

**Hochschule Karlsruhe
Technik und Wirtschaft**
UNIVERSITY OF APPLIED SCIENCES

GENETISCHE ALGORITHMEN ZUR OPTIMIERUNG VON HYPERPARAMETERN EINES KÜNSTLICHEN NEURONALEN NETZES

Fakultät für Maschinenbau und Mechatronik
der Hochschule Karlsruhe
Technik und Wirtschaft

Bachelorarbeit

vom 01.03.2018 bis zum 31.08.2018
vorgelegt von

Christian Heinzmann

geboren am 18.02.1995 in Heilbronn
Matrikelnummer: 52550

Winter Semester 2019

Professor	Prof. Dr.-Ing. habil. Burghart
Co-Professor	Prof. Dr.-Ing. Olawsky
Betreuer FZI:	M. Sc. Kohout

Eigenständigkeitserklärung und Hinweis auf verwendete Hilfsmittel

Eigenständigkeitserklärung und Hinweis auf verwendete Hilfsmittel. Hiermit bestätige ich, dass ich den vorliegenden Praxissemesterbericht selbständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen des Berichts, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

Datum: _____ Unterschrift: _____

Danksagungen

An dieser Stelle möchte ich mich zuerst bei dem Forschungszentrum Informatik bedanken, durch die ich die Möglichkeit bekommen habe, mich in einem innovativen Forschungszentrum auf meiner persönlichen und fachlichen Ebene weiter zu entwickeln. Außerdem bedanke ich mich bei sämtlichen Kollegen der Abteilung Embedded Systems and Sensors Engineering, die mich jederzeit hilfsbereit und kompetent unterstützt haben. Mein besonderer Dank gilt Herrn Lukas Kohout, welcher sich als Betreuer immer die nötige Zeit nahm, Sachverhalte zu erklären und meine Fragen zu beantworten.

Ausschreibung

BACHELORARBEIT

Genetische Algorithmen zur Optimierung von Hyperparametern eines künstlichen neuronalen Netzes

Zur Vermeidung der weiteren Ausbreitung von multiresistenten Keimen in Einrichtungen des Gesundheitswesens (insb. Krankenhäuser) werden am FZI Systeme und Methoden entwickelt, welche helfen sollen die Händehygiene-Compliance von Mitarbeitern dort zu erhöhen. Durch technische Unterstützung bei häufig wiederkehrenden Maßnahmen, soll mehr Zeit geschaffen und gleichzeitig auf die Wichtigkeit von Desinfektionsmaßnahmen aufmerksam gemacht werden. Dies soll mit Hilfe von Augmented-Reality umgesetzt werden. Dabei ist das Ziel, bekannte Prozesse, wie beispielsweise den Wechsel von postoperativen Wundverbänden, visuell zu unterstützen und automatisch zu dokumentieren.

AUFGABEN

Im Kontext der Detektion von Aktionen und Objekten, soll die Optimierung von Neuronalen Netzen mit Genetischen Algorithmen durchgeführt werden. Das Ziel hierbei ist es, ein Framework zu entwickeln und zu implementieren, in welchem künstliche neuronale Netze automatisiert trainiert werden. Unter anderem sollen die Hyperparameter mit Hilfe von Genetischen Algorithmen intelligent angepasst und das trainierte Netz anschließend ausgewertet werden. Diese Aufgaben sollen voll automatisiert ablaufen. Daraus ergeben sich folgende Aufgaben:

- Literaturrecherche über aktuelle Genetische Algorithmen und aktuelle Neuronale Netze
- Einarbeiten in vorhandene Frameworks für Genetische Algorithmen und Neuronale Netze
- Konzeptionierung und Implementierung des ausgewählten Ansatzes zur Optimierung von Hyperparameter des Neuronalen Netzes
- Evaluation und Auswertung speziell unter der Beachtung geringer Datenmengen
- Wissenschaftliche Aufbereitung und Dokumentation des Projekts

WIR BIETEN

- Aktuelle Softwaretools im täglichen wissenschaftlichen Einsatz
- eine angenehme Arbeitsatmosphäre
- konstruktive Zusammenarbeit

