

Klassifizierung Multidimensionaler Zeitreihen mithilfe von Deep Learning

Manuel Meyer

Hamburg University of Applied Sciences, Dept. Computer Science,
Berliner Tor 7

20099 Hamburg, Deutschland

Email: manuel.meyer@haw-hamburg.de

Zusammenfassung. Gegenstand dieser Arbeit ist die Vorstellung eines Lösungsansatzes zur Klassifikation von Heterogenen Sensordaten zur Vorhersage von Zeitreihen auf Basis von tiefen neuronalen Netzen. Es werden auf notwendige Grundlagen eingegangen wie der Aufbau eines Neuron, verschiedene Netztopologien und eine bereits verwendete Technik zur Prognose von Zeitreihen. Darüber hinaus werden ähnliche wissenschaftliche Arbeiten zum aktuellen Stand der Technik, die diesen Bereich betreffen, präsentiert.

Schlüsselwörter: recurrent neural network, Vorhersage, Zeitreihen, Klassifikation, convolutional neural network, Faltungsnetzwerke, deep learning, heterogene Sensordaten, multidimensional

1 Einführung

Eine Abfolge von Werten oder Daten wird auch als Zeitreihe bezeichnet. Die täglich gemessenen Temperaturwerte oder die Aktienwerte können eine Zeitreihe darstellen. Bei Beobachtungen von verschiedenen Zeitreihen wächst das Interesse wie zukünftige Werte oder Ereignisse aussehen könnten, bspw. die Wettervorhersage oder Börsenkurse. Bei der Wettervorhersage geben Sensoren in bestimmten zeitlichen Abständen Zeitsignale, welche mittels Verfahren verarbeitet und im Anschluss Aussagen über die Zukunft liefern können. Eine sicherer Prognose über die Zukunft ist zwar in der Regel nicht möglich, aber wendet man unterschiedliche Methoden an, können zeitliche Entwicklungen einigermaßen genau prognostiziert werden. Seitdem großen Aufschwung der neuronalen Netzen zählen diese sicherlich zu den stärksten Konkurrenten für die Analyse und Prognose von Zeitreihen gegenüber althergebrachte statische Verfahren. Obwohl sich diese Arbeit nicht auf ein spezifisches Anwendungsproblem bezieht, soll sie einen ersten Ansatz vorstellen, wie multidimensionale Zeitreihen auf Basis tiefer neuronaler Netze (Deep Learning) klassifiziert werden können. Ziel von Anwendungen könnten später komplexe technische Systeme sein, welche mit vielen Sensoren ausgestattet sind. Die Medizintechnik bietet ein breites Spektrum von Anwendungen, wie beispielsweise die Vorhersage einer Krankheit durch die Analyse von verschiedenen heterogenen Daten mittels Vorsorgeuntersuchungen wie

EKG, MRT, Blutuntersuchung, CT. Eine weitere Anwendungsmöglichkeit wäre ein UAV-System. Sie bestehen oftmals aus verschiedenen heterogenen Sensoren. Würde es eine Möglichkeit der Klassifikation der Sensordaten geben, könnte beispielweise ein Verfahren zur Kollisionsvermeidung entwickelt werden.

2 Künstliche Neuronaler Netze für zeitveränderliche Muster

Der Einsatz Neuronaler Netze als Methode zur Analyse- und Prognose von Zeitreihen haben sich mit der Zeit als vielversprechend erwiesen. Künstliche Neuronale Netze sind in der Lage ohne Vorwissen nichtlineare Zusammenhänge zwischen Ein- und Ausgaben herzustellen, weshalb sie sich auch in diesem Bereich als Vorgehen bewährt haben [17]. Zur Entwicklung neuer Modelle zur Vorhersage von Zeitreihen kommen dabei verschiedene Architekturen in Frage. In diesem Kapitel werden deshalb die für diese Arbeit wichtigen Netz-Typen mit ihren jeweiligen Besonderheiten vorgestellt.

