



NLP

Maschinelle Übersetzung

# Tilman und Anna präsentieren

Englisch ▾



Fun with translations

Deutsch ▾



Spaß mit  
Übersetzungen

*„Es ist die Aufgabe des Übersetzers, jene reine Sprache, die unter dem Zauber einer anderen liegt, in die eigene Sprache freizusetzen, die Sprache, die in einem Werk gefangen ist, in seiner Neuerschaffung jenes Werkes zu befreien.“*

(Walter Benjamin)

# Lost in translation

Zitat von Walter Benjamin:

- Andere Perspektive auf Übersetzung als wir im Lab
- Ähnlicher Grundgedanke:  
"Kern"/(abstrakter) Gedanke eines Textes wird je nach Sprache unterschiedlich ausgedrückt.

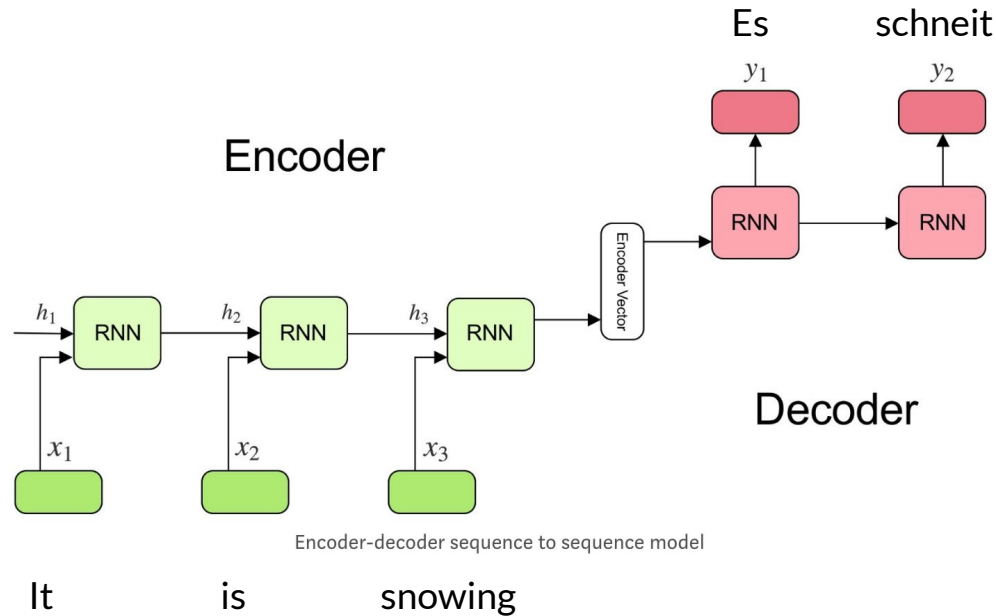
"Katze" =>



=> "cat"

# Übersetzung aus ML-Sicht

## Many-to-many sequence prediction



<https://towardsdatascience.com/understanding-encoder-decoder-sequence-to-sequence-model-679e04af4346>

# Encoder-Decoder

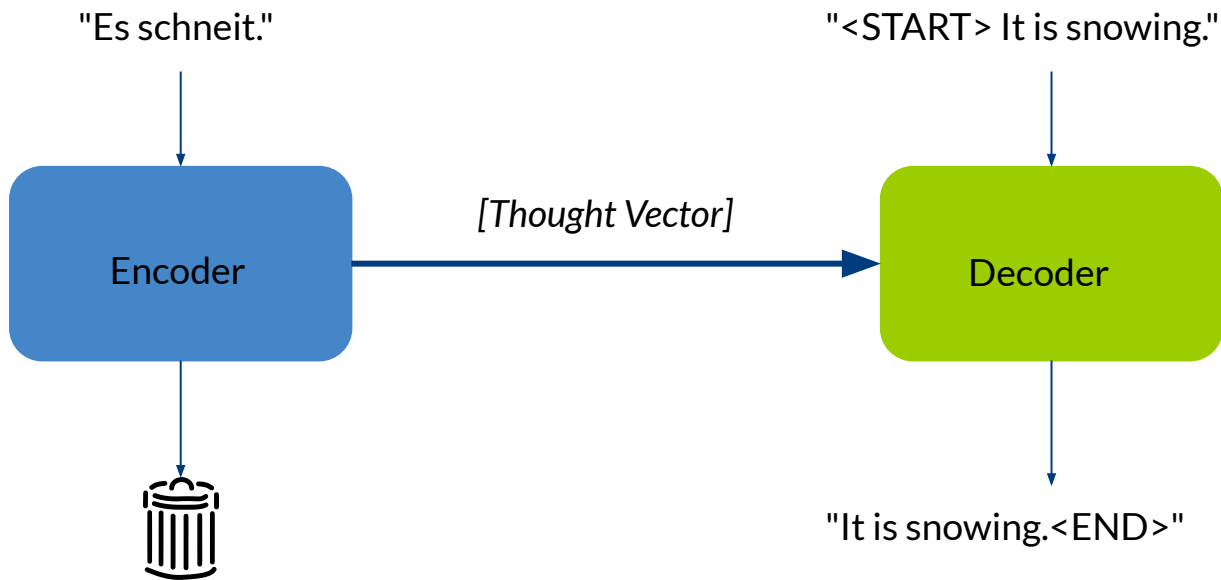
*... RNN Encoder-Decoder, consists of two recurrent neural networks (RNN) that act as an encoder and a decoder pair. The encoder maps a **variable-length source** sequence to a **fixed-length vector**, and the decoder maps the vector representation back to a **variable-length target** sequence.*

([Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation](#))

# Encoder-Decoder in Stichworten

- Encoder wandelt Eingabe variabler Länge in *thought vector*.
  - Ausgabe des Encoders wird verworfen.
  - Nur innerer Zustand des Encoders interessant (vgl. Embeddings)
- Decoder wird mit *thought vector* initialisiert und erzeugt daraus Ausgabe ebenfalls variabler Länge.
- "Sequence Embedding": *Thought vector* hat feste Länge.
- Eingabe- und Ausgabelänge können sich unterscheiden.

