Data Science im Unternehmen
Preprocessing und Datenanalyse

Dr. Jonas Rende and Thomas Stadelmann, DATEV eG

Ansprechpartner

Dr. Jonas Rende

- Jonas Rende is a senior data scientist at DATEV eG working in the customer centric design department.
- He uses machine learning methods to extract customer needs out of vast text and user behavior data. Together with Thomas Stadelmann, he is working on automatically generating insights out of customer feedback.
  In addition, he is laying the foundations for a customer experience platform.
- Before joining DATEV e.G. Jonas was a research and teaching assistant at the department of statistics and econometrics at the University of Erlangen-Nürnberg. Jonas holds a master's degree in economics and a PhD in statistics from the University of Erlangen-Nürnberg.

Thomas Stadelmann

- Thomas Stadelmann is a senior data scientist at DATEV eG., where his work focuses on Information Retrieval, Neural Search, Query Log Analysis and A/B-Testing. 10 years ago Thomas started as a software engineer at DATEV and over the time his passion for data science continuously increased.
- His current research interests include Natural Language Processing, Machine Learning with focus on Deep Learning, Search Evaluation, Datastructures, Systems Reengineering and anything computable. Following his Bachelor's degree in the dual system, Thomas got his Master's degree with distinction from the Otto-Friedrich University of Bamberg. He was awarded the Best Short Paper Award of the BTW 2015 conference.

DATEV eG

Genossenschaftlicher IT-Dienstleister

für Steuerberater, Wirtschaftsprüfer und Rechtsanwälte sowie deren Mandanten

Digitalisierung von betriebswirtschaftlichen Prozessen

Eines der größten Softwarehäuser Europas im sich veränderten Marktumfeld

Das alles ist DATEV

Wir bieten
Lösungen zu
Rechnungswesen, Personalwirtschaft, Kanzleiorganisation, Steuern, betriebswirtschaftliche Beratung

Bei uns arbeiten ca.
8.193
Mitarbeiterinnen und Mitarbeiter

DATEV wurde
1966
gegründet

Rund
13,5 Mio.
Lohn- und Gehaltsabrechnungen werden jeden Monat mit unseren Lösungen erstellt

Rund
440.000
Kundinnen und Kunden vertrauen auf unsere Lösungen

Wir entwickeln und vertreiben über
200
Software-Produkte und IT-Dienstleistungen

Unser Motto ist
Zukunft gestalten, gemeinsam.

Wir sind eines der **größten Softwarehäuser** in Europa

Unser Hauptsitz ist in
Nürnberg

2020 haben wir einen Umsatz von ca.
1,1 Mrd.
Euro erzielt

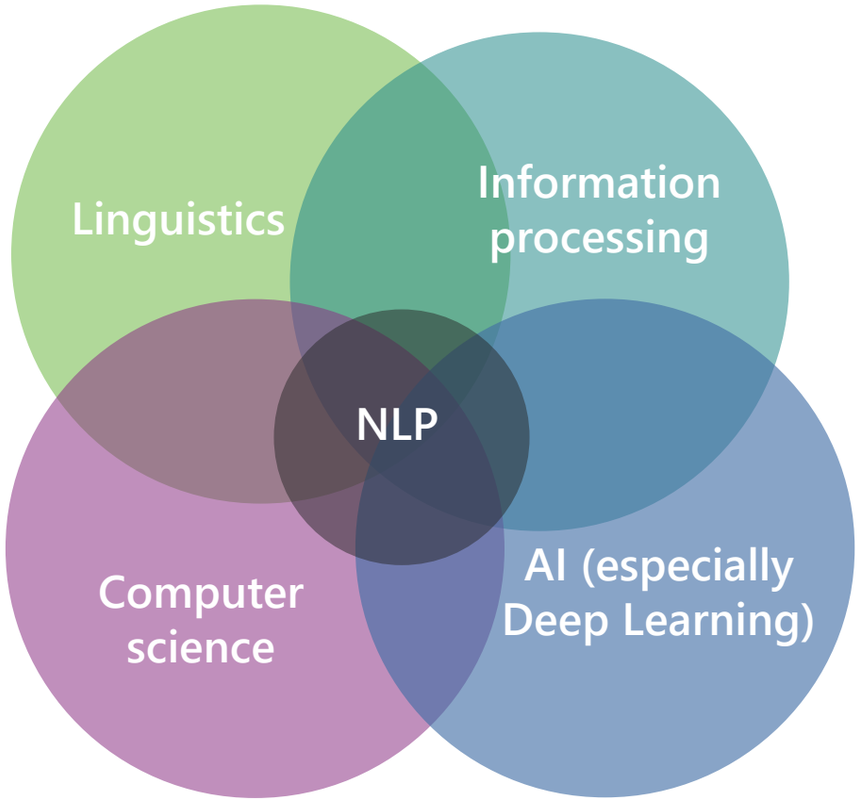
Wir sind eine
Genossenschaft für
Steuerberater, Wirtschaftsprüfer, Rechtsanwälte und deren Mandanten

DATEV

Agenda

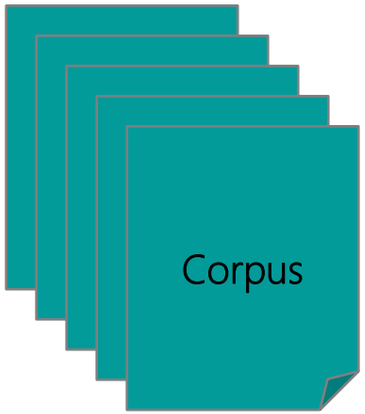
1. Introduction to NLP („gentle“): Basic Idea of a language model
2. Sentiment Analysis
3. Hands-On Coding Session: Preprocessing and insight extraction utilizing pre-trained sentiment analysis models
4. Why domain language matters: A short teaser

A management proofed definition of NLP



Algorithmic processing and analysis of natural language

The (very very) Big Picture



- Textual basis for a linguistic model (language model)
- Public german text data
- Public domain data
- DATEV-own domain data

DATEV-LM

- Spoken language is evolving over time and can not be fully defined by formal criteria in contrast to a programming language
- Data driven approach: learning a language by observing real world examples (corpus)
- Statistical language models: (SOTA: Deep Learning):

Downstream Tasks

Tasks („head“) we want to solve with the corresponding language model e.g. problem classification

Probabilistic Language Models

Definition *(Page 105, Neural Network Methods in Natural Language Processing, 2017)*

*Language modeling is the task of assigning a **probability** to **sentences** in a language. […] Besides assigning a probability to each sequence of words, the language models also assigns a probability for the **likelihood of a given word** (or a sequence of words) to follow a sequence of words*

Mathematical

Let w be a sequence of words with the length m

$$P(w) = P(w_1, w_2, \dots, w_{m-1}, w_m)$$

$$P(w_m | w_1, w_2, \dots, w_{m-2}, w_{m-1})$$

Outcome for further processing

Vector representation of words

- Bag of Words
- Embeddings
- Contextualized Embeddings

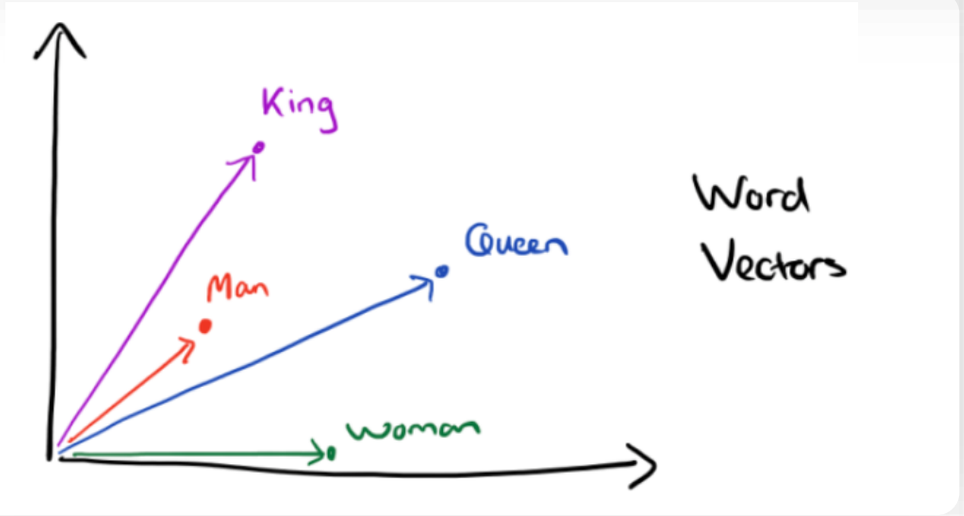
Embeddings (Word2Vec, Glove)

