

# Einsatz effektiverer Transformer-Encoder innerhalb der OIE

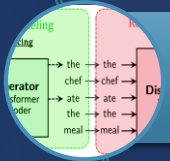
Präsentation von Henning Beyer

58. Jugend forscht Landeswettbewerb Sachsen

# Gliederung



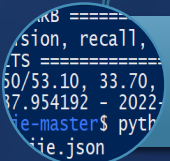
Einführung, Motivation



Vortrainingsmethoden



DetIE-Modell



Experimente und Ergebnisse



Ausblick und weitere Forschung

# Einleitung, Motivation

- Bereich der Open Information Extraction (OIE)
  - Teilbereich des NLP
  - Extrahieren von strukturierten Fakten
- Nutzen der OIE
  - automatischer Informationsgewinn
  - Wissensdatenbanken
- Encoder-Austausch am DetIE-Modell
  - BERT-Encoder ersetzt

# Ergebnisse

## Praxis

- Erzielen großer Leistungsverbesserungen
- Anwendbarkeit für zukünftige OIE-Modelle:
  - keine Minderung der Extraktionsgeschwindigkeit
  - für alle OIE-Modelle anwendbar

## Theorie

- Zusammenfassung relevanter BERT-Varianten
- Ermittlung allgemeiner Verbesserungskriterien:
  - Schwierigkeit der Aufgabe
  - Menge an Trainingsdaten

