Delexikalisiertes Cross-Linguales Konstituenz-Parsing für Mittelhochdeutsch (sowie Frühneuhochdeutsch)

Ercong Nie

Centrum für Informations- und Sprachverarbeitung (CIS)
Ludwig-Maximilian-Universität München (LMU)

4. März 2024

Outline

Ressourcen zur historischen deutschen Sprache

- **Korpora annotiert auf der Token-Ebene**
  - Deutsche Referenzkorpora[1]

[1] https://www.deutschdiachrondigital.de/

[2] https://korpling.german.hu-berlin.de/ddb-doku/index.htm

[3] https://www.uni-potsdam.de/de/guvdds/baumbankup

[4] https://ipchg.iu.edu/index.html

[5] https://www.chlg.ugent.be/

Ressourcen zur historischen deutschen Sprache

- **Korpora annotiert auf der Token-Ebene**
  - Deutsche Referenzkorpora[1]

- **Syntaktisch annotierte Korpora**

Id.	Name	Sprachen	Stil	#Wort
DDB[2]	Deutsche Diachrone Baumbank	AHD, MHD, FNHD	Tiger	8,580
ReF[3]	Referenzkorpus Frühneuhochdeutsch: Baumbank.UP	FNHD	Tiger	∼500,000
IPCHG[4]	Indiana-Baumbank des historischen Hochdeutschen	AHD, MHD, FNHD	PTB	∼10,000
CHLG[5]	Korpus des historischen Niederdeutschen	AND, MND	PTB	∼200,000

[1] https://www.deutschdiachrondigital.de/

[2] https://korpling.german.hu-berlin.de/ddb-doku/index.htm

[3] https://www.uni-potsdam.de/de/guvdds/baumbankup

[4] https://ipchg.iu.edu/index.html

[5] https://www.chlg.ugent.be/

- Für das Training eines automatischen Parsing-Systems ist in der Regel ein großes syntaktisch annotiertes Korpus (auch bekannt als **Baumbank**) erforderlich.

Automatisches Parsing für Historische Sprachen

- Für das Training eines automatischen Parsing-Systems ist in der Regel ein großes syntaktisch annotiertes Korpus (auch bekannt als **Baumbank**) erforderlich.

- Allerdings treten erhebliche **Schwierigkeiten** beim Aufbau einer großen Baumbank für **historische Sprachen** auf.
  - Knappheit an digitalen Textressourcen
  - Hoher Bedarf an linguistischer Expertise bei der Annotation
  - Großer manueller Aufwand

Automatisches Parsing für Historische Sprachen

- Für das Training eines automatischen Parsing-Systems ist in der Regel ein großes syntaktisch annotiertes Korpus (auch bekannt als **Baumbank**) erforderlich.

- Allerdings treten erhebliche **Schwierigkeiten** beim Aufbau einer großen Baumbank für **historische Sprachen** auf.
  - Knappheit an digitalen Textressourcen
  - Hoher Bedarf an linguistischer Expertise bei der Annotation
  - Großer manueller Aufwand

→ **Ausweg**: Training eines automatischen Systems zur syntaktischen Analyse unter Verwendung von Techniken des **cross-lingualen Transfers**.

Outline

Motivation

Hauptidee:

- Anstatt eine Sequenz von Wörtern zu parsen, erstellt der Delexikalisierungsparser einen Parse-Baum basierend auf einer Sequenz von Wortart-Tags.