WIR ERWARTEN

- Grundkenntnisse in maschinellem Lernen
- Kenntnisse in folgenden Bereichen sind von Vorteil: Python, Tensorflow, C/C++
- selbständiges Denken und Arbeiten
- sehr gute Deutsch- oder Englischkenntnisse
- Motivation und Engagement

ERFORDERLICHE UNTERLAGEN

Wir freuen uns auf Ihre PDF-Bewerbung an Herrn Lukas Kohout, kohout@fzi.de, mit folgenden Unterlagen:

- aktueller Notenauszug
- tabellarischer Lebenslauf

WEITERE INFORMATIONEN

- Start: ab sofort

Inhaltsverzeichnis

Eigenständigkeitserklärung und Hinweis auf verwendete Hilfsmittel	2
Danksagungen	2
Ausschreibung	3
1 Einleitung	6
1.1 Forschungszentrum Informatik	6
1.2 Der Forschungsbereich ESS	6
1.3 Einleitung zum Projekt	7
1.4 Aufgabenstellung	8
1.5 Motivation	8
1.6 Aufbau der Arbeit	8
2 Grundlagen	9
2.1 Einleitung	9
2.2 Genetische Algorithmen	9
2.3 Aufbau eines GA in Einzelschritten	9
2.3.1 Initialisierung der Population	9
2.3.2 Grade	9
2.3.3 Evolve	9
2.3.4 Loop	9
2.4 Künstliche Neuronale Netze	10
2.5 Aufbau eines Neurons	11
2.5.1 Eingang aka Input	11
2.5.2 Offset aka bias	11
2.5.3 Aktivierungs Funktion	11
2.5.4 Ausgang aka Output	12
2.6 Verlustfunktion aka lossfunktion	12
2.7 Optimierer alt Gradientenabstieg	12
2.8 Lern Methoden des Maschine Learnings	12
2.8.1 Überwachtes Lernen aka Supervised Learning	13
2.8.2 Unüberwachtes Lernen aka Unsupervised Learning	13
2.9 Hyperparameter	13
2.10 Zusammenfassung	13
3 Stand der Technik	14
3.1 Einleitung	14
3.2 Forschung	14
3.3 Anwendungen	14
3.4 Zusammenfassung	14

4	Konzept	15
4.1	Einleitung	15
4.2	Anforderungsanalyse	15
4.3	Zusammenfassung	15
5	Implementierung	16
5.1	Einleitung	16
5.2	Systemaufbau	16
5.3	Zusammenfassung	16
6	Evaluation und Tests	17
6.1	Einleitung	17
6.2	Testszenarien	17
6.3	Evaluation	17
6.4	Ergebniss und Interpretation	17
6.5	Zusammenfassung	17
7	Zusammenfassung und Ausblick	18
7.1	Einleitung	18
7.2	Zusammenfassung	18
7.3	Bedeutung der Arbeit	18
7.4	Ausblick	18

Abbildungsverzeichnis

1	Ablaufdiagramm Verbandswechsel [?]	7
2	Neuronales Netz [?]	10
3	Aufbau eines Neurons [?]	11

Abkürzungsverzeichnis

1 Einleitung

1.1 Forschungszentrum Informatik

„Das FZI Forschungszentrum Informatik am Karlsruher Institut für Technologie ist eine gemeinnützige Einrichtung für Informatik-Anwendungsforschung und Technologietransfer. Es bringt die neuesten wissenschaftlichen Erkenntnisse der Informationstechnologie in Unternehmen und öffentliche Einrichtungen und qualifiziert junge Menschen für eine akademische und wirtschaftliche Karriere oder den Sprung in die Selbstständigkeit.“
[?]

1.2 Der Forschungsbereich ESS

„Der Forschungsbereich Embedded Systems and Sensors Engineering (ESS) beschäftigt sich mit innovativen Technologien, Entwurfsmethoden und Anwendungen für und von eingebetteten Systemen. Von modellbasierten Entwurfsmethoden und -werkzeugen über technologieorientierte Forschung bis hin zu anwendungsorientierten Forschungs- und Entwicklungsprojekten – wir gestalten und entwickeln praxistaugliche Anwendungen rund um eingebettete Systeme und evaluieren diese.