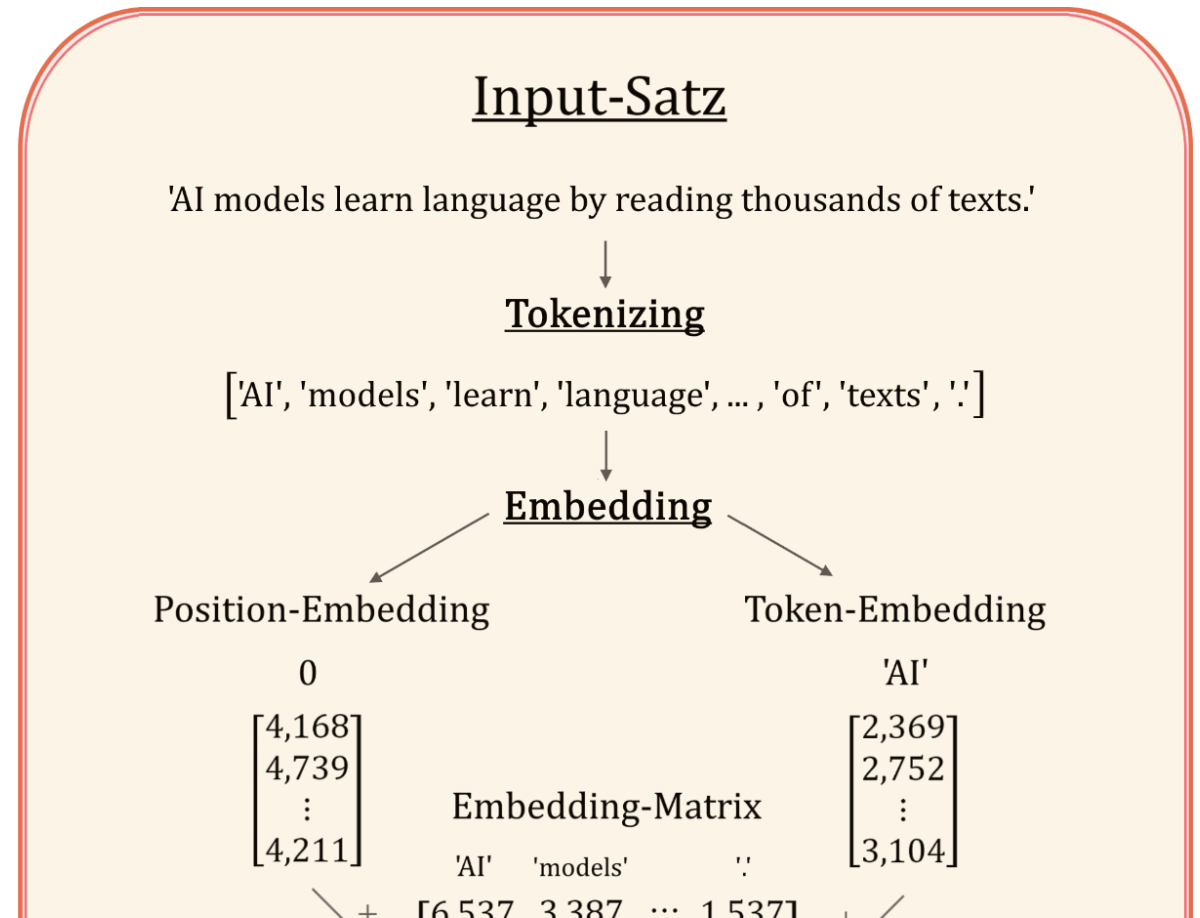


# Einführung in die Sprachverarbeitung

*Siehe Plakat*

Verarbeitungsschritte  
von NLP und OIE

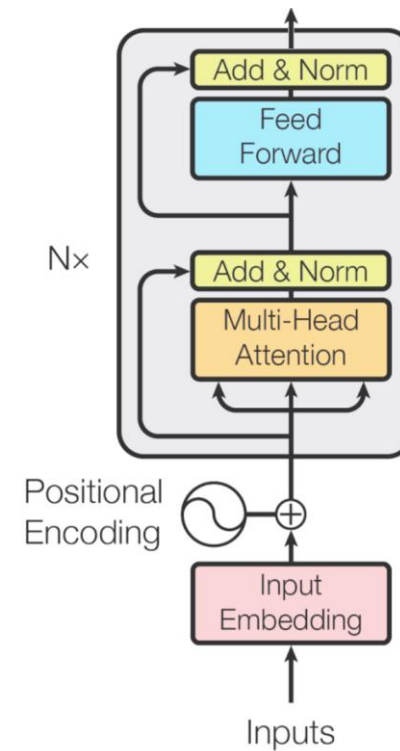
Erklärung  
von NLP und OIE



# Vortrainingsmethoden

- BERT-Varianten
  - sind alle Transformer-Encoder
  - unterschiedliches Vortraining
- betrachtete BERT-Varianten
  - BERT
  - RoBERTa
  - ELECTRA
  - DeBERTa
  - DeBERTaV3

*Transformer-Encoder*



# Vortrainingsmethoden – BERT und RoBERTa

## ■ BERT-Varianten

- sind alle Transformer-Encoder
- unterschiedliches Vortraining

## ■ betrachtete BERT-Varianten

- BERT
- RoBERTa
- ELECTRA
- DeBERTa
- DeBERTaV3

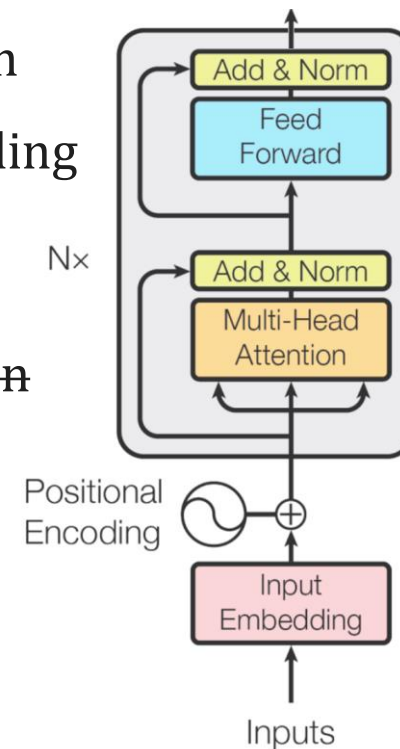
## ■ BERT-Modelle

- Next-Sentence-Prediction
- Masked-Language-Modeling

## ■ RoBERTa-Modelle

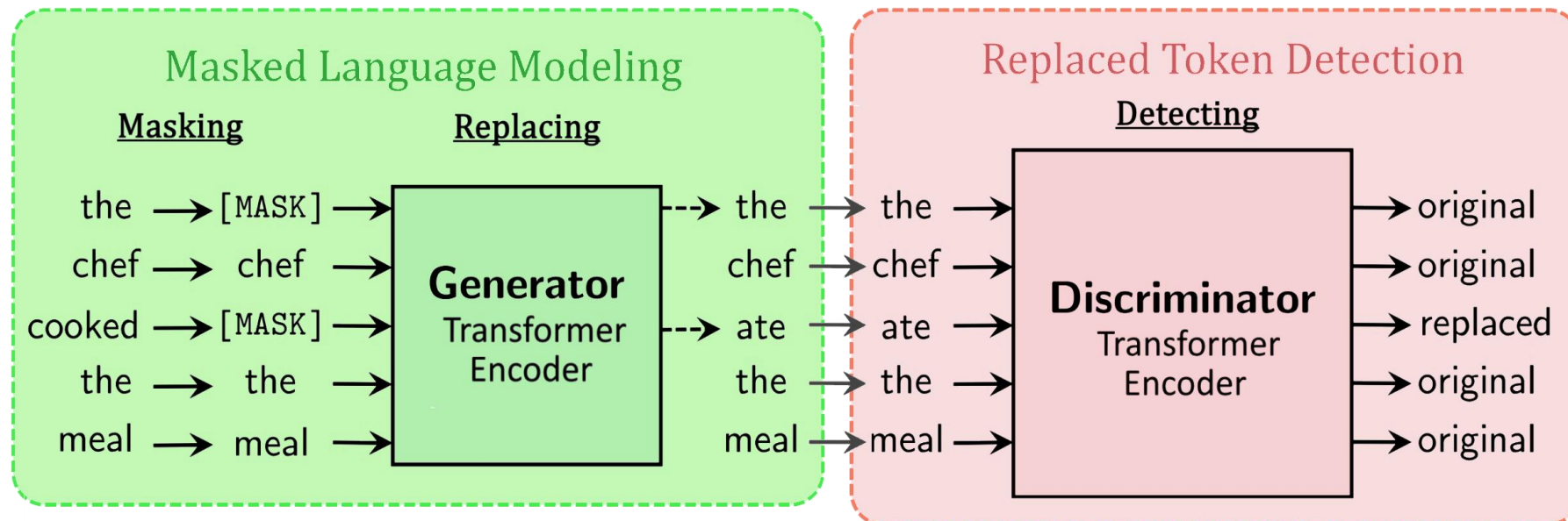
- ~~Next-Sentence-Prediction~~
- Dynamic Masking
- mehr Trainingsdaten