Motivation

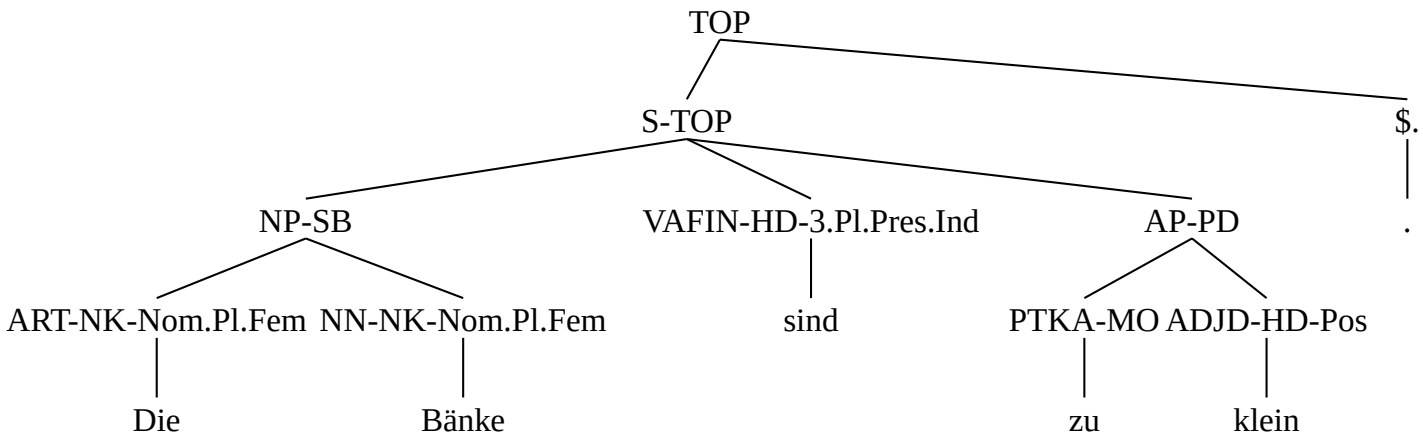
Hauptidee:

- Anstatt eine Sequenz von Wörtern zu parsen, erstellt der Delexikalisierungsparser einen Parse-Baum basierend auf einer Sequenz von Wortart-Tags.

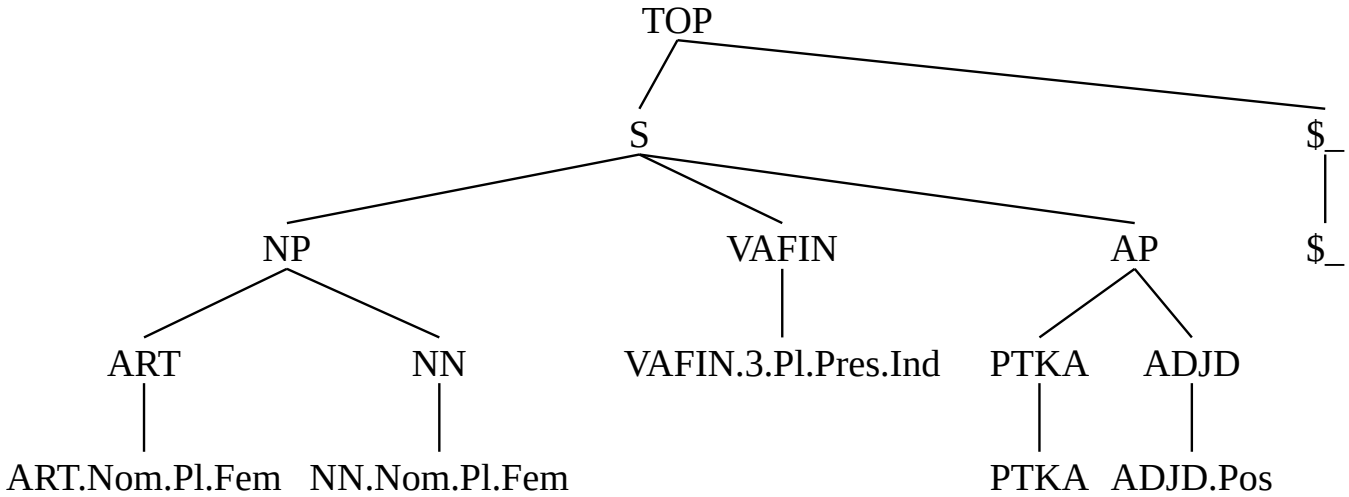
Motivation der Delexikalisierungsmethode:

- Die **Kontinuität** im Prozess der Sprachevolution führt zu **linguistischen Ähnlichkeiten** zwischen modernem Deutsch (MD) und MHD.
  - Ähnliche Satzstruktur
  - Ähnliche Wortstellung
  - ...
- Reichhaltige Ressourcen von MD-Texten mit syntaktischen Annotationen.
  - Tiger Corpus (Smith, 2003), usw.

Ein Beispiel der Delexikalisierung



(a) Originaler MD-Parse



(b) Delexikalisierter MD-Parse

Figure: Ein Beispiel, das den Delexikalisierungsprozess eines MD-Baumes veranschaulicht.