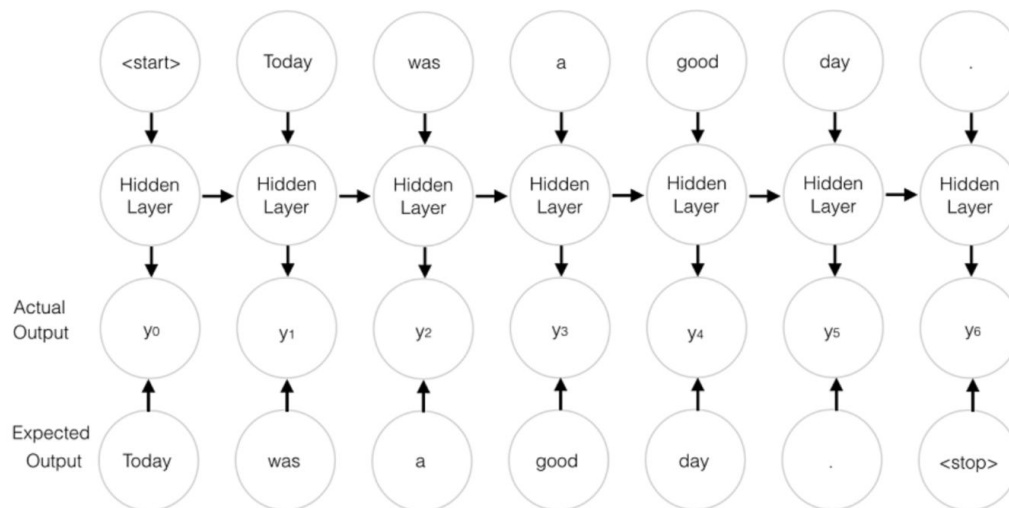
# Training





# Training

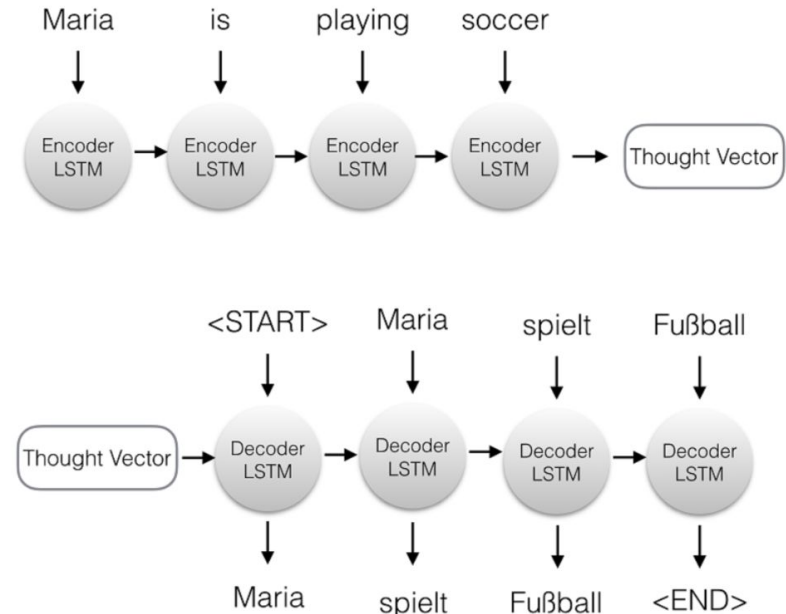
- Encoder und Decoder werden *gemeinsam trainiert*
  - Trainingsdaten: Encoder-Input, Decoder-Input und Decoder-Target
- *Teacher-Forcing* für Decoder: Input und Target selbe Sequenz, aber um einen Zeitschritt verschoben



Expected output is the next token in the sample. Shown here on word level.

# Anwendung

1. Eingabe enkodieren  
=> Thought Vector
2. Decoder mit Start-Token und Thought Vector initialisieren  
=> Nächstes Token
3. Decoder mit Start-Token, bisher generierten Tokens und aktualisiertem Thought Vector (innerer Decoder-Zustand) füttern  
=> Nächstes Token
4. Schritt 3 wiederholen, bis End-Token generiert wird.



**Figure 10.3 Unrolled encoder-decoder**

Hobson Lane et al.: Natural Language Processing in Action, Kap. 10, S. 372

Und nun ...

... Spaß mit Seq2Seq-Learning

[https://github.com/hskaailabnlp/machine translation](https://github.com/hskaailabnlp/machine_translation)

# Vielen Dank

Anna Weißhaar

[aweisshaar@inovex.de](mailto:aweisshaar@inovex.de)

Tilman Wittl

[twittl@inovex.de](mailto:twittl@inovex.de)