## *Transformer-Encoder*



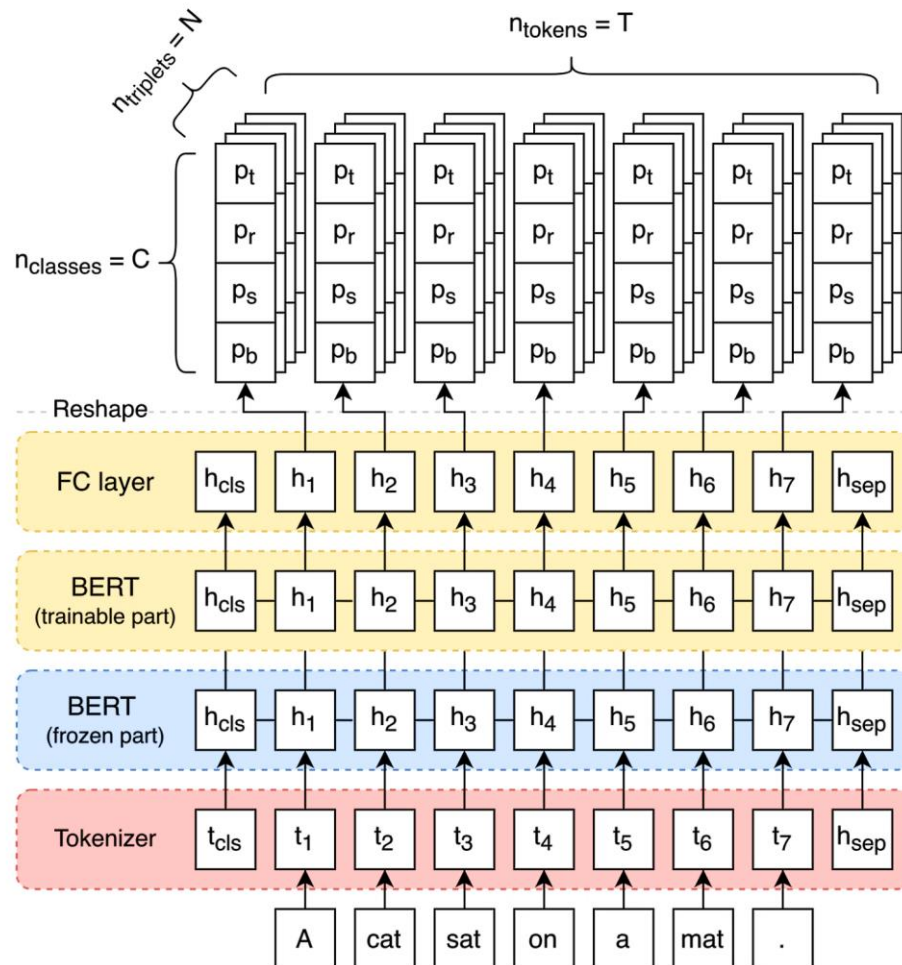
# Vortrainingsmethoden – ELECTRA

- ELECTRA-Modelle
  - GAN-Struktur
  - Replaced Token Detection





# DetIE-Modell



## ■ DetIE-Modelle

- aktuell leistungsfähigste Modell
- Vorhersage in einem Durchlauf
  - hohe Extraktionsgeschwindigkeit

## ■ Output

$$P_{out} \in \mathbb{R}^{T \times N \times C} \xrightarrow{\arg \max} C_{out} \in \mathbb{R}^{T \times N} \xrightarrow{filtern} C_{out} \in \mathbb{R}^{T \times n}$$