Die breite Technologie- und Systemkompetenz aus Elektronik, Software-Engineering, Optik und Optoelektronik, Mikrosystemtechnik und Sensorik ist ein Alleinstellungsmerkmal des Bereiches. Schwerpunkte der Arbeiten bilden dementsprechend vor allem stark interdisziplinäre, Technologieübergreifende Forschungsprojekte und Anwendungen von eingebetteten Systemen in der Automobilelektronik, der Industrieautomation und im Gesundheits- und Sozialwesen.

Der Bereich ESS deckt mit seinen verfügbaren Kompetenzen dabei das komplette Spektrum der Entwicklung eingebetteter Systeme und Cyber Physical Systems (CPS) mit heterogenen Komponenten aus Mikroelektronik, Mikrooptik, Mikromechanik und Telematik ab.“ [?]

1.3 Einleitung zum Projekt

”Nach einer Studie der Charité aus dem Jahr 2015 sterben in Europa jährlich 23.000 Menschen an den Folgen einer Infektion mit multiresistenten Keimen. Die Tendenz ist dabei steigend. Hauptursache für die Ausbreitung dieser Keime, wie beispielsweise MR-SA, ist eine mangelnde Hygiene der Angestellten in den Versorgungseinrichtungen beim Umgang mit den Patienten. Gründe dafür liegen im fehlenden Problembewusstsein, der zu hohen Arbeitsdichte und damit verbundenem Zeitmangel und der mangelnden Qualifikation der beteiligten Pflegekräfte.

Ziel des Projekts HEIKE ist es neue, technikgestützte Möglichkeiten zu entwickeln, welche die behandelnden Mitarbeiter im Krankenhausumfeld bei Maßnahmen am Patienten unterstützen und dadurch deren Compliance in Bezug auf die Händedesinfektion erhöhen.

Die Grundlage bilden ein mobiler, vernetzter Desinfektionsspender sowie Augmented-Reality-Technik. Die Technologien werden in dem Projekt weiterentwickelt und in einem System integriert, welches automatisch die durchgeführten Handlungen am Patienten erkennt und basierend darauf zusätzliche Informationen zur Verfügung stellt. Schließlich werden die durchgeführten Maßnahmen automatisch im System dokumentiert, was den Verwaltungsoverhead für das operative Personal verringert.” [?]

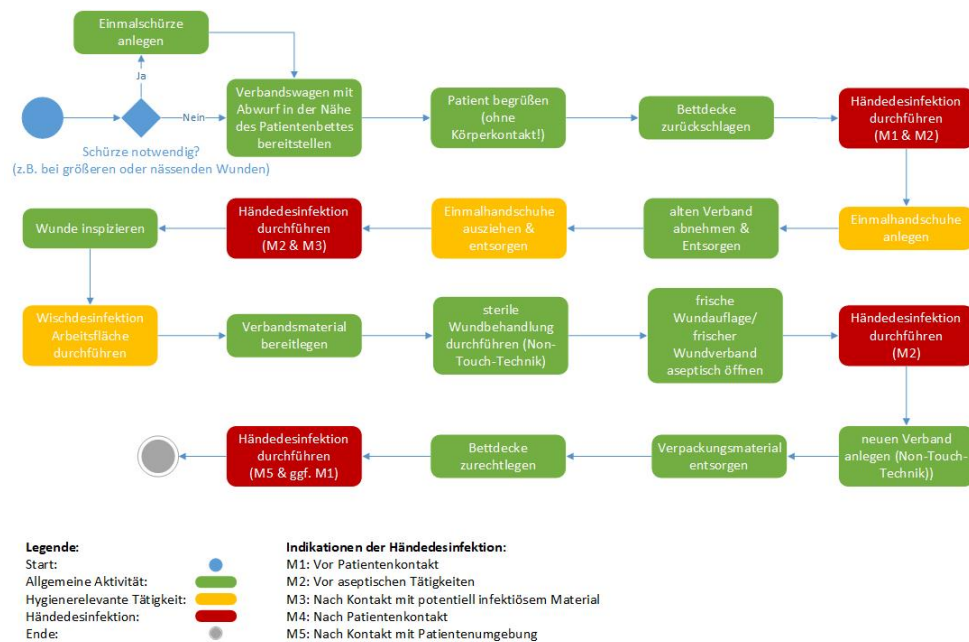


Abbildung 1: Ablaufdiagramm Verbandswechsel [?]

