

Entwicklung eines KI-Modells mittels Machine Learning

Annika Blank [REDACTED], Jennifer Mis-Lehmann [REDACTED], Anne Seidler-Wolf [REDACTED]

Einleitung

Machine Learning ist eine Anwendung von künstlicher Intelligenz, welche in der Lage ist auf der Basis von vorhandenen Daten und mit Hilfe von Algorithmen Muster und Zusammenhänge zu erkennen. Diese gewonnenen Erkenntnisse lassen sich verallgemeinert auf neue Analysen von bisher unbekanntem Daten anwenden.[1]

Voraussetzung hierfür ist, dass der Algorithmus mit vorhandenen Datensätzen versorgt und für die Analyse der Daten eine bestimmte Klassifizierung vorgegeben wird, nach der Muster erkannt werden können. Im Bereich der zerstörungsfreien Bauwerksdiagnostik ergeben sich bei der Anwendung des Impact-Echo Verfahrens Daten mit hohem Informationsgehalt, die für die Analyse und Interpretation einer bestimmten Fragestellung herangezogen werden. Die Motivation für die Analyse und Interpretation dieser Daten ist die Anwendung des Machine Learnings.

Das Potenzial hierfür liegt darin, dass die konventionelle Analyse der Impact-Echo-Daten aufgrund eines Signalmerkmals der dominanten Frequenz (Frequenz mit höchster Amplitude) genutzt wird. Die Machine Learning Technologie ist in der Lage den gesamten Signalinhalt zu nutzen und in die Auswertung einfließen zu lassen.

Entwicklung des KI-Modells

Im Rahmen dieser Arbeit wurde für die Anwendung des Machine Learnings das Modul Echolyst A.I. verwendet. Die Datensätze stammen von Impact-Echo-Messungen an Brückenfahrbahnen des NDE-Labors *Federal Highway Administration (FHWA) Advanced Sensing Technology (FAST)*. [2]

Der Versuch wurde an acht Probekörpern mit einer Länge von 3 m, einer Breite von 1 m und einer Höhe von 0,2 m durchgeführt. Die Platten wurden in Normalbeton C25/30 hergestellt. Vor dem Betoniervorgang wurden 4 verschiedene Fehlerquellen und ein künstlich erzeugter Trennriss in die Platten eingebaut. Im Abstand von 30 cm befinden sich flache Delaminationen (ausgehend von oberer Bewehrungslage), tiefe Delaminationen (ausgehend von unterer Bewehrungslage), eine Wabenstruktur sowie ein Hohlraum (Abbildung 1). Auf der Probenoberfläche wurde für die Impact-Echo-Messung ein Messraster von 100x100mm aufgetragen, wodurch 261 Gitternetzpunkte entstehen. [3]

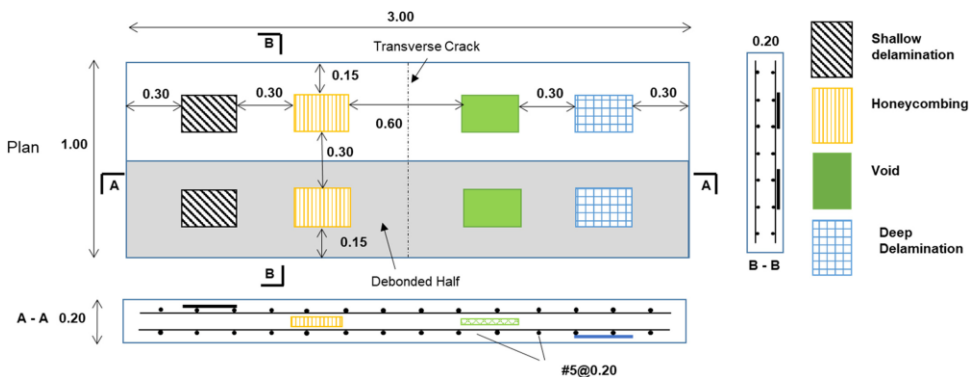


Abbildung 1: Probekörperabmessung und Anordnung der Fehlstellen
Quelle: Dorafshan, Azari, 2020 [3]

Für die eigene Bearbeitung werden die Versuchsdaten der 8 Testkörper den Impact-Echo-Messung entnommen und in das Machine-Learning-Tool Echolyst A.I. importiert.