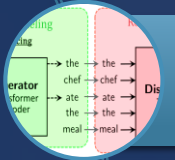
## ■ Optimierung

- optimale Zuordnung von Extraktionen und Lösungen

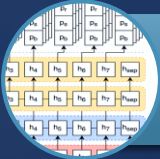
# Gliederung



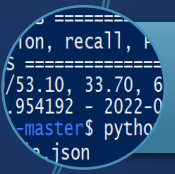
Einführung, Motivation



Vortrainingsmethoden



DetIE-Modell



Experimente und Ergebnisse



Ausblick und weitere Forschung

## Untergliederung

- Experimentelles Setup
- Datensätze
- Bewertungsschemas
- Ergebnisse
- Auswertung

# Experimentelles Setup

- Server mit 2 GPUs
  - TITAN V (12 GB RAM)
  - TITAN RTX (24 GB RAM)
- Original-Code benutzt
  - Trainingsdatensätze, Bewertungsschemas
  - PyTorch, Transformers, Hydra
  - Ca. 2000 Zeilen Code



UNIVERSITÄT  
LEIPZIG

```
ons: torch.Tensor,  
ict[str, torch.Tensor]) -> Tuple  
length = state["source_to_target"].size()  
  
selective_weights = self._get_input_and_selective  
der_step(input_choices, selective_weights, sta  
= self._get_generation_scores(state)  
  
hbeyer@mlserv1: ~  
===== EVALUATING OVER CARB =====  
AUC: 0.323      Optimal (precision, recall, F1): (0.672, 0.439  
===== COMPILING RESULTS =====  
Best sum Scores = (61.90/46.50/53.10, 33.70, 61.00/46.60/52.80  
total time: 2022-02-19 21:05:37.954192 - 2022-02-19 20:58:06.806  
(imojie) hbeyer@mlserv1:~/imojie-master$ python allennlp_script  
--param_path imojie/configs/imojie.json
```

# Datensätze

## ■ Übersicht

Verwendung	Datensatz	Sätze	Tupel
Training	IMoJIE	91.725	190.611
	LSOIE	34.780	100.862
Test	CaRB	641	2.715
	OIE2016	3.200	10.359

### ■ LSOIE-Datensatz

- Aus QA-SRL-Datensatz
- Konvertierungs-Algorithmus

### ■ IMoJIE-Datensatz

- Aus Wikipedia-Artikeln
- Nutzung von OIE-Modellen

# Bewertungsschemas

- Berechnung der Metriken

- $$Prec = \frac{\sum_t |G_t \cap P_t|}{\sum_t |P_t|} \quad Rec = \frac{\sum_t |G_t \cap P_t|}{\sum_t |G_t|} \quad F1 = \frac{2 * Prec * Rec}{Prec + Rec}$$

- AUC: Fläche unter der Precision-Recall-Curve

- Bewertungsschemas

- OIE2016
- WiRe57
- CaRB
- CaRB(1-1)

- Prozess der Messwertaufnahme



# Ergebnisse

## IMoJIE-Datensatz (92.000 Trainingssätze)

- Ø-Verbesserungen
- ohne IGL-CA: 1,0 F1- und 1,3 AUC-Punkte
  - mit IGL-CA: *keine Verbesserung*

Modell	Encoder	CaRB-Bewertungsschemas							
		CaRB		CaRB(1-1)		OIE16-C		WiRe57-C	
		F1	AUC	F1	AUC	F1	AUC	F1	
DetIE <sub>IMoJIE</sub>	BERT	52,1	37,0	40,7	24,7	56,3	39,0	36,4	
DetIE <sub>IMoJIE+IGL-CA</sub>	BERT	47,3	35,7	<u>43,2</u>	<u>29,8</u>	<u>67,4</u>	<u>54,2</u>	<u>37,7</u>	
DetIE <sub>IMoJIE</sub>	RoBERTa	<u>52,6</u>	<u>37,2</u>	41,0	24,8	56,9	39,5	37,1	
DetIE <sub>IMoJIE+IGL-CA</sub>	RoBERTa	48,1	35,9	<b>43,8</b>	<b>30,0</b>	<b>67,7</b>	54,0	<b>39,0</b>	
DetIE <sub>IMoJIE</sub>	ELECTRA	<b>52,7</b>	<b>37,9</b>	41,5	25,5	58,3	41,1	37,0	
DetIE <sub>IMoJIE+IGL-CA</sub>	ELECTRA	46,8	35,7	43,0	<b>30,0</b>	<b>67,7</b>	<b>55,1</b>	37,5	

# Ergebnisse

## LSOIE-Datensatz (35.000 Trainingssätze)

Ø-Verbesserungen • ohne IGL-CA: *keine Verbesserung*  
• mit IGL-CA: 1,1 F1- und 1,3 AUC-Punkte