1.4 Aufgabenstellung

Um diese Automatischen System, meist Deep Learning Methoden, zu trainieren braucht es sehr große Datenset und viele verschiedene Parameter, welche irgendwie nach groben ermessen und herum probieren vom Programmier ausgewählt werden. Diese Auswählen beansprucht sehr viel Zeit und Mühe.

Um dies dem Benutzer zu vereinfachen soll ein Konzept geschaffen werden, welches diese Vorgänge automatisiert und zusätzlich noch mit einen Optimierungsalgorithmus verbessert. Mit diesem Ansatz kann zum Beispiel die dimensionierung eines Netzes einfacher umgesetzt werden. Ein weiter Anwendungsfall ist die Hyperparameterauswahl. Mit hilfe diesem Werkzeug soll eine einfachere und bessere Auswahl der Hyperparameter erfolgen.

Diese berechneten Werte sollen gespeichert und anschließend übersichtlich und intuitiv angezeigt werden. Wodurch sich die idealen Parameter herausbilden. Mit diesen Parameter wird möglicherweise die Richtigkeit des Netzes erhöht, sodass dieses auf ein besseres Ergebnisse kommt. Dieses Werkzeug soll Konzeptioniert und Implementiert werden. Anschließend soll eine Evaluation und Auswertung über die mögliche Verbesserung durchgeführt werden.

1.5 Motivation

1.6 Aufbau der Arbeit

2 Grundlagen

2.1 Einleitung

2.2 Genetische Algorithmen

2.3 Aufbau eines GA in Einzelschritten

2.3.1 Initialisierung der Population

2.3.2 Grade

2.3.3 Evolve

Select Parents

Breed

Exchange

2.3.4 Loop

2.4 Künstliche Neuronale Netze

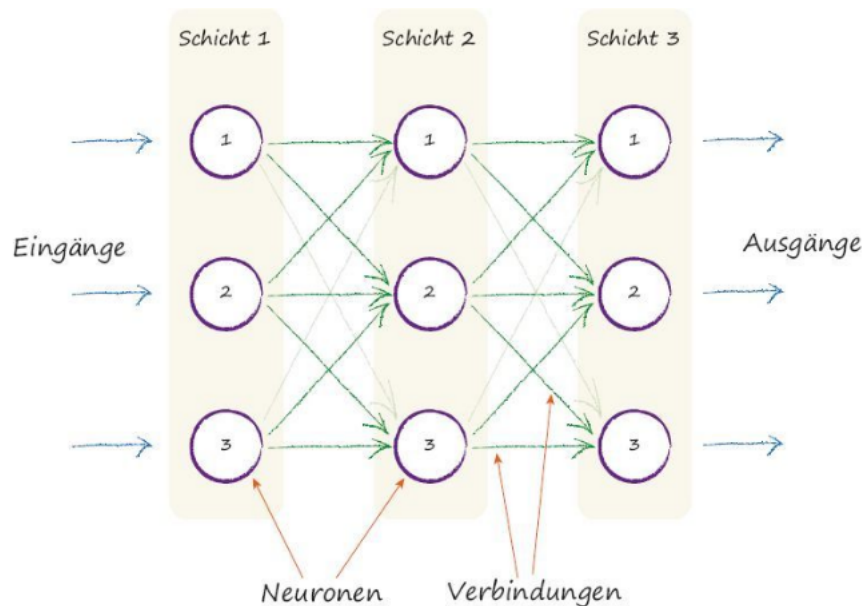


Abbildung 2: Neuronales Netz [?]