2.1 Multilayer-Perzeptrons

Das Multilayer Perceptron (MLP) ist ein mehrschichtiges Feedforward-Netz (ffn) mit mindestens einer verborgenen Schicht. Dabei wird das Netz folgendermaßen unterteilt: Eine Eingabeschicht (Input-Layer), n versteckte Verarbeitungsschichten (*engl. Hidden-Layer*) und eine Ausgabeschicht (Output-Layer). Das Feedforward-Multilayer Perceptron ist so strukturiert, dass Neuronen einer Schicht mit den Neuronen der nächsthöheren Schicht verknüpft werden und eine einfache Netzstruktur bzw. Netztopologie bildet. Das Neuron einer Schicht ist immer mit allen Neuronen der darauffolgenden Schicht verbunden. Es gelten lediglich folgende Voraussetzungen: Die Eingabeschicht darf keinen Vorgänger und die Ausgabeschicht keinen Nachfolger haben. In Abb.2 ist das eben beschriebene ffn dargestellt.

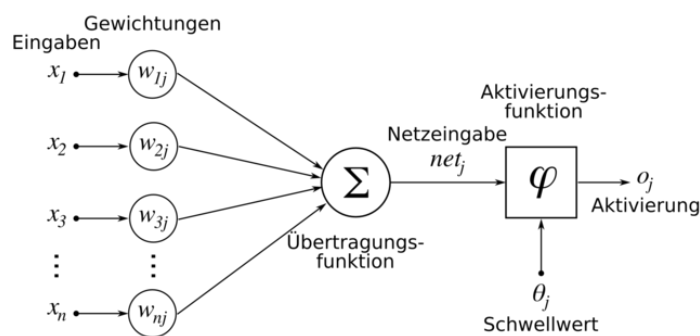


Abb. 1. Der Aufbau eines Neuron, [2].

In Abbildung 1 wird ein allgemeines Neuron als Grundeinheit der neuronalen Netze dargestellt. Die Eingabeschicht mit den entsprechenden Eingabewerten x_1, x_2, \dots, x_n mit deren Gewichten $w_{1j}, w_{2j}, \dots, w_{nj}$. Die Gewichte w_{nj} des Neurons j werden zusammen mit den Eingabewerten x_n des Neurons j multipliziert und im Anschluss aufsummiert. Auf das Ergebnis wird anschließend eine Aktivierungsfunktion angewendet, um zum Schluss den Ausgangswert o_j bestimmen zu können. Für die Netzeingabe net_j gilt dann:

$$net_j = \sum_{i=1}^n x_i w_{ij} + \theta_j, \tag{1}$$

mit θ_j als Neuron-eigenen Schwellwert.

Das Feedforward-Netz gehört zu den bekanntesten und am häufigsten eingesetzten Künstlichen Neuronalen Netzen [19]. Es wird oftmals als das Standardnetz angesehen und verstärkt bei Zeitreihenvorhersagen verwendet. Wie in Abb.2 dargestellt, handelt es sich dabei ausschließlich um vorwärtsbetriebene Netze, somit können nur auf aktuell anliegende Datensätze zugegriffen werden. Für die Prognose zukünftiger Ereignisse, kann es aber oftmals notwendig sein auch zeitlich vergangene Datensätze mit zu berücksichtigen. Im nächsten Abschnitt wird eine Technik vorgestellt die zur Prognose von Zeitreihen angewandt werden kann.

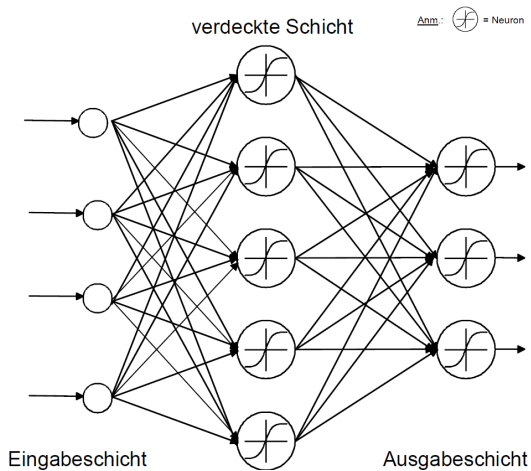


Abb. 2. Allgemeine Struktur eines Feedforward-Netz, [12].