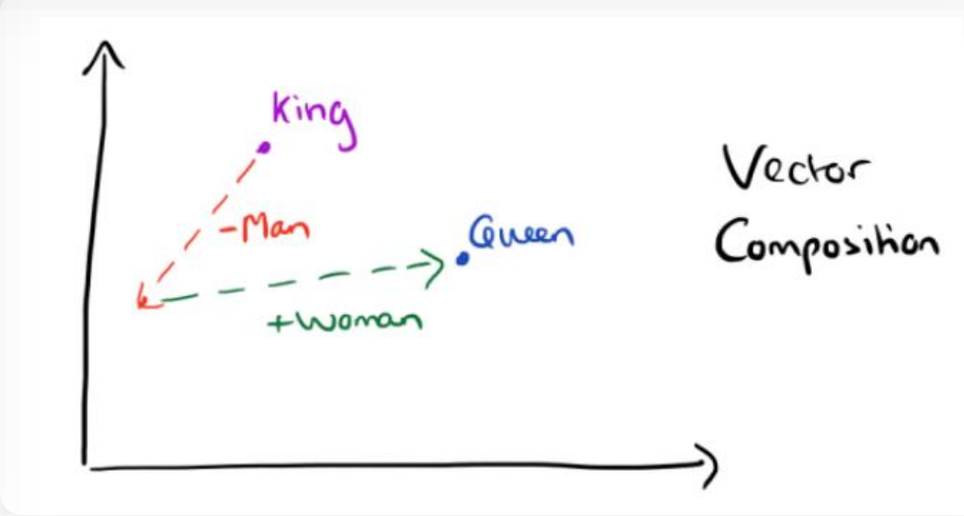
- Word Embeddings are real-valued dense vector representations ($1xd$) for words you can perform mathematical operations on
- e.g. bat

0.2	0,5	0,1	0,0	3,1	1,2	2,3	0,5

- Based on co-occurence statistics which allows to capture semantic similarities

Most famous example: King – Man + Woman = Queen





Source: https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/

Contextualized Embeddings (e.g. Transformer)

Embeddings are not context sensitive

The bat hits the baseball → bat

The bat is sleeping in the cave → bat

Models build up on the transformer architecture (e.g. BERT) are solving this problem

The bat hits the baseball → bat

The bat is sleeping in the cave → bat

Currently we are focusing on BERT

Extracting an atmospheric picture out of text data (1/2)

- The goal of a sentiment analysis is to assign a tonality class to a statement
- Common tonality classes are good, bad, neutral and mixed (Side note: more modern approaches work aspect-based)
- To generate insights it is necessary to aggregate tonality classes along several dimensions to extract an atmospheric picture
  - e.g., sentiment distribution over time
  - e.g., identify customers with repeated bad / good tonality
  - e.g., shitstorm detection on social media
  - …

Extracting an atmospheric picture out of text data (2/2)

Hands-On Coding in Google Colab with Thomas

Domain knowledge is the key to success

Why domain knowledge?

- Publicly available language models are pretrained on public text sources e.g. on the entire articles on Wikipedia
- Freely available language models show weaknesses if the documents contain a lot of domain knowledge e.g. tax or legal terms

How to take domain knowledge into account?

- Benchmark: GermanBERT by deepset.ai (no domain knowledge)
- Language Modell Adaption: An existing Model is enriched with domain knowledge (up to 10% of vocabulary at BERT)
- From Scratch: The Model ist trained from scratch so that the proportion of domain knowledge can vary between 0% and 100%

We started with LM adaption. Currently, we evaluate training from scratch

Context and domain language matters

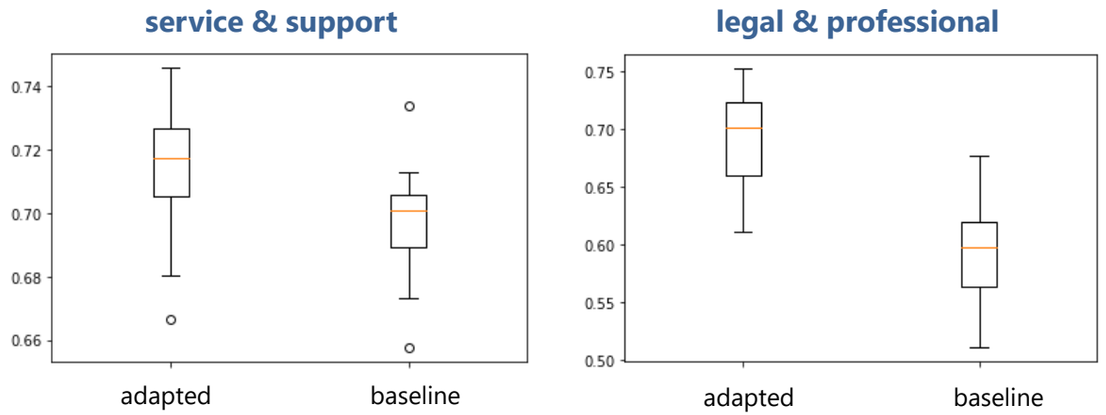
- Hallo in die Runde, ich hatte folgende Lösung gefunden: man speichert den in DUO Bank online stehenden Zahlungssatz einfach nochmal neu ab (über Bank kann man zB das Fälligkeitsdatum ändern) und dann klappt es mit dem Import des exportierten Datensatzes. Das habe ich in einem Mitarbeiter zu meinem Servicekontakt berichtet. Danach bekam ich einen Rückruf (da der Export/Import auch im DATEV Zahlungsverkehr nicht geklappt hat) und bekam folgende Lösung: Häkchen setzen im DATEV Zahlungsverkehr unter Stammdaten/Bank/Konto, direkt unter den Einstellungen zum Batchbooking, dass die Umlaute angepasst werden sollen. Siehe da: jetzt klappt alles! Was so ein schnödes Häkchen ausmacht! Super DATEV

- @... Wie das mit dem KAG weitergeht entscheiden am Ende doch das Gericht!

Proof of concept: Key insights (1/3)

Does incorporating our domain data improve the results of our downstream tasks?

f1-macro scores of evaluation tasks



Details

- Magnitude of improvement on par with previous results in literature (~2% p. f1-macro improvement) for one use case (n~1100) and vastly exceeding on the other (n~350)
- BERT-based approaches perform way better than fasttext
- Corpus composition is essential (especially if you have subdomains)
- There is not one language model to solve all tasks
- MLM Head (much) more important than NSP Head

Proof of concept: Key insights (2/3)

How to set up a domain specific corpus?

Answers

- Don’t create one big file, keep it modular
- Split corpus by subdomains, document type, etc.
- Updating/Versioning of corpus needed as completely new topics may arise (e.g. corona)
- Common preprocessing logic using adapters for different data sources
- Sentence splitting is one of the hardest but not the most important preprocessing step

Proof of concept: Key insights (3/3)

How mature are current open-source frameworks for NLP transfer learning?