Delexikalisierungsparsing-System für MHD

Das Delexikalisierungsparsing-System für MHD besteht aus drei Modulen: *Wortart-Tagger*, *Tag-Mapper* und *Delexikalisierter Parser*.

Delexikalisierungsparsing-System für MHD

Das Delexikalisierungsparsing-System für MHD besteht aus drei Modulen: *Wortart-Tagger*, *Tag-Mapper* und *Delexikalisierter Parser*.

- **Wortart-Tagger**
  - Annotiert eine Sequenz von MHD-Tokens mit Wortart- und morphologischen Tags.
  - Wird auf dem ReM-Korpus unter Verwendung des RNNTaggers trainiert. (Schmid, 2019).

Delexikalisierungsparsing-System für MHD

Das Delexikalisierungsparsing-System für MHD besteht aus drei Modulen: *Wortart-Tagger*, *Tag-Mapper* und *Delexikalisierter Parser*.

- **Wortart-Tagger**
  - Annotiert eine Sequenz von MHD-Tokens mit Wortart- und morphologischen Tags.
  - Wird auf dem ReM-Korpus unter Verwendung des RNNTaggers trainiert. (Schmid, 2019).
- **Tag-Mapper**

  Ordnet Tags aus dem HiTS-Tagset (für ReM) dem STTS-Tagset (für MD-Baumbanken) zu.

MHD-Tag	MD-Tag
CARDD	CARD
DDART	ART
NA	NN

Delexikalisierungsparsing-System für MHD

Das Delexikalisierungsparsing-System für MHD besteht aus drei Modulen: *Wortart-Tagger*, *Tag-Mapper* und *Delexikalisierter Parser*.

- **Wortart-Tagger**
  - Annotiert eine Sequenz von MHD-Tokens mit Wortart- und morphologischen Tags.
  - Wird auf dem ReM-Korpus unter Verwendung des RNNTaggers trainiert. (Schmid, 2019).
- **Tag-Mapper**

  Ordnet Tags aus dem HiTS-Tagset (für ReM) dem STTS-Tagset (für MD-Baumbanken) zu.

MHD-Tag	MD-Tag
CARDD	CARD
DDART	ART
NA	NN

- **Delexikalisierter Parser**
  - Basiert auf dem Berkeley Neural Parser (Benepar) (Kitaev and Klein, 2018).
  - Wird auf der delexikalisierten Tiger Treebank (50.474 MD Parse-Bäume) trainiert.