Sliding-Window oder auch **gleitendes Fenster** ist ein Verfahren zur Prognose von Zeitreihen mittels Multilayer Perceptron. Zur praktischen Umsetzung wird das gleitende Zeitfenster der Breite n über die Zeitreihe gelegt, wobei n

den Vergangenheitswerten entspricht. Das Verfahren soll anhand der zeitlichen Entwicklung die Zukunft der Zeitreihe vorhersagen. Diese Art des Vorgehens befindet sich bereits in Anwendung sowohl in den Bereichen der Ökonomie, Finanzwirtschaft als auch zur Analyse beim Verbrauch von Strom und Fernwärme [16]. Das Anwenden der *sliding window* Technik auf Multi-Layer Perceptrons hat zur Folge das die Datensätze neben den aktuellen Werten auch alle notwendigen vergangenen Werten von Datensätze beinhalten muss. Dementsprechend muss es eine effiziente Vorverarbeitung der Daten geben, wie beispielsweise die Zusammenstellung von Werten einer vergangenen Zeitreihe als Inputvektor. Die nachfolgende Abbildung 3 zeigt, wie das *sliding window* Verfahren zur Analyse und Prognose eingesetzt werden kann, dabei ist zu erkennen, dass die letzten n Zeitsignale der Zeitreihe an die Eingabeschicht übergeben und folglich ein neuer Wert prognostiziert wird.

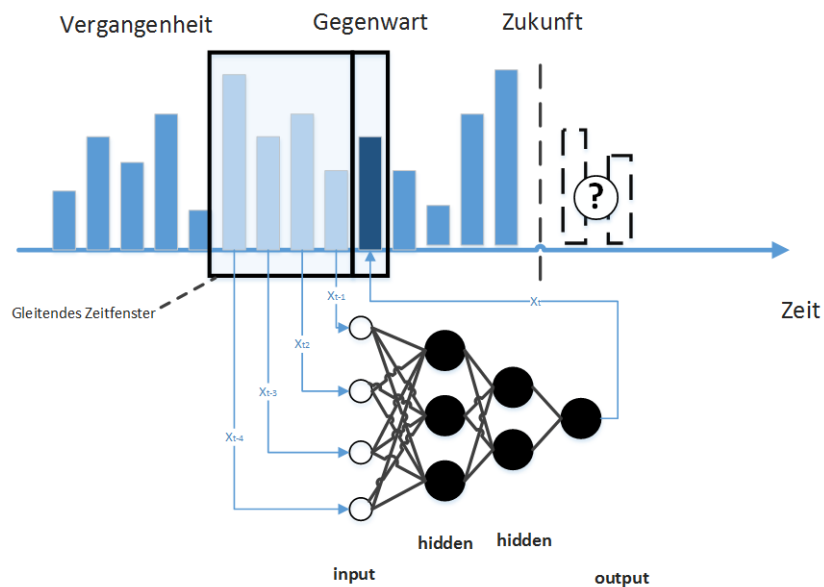


Abb. 3. Zeitreihenvorhersage mit Feedforward Multilayer Perceptron (MLP) und dem sliding window.

2.2 Recurrent Neural Networks

Im Gegensatz zu den Feedforward-Netzen können recurrent networks (RNN) von der allgemeinen Verarbeitungsrichtung abweichen, dadurch sind sie in der Lage sich mittels Rückkopplungen selbst zu beeinflussen. Das macht sie zu flexiblen und leistungsfähigen Werkzeugen zur Modellierung von Sequenzen. Beispiele für

recurrent neural networks sind: das Elman-Netz, das Jordan-Netz, das Hopfield-Netz, LSTM (Long short-term memory), sowie das vollständig verschachtelt neuronale Netz. Die recurrent neural networks (RNN) eignen sich zum Erkennen von Systematiken in Zeitfunktionen. Anwendungsfelder sind beispielsweise: Zeitreihenvorhersage (Prognosen in der Finanzwirtschaft, Wettervorhersage) [13] sowie die Spracherkennung (Sprachverarbeitung, Satzverarbeitung, Reading aloud) [1,3]. RNN lassen sich in verschiedene Arten von Rückkopplungen unterteilen, diese werden nachfolgende beschrieben und in Abb.4 beispielhaft dargestellt.

- **direkte Rückkopplungen (direct feedback):**
Der Neuronenausgang wird als weiterer Eingang genutzt.
- **indirekte Rückkopplungen (indirect feedback):**
verbindet den Ausgang mit einem Neuronen der vorhergehenden Schicht.
- **seitlichen Rückkopplungen (lateral feedback):**
Der Ausgang des Neuronen wird mit einem Neuronen der selben Schicht verbunden.
- **vollständigen Verbindungen:**
Jeder Neuronenausgang hat eine Verbindung zu jedem anderen existierenden Neuron.