Durch die Wahl der Verarbeitung der Daten wird die Ausgabe der Ergebnisse bestimmt. In dieser Arbeit werden die Originalsequenz, die Fast-Fourier-Transformation (FFT) und die Short-Time-Fourier-Transformation (STFT) miteinander verglichen. Die Verschiedenen Sequenzierungen unterscheiden sich vor allem in ihren Darstellungsformen.

Die Originalsequenz liefert ausschließlich das Zeitsignal der Messdaten. Diese können mittels Fast-Fourier-Transformation in den Frequenzbereich transformiert werden und liefern damit das Frequenzspektrum des gesamten Signals, jedoch keine Information über eine mögliche zeitliche Änderung. Die Short Time Fourier Transformation ist in der Lage den Frequenzgehalt in Abhängigkeit der zeitlichen Veränderung darzustellen, was insbesondere bei Impact-Echo- Daten als sinnvoll erscheint, da sich der Frequenzgehalt dieser Signale über die Zeit verändert und nicht konstant bleiben. [4]

Unter dem Punkt Explore lassen sich die unterschiedlichen Verarbeitungen für die Daten einzeln anzeigen.

Um das Training des KI-Modells zu beginnen, müssen die Versuchsdaten gelabelt werden. Die Klassifizierung der Daten erfolgt in diesem Fall durch eine binäre Zuweisung. Entsprechend dem Messraster werden die Messpunkte in die Klasse 1: Fehlstelle und Klasse 2: keine Fehlstelle eingeordnet. Weiterhin ist eine Zuweisung in weitere Klassen möglich.

Die klassifizierten Daten werden dem Algorithmus in Trainings- und Testdaten aufgeteilt übergeben. Beim sogenannten *Supervised Learning* wird ein Teil der Daten dem Training des Modells mit bekannten Zielvariablen und der andere Teil der Validierung des Modells übergeben. Um die Genauigkeit der späteren Ergebnisse zu erhöhen, kann die prozentuale Gewichtung der Trainings- und Testdaten angepasst werden. Es ist sinnvoll einen höheren Anteil an Trainingsdaten zu wählen, damit die Akkuranz des Modells erhöht wird. Jedoch müssen ausreichend Testdaten zur Validierung zur Verfügung stehen.

In dieser Arbeit wurde eine Aufteilung von 80% Trainingsdaten und 20% Testdaten, sowie eine Gewichtung von 70% und 30% gewählt. In der Übersicht werden die Trainingsdaten mit einer 0, die Testdaten mit einer 1 und die Datensätze die deaktiviert werden mit einer -1 gekennzeichnet.

Das KI-Modell wird schlussendlich durch die Machine Learning Technologie, *Deep Learning*, der Anwendung eines Algorithmus, erzeugt. Die Algorithmen, die auf neuronalen mehrschichtigen Netzen basieren, werden durch Multi-Layer-Perceptron (MLP)-Architekturen oder durch die erweiterte Form davon, der Convolutional Neural Networks (CNN's) realisiert.[2] Die CNN's werden insbesondere in der Bilderkennung angewandt, daher liegt es Nahe diese auf die Sequenzierung der Short-Time- Fourier-Transformation anzuwenden.[5] Mittels der STFT können Spektrogramme erzeugt werden, die eine bildbasierte Darstellung der Zeit-Frequenz-Inhalte der Amplitude visualisieren. Die Algorithmen laufen über die Funktionen der Python-Bibliothek Tensorflow, die sich in der Deep-Learning-Community etabliert hat. [6]

Nach Wahl des Algorithmus, sprich MLP oder CNN, lassen sich weitere Anpassungen vornehmen. Beispielsweise werden durch eine höhere Anzahl an Epochen die Durchgänge der Trainingsdaten gesteigert. So kann die Genauigkeit des Modells angepasst werden. [7] Im Folgenden werden die Daten mit Epochen von 30 bzw. 100 verarbeitet und verglichen.

In dem die Daten dem Algorithmus übergeben werden, kann eine Lernkurve des KI-Modells erzeugt werden. Mithilfe dieser wird die Trainingsgenauigkeit und die Genauigkeit der ausgewerteten Testdaten grafisch dargestellt. Durch Anpassung der vorher genannten Punkte wird versucht, dass die Trainings- und Validierungskurve auf einem hohen Niveau liegen und gegen 100% streben.

