

Analyse eines Forschungsthemas INF-D-960

Texterkennung in topographischen Karten: Untersuchungen mit dem
Deep-Learning-Framework Keras in einer HPC-Systemumgebung

Jan Stephan

Betreuender Hochschullehrer: Prof. Dr. Wolfgang E. Nagel

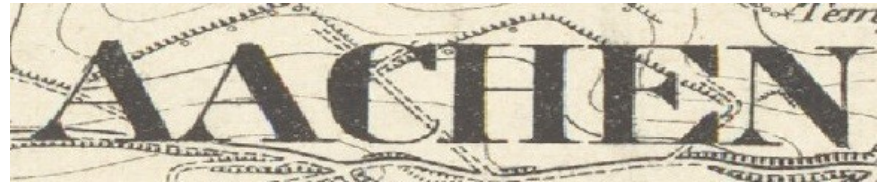
Betreuer: Dr. Peter Winkler

02. Oktober 2018

- Einleitung
 - Motivation
 - Forschungsstand
 - Ziel
- Daten und Methoden
 - Reale und künstliche Daten
 - Bilderkennung
 - Textklassifizierung
 - Umsetzung auf dem *Taurus*
- Ergebnisse
 - Trainings- und Validierungsdaten
 - Erkennungsrate und Losses
 - Performance
- Ausblick

Motivation

- Analyse historischer topographischer Karten
 - Raum- und Landschaftsplanung
 - Ziel: automatisierte Texterkennung
- Methode: *Optical Character Recognition* (OCR)
 - Automatisierte Erkennung von Kennzeichen
 - Digitalisierung von Büchern
 - Informationsextraktion aus nicht digitalisierten Dokumenten
- Problem: „Informationsflut“



● Vorstellen?

- Texterkennung und -extraktion in topographischen Karten
- Ausführung auf dem Taurus
- Skalierungsverhalten des Netzes

Daten und Methoden



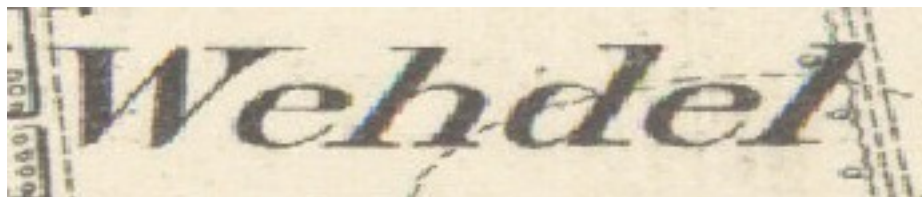
Reale und künstliche Daten

- Training mittels künstlicher Daten
- zufällige Breite und Höhe
- Datengenerator des ScaDS-Projekts



- Beschränkung auf 6 Zeichen pro Wort
- Skalierung der Eingangsdaten
 - 192 x 64
- Greyscale
- float32

● TODO: Model plot



W W W ϵ e ϵ h h d d ϵ e e ϵ l l

W e h d e l

Wehdel

Connectionist Temporal Classifier (CTC)

(Graves et al., 2006)

- Nicht nativ in *Keras* enthalten
- Teil von *TensorFlows* `tf.keras.backend`
- Training: `ctc_batch_cost` (ersetzt Loss-Funktion)
- Inferenz: `ctc_decode`

Umsetzung auf dem *Taurus*

- Training und Inferenz: **gpu2-Partition**
 - 4 NVIDIA Tesla K80
 - 62 GiB Arbeitsspeicher
- Datengenerator: beliebiger CPU-Knoten
 - in der Regel **sandy-Partition**
 - 16 CPU-Kerne
 - 30 GiB Arbeitsspeicher
- SCS5-Modulumgebung (Training und Inferenz)
 - *Keras 2.2.0-foss-2018a-Python-3.6.4*
 - *OpenCV 3.4.1-foss-2018a-Python-3.6.4*
- venv (Datengenerator)
 - `/projects/p_scads/keras/new_venv2`