Modell	Encoder	CaRB-Bewertungsschemas						
		CaRB		CaRB(1-1)		OIE16-C		WiRe57-C
		F1	AUC	F1	AUC	F1	AUC	F1
DetI <sub>E</sub> <sub>LSOIE</sub>	BERT	35,1	24,6	32,2	20,7	<b>65,9</b>	<u>51,4</u>	28,4
DetI <sub>E</sub> <sub>LSOIE+IGL-CA</sub>	BERT	36,9	<u>26,9</u>	34,2	23,0	63,4	50,4	30,1
DetI <sub>E</sub> <sub>LSOIE</sub>	RoBERTa	35,4	24,0	32,8	20,8	63,0	47,9	29,6
DetI <sub>E</sub> <sub>LSOIE+IGL-CA</sub>	RoBERTa	<b>38,1</b>	26,6	<b>35,4</b>	<u>23,1</u>	63,7	50,0	<b>31,5</b>
DetI <sub>E</sub> <sub>LSOIE</sub>	ELECTRA	35,0	24,6	32,3	20,9	<u>65,7</u>	51,2	29,1
DetI <sub>E</sub> <sub>LSOIE+IGL-CA</sub>	ELECTRA	<u>37,1</u>	<b>27,2</b>	<u>34,7</u>	<b>23,6</b>	<b>65,9</b>	<b>53,5</b>	<u>31,1</u>

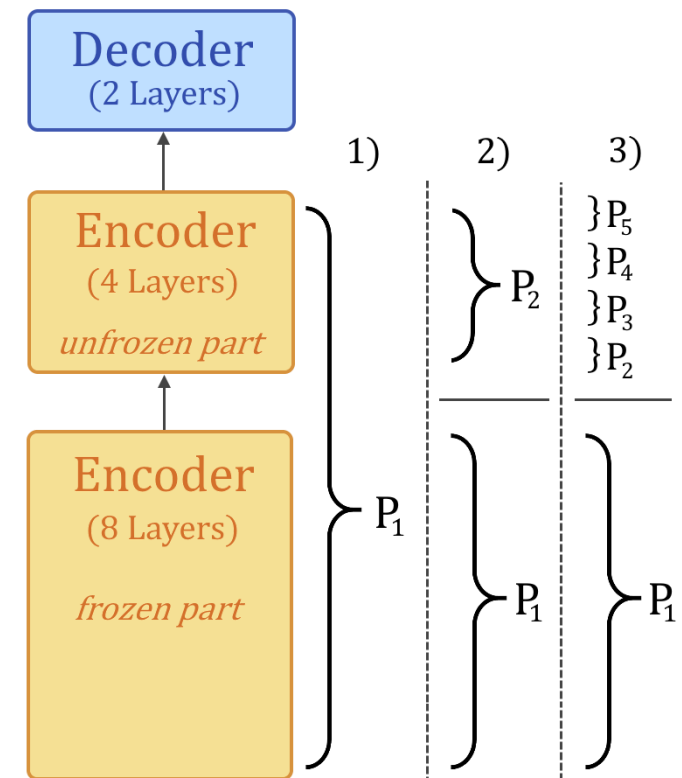
# Ergebnisse

## Endresultate (Vergleich mit bisherigen Modellen)

Modell	Encoder	CaRB-Bewertungsschemas							Geschwind. (Sätze je Sek.)
		CaRB		CaRB(1-1)		OIE16-C		WiRe57-C	
		F1	AUC	F1	AUC	F1	AUC	F1	
IMoJIE *	BERT	<u>53,5</u>	33,3	41,4	22,2	56,8	39,6	36,0	2,6
IGL-OIE *	BERT	52,4	33,7	41,1	22,9	55,0	36,0	34,9	142,0
CIGL-OIE *	BERT	<b>54,0</b>	35,7	42,8	24,6	59,2	40,0	36,8	142,0
OpenIE6 *	BERT	52,7	33,7	<b>46,4</b>	26,8	65,6	48,4	<b>40,0</b>	31,7
DetIE <sub>IMoJIE</sub>	BERT	52,1	37,0	40,7	24,7	56,3	39,0	36,4	500,0
DetIE <sub>IMoJIE+IGL-CA</sub>	BERT	47,3	35,7	43,2	<u>29,8</u>	<u>67,4</u>	<u>54,2</u>	37,7	203,2
DetIE <sub>IMoJIE</sub>	RoBERTa	52,6	<u>37,2</u>	41,0	24,8	56,9	39,5	37,1	522,1
DetIE <sub>IMoJIE+IGL-CA</sub>	RoBERTa	48,1	35,9	<u>43,8</u>	<b>30,0</b>	<b>67,7</b>	54,0	<u>39,0</u>	206,4
DetIE <sub>IMoJIE</sub>	ELECTRA	52,7	<b>37,9</b>	41,5	25,5	58,3	41,1	37,0	511,9
DetIE <sub>IMoJIE+IGL-CA</sub>	ELECTRA	46,8	35,7	43,0	<b>30,0</b>	<b>67,7</b>	<b>55,1</b>	37,5	205,9

# Ausblick

- eigene Experimente beendet
  - relevante Verbesserungen: *1,1 F1 1,3 AUC*
  - Verbesserungskriterien
  - DeBERTa-Modelle nicht für die OIE anwendbar
- zukünftige Forschung
  - erneutes Vortraining für DeBERTa-Encoder
  - absolutes und relatives Embedding
  - Verbesserungen in NLP und OIE erwartet
  - Vortraining: 2,6 und 16 TB GPU RAM nötig



DeBERTa-Experimente



Vielen Dank!

