

Woche 05 Theorie: Überwachtes Lernen

# Skript

Erarbeitet von  
Joana Grah

Lernziele .....	1
Inhalt .....	2
Einstieg.....	2
Lernen aus Input-Output-Paaren.....	2
Qualitätsmaße .....	3
Aufteilung in Trainings-, Validierungs- und Testdaten.....	4
Take-Home Message .....	5
Quellen .....	5
Weiterführendes Material.....	6
Disclaimer .....	6

## Lernziele

- Das Ziel und den groben Ablauf des Überwachten Lernens benennen können
- Beispiele für Input-Output-Paare im Überwachten Lernen nennen können
- Den Begriff Training im Rahmen des Überwachten Lernens einordnen und erläutern können
- Beschreiben können, wie Qualitätsmaße im Überwachten Lernen zur Quantifizierung eines erfolgreichen Trainings eingesetzt werden
- Beispiele für Qualitätsmaße nennen können
- Die Aufteilung in Trainings-, Validierungs- und Testdaten beim Training erläutern, begründen und ihre Notwendigkeit und Wichtigkeit herausstellen können

## Inhalt

### Einstieg

Heute wollen wir uns Supervised Learning bzw. Überwachtes Lernen genauer anschauen. Diese Methodik wird in Algorithmen des Maschinellen Lernens verwendet. Ein zentraler Begriff dabei ist der des Trainings.

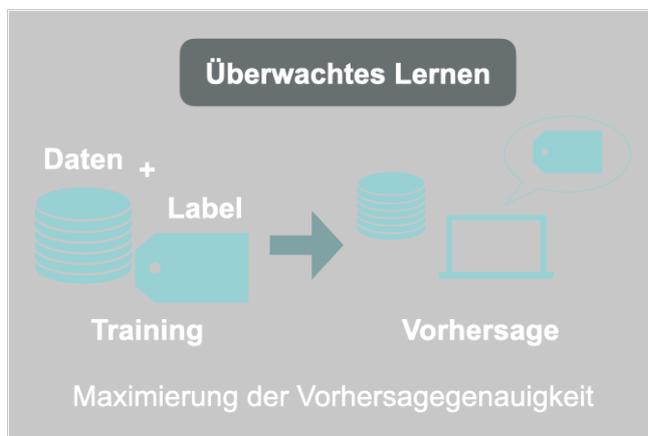
Einblendung TikTok-Video Arnold Schwarzenegger beim Training

### Quelle [1]

Schon wieder ... nein, also wir meinen mit Training die erste von zwei Phasen des Überwachten Lernens.

### Lernen aus Input-Output-Paaren

Einblendung Grafik Überwachtes Lernen



Wir möchten aus gelabelten Daten lernen, d.h. wir kennen zu jedem Input auch schon den gewünschten Output.

Einblendung Hunde- und Katzenbilder und Labels

Wenn wir zum Beispiel Bilder von Hunden und Katzen unterscheiden wollen, besitzt jedes Bild in den Trainingsdaten das Label „Hund“ oder „Katze“, was im Computerprogramm meistens durch die numerischen Werte „0“ und „1“ dargestellt wird.

Die Labels oder gewünschten Outputs können aber auch vom gleichen Datentyp wie der Input sein.

Einblendung verrauschte und dazugehörige entrauschte Bilder

Im Beispiel der Bildentrauschung sind die Input-Output-Paare verrauschte Bilder und die dazugehörigen entrauschten oder sauberen Bilder.

Ein Textverarbeitungsbeispiel ist die sogenannte Sentiment-Analyse. Dabei versucht man zu bestimmen, ob ein gegebener Text positiv oder negativ konnotiert ist.

Einblendung Tweets über KI mit positiver bzw. negativer Konnotation

## Quellen [2-5]

Inputs könnten z. B. Tweets über Künstliche Intelligenz sein und Outputs eine eher positive oder eher negative Haltung.

### Qualitätsmaße

Einblendung Grafik Überwachtes Lernen

Das Ziel des Überwachten Lernens ist im 2. Schritt, wenn wir neue Input-Daten gegeben haben, eine gute Vorhersage für die Outputs zu treffen.

Aber was heißt eigentlich gut? Das müssen wir natürlich erstmal quantifizieren und damit auch maschinenlesbar machen.

In Klassifikationsproblemen wird z. B. typischerweise die Genauigkeit oder Accuracy als Maß dafür verwendet, wie gut ein Vorhersagemodell ist. Die Accuracy beschreibt das Verhältnis der korrekten Klassifikationen zu allen vorgenommenen Klassifikationen. Werden z. B. 19 von 20 Hunde- und Katzenbildern von einem Machine-Learning-Modell korrekt mit dem Label „Hund“ oder „Katze“ versehen – oder 19 von 20 Tweets dem korrekten zugrundeliegenden Sentiment zugeordnet, liegt die Accuracy bei 95 %.

Beim Bildentrauschen brauchen wir ein anderes Qualitätsmaß, das den Unterschied zwischen einem verrauschten Bild und dessen entrauschter Version quantifiziert. In der Praxis wird häufig die sogenannte PSNR – also Peak Signal-to-Noise Ratio – verwendet und in Dezibel angegeben.

Einblendung Bildbeispiele PSNR

Je höher der Wert, desto besser die Bildqualität.

Eine immer häufiger verwendete Alternative ist der SSIM – Structure Similarity Index Measure. Wie der Name schon verrät, misst er die strukturelle Ähnlichkeit zweier Bilder.

Einblendung Bildbeispiele SSIM

Während die PSNR absolute Fehler angibt, wurde der SSIM basierend auf der Funktionalität menschlicher visueller Wahrnehmung entwickelt.