Es ist jedoch darauf zu achten, dass ein *Overfitting* vermieden wird. Dabei wird das Modell an die Trainingsdaten überangepasst und die Vorhersage neuer Daten ist nicht aussagekräftig. [8]

Wenn die Anpassungen zu einer zufriedenstellenden Lernkurve geführt haben, kann das trainierte KI-Modell auf die Daten angewandt werden. Es erfolgt die Ausgabe der Ergebnisse in grafischer Form, wodurch die Lage der Fehlstellen ablesbar wird.

Auswertung

Nach Auswertung der Lernkurven lässt sich feststellen, dass die Verarbeitung als STFT unter Anwendung der Algorithmus Architektur des Convolutional Neural Networks (CNN) die genauesten Ergebnisse liefert (Tabelle 1/2). Eine Erklärung hierfür liefert die Darstellungsstruktur der Verarbeitung. Diese basiert auf einer Zeit-Frequenz-Darstellung welche der Impact-Echo-Erfassung entspricht.

Tabelle 1: Trainingsergebnisse bei Epochs:30

Processing	Split [%-Anteilen]	Algorithm	Lernkurve – Trainingsdaten [%]	Lernkurve – Validation (Max.-Wert) [%]
Original Sequence	80 / 20	MLP	95	77
Original Sequence	70 / 30	MLP	95	80
FFT	80 / 20	MLP	95	89
FFT	70 / 30	MLP	95	90,5
STFT	80 / 20	MLP	96	91,5
STFT	70 / 30	MLP	95	90,5
STFT	80 / 20	CNN	96	93
STFT	70 / 30	CNN	95	92

Tabelle 2: Trainingsergebnisse bei Epochs:100

Processing	Split [%-Anteilen]	Algorithm	Lernkurve – Trainingsdaten [%]	Lernkurve – Validation (Max.-Wert) [%]
Original Sequence	80 / 20	MLP	100	77
Original Sequence	70 / 30	MLP	100	81
FFT	80 / 20	MLP	98	90,5
FFT	70 / 30	MLP	97,5	90
STFT	80 / 20	MLP	100	92
STFT	70 / 30	MLP	99	92
STFT	80 / 20	CNN	100	93
STFT	70 / 30	CNN	100	93

Zudem ist festzustellen, dass eine Erhöhung der Epochen zu einer gesteigerten Genauigkeit der Lernkurve führt. Durch die höhere Anzahl der Epochen ergeben sich mehrfach wiederholte Trainingsprozesse der Daten, welche einen erhöhten Rechenaufwand zur Folge haben.

Das Splitting mit einer Aufteilung von 80/20 und 70/30 hat keine signifikanten Unterschiede geliefert.

Auch wenn das trainierte Modell für die vorliegende Problemstellung valide Ergebnisse liefert, handelt es sich um kein allgemeingültiges KI-Modell und ist daher nicht auf andere übertragbar.

Quellenangabe:

- [1] Luber; Litzel. (2016). Was ist Machine Learning? <https://www.bigdata-insider.de/was-ist-machine-learning-a-592092/>
- [2] Algernon; Feistkorn; Hagenbruch; Kicherer; Rösseler; Scherrer. (2020). Echolyst-Entwicklung eines Impact-Echo-Systems für Scanning, Analyse und Machine Learning, ZfP-Zeitung (172). www.svti.ch
- [3] Dorafshan, S; Azari, H. (2020): Automation in Construction, Elsevier, www.elsevier.com/locate/autcon
- [4] Wiggerhauser. (2005). Anwendung der Hilbert-Huang-Transformation zur Auswertung von Impact-Echo-Datensätzen. <https://www.ndt.net/search/docs.php?id=3160>
- [5] Jochen K. (2021). Convolutional Neural Network. <https://datascientest.com/de/convolutional-neural-network-2>
- [6] Campagna, (2021). Python - Bibliotheken für Machine Learning [.https://www.startboosters.de/post/die-wichtigsten-python-bibliotheken-f%C3%BCr-machine-learning](https://www.startboosters.de/post/die-wichtigsten-python-bibliotheken-f%C3%BCr-machine-learning)
- [7] Brownlee, J (2018). Difference Between a Batch and an Epoch in a Neural Network <https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/>
- [8] Hunts. (2021). Modellvalidierung: Problembereiche und Lösungen - Overfitting und Underfitting <https://rocketloop.de/de/blog/modellvalidierung-overfitting-underfitting/#section-two>