Hier zu sehen ist ein Künstliches Neuronales Netz mit drei Schichten. Dies wurde dem natürlichen Vorbild der neuronalen Netze im Gehirn nachempfunden. Die Kreise nennt man Neuronen, mehrere Neuronen zusammen ergeben eine Schicht oder auch Layer genannt. Die Verbindungen repräsentieren die Gewichte, über diese kann einem Netz verschiedene Zusammenhänge von Input und Output antrainiert bzw. angelernt werden. Zum Training werden viele Daten benötigt, aus welchen das Netz „Lernt“. Dafür ist es wichtig, viele aufbereitete Daten zu besitzen, denn diese Netze brauchen viele Trainingsiterationen, bis das gewünschte Ergebnis zustande kommt. Ein Neuron besteht aus Eingängen, Gewichten und einer Aktivierungsfunktion sowie einem Ausgang. Die Vernetzung mehrerer Neuronenschichten lässt ein Neuronales Netz entstehen.

2.5 Aufbau eines Neurons

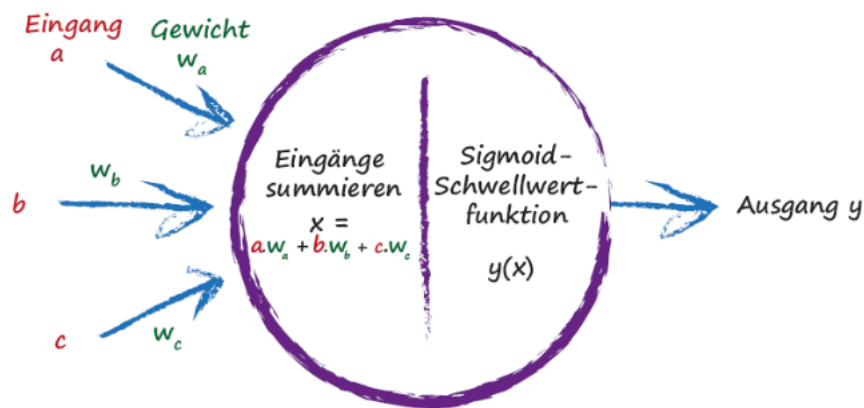


Abbildung 3: Aufbau eines Neurons [?]

2.5.1 Eingang aka Input

Bei dem Input handelt es sich um einfache xxxFloatwer dieser wird mit den einzelnen Gewichten verrechnet. Ein Neuron hat meist mehrere Eingangsgrößen welche alle zusammen mit den gewichten aufsummiert werden. Diese Werte werden zufällig initialisiert und per Training verbessert, somit handelt es sich um einen angelerten Werte, welche durch die Backproagation (Fehlerrückführung) verbessert werden.

2.5.2 Offset aka bias

Auf dieses Aufsummiertes Ergebniss wird anschließend ein Bias gerechnet, dieser führt zu einem besseren Verhalten beim Trainieren. Bei diesen Werten handelt es sich um angelernete Werte, die per Backpropagation verbessert werden und die Flexibitlität der Netze erhöht.

2.5.3 Aktivierungs Funktion

Die Aktivierungsfunktion kann man sich als Schwellwert vorstellen, ab wann das Neuron den Input weiter gibt. Es gibt verschiedene Funktionen, um diesen Schwellwert zu definieren. Je nach Aufgabe des Neuronalen Netze werden andere Aktivierungsfunktionen verwendet. Bei Klassifizierungen werden heute meist ReLu-Layer oder ein Weakly-ReLu

Layer benutzt, diese verhindern das Vanishing- bzw. Exploding- gradientproblem beim Trainieren.

2.5.4 Ausgang aka Output

Wenn der Schwellwert überschritten wird, wird am Output durchgeschaltet. Dieser Output kann entweder mit einer neuen Schicht Neuronen verbunden sein oder direkt als Ausgang gesehen werden. Über welchen man anhand von xxxVariablenwerten/Kommawerten die Von Input nach Output nennt sich ein Single-Forward-Pass. Wie hier beschrieben wird, kann ein Netz verschieden viele Layer besitzen mit verschiedenen Anzahlen von Neuronen.