Answers

- All frameworks seem to be pretty stable (e.g. almost no breaking changes or rough edges)
- But keep in mind: There is a trade-off between rapid progress and stability …and progress is very fast
- Biggest pain are still varying environments (e.g. linux vs. windows)
- if it‘s broken, you can fix it

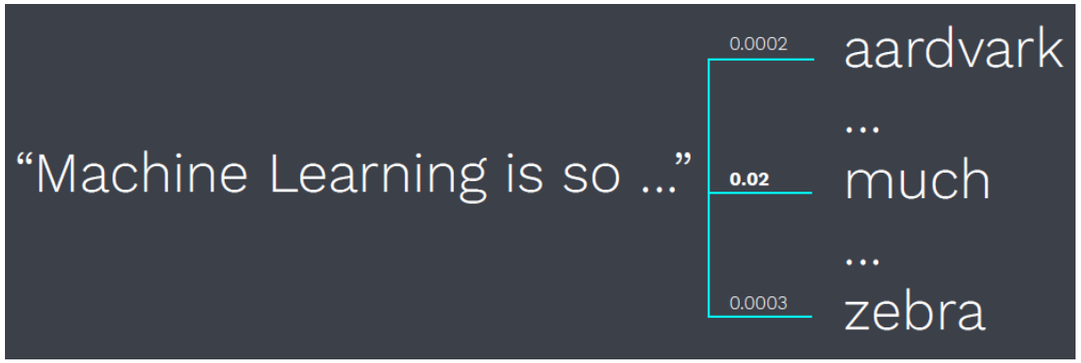
Dr. Jonas Rende
Thomas Stadelmann
DATEV eG

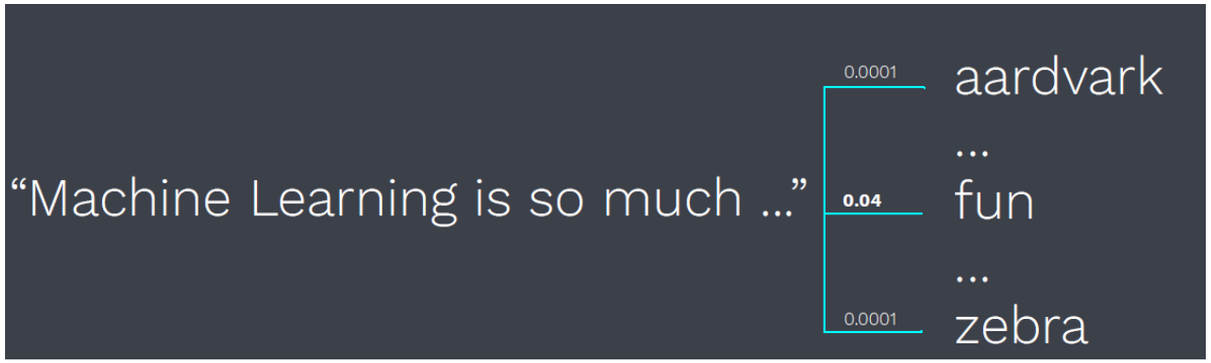
Jonas.rende@datev.de
Thomas.Stadelmann@datev.de

Shaping the future – together.

The Tasks of a Language Model: An example

*Task 1: Assigning a probability for the **likelihood of a given word** (or a sequence of words) to follow a sequence of words*



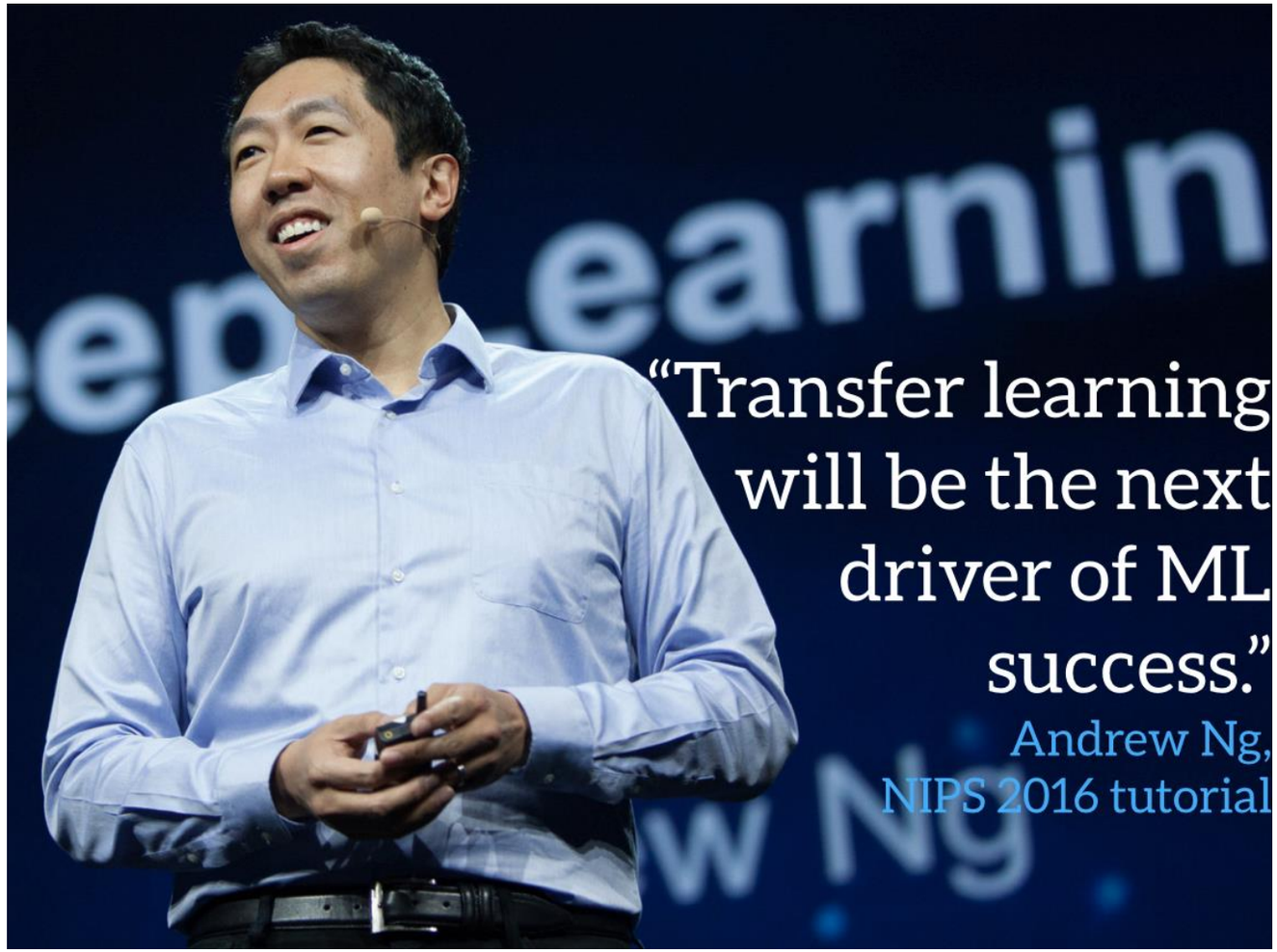


*Task 2: Assigning a **probability** to a **sentence***

P("Machine Learning is so much fun") = 0.2

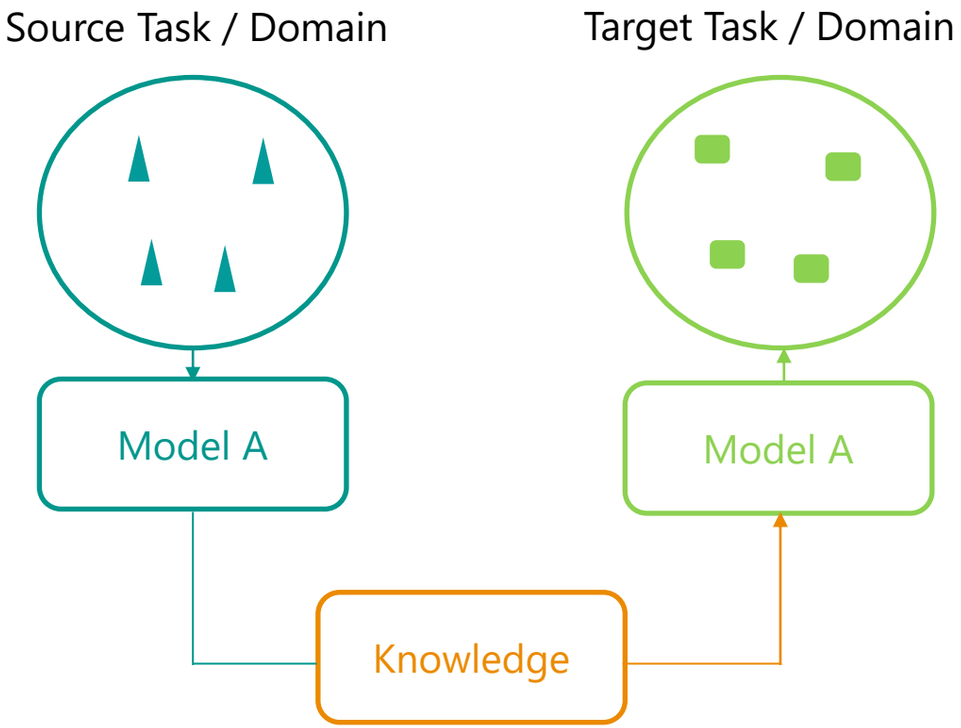
Source: Example taken from deepset.ai

The Game Changer: Transfer Learning



Source: https://ruder.io/transfer-learning/index.html#whatistransferlearning

“Transfer learning is a means to extract knowledge from a source setting and apply it to a different target setting.”