Aufteilung in Trainings-, Validierungs- und Testdaten

Einblendung Grafik Splitting in Trainings- und Testdaten

Kommen wir noch einmal zurück zum Trainingsprozess und dessen elementarem Bestandteil – nämlich den Daten! Damit wir überhaupt zwischen dem ersten und zweiten Schritt unterscheiden und für die Vorhersagequalität unseres Machine-Learning-Algorithmus sinnvolle Aussagen treffen können, müssen wir unseren zugrundeliegenden Datensatz aufteilen – in Trainings- und Testdaten. Das ist ein sehr wichtiger Schritt, der immer sorgfältig durchgeführt werden muss, um eine gute wissenschaftliche Praxis sicherzustellen. Die Menge der Trainings- und Testdaten darf sich auf keinen Fall überschneiden – die Trainingsdaten werden ausschließlich im ersten Schritt des Trainings verwendet und die Testdaten werden erst nach abgeschlossenem Training für die Evaluation im Vorhersage-Schritt benutzt. Damit stellen wir sicher, dass unser Modell generalisieren kann.

Typischerweise ist die Menge der Trainingsdaten viel größer als die der Testdaten. Vor allem für Deep-Learning-Modelle ist es wichtig, dass sowohl die Qualität als auch die Quantität der Daten hoch ist. Zum Testen genügt dann ein kleinerer Datensatz, um Vorhersagen des Modells zu evaluieren.

Einblendung Grafik Splitting in Trainings-, Validierungs- und Testdaten

In der Praxis ist es noch ein wenig komplexer: eigentlich wird meistens mit drei Datensätzen gearbeitet. Aus der Menge der Trainingsdaten wird ein kleiner Anteil für die Validierung während des Trainings verwendet – sozusagen zum Testen während des Trainings. So können bestimmte Modellparameter angepasst und optimiert werden. Noch einmal zur Sicherheit: Die eigentlichen Testdaten, die wir im Vorhinein festgelegt haben, bleiben bis zur Evaluation unberührt.

Ein wichtiger Aspekt beim Aufteilen eines Datensatzes in Training-, Validierungs- und Testdaten ist sicherzustellen, dass alle drei Mengen repräsentativ für die Gesamtmenge der Daten sind, oder mathematisch gesprochen: die gleiche Verteilung haben. Wenn wir beim Hunde-Katzen-Klassifikationsproblem einen viel größeren Anteil an Hundebildern während des Trainings haben, im Testdatensatz aber vor allem Katzenbilder vorkommen, wird unser

Modell ziemlich gut darin sein, Hunde zu klassifizieren, aber in der Evaluation wahrscheinlich nicht besonders gut dabei abschneiden, Katzenbilder zu erkennen.

### Take-Home Message

Fassen wir noch einmal zusammen, was wir heute über Supervised Learning und den Trainingsprozess gelernt haben:

- 1) Während des Trainings sieht der Algorithmus Paare von Inputdaten und dem dazugehörigen gewünschten Output.
- 2) Ziel ist es, ein Machine-Learning-Modell so zu trainieren, dass es für neue, ungesehene Input-Daten eine gute Vorhersage für die Outputs trifft.
- 3) Was gut bedeutet, bestimmen wir selbst, indem wir ein passendes Qualitätsmaß wählen.
- 4) Als Vorverarbeitungsschritt müssen die Daten in Trainings-, Validierungs- und Testdaten aufgeteilt werden.

### Quellen

- Quelle [1] arnoldschnitzel. (2019, 16. März). *Am I doing this right?* [Video]. TikTok.
- Quelle [2] Elon Musk [@elonmusk]. (2020, 18. Februar). *OpenAI should be more open imo* [Tweet]. Twitter.
- Quelle [3] Jürgen Schmidhuber [@SchmidhuberAI]. (2022, 22. November). *LeCun's "5 best ideas 2012-22" are mostly from my lab, and older: 1 Self-supervised 1991 RNN stack; 2 ResNet = open-gated...* [Tweet]. Twitter.
- Quelle [4] Timnit Gebru [@timnitGebru]. (2020, 21. Juni). *I'm sick of this framing. Tired of it. Many people have tried to explain, many scholars. Listen to us. You...* [Tweet]. Twitter.
- Quelle [5] Elon Musk [@elonmusk]. (2022, 30. Mai). *2029 feels like a pivotal year. I'd be surprised if we don't have AGI by then. Hopefully, people on Mars...* [Tweet]. Twitter.

## Weiterführendes Material

### Plus Magazine.

*Maths in a minute: Semi-supervised machine learning* (2021, 8. Oktober)  
<https://plus.maths.org/content/maths-minute-semi-supervised-machine-learning>

### SSIM.

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.

### Podcast InsideHeiCAD.

Staffel 1, Folge 7. <https://www.heicad.hhu.de/aktivitaeten/der-heicadpodcast>  
Grah, J. (Moderatorin), Dietze, S. (Gast). (2021, 22. September). #7: Wie untersucht man 10 Milliarden Tweets wissenschaftlich? [Audio-Podcast]. In *InsideHeiCAD*. Heine Center for Artificial Intelligence and Data Science.

## Disclaimer

Transkript zu dem Video „Woche 05 Theorie: Überwachtes Lernen“, Joana Grah.  
Dieses Transkript wurde im Rahmen des Projekts ai4all des Heine Center for Artificial Intelligence and Data Science (HeiCAD) an der Heinrich-Heine-Universität Düsseldorf unter der Creative Commons Lizenz [CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/) veröffentlicht. Ausgenommen von der Lizenz sind die verwendeten Logos, alle in den Quellen ausgewiesenen Fremdmaterialien sowie alle als Quellen gekennzeichneten Elemente.