- Fragen und Diskussion



# Vortrainingsmethoden – DeBERTa, DeBERTaV3

## ■ DeBERTa-Modelle

- Disentangled Attention

$$\tilde{A} = A_{c \rightarrow c} + A_{c \rightarrow p} + A_{p \rightarrow c}$$

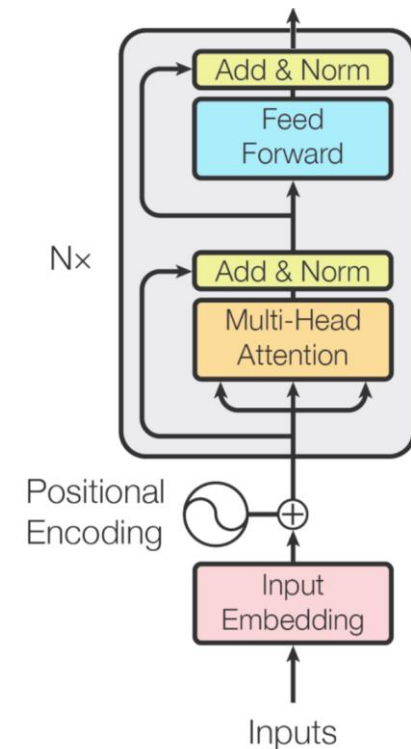
- Content-Matrizen:  $Q_c = HW, K_c = HW$
- Positions-Matrizen:  $Q_p = PW, K_p = PW$

- Embedding  $P \in \mathbb{R}^{2k \times d}$
- vektorweise Berechnungen (langsamer)
- höhere Leistungen