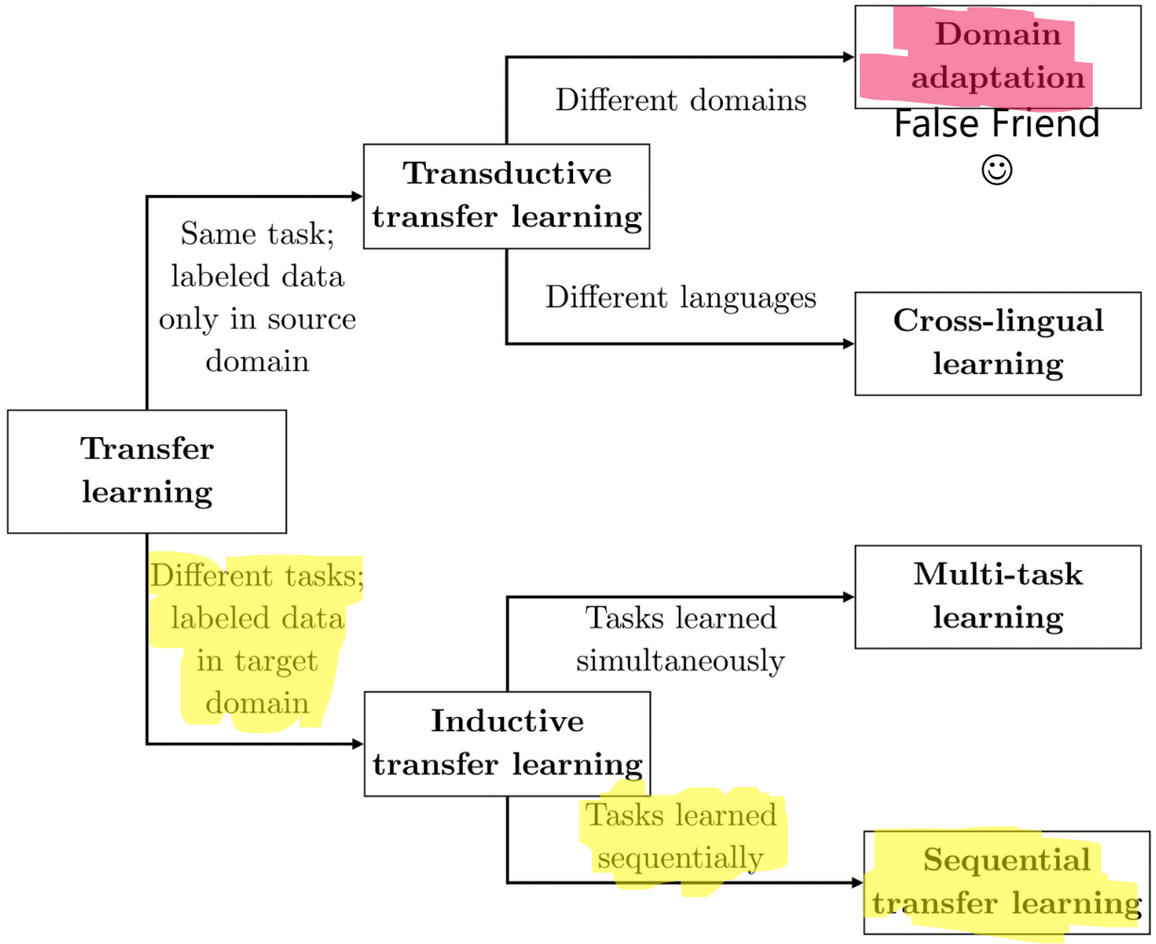


Why transfer learning?

- Less labels required
- Re-usable for multiple problems
- Solve problems in new domains
- Faster development
- Performance

Source: https://ruder.io/state-of-transfer-learning-in-nlp/ and deepset.ai

Transfer Learning in NLP



Source: Ruder (2019): A taxonomy for transfer learning in NLP

Source data

- **Pretrain** (vector) representations on a large **unlabelled** text corpus with a suitable method (e.g. word2vec, BERT, …)
- Unlabelled text corpus examples: Public german text data (e.g. Wikipedia), Public domain data (e.g. german legal), DATEV-own domain data

Knowledge

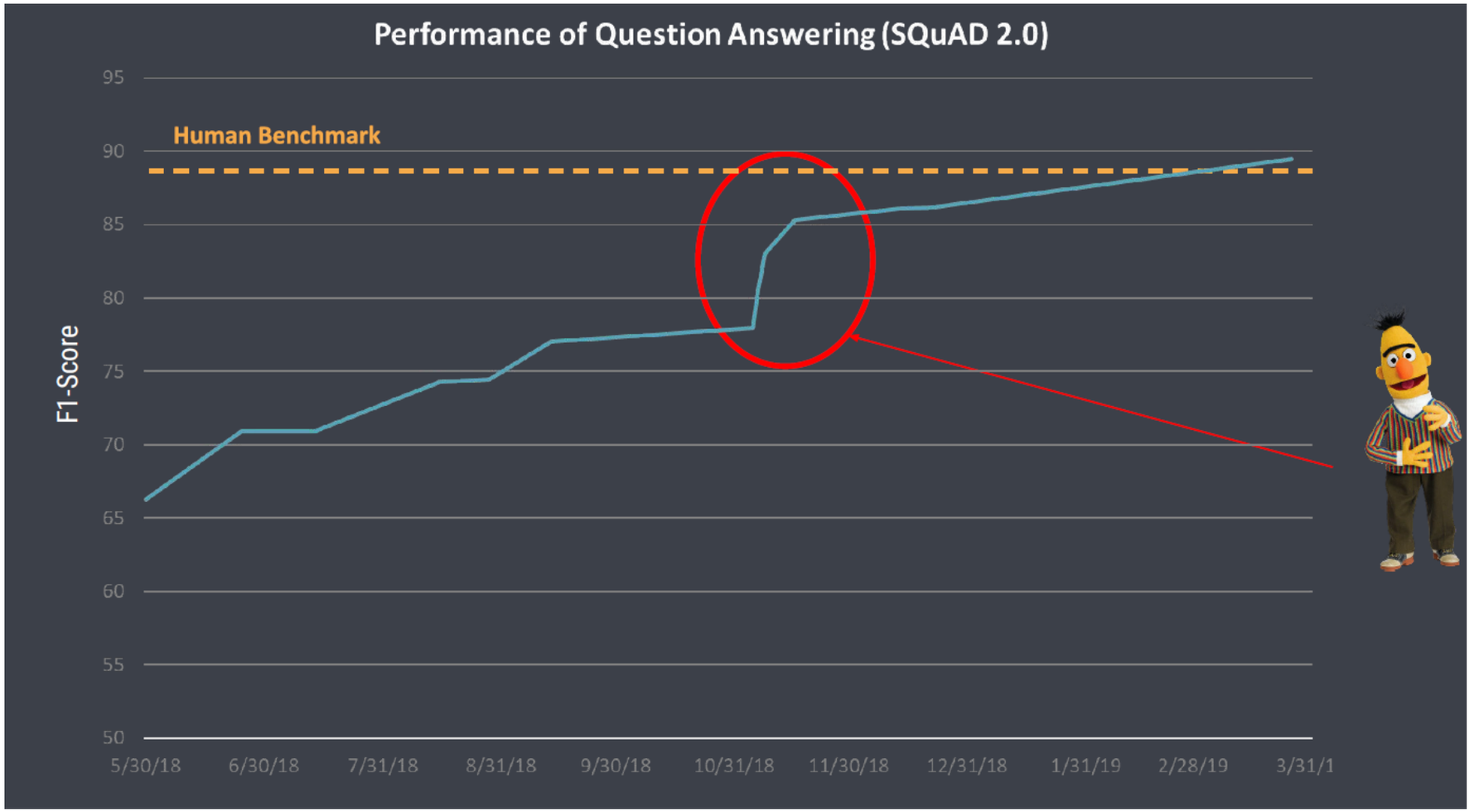
Extract pretrained model (e.g., representations and parameters)