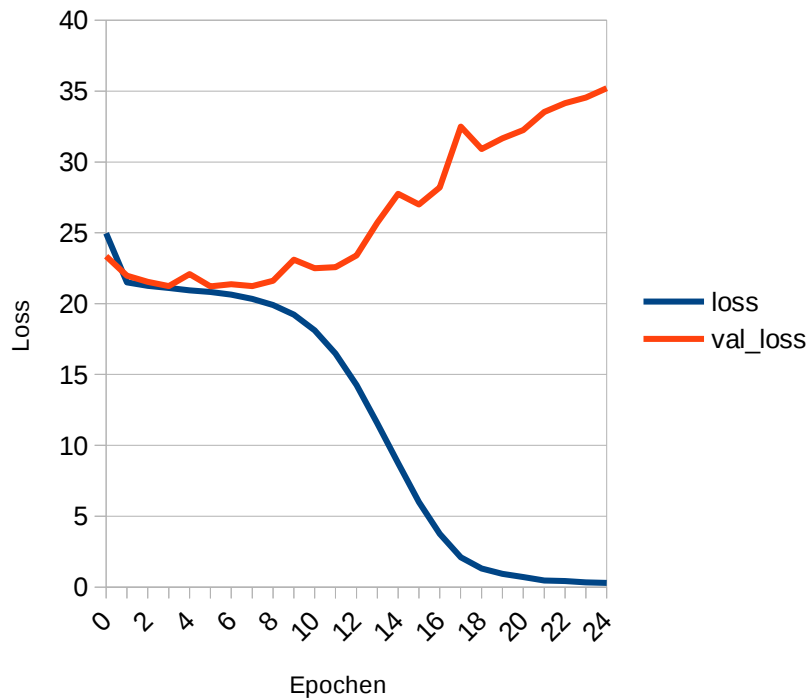
Ergebnisse

- Wordscore W
 - 0 oder 1
- Charscore C
 - korrekte Buchstaben / Wortlänge
 - zwischen 0 und 1
- Wenn $W = 1$, dann $C = 1$
- Wenn $C < 1$, dann $W = 0$
- Beispiele:
 - Real: „Bramel“. Erkannt: „Bramel“. $W = C = 1$
 - Real: „Tannen“. Erkannt: „Tunen“. $W = 0$, $C = 0,8333$
- Sonderfall:
 - Real: „Wehdel“. Erkannt: „NWehdel“. $W = C = 0$