Delexikalisierungsparsing-System für MHG

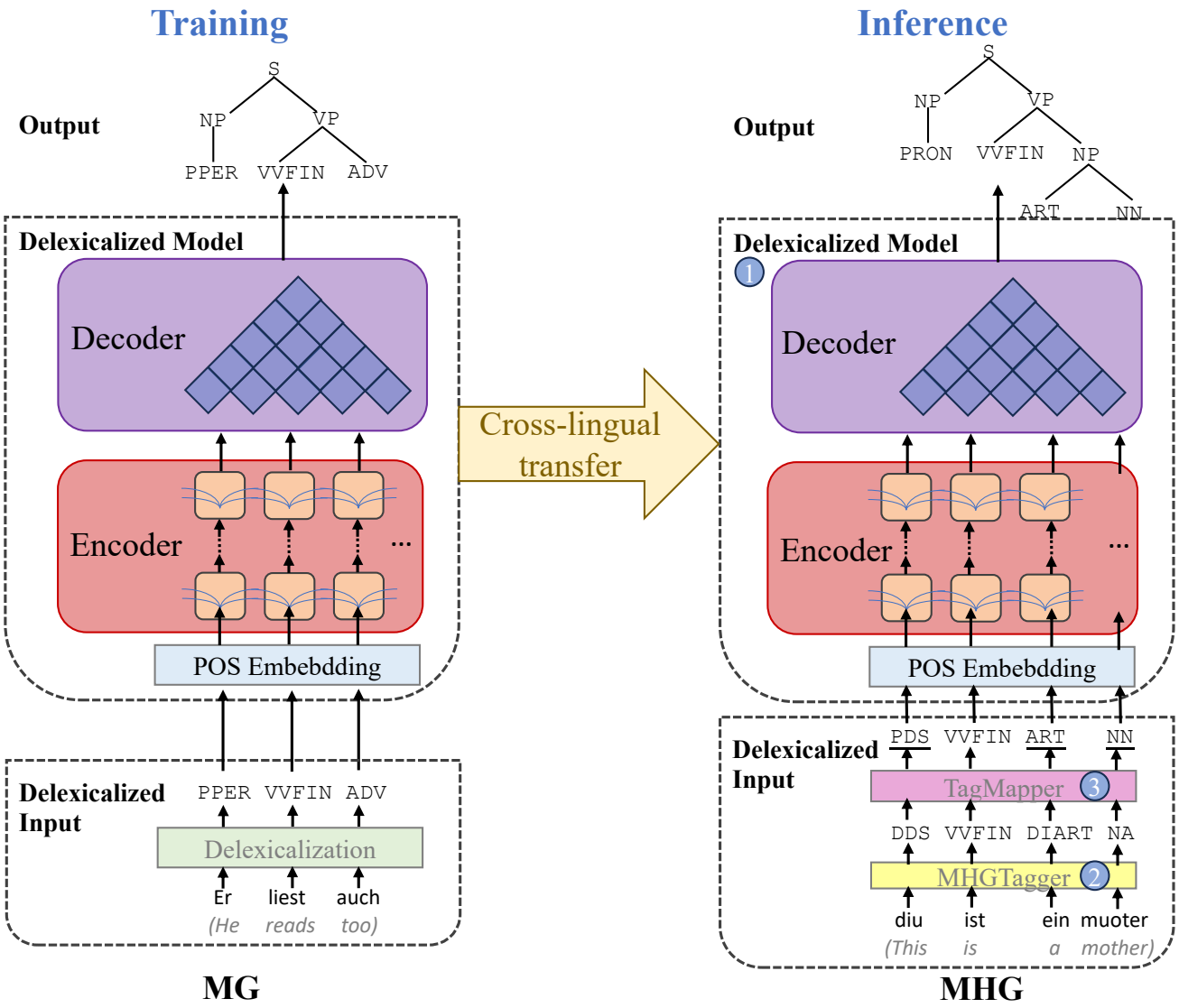


Figure: Überblick über das cross-linguale delexikalisierte Parsing-System für MHD.

Outline

Baselines

- **Vanilla Benepar**: Führt einen einfachen Zero-Shot Cross-Lingual Transfer durch, indem ein Benepar-Modell auf MD-Baumbanken ohne Delexikalisierung trainiert und dann direkt auf das Parsen von MHD-Eingabesätzen angewendet wird.

- **Tetra-Tagging mit Vortrainierten Sprachmodellen (PLMs)**: Eine Technik, die das Konstituenten-Parsing auf Sequenz-Labeling reduziert (Kitaev and Klein, 2020).
  - **gBERT**: Tetra-Tagging mit dem deutschen BERT-Modell (Chan et al., 2020)
  - **mBERT**: Tetra-Tagging mit dem multilingualen BERT-Modell (Devlin et al., 2019)

Hauptergebnisse

	Recall		Precision		FScore		CM	
	MG	MHG	MG	MHG	MG	MHG	MG	MHG
Baselines								
Vanilla Benepar	84.18	34.41	87.57	44.40	85.84	38.77	45.80	0.00
Tetra-gBERT	**86.31**	23.20	**88.19**	29.53	**87.24**	25.98	**51.70**	3.12
Tetra-mBERT	60.68	19.69	65.61	23.25	63.15	21.32	21.35	0.00
Unsere Methode								
Dexparser	81.39	**64.72**	84.89	**70.19**	83.10	**67.34**	39.03	**12.50**

Table: Parsing-Ergebnisse verschiedener Methoden des Cross-Lingual-Transfers. **CM** steht für vollständige Übereinstimmung“. Der beste Wert jeder Spalte ist **fett** markiert.

- Dexparser zeigt deutliche Vorteile beim Parsen von MHD im Vergleich zu anderen Baselines
- Dexparser erzielt auch vergleichbare Ergebnisse bei MD.