## ■ DeBERTaV3-Modelle

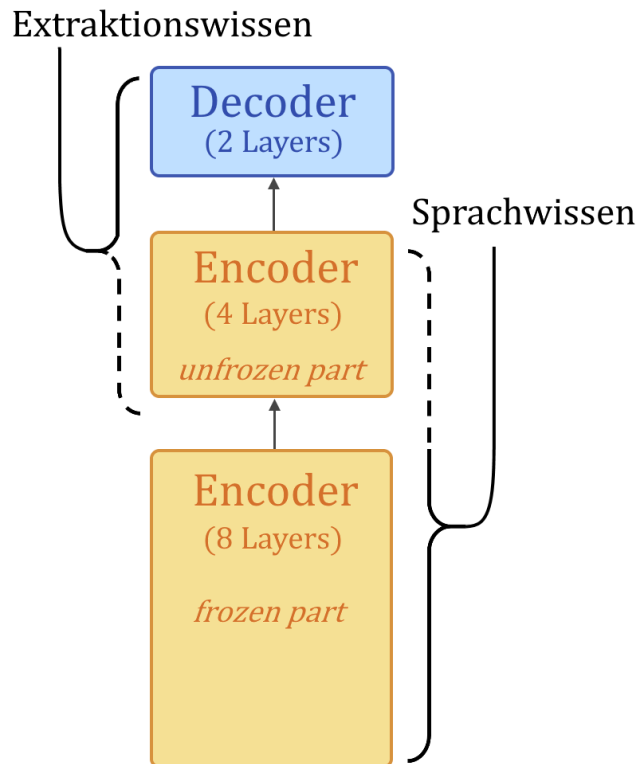
- Replaced Token Detection

*Transformer-Encoder*

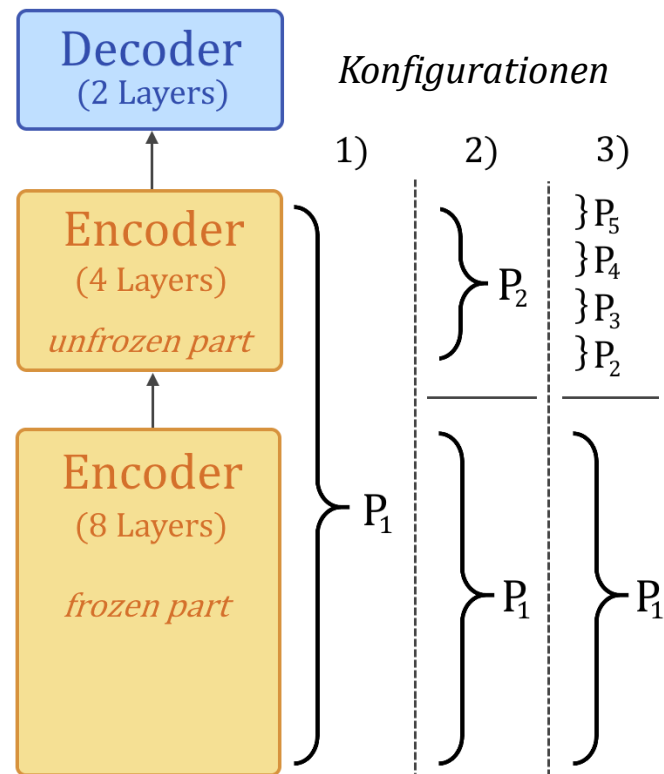


# DeBERTa-Experimente

## DetIE-Struktur



## Experiment-Konfigurationen



## ■ Ergebnisse

- immer noch sehr niedrige Leistungen

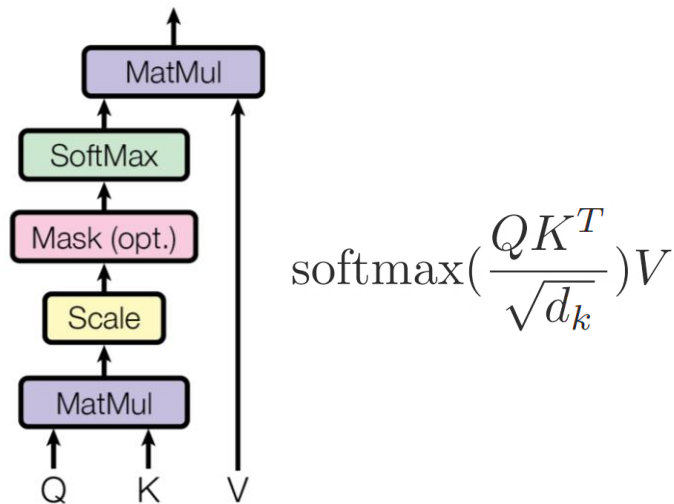
➤ fehlendes absolutes Position-Embedding

# Sprachverarbeitung in Transformer-Modellen

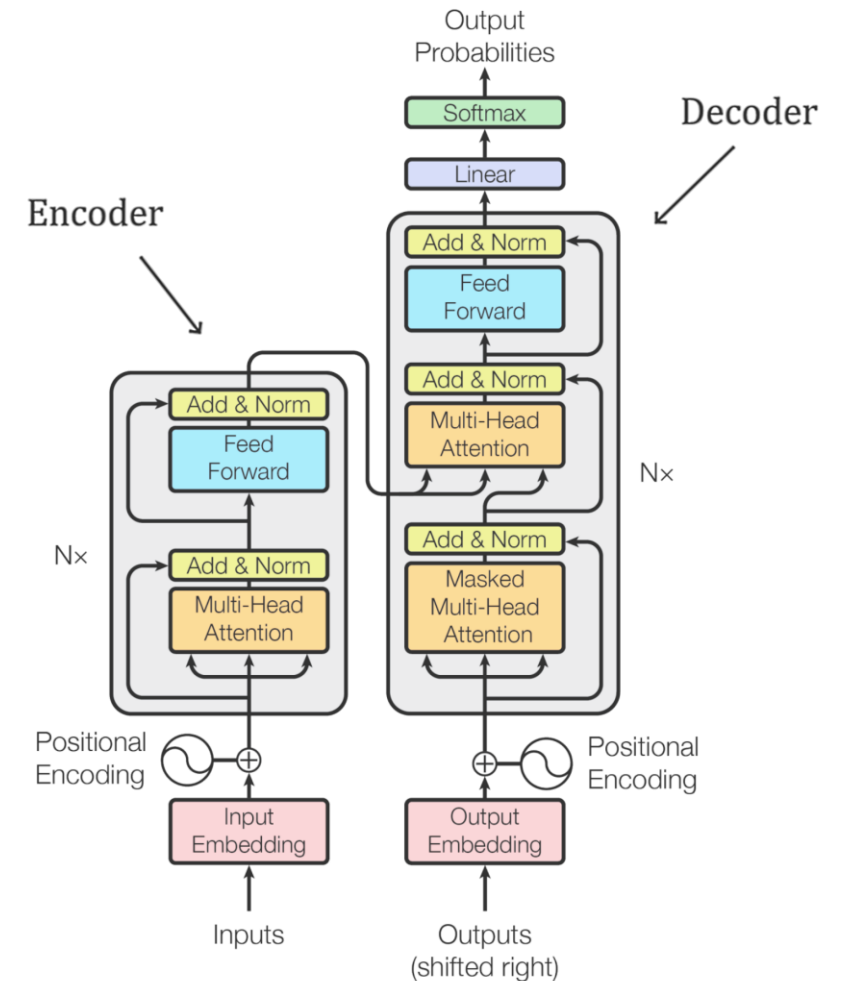
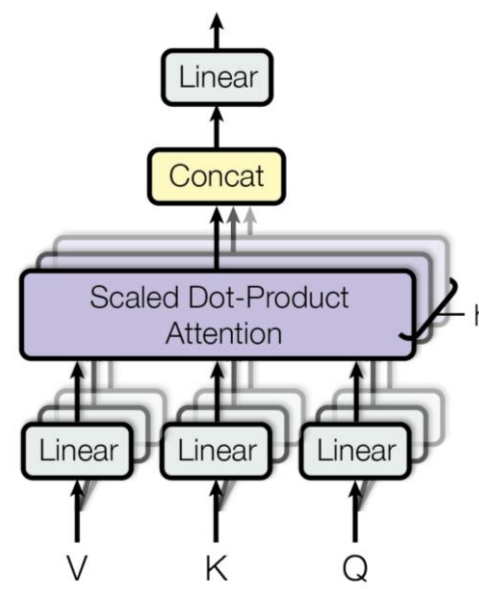
- Transformer-Modelle