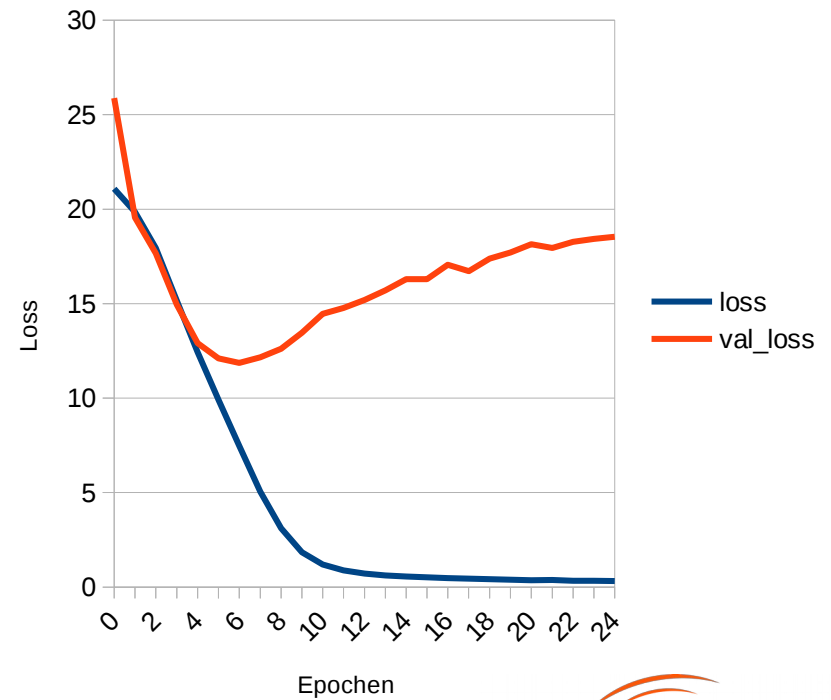
Overfitting

- allgemeines ML-Problem
- CTC besonders anfällig
- Kriterium: Validierungsloss

Loss - 1.000 Bilder



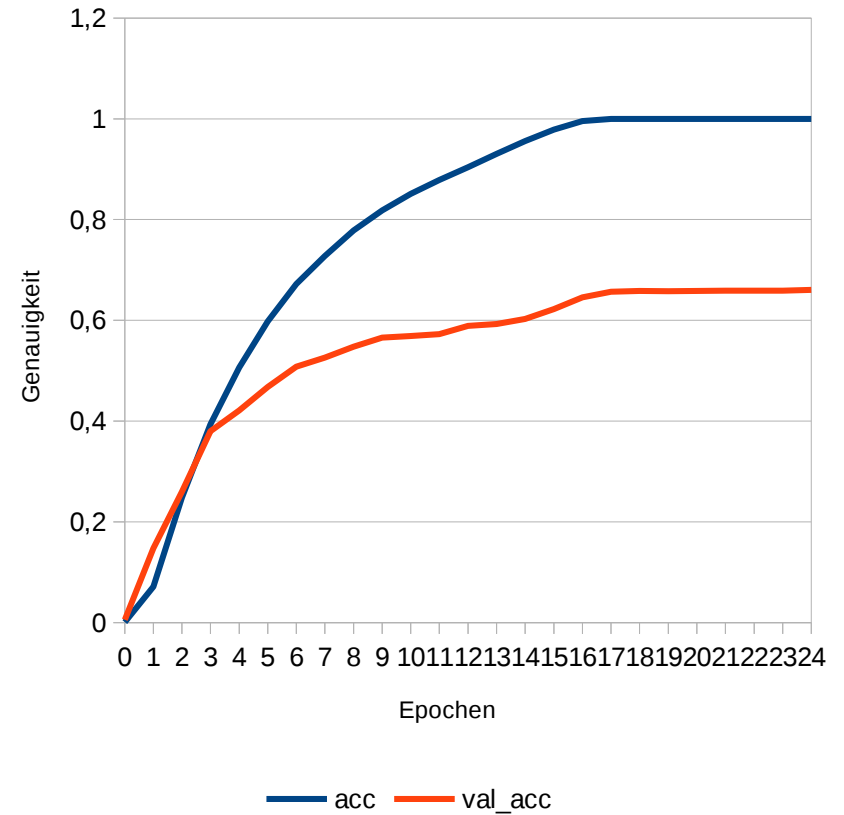
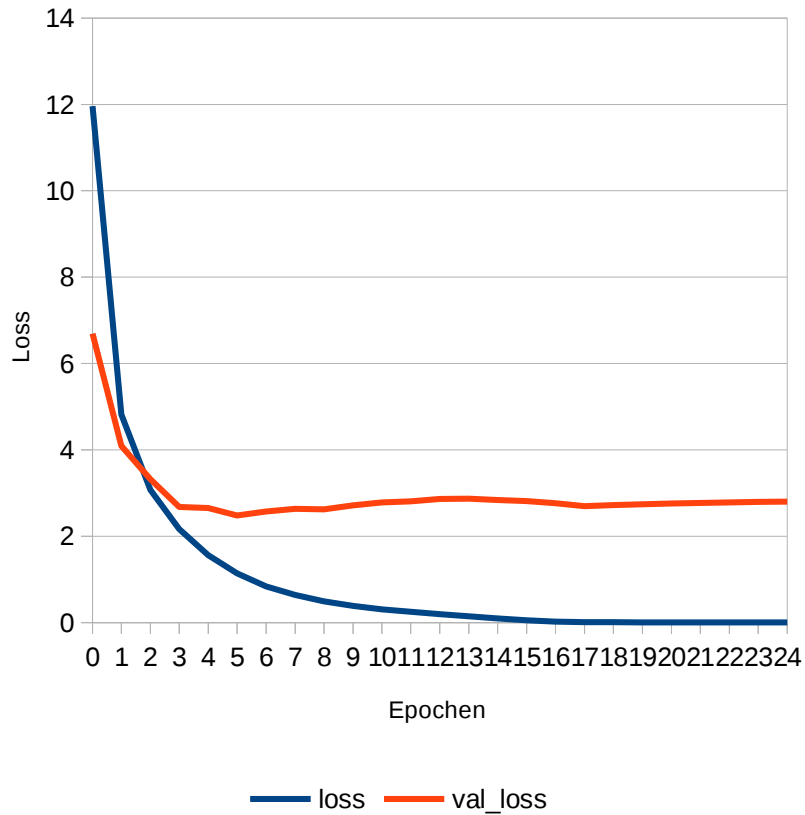
Loss - 10.000 Bilder