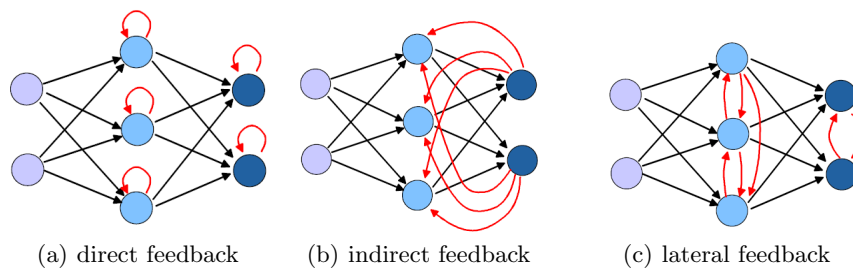


Abb. 4. Darstellung der verschiedenen Arten von Rückkopplungen der RNNs, [11] .

3 Convolutional Neuronal Networks

Bei den Convolutional Neuronal Networks (CNN) (dt. „Faltungsnetzwerk“) handelt es sich um eine Sonderform des Multilayer Perceptron mit einer speziellen Architektur. Die CNNs sind eine Kombination der Multilayer Perceptrons und der aus der Signalverarbeitung stammende Technik zur Filterung von Daten, diese Technik wird „Faltung“ genannt. Die Idee der convolutional networks (CNN) kommt aus der 80ern [4] und wurde seitdem kontinuierlich weiterentwickelt. Yann LeCun [7] konnte mit seinem Forschungsteam erstmals zeigen, dass einfache Lernalgorithmen wie der Backprogration-Lernzyklus auf die Faltungsnetzwerke angewandt werden können. Faltungsnetzwerke verzeichnen große Erfolge in den

Bereichen der Bildklassifizierung [15], darunter zählen vor allem die LeNet-5 Systeme für OCR und Handschrifterkennung [6],[8]. Die Top Wissenschaftler in diesen Bereichen sind Geoffrey Hinton und Yann LeCun.

3.1 Funktionsweise und Aufbau

Die Funktionsweise basiert auf der Filterung der Eingabedaten. Dabei durchlaufen die Daten verschiedene Schichten (Layer) des Faltungsnetzwerks. In jeder Schicht findet eine weitere Filterung statt. Mit der steigenden Anzahl von Schichten, steigert sich auch die Komplexität im Layer.

Im Bereich der digitalen Bildverarbeitung wird die Faltung bei der Bildglättung, Bildschärfung sowie als Kantensfilter angewendet. Dabei werden für die praktische Umsetzung die Eingabedaten mittels Faltungsmaske bearbeitet, womit sich bestimmte Merkmale hervorheben oder unterdrücken lassen, somit ist die Faltung eine Form der Merkmals- ‚Bildfilterung‘. Die Faltungsmaske wird auch Faltungskern oder Faltungsmatrix genannt. Abbildung 5 zeigt die Anwendung der Faltungsmaske. Es wird deutlich, wie eine entsprechende Faltungsmatrix auf einen Abschnitt des Quellbildes angewandt wird. Durch die Berechnung von korrespondierender Punkte wird im Anschluss ein neuer Pixel erzeugt.

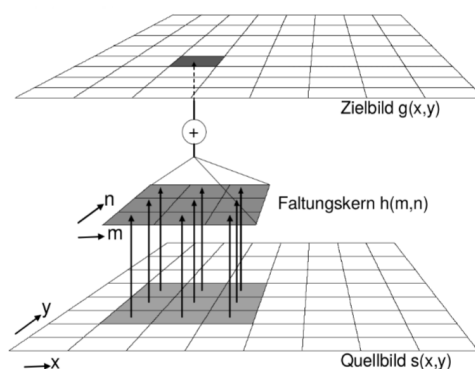


Abb. 5. Anwendungsbeispiel der Faltungsmaske, [10].

Beschreiben lässt sich die Faltung wie folgt:

$$g(x, y) = \sum_{m=-b}^a \sum_{n=-b}^b h(m, n) \cdot s(x - m, y - n) \quad (2)$$

mit $h(m, n)$ als Faltungskern.