Target data

- Adapt your extracted (vector) representations to a **supervised learning** target task, using your labelled data
- Examples for tasks: Classification, sentiment analysis, named entity recognition, intent recognition, neural search, question answering, text summariziation, …

Source: https://ruder.io/state-of-transfer-learning-in-nlp/

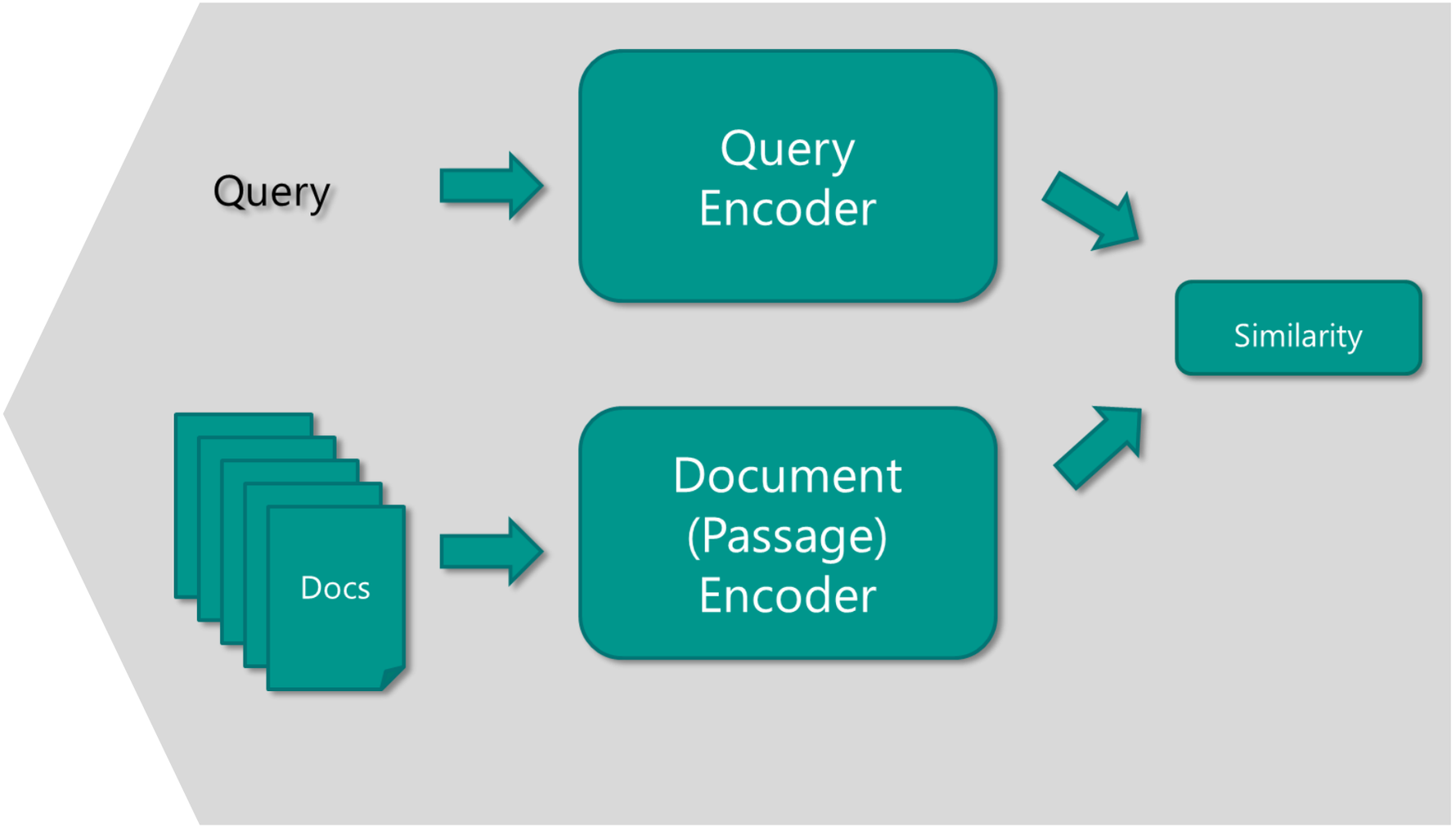
Transfer Learning is changing Research



Source: Taken from deepset.ai

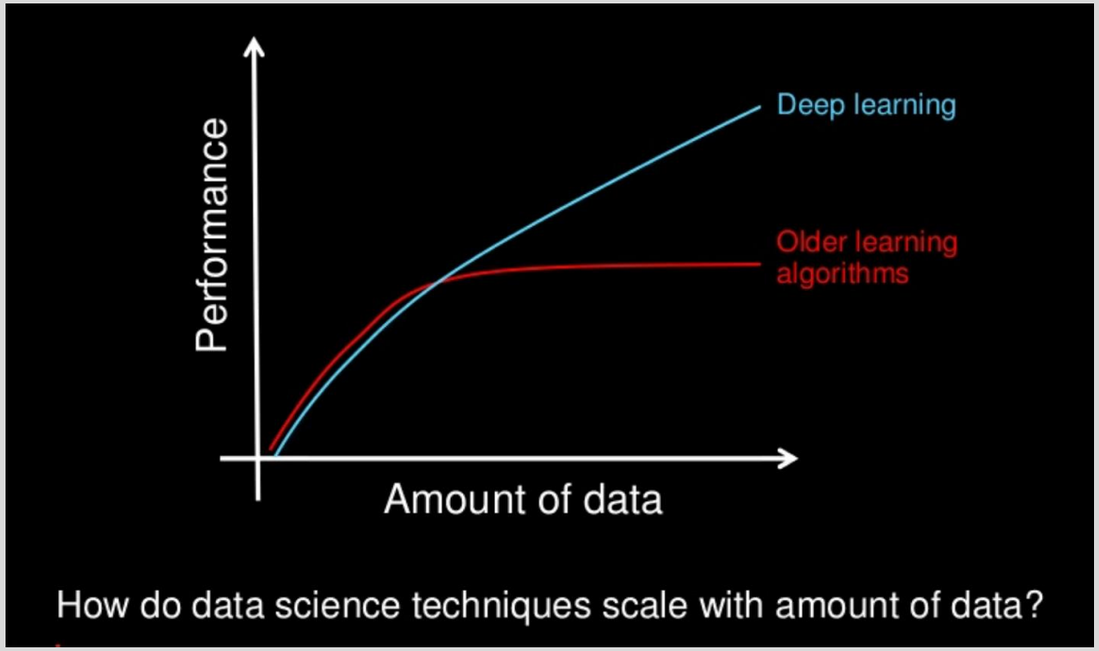
Further Use Cases

- Sentiment analysis (n ~ 19000)
- Neural Search
  - Bi-Encoder



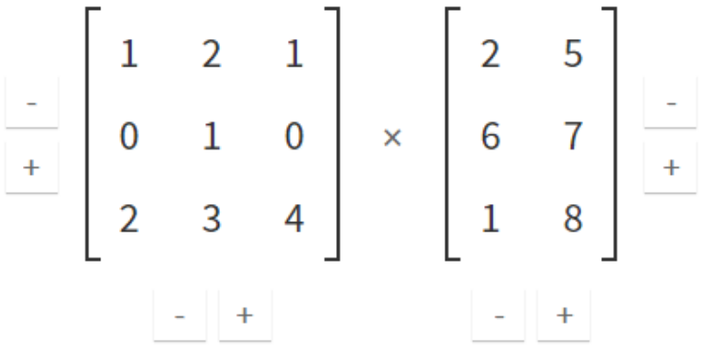
Why Deep Learning?