- Encoder-Decoder-Struktur

- Scaled Dot-Product Attention

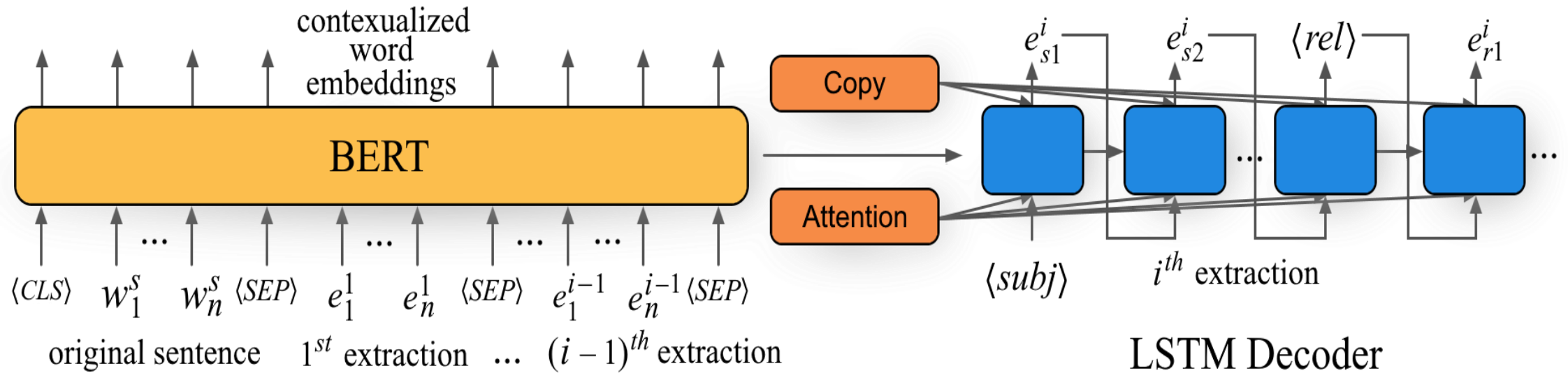


- Multi-Head Attention



# Quellenverzeichnis I/I

Transformer:	A. Vaswani, N. Shazeer, P. Niki, U. Jakob, J. Llion, G. N. Aidan, K. Lukasz und P. Illia, „Attention Is All You Need“, arXiv:1706.03762v5, 2017.
ELECTRA:	K. Clark, M.-T. Luong, Q.V. Le und C. D. Manning, „ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators“, arXiv:2003.10555, 2020.
DeBERTa:	P. He, X. Liu, J. Gao und W. Chen, „DeBERTa: Decoding-enhanced BERT with Disentangled Attention“, arXiv:2006.03654v6, 2021.
DetIE:	M. Vasilkovsky, A. Alekseev, V. Malykh, I. Shenbin, E. Tutubalina, D. Salikhov, M. Stepanov, A. Chertok und S. Nikolenko, „DetIE: Multilingual Open Information Extraction Inspired by Object Detection“, arXiv:2206.12514, 2022.
Uni Leipzig Logo:	2880px-Universität_Leipzig_logo - Institut für Testforschung und Testentwicklung e.V. (ITT) (itt-leipzig.de). [Online]. <a href="https://itt-leipzig.de/ueberuns/2880px-universitaet_leipzig_logo/">https://itt-leipzig.de/ueberuns/2880px-universitaet_leipzig_logo/</a> [16.02.2022].
IMoJIE:	K. Kolluru, S. Aggarwal, V. Rathore, Mausam und S. Chakrabarti, „IMoJIE: Iterative Memory-Based Joint Open Information Extraction“, arXiv:2005.08178v1, 2020.



- Encoder-Decoder-Modell
- Additive und Copy-Attention
- Iterativer Mechanismus

## Das IMojIE-Modell