Faltungskerne werden trainiert, dabei werden sie immer komplexer und können somit neue Features identifizieren. Bei einer Faltungsmaske kann es sich üblicherweise

um ein einzelnes Neuron mit einer entsprechenden Aktivierungsfunktion handeln.

Jedes Objekt besteht aus verschiedenen Merkmalen (Kante, Linien, Kreise), dadurch kann eine Einordnung der Merkmale eines Objektes festgelegt bzw. kategorisiert werden. Im Bereich der Objektklassifizierung bekommt das Netz zwei- oder dreidimensionale Bilder in Form von Pixeldaten als Eingabewert in die Eingabeschicht. Nach und nach werden durch die verschiedenen Schichten Ausgabewerte berechnet, welche im Anschluss einer bekannten Klasse zugeordnet werden können. Die Architektur eines CNNs besteht aus mehreren Layern, welche sich in verschiedene Aufgabenbereiche einteilen lässt:

- **C-Layer:** Der C-Layer ist der Convolutional-Layer, hier werden die Faltungsmasken auf die Eingabedaten angewandt.
- **S-Layer:** Der Subsampling-Layer rechnet die Auflösung herunter, um das Faltungsnetz stabiler gegenüber Störungen zu machen.
- **F-Layer:** Der Fully-Connected-Layer übernimmt die Klassifikation selber. Hier werden Neuronen einer Schicht mit den Neuronen alle nachfolgenden Schichten verbunden.

Weiterhin lässt sich jeder dieser Layer in verschiedenen Ebenen einteilen. Der Output dieser Ebenen wird Feature-Maps genannt. Die Feature-Maps sind Merkmalskarten und extrahieren pro Stufe immer wieder neue Merkmale aus den neuen Eingabedaten. Mit steigender Anzahl von Layern, erhöht sich auch die Komplexität Merkmale höherer Ordnung zu lernen. Des weiteren existieren mehrere Feature-Phasen, wobei jede Phase aus den folgenden drei Layer besteht: Filter Bank Layer, Non-Linearity Layer, Feature Pooling Layer. Weitere Erklärungen zu den Faltungsnetzwerken sind in der Veröffentlichung von Yann LeCun et.al. detailliert und anschaulich erklärt [8]. Die gesamte Convolutional Neural Network Architektur zeigt die Abbildung 6.

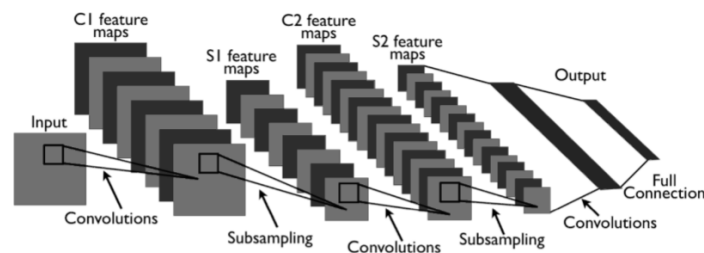


Abb. 6. Allgemeine Struktur eines Faltungsnetzes, LeCun et.al. [8].

Die Darstellung (Abb.7) stammt von der NVIDIA GPU Technology-Konferenz vom Juni 2015 aus dem Bereich Deep Learning & Artificial Intelligence und wurde von Larry Brown vorgestellt. Sie zeigt den schematischen Aufbau eines NN

mithilfe von Deep Learning. Es ist zu erkennen, dass der Aufbau verschiedene Layer beinhaltet und das sich durch die steigende Anzahl auch die Komplexität der erlernten Merkmale erhöht. Am Anfang sind nur einfache Features zu erkennen, während im letzten Abschnitt schon ganze Konturen von Gesichtern zu erkennen sind.