- Universal Approximation Theorem (Hornik 1991)
- Availability of vast amounts of data
- Availability of compute power (GPUs)
- Obtain good results with ease in comparision to excessive feature engineering
- SOTA-Results in most ML-disciplines

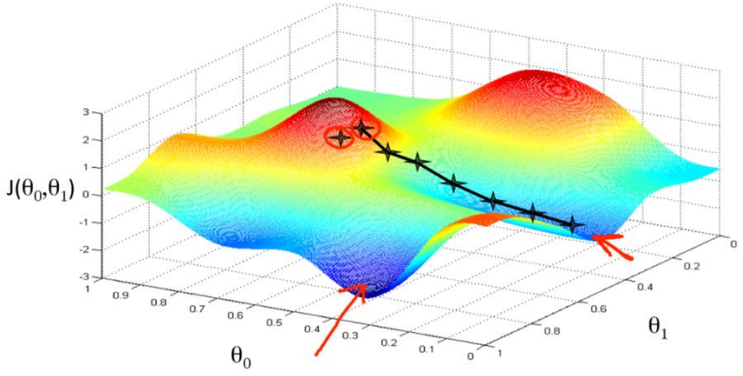


https://www.slideshare.net/ExtractConf (Slide 30) by Andrew Ng

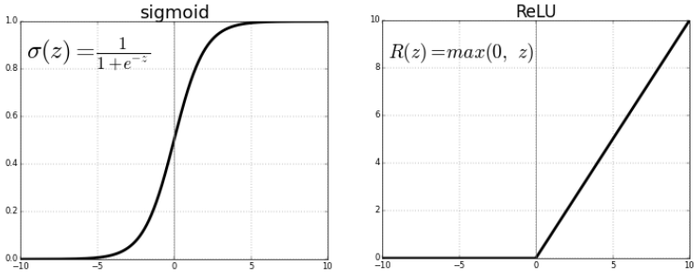
Neural Network Building Blocks



http://matrixmultiplication.xyz/



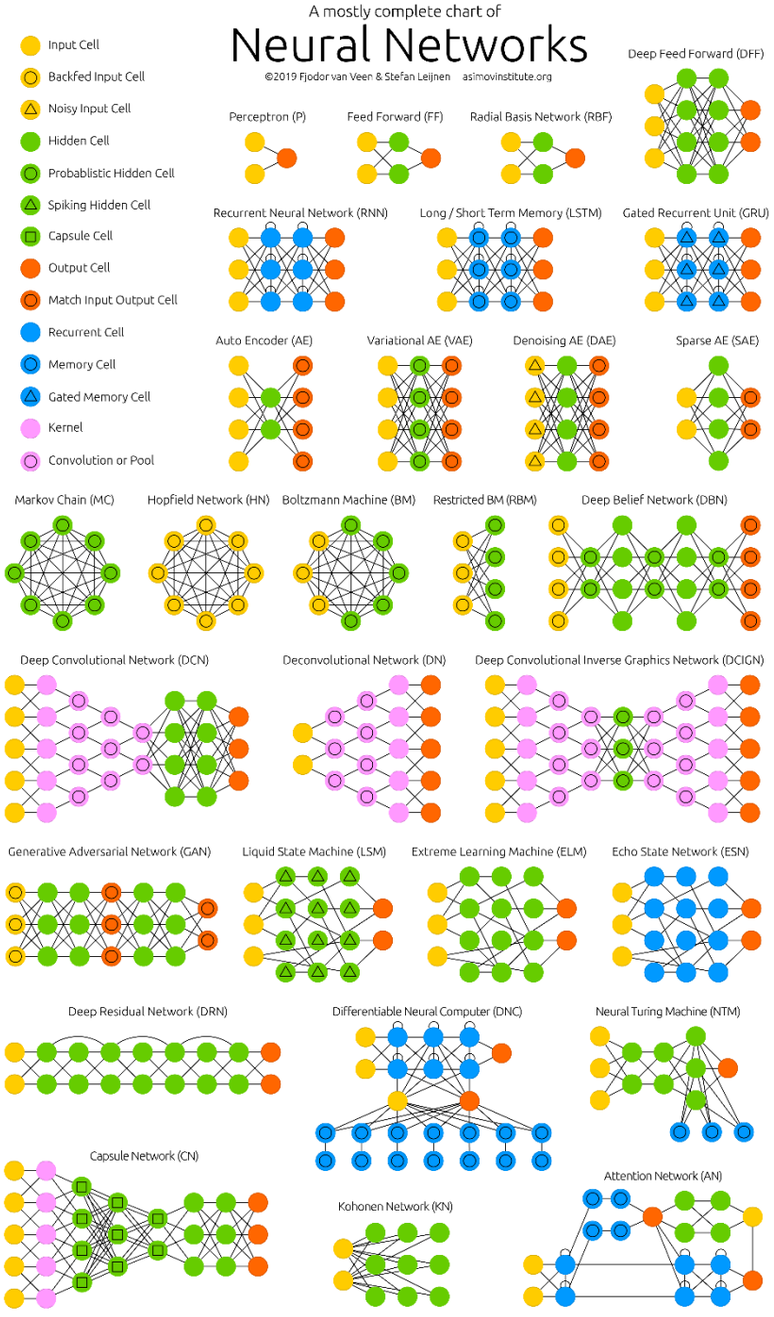
https://hackernoon.com/gradient-descent-aynk-7cbe95a778da



https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6?gi=b4754d64054

Operating principle of neural networks visualised

http://neuralnetworksanddeeplearning.com/chap4.html



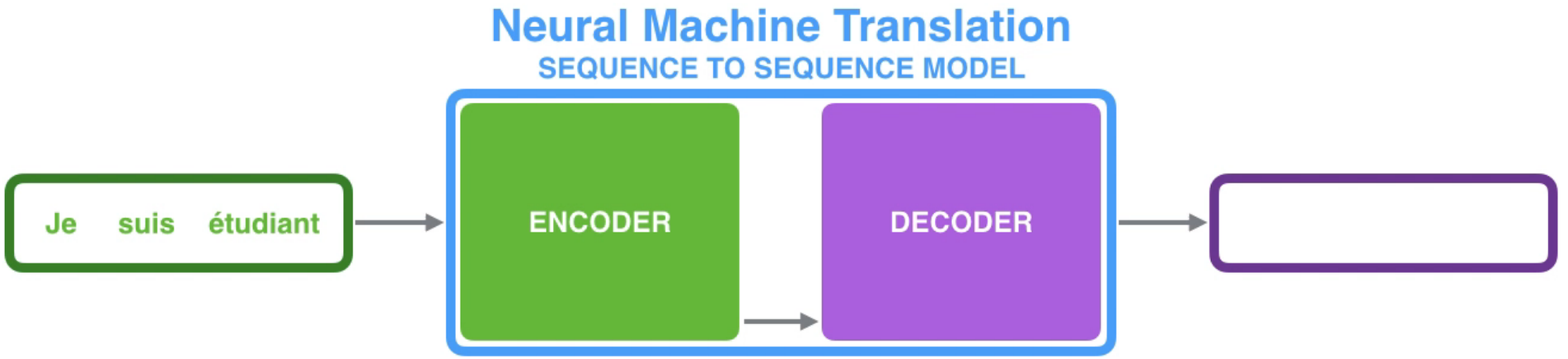
NLP Network Architecture

- Key idea: Using the same information multiple times by applying different transformations
- Context: locality, attention

A brief history of Deep Learning Network Architectures:

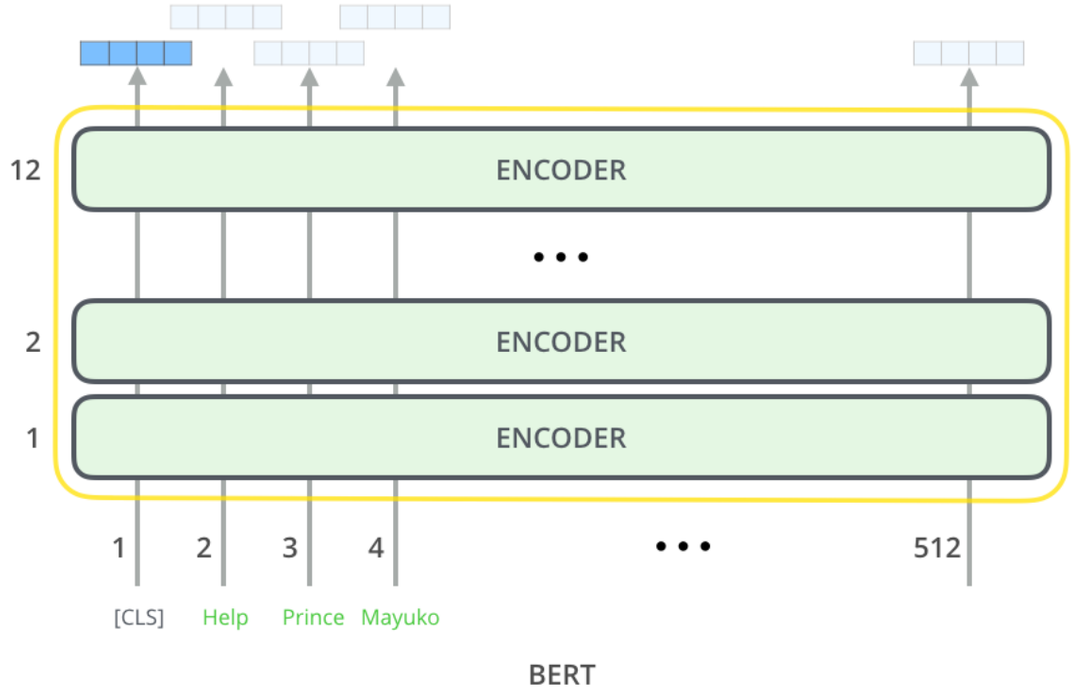
- Convolutional Neural Networks (CNNs) (Krizhevsky 2012)
- Recurrent Neural Networks (RNNs) (Mikolov 2010)
- ResNet (He 2016), LSTMs (Merity 2017), Transformers (Vaswani 2017), ELMo (Peters 2018), ULMFit (Howard 2018), BERT (Devlin 2018), …