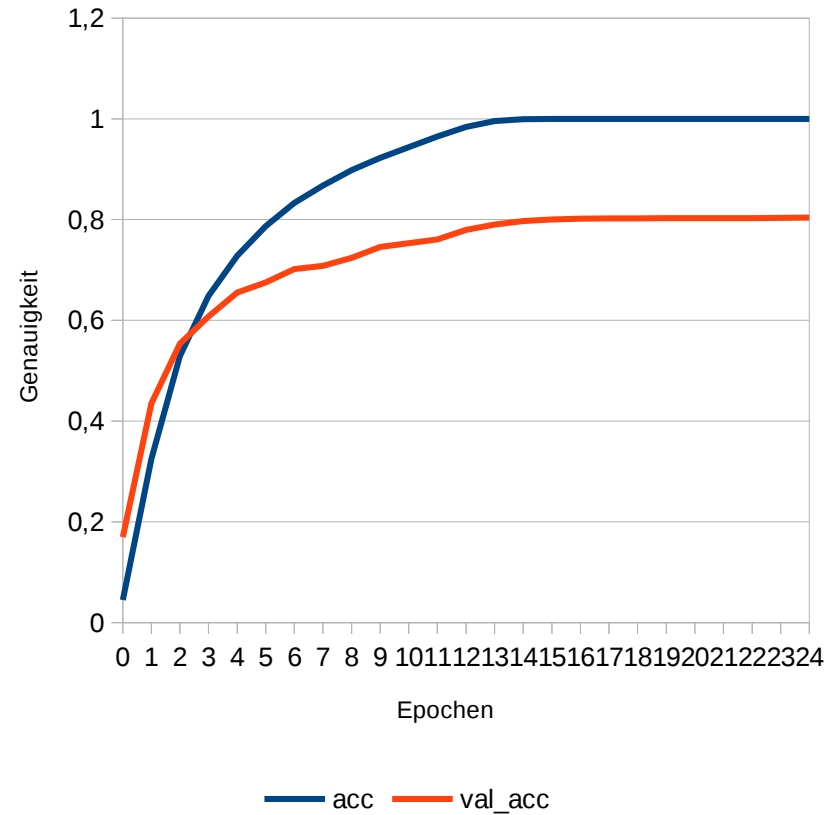
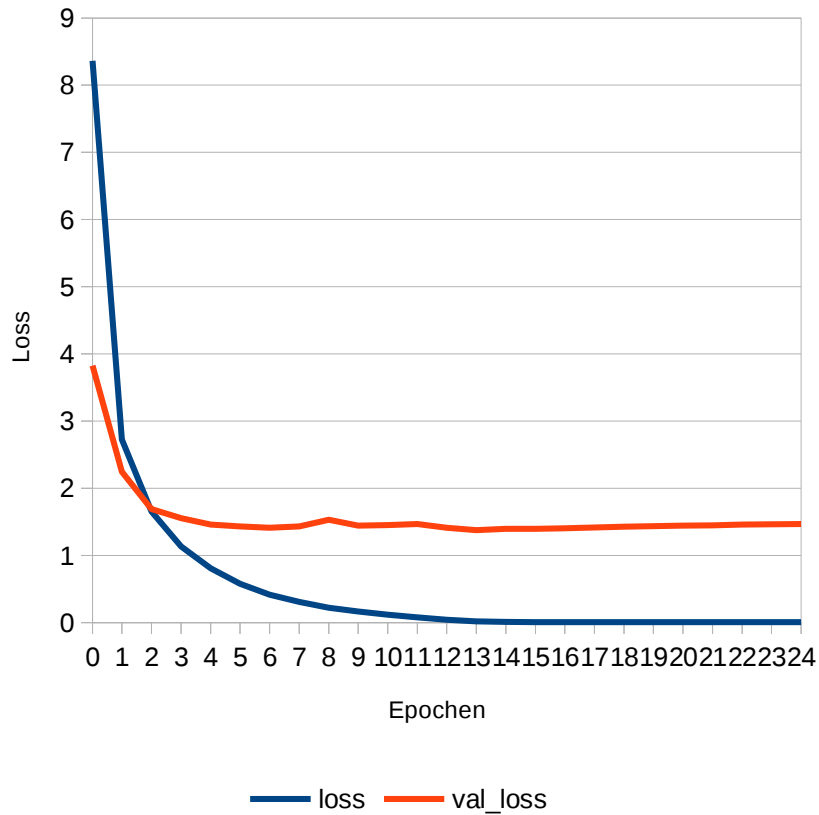
Trainings- und Validierungsdaten