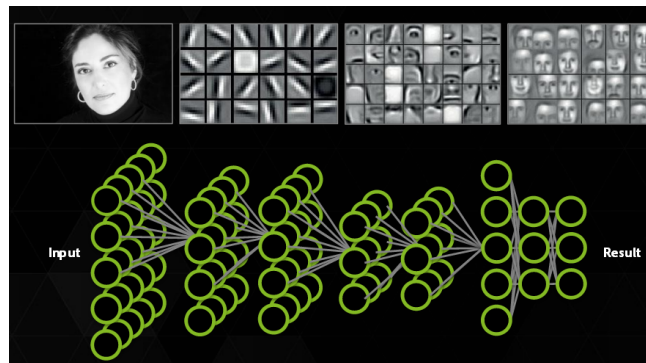


Abb. 7. Darstellung von erlernten Merkmalen eines tiefen-Faltungszusammenhanges mithilfe von Deep Learning, [14].

4 Verwandte Wissenschaftliche Arbeiten

Direkt vergleichbare Arbeiten gibt es nach dem aktuellen Stand nicht. Es kommen lediglich einige Arbeiten infrage die zum Teil als Basis dienen könnten.

4.1 Klassifikation von Zeitreihen

Aus dem Themenbereich der Klassifikation von Zeitreihen eignet sich die aktuelle Arbeit vom März 2016 „*Classifying Time Series Using Local Descriptor with Hybrid Sampling*“. In diesem Beitrag wurde auf Basis des BoW-Framework (back-of-words) Zeitreihen klassifiziert und dabei ein Hybrides-Sampling Verfahren verwendet. Das Verfahren ist eine Kombination aus einer zufälligen- und periodischen Merkmalsextraktion (feature points extraction). Das Resultat des Experiments ergab, dass mit der Verknüpfung beider Sampling Methoden bessere Ergebnisse erzielt werden konnten [20].

4.2 Klassifikation von Sensordaten

Die Arbeit „*Adaptive Activity Recognition with Dynamic Heterogeneous Sensor Fusion*“ beschäftigte sich mit der Entwicklung eines Frameworks zur Klassifizierung von Sensordaten für eine dynamische Aktivitätserkennung. In Abbildung 8

wird die Basisstruktur des Frameworks zur Aktivitätserkennung von Sensordaten mittels Sensorfusion vorgestellt. Das Forschungsteam der Carnegie Mellon University setzt dabei auf die Integration folgender Merkmals-Transformationsverfahren (Feature transformation algorithmen): Linear Discriminant analysis (LDA), Marginal Fisher's Analysis (MFA) und Maximum Mutual Information (MMI) algorithm. Das Team schreibt von vielversprechenden Ergebnissen aufgrund weniger Heterogener Sensordaten. In dem Beitrag war dabei nur einmal die Rede zwei verwendeten Sensoren: den „*motion*“ und „*location*“ Sensor. Für zukünftige Arbeiten wollen sie die Untersuchungen mit weitaus mehr Datensätzen und komplizierteren Erkennungsaufgaben prüfen und weiterentwickeln.

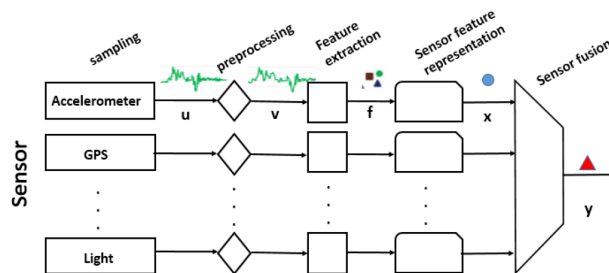


Abb. 8. Darstellung des Sensor Fusion Framework, [18].

Weitere aktuelle Veröffentlichungen ebenfalls aus dem Bereich der Aktivitätserkennung ist die Arbeit „*Multi-modal Convolutional Neural Networks for Activity Recognition*“. Bei dieser Arbeit handelt es sich um eine der ersten Arbeiten im Bereich Activity Recognition, wo zur Lösung des Problems ein convolutional neural network (mit einem 1D/2D-Kern) zum Einsatz kommt [5]. Zur Klassifizierung richtiger menschlichen Tätigkeiten wurden dafür ein: 3-Achsen-Beschleunigungsmesser und Gyroskope verwendet.