2.6 Verlustfunktion aka lossfunktion

Die Verlustfunktion stellt ein ausgesuchtes Maß der Diskrepanz zwischen den beobachteten und den vorhergesagten Daten dar. Sie bestimmt die Leistungsfähigkeit des neuronalen Netzes während des Trainings und der Ausführung. Ziel ist es, im laufenden Prozess der Modellanpassung, die Verlustfunktion zu minimieren.

2.7 Optimierer alt Gradientenabstieg

Um die Fehlerfunktion zu minimieren wird als Werkzeug der Gradienten Abstieg benutzt. Diese ist nur möglich da ein Künstliches Neuronales Netz aus verketteten differenzierbaren Gewichte der Neuronen(Tensoroperationen) aufgebaut ist, die es erlauben durch anwendung der Kettenregel die Gradientenfunktion zu finden, die den aktuellen Parametern des Datenstapels werte des Gradienten zuordnet. Es gibt auch hier verschiedene Ansätze von Optimierern, welche die genauen Regeln wie der Gradient der Verlustfunktion zu Aktualisierung der Parameter verwendet wird hier könnte Beispielsweise den RMSProp-Optimierer, der die Trägheit des Gradientenabstiegsverfahren berücksichtigt. Seite 83 - Deep Learning chollat

2.8 Lern Methoden des Maschine Learnings

Es gibt vier verschiedene Lernmethoden beim Maschinellen Lernens, überwachtes, unüberwachtes, selbstüberwachtes und verstärkendes Lernen. Es werden aus Zeitgründen nur die wichtigsten zwei Besprochen.

2.8.1 Überwachtes Lernen aka Supervised Learning

Überwachtes Lernen wird dafür benutzt, eine Funktion zu finden, Daten einem Wert zuzuweisen. Dennoch müssen dafür alle Daten meist von einem Menschen vorverarbeitet werden und eine Beschriftung/Zielwert(eng. Label) zugeordnet werden. Damit das Netz auch eine Aussage über das Ergebnis, während des Trainings, geben kann. Anwendungsfälle sind Regression, Klassifikation und Empfehlungen.

2.8.2 Unüberwachtes Lernen aka Unsupervised Learning

Im Vergleich zum überwachten Lernen liegen hier keine Zielinformationen (eng. Label-information) vor. Weshalb dieser Ansatz eher zum Erkennen von Mustern und Ableiten von Regeln da ist. Für unsupervised Learning Algorithmen sind in der Regel sehr viele Daten nötig. Anwendungsgebiete für Unüberwachtes lernen sind das Clustering und die Dimensionsreduktion.

2.9 Hyperparameter

Als Hyperparameter werden, in Bezug auf KNN's, meist die Anfangsbedingungen bezeichnet. Im klassischen Fall wären dies die Learnrate (eng. Learningrate), der Abdeckungsgrad(eng. Dropout), die Verlustfunktion oder auch der Optimizer. In selten Fällen kann man die Modelarchitektur auch als Hyperparameter bezeichnen. Für diese Hyperparameter gelten keine universellen Werte sondern müssen je nach Daten und Funktion (oder KNN) speziell angepasst und verändert werden. Deshalb gibt es nur einige Regeln und grobe Abschätzungen wie diese Hyperparameter aussehen sollen.

2.10 Zusammenfassung

3 Stand der Technik

3.1 Einleitung

3.2 Forschung

3.3 Anwendungen

3.4 Zusammenfassung

4 Konzept

4.1 Einleitung

4.2 Anforderungsanalyse

4.3 Zusammenfassung

5 Implementierung

5.1 Einleitung

5.2 Systemaufbau

5.3 Zusammenfassung

6 Evaluation und Tests

6.1 Einleitung

6.2 Testszenarien

6.3 Evaluation

6.4 Ergebniss und Interpretation

6.5 Zusammenfassung

7 Zusammenfassung und Ausblick

7.1 Einleitung

7.2 Zusammenfassung

7.3 Bedeutung der Arbeit

7.4 Ausblick