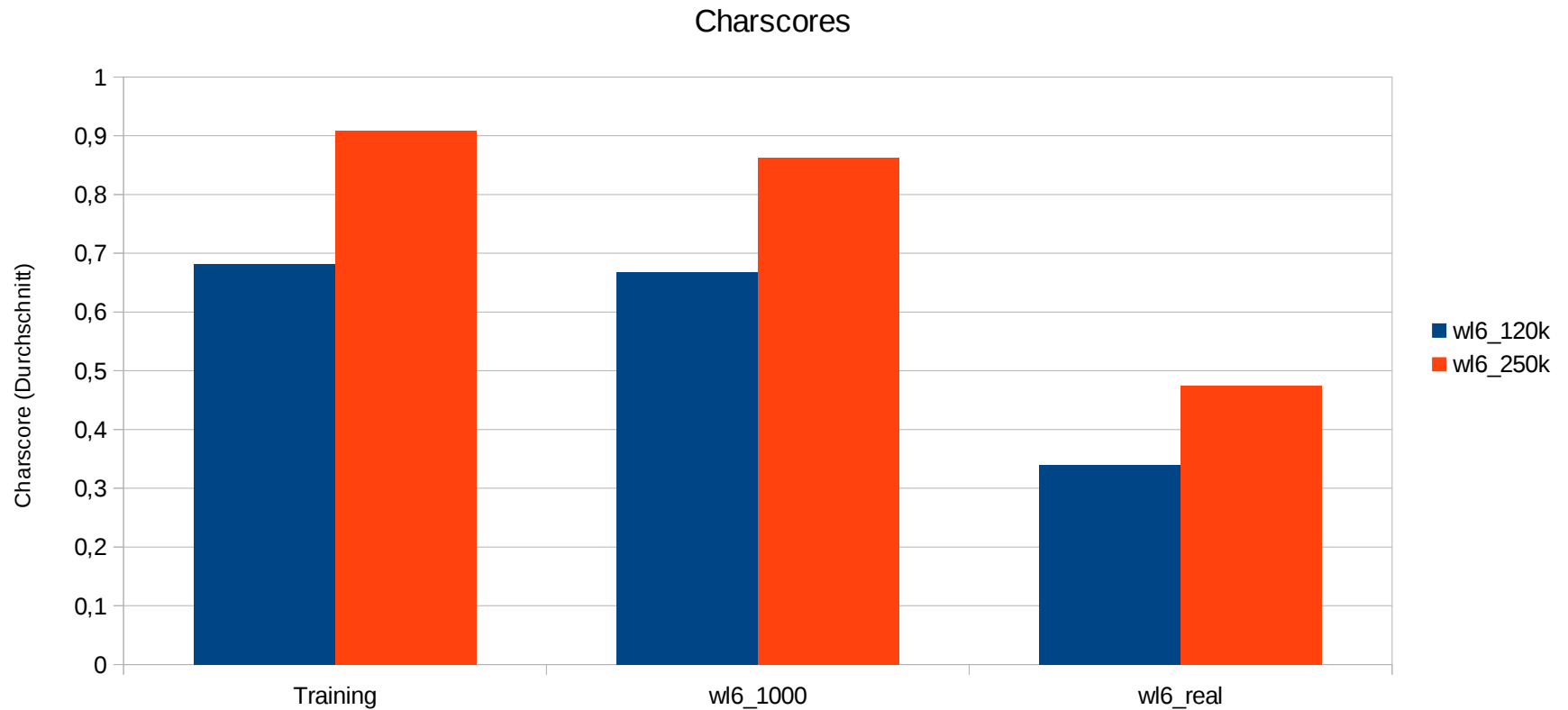
- Zwei Datensätze für das Training
 - Wortlänge 6 Buchstaben
 - wl6_120k: 120.164 generierte Bilder
 - wl6_250k: 250.000 generierte Bilder
- Zwei Datensätze für die Validierung
 - Wortlänge 6 Buchstaben
 - wl6_1000: 1.000 generierte Bilder
 - wl6_real: 27 reale Bilder (aus Kartenmaterial ausgeschnitten)
- Validierung der Netzwerke in drei Schritten
 1. Trainingsdatensatz
 2. Validierungsdatensatz: wl6_1000
 3. Validierungsdatensatz: wl6_real