4.3 Kombination von Recurrent und Convolutional Neural Network

Das Forschungsteam von der Tsinghua University ist von den großen Erfolgen der CNNs im Bereich der Computer-Vision sehr beeindruckt. Die CNNs arbeiten auf Basis von Feed-Forward Architekturen, während aber das Visuelle-System im Gehirn mit Rückkopplungen arbeitet. Inspiriert von dieser Tatsache schlugen sie eine Kombination aus recurrent und convolutional neural networks (RCNN) für die Objekterkennung vor. Die Arbeit „*Recurrent convolutional neural network for object recognition*“ [9] beschreibt dabei zum einen wie eine mögliche Kombination mit den Stärken beider Netze aufgebaut werden kann und dass der Einsatz von rekurrenten Netzen viel mehr Vorteile einbringt als der Einsatz von einfachen Feed-Forward Architekturen. Die Grundidee zur Umsetzung war es dabei in jeder Faltungsschicht des Feed-Forward-CNN eine Rückkopplung einzubauen,

was zum einen die Tiefe des Netzes erhöht und zum anderen soll durch eine Erhöhung der Parameter eine bessere Leistung gewonnen werden.

4.4 Abgrenzung

Die vorgestellten wissenschaftlichen verwandten Arbeiten sind nur Teilaspekte und können insofern in Betracht gezogen werden, dass sie für den Anfang eine Hilfestellung oder Anregungen liefern könnten.

5 Eigene Lösungsansätze

Zur Klassifizierung von Zeitreihen multidimensionaler Sensordaten soll eine Kombination aus RNN und CNN zum Einsatz kommen. Die recurrent networks eignen sich optimal für die Speicherung und den Zugriff auf Signale von vorgehenden Datensätzen und somit zur Prognose von Zeitreihen. Die Art der Rekurrenz gibt dabei vor, wie weit Informationen in die Vergangenheit reichen dürfen. Außerdem sollen die Convolutional Neural Networks hinzugezogen werden. Durch ihre hervorragenden Erfolge vor allem im Bereich der Objektklassifizierung sollen sie gemeinsam mit den RNNs die Basis zur Zeitreihenvorhersage für Heterogene Sensorsysteme bilden. Es sollen dabei aktuelle Verfahren der Mustererkennung zum Einsatz kommen, dabei sollen auf die in Kapitel 5 verwandten wissenschaftlichen Arbeiten zurückgegriffen werden. Das Training des Netzes soll mithilfe des überwachten Lernens erfolgen.

6 Ziele für das Grundprojekt 1

Im Rahmen des Grundprojekts 1 müssen die Kenntnisse über den aktuellen Stand der Technik in den entsprechenden Bereichen vertieft werden. Dazu zählen neben der Klassifikation von Zeitreihen auch die Verarbeitung von Heterogener Sensordaten mit den aktuellen Verfahren der Mustererkennung. Dabei müssen sich besonders weitere Grundlagen der CNNs und RNNs angeeignet werden. Neben der Beschaffung von Trainingsdatensätzen wird es ein weiteres Ziel werden, ein Anwendungsszenario zu finden, welches als Grundlage der Testumgebung dienen kann. Die Testumgebung soll im Grundprojekt erstmals aus einer eigenen Trainingsumgebung mit den entsprechenden Trainings- und Testdatensätzen bestehen. Dabei sollen verschiedene Frameworks miteinander verglichen werden. Bei den Frameworks könnte es sich um folgende handeln: MatLab, Caffe, BoW-Framework, Tensorflow von Google.

Mögliche Risiken während der Durchführung des Projektes können Wechsel der Arbeitsumgebung bzw. Frameworks werden, sowie die Beschaffung der richtigen Trainingsdatensätze (Menge, Qualität, Aussagekraft). Für das Hauptprojekt soll sich auf ein Szenario festgelegt werden. Darüber hinaus soll der Aufbau eines eigenen Multisensorsystem erfolgen und somit eine Grundlage für weitere Experimente geschaffen werden.

