis

- [BPC63] BAYES, Thomas ; PRICE, Richard ; CANTON, John: An essay towards solving a problem in the doctrine of chances. (1763)
- [Dha17] Aufgerufen: 03/12/18
- [Goo17] GOOGLE INC.: *Google Assistant*. <https://assistant.google.com/>. Version: 2017. – Aufgerufen: 09.01.2017
- [int18] Generated on 27.11.18
- [Leg05] LEGENDRE, Adrien M.: *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805
- [Mac50] MACHINERY, Computing: Computing machinery and intelligence-AM Turing. In: *Mind* 59 (1950), Nr. 236, S. 433
- [Mar06] MARKOV, Andreï A.: An example of statistical investigation of the text Eugene Onegin concerning the connection of samples in chains. In: *Science in Context* 19 (2006), Nr. 4, S. 591–600
- [MM51] Haven't been able to locate their original thesis, Website opened 09.12.18
- [Neu] *Neuronales Netz Bild*. <https://machine-learning-blog.de/2017/11/02/was-ist-deep-learning/>. – Aufgerufen: 09.01.2017
- [Ros58] ROSENBLATT, Frank: The perceptron: a probabilistic model for information storage and organization in the brain. In: *Psychological review* 65 (1958), Nr. 6, S. 386
- [Shaa] SHAIKH, Faizan: *audiofile kids playing*. <https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2017/08/23193324/02.png>. – Aufgerufen: 09.01.2017
- [Shab] SHAIKH, Faizan: *audiofile Presslufthammer*. <https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2017/08/23194100/jack.png>. – Aufgerufen: 09.01.2017