Loss & Genauigkeit: w16_120k



Loss & Genauigkeit: w16_250k





Beispiel: Haaren

wl6_120k: „Haaren“

$W = 1$

$C = 1$

wl6_250k: „Haaren“

$W = 1$

$C = 1$



Beispiel: Ostiem

wl6_120k: „Ostion“

$W = 0$

$C = 0$

wl6_250k: „Ostiem“

$W = 1$

$C = 1$



Beispiel: Aachen

wl6_120k: „gAacli“

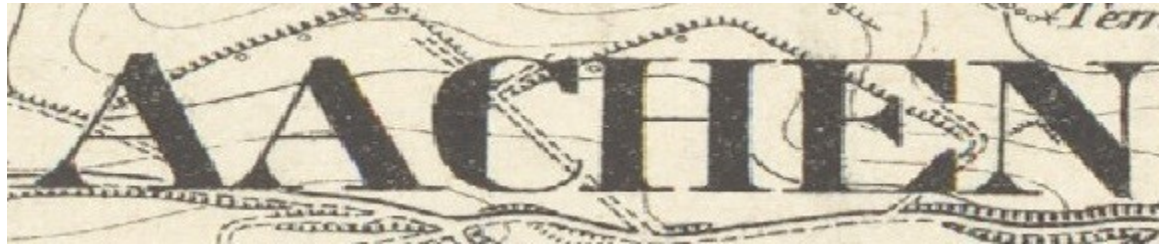
$W = 0$

$C = 0$

wl6_250k: „Aatiz“

$W = 0$

$C = 0,3333$



Beispiel: Neuhof

wl6_120k: „gNeuhof“

$W = 0$

$C = 0$

wl6_250k: „Neuhol“

$W = 0$

$C = 0,8333$



	Gesamtlaufzeit	Epochenlaufzeit
wl6_120k	4h 20min 54s	10min 39s
wl6_250k	9h 3min 22s	21min 44s

	Gesamtlaufzeit	Laufzeit pro Bild
wl6_120k	19min 25s	9,7ms
wl6_250k	40min 17s	9,7ms

Fazit und Ausblick

- Texterkennung und -extraktion aus topographischen Karten ist möglich
- Vorbedingung: Trainingsdaten in guter Qualität und hoher Menge
- einfache GPU-Nutzung

Ab € 500,99 inkl. MwSt.

- historisch genaue Trainingsdaten schwierig
- Verbesserung durch Originalschriftarten wahrscheinlich
 - Roemisch*
 - Kursivschrift*

Ab € 418,88 inkl. MwSt.

grosse



Apeler



- Anderer Ansatz: Trennung von Vorder- und Hintergrund (ScaDS Leipzig)
- Problem: Einbruch der Erkennungsrate (große Datensätze)

- Netzwerk grundsätzlich geeignet
- Problem: feste Bildbreite für variablen Text

- Training auch auf einer GPU dauert sehr lange
- Multi-GPU mit Keras prinzipiell möglich
- Problem: erhoffter Speedup bleibt aus

- Nutzbarkeit von *Transfer Learning*?
- verbesserte Score-Berechnung (Sonderfälle!)

Vielen Dank!

A. Graves, S. Fernández, F. Gomez, J. Schmidhuber: *Connectionist Temporal Classification: Labelling Unsegmented Data with Recurrent Neural Networks*. In: Proceedings of the 23rd International Conference on Machine Learning, 2006. S. 369 – 376

B. Shi, X. Bai, C. Yao: *An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition*. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 39, Nov. 2017. S. 2298 – 2304

E. Schölzel: *genSet3.py*, 2017. Unveröffentlicht.