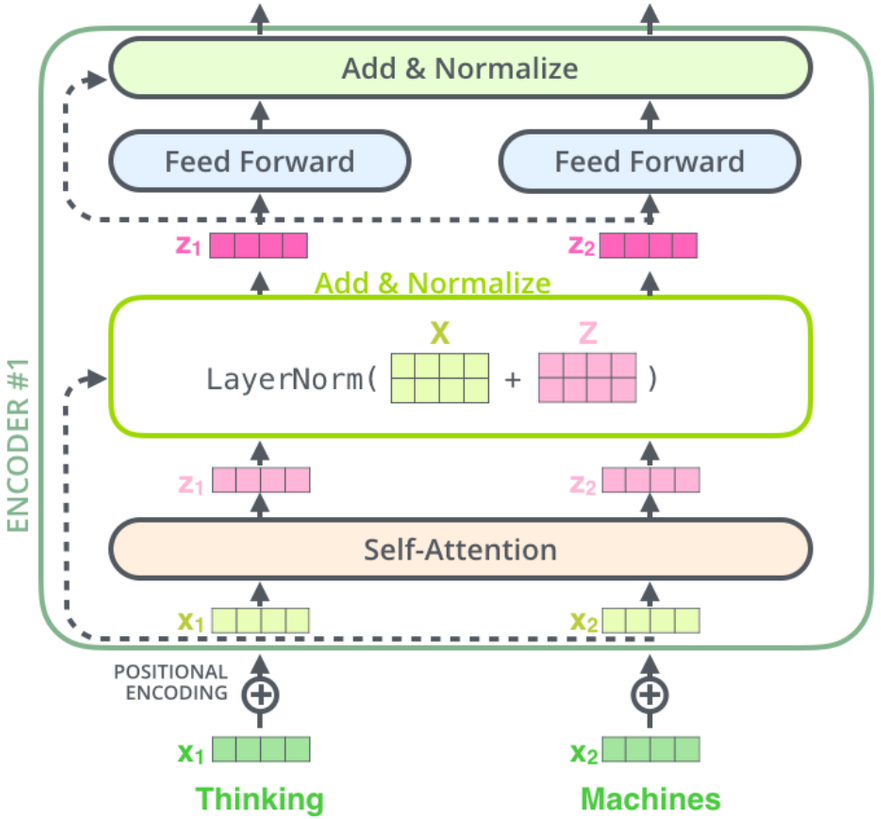
BERT and its origin



https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/

BERT Architecture

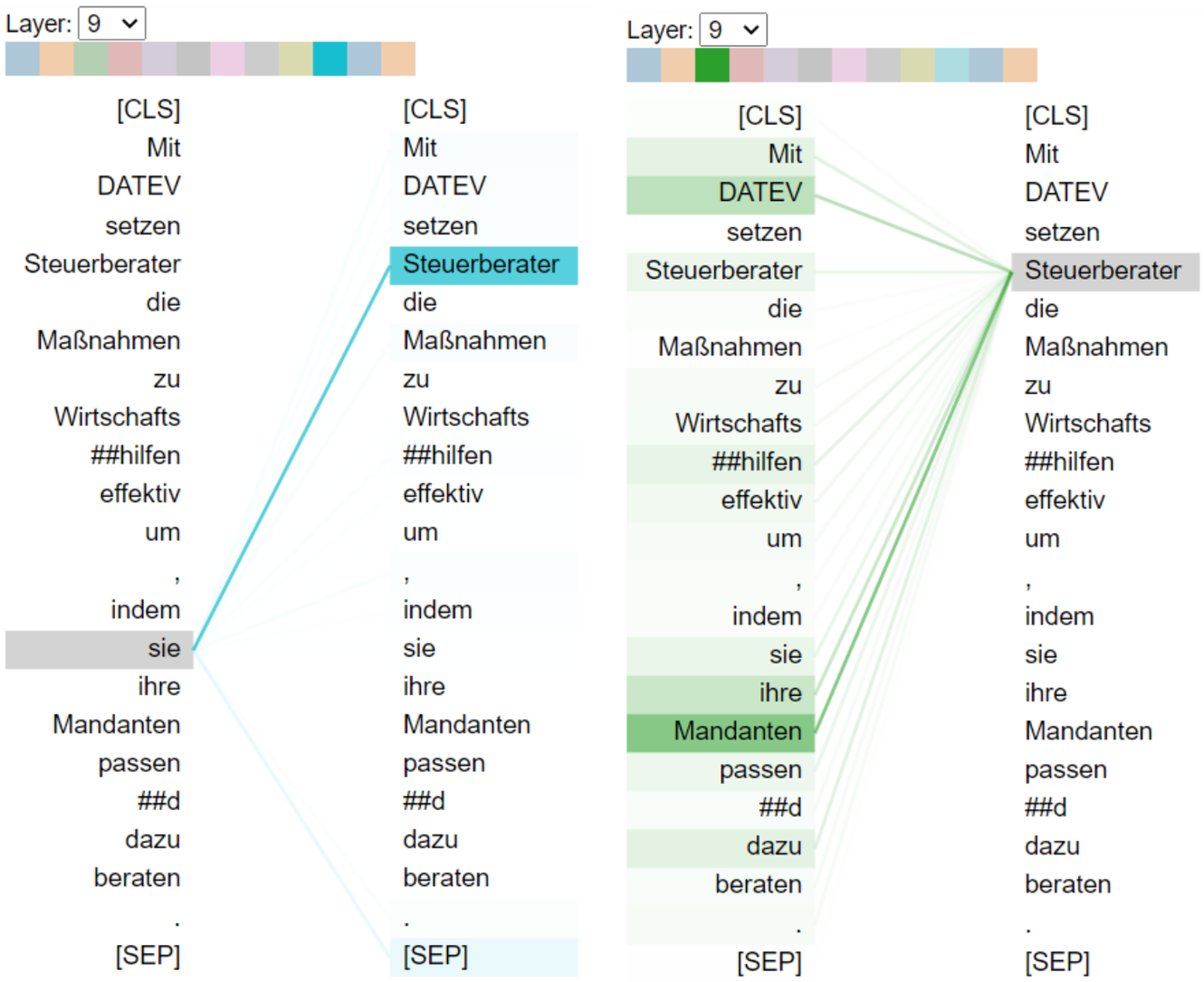




http://jalammar.github.io/illustrated-bert/

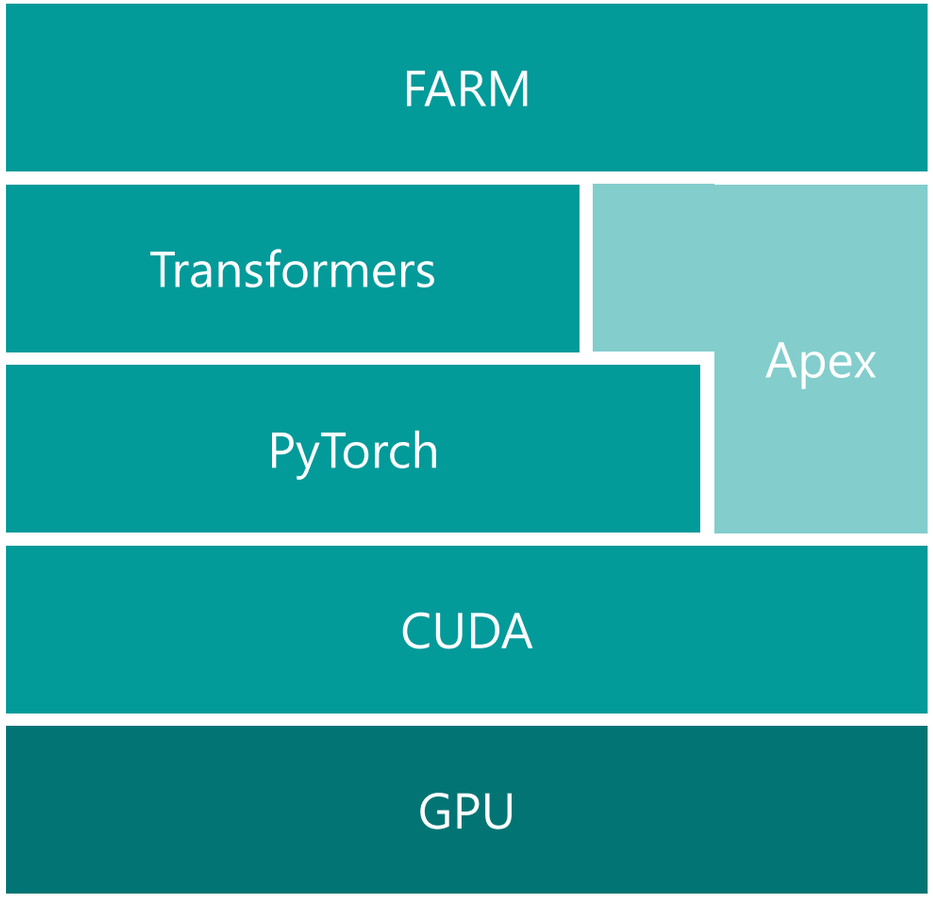
Context in NLP: Attention

- Semantic meaning of words is context dependent
- Calculate scores that depict the corresponding influence of surrounding words
- Incorporate the meaning of surrounding words according to this score



Created with bertviz: https://github.com/jessevig/bertviz.git

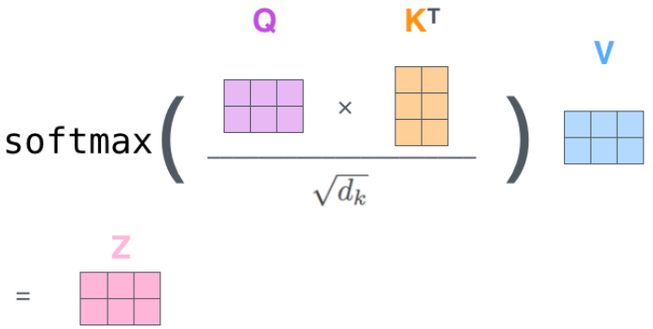
Hardware and Software stack

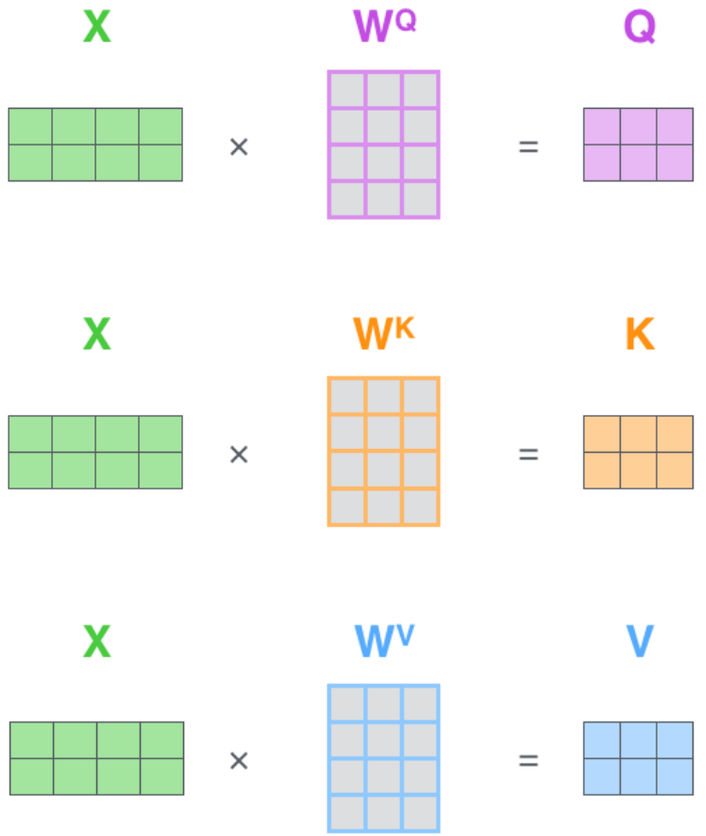


- https://github.com/deepset-ai/FARM
- https://github.com/huggingface/transformers
- https://github.com/pytorch/pytorch
- https://github.com/NVIDIA/apex
- We use GermanBERT as initial model which has 10% sentence piece tokens non-allocated
- We insert the top 3000 most important sentence pieces of our corpus
- And train on…
- 1,2 GB of raw domain specific text

Self-Attention

- 3 Transformations (Matrices)
- Key: does the Query word's meaning depend on me?
- Query: does the Key word influence my meaning?
- Value: word's raw meaning





http://jalammar.github.io/illustrated-transformer/

References (1/3)

- Hornik, Kurt. 1991.
  Approximation capabilities of multilayer feedforward networks. In: Neural Networks 4. Jg., Nr. 2, S. 251-257.
  http://www.vision.jhu.edu/teaching/learning/deeplearning18/assets/Hornik-91.pdf
- Merity, Stephen; Keskar, Nitish Shirish; Socher, Richard. 2017.
  Regularizing and optimizing LSTM language models. arXiv preprint.
  https://arxiv.org/abs/1708.02182
- Devlin, Jacob; Chang, Ming-Wie; Lee, Kenton; Toutanova, Kristina. 2018.
  BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
  https://arxiv.org/abs/1810.04805
- Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Lukasz; Polosukhin, Illia. 2017. Attention Is All You Need.
  https://arxiv.org/abs/1706.03762
- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. 2018.
  Deep contextualized word representations. arXiv preprint arXiv:1802.05365.
  https://arxiv.org/abs/1802.05365

References (2/3)

- He, K., Zhang, X., Ren, S., & Sun, J. 2016.
  Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
  https://arxiv.org/abs/1512.03385
- Howard, Jeremy; Ruder, Sebastian. 2018.
  Universal Language Model Fine-tuning for Text Classification.
  https://arxiv.org/abs/1801.06146
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012.
  Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
  http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf
- Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S. 2010.
  Recurrent neural network based language model. In Eleventh annual conference of the international speech communication association.
  http://www.fit.vutbr.cz/research/groups/speech/servite/2010/rnnlm_mikolov.pdf

References (3/3)

- Allamar, Jay
  Visualizing NLP Blog
  http://jalammar.github.io/
- Ruder, Sebastian. 2019.
  Neural Transfer Learning for Natural Language Processing.
  https://ruder.io/thesis/
- Ruder, Sebastian.
  NLP Transfer Learning Blog.
  https://ruder.io/
- Stanford NLP with Deep Learning Kurs
- fast.ai: Practical Deep Learning for Coders.
- fast.ai: NLP Kurs