Literatur

- [1] CHIEN, J.-T. ; KU, Y.-C. : Bayesian Recurrent Neural Network for Language Modeling. In: *IEEE Transactions on Neural Networks and Learning Systems* 27 (2016), Feb, Nr. 2, S. 361–374. <http://dx.doi.org/10.1109/TNNLS.2015.2499302>. – DOI 10.1109/TNNLS.2015.2499302. – ISSN 2162–237X
- [2] CHRISLB: *Schema eines künstlichen neurons*. https://de.wikipedia.org/wiki/K%C3%BCnstliches_neuronales_Netz#/media/File:ArtificialNeuronModel_deutsch.png. – Zuletzt besucht am 29.02.2016
- [3] FERRONI, G. ; BONFIGLI, R. ; PRINCIPI, E. ; SQUARTINI, S. ; PIAZZA, F. : A Deep Neural Network approach for Voice Activity Detection in multi-room domestic scenarios. In: *Neural Networks (IJCNN), 2015 International Joint Conference on*, 2015, S. 1–8
- [4] FUKUSHIMA, K. : Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. In: *Biological Cybernetics* 36 (1980), S. 193–202
- [5] HA, S. ; YUN, J.-M. ; CHOI, S. : Multi-modal Convolutional Neural Networks for Activity Recognition. In: *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*, 2015, S. 3017–3022
- [6] LECUN, Y. : *LeNet-5, convolutional neural networks*. <http://yann.lecun.com/exdb/lenet/>. Version: 2016. – Letzer Zugriff 10. März 2016
- [7] LECUN, Y. ; BOTTOU, L. ; BENGIO, Y. ; HAFFNER, P. : Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE* 86 (1998), Nov, Nr. 11, S. 2278–2324. <http://dx.doi.org/10.1109/5.726791>. – DOI 10.1109/5.726791. – ISSN 0018–9219
- [8] LECUN, Y. ; KAVUKCUOGLU, K. ; FARABET, C. : Convolutional networks and applications in vision. In: *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 2010, S. 253–256
- [9] LIANG, M. ; HU, X. : Recurrent convolutional neural network for object recognition. In: *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015, S. 3367–3375
- [10] MEISEL, A. : *Faltung mit einem Faltungskern*. Vorlesungsskript 'Robot Vision', 2014
- [11] MEISEL, A. : *Getaktete Netze mit $R\tilde{A}^{\frac{1}{4}}$ ckkopplung (zustandsbehaftete Netze)*. Vorlesungsskript 'Robot Vision', 2014
- [12] MEISEL, A. : *Verbindungsnetzwerk (feed forward NN)*. Vorlesungsskript 'Robot Vision', 2014
- [13] ONG, B. T. ; SUGIURA, K. ; ZETTSU, K. : Dynamic pre-training of Deep Recurrent Neural Networks for predicting environmental monitoring data. In: *Big Data (Big Data), 2014 IEEE International Conference on*, 2014, S. 760–765
- [14] PH.D., L. B.: *DEEP LEARNING WITH GPUS*. http://www.nvidia.com/content/events/geoInt2015/LBrown_DL.pdf. Version: 2015. – Letzer Zugriff 10. März 2016
- [15] SIMONYAN, K. ; ZISSERMAN, A. : Very Deep Convolutional Networks for Large-Scale Image Recognition. In: *CoRR* abs/1409.1556 (2014). <http://arxiv.org/abs/1409.1556>
- [16] VEMURI, V. R. ; ROGERS, R. D.: *Artificial Neural Networks - Forecasting Time Series*. Los Alamitos, CA, USA : IEEE Computer Society Press, 1994. – ISBN 0818651202

- [17] WIKIPEDIA: *Künstliches neuronales Netz* — *Wikipedia, Die freie Enzyklopädie*. https://de.wikipedia.org/w/index.php?title=K%C3%BCnstliches_neuronales_Netz&oldid=151898880. Version: 2016. – Letzer Zugriff 10. März 2016
- [18] ZENG, M. ; WANG, X. ; NGUYEN, L. T. ; WU, P. ; MENGSHOEL, O. J. ; ZHANG, J. : Adaptive activity recognition with dynamic heterogeneous sensor fusion. In: *Mobile Computing, Applications and Services (MobiCASE), 2014 6th International Conference on*, 2014, S. 189–196
- [19] ZHAN, G. ; PATUW, B. E. ; MICHAEL Y. H. d: Forecasting with artificial neural network. In: *International Journal of Forecasting* 14 (1998), S. 35–62
- [20] ZHAO, J. ; ITTI, L. : Classifying Time Series Using Local Descriptors with Hybrid Sampling. In: *IEEE Transactions on Knowledge and Data Engineering* 28 (2016), March, Nr. 3, S. 623–637. <http://dx.doi.org/10.1109/TKDE.2015.2492558>. – DOI 10.1109/TKDE.2015.2492558. – ISSN 1041–4347