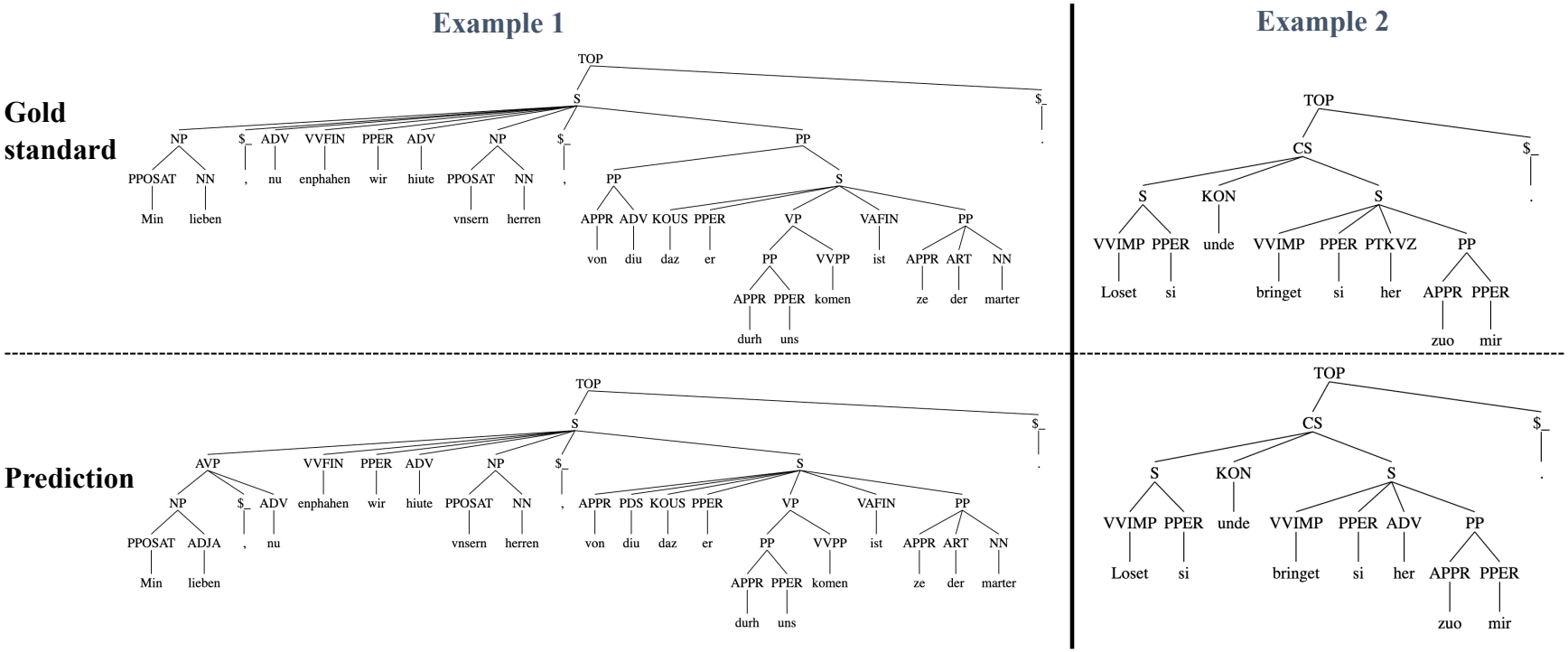
Ablationsstudie

	Recall	Precision	FScore	CM
Delexikalisierter Parser unter Verwendung von Gold-Tags	**66.18**	**71.17**	**68.59**	**14.58**
- *unter Verwendung von vorhergesagten Tags*	64.72	70.19	67.34	12.50
- *ohne Mapping*	59.16	68.82	63.63	7.29
- *ohne morphologische Information*	48.66	65.38	55.8	9.28

Table: Die MHD-Parsing-Ergebnisse unter Verwendung des delexikalisierten Parsers in der Ablationsstudie.

- Die **Qualität der Wortart-Annotation**, das **Tagset-Mapping** und die **Annotation morphologischer Informationen** tragen gemeinsam zur Leistung des Delexikalisierungsparsers bei MHD bei.

Fallstudie



Zwei Beispiele für die Bäume, die von unserem delexikalisierten Parser erzeugt wurden, im Vergleich zu den Referenz-Parsebäumern.

Outline

Verwendung des Delexikalisierungsparsings auf FNHD

Trainingdatensatz	Recall	Precision	FScore	CM
Nur MD	41.11	51.42	45.69	7.80
Nur FNHD	56.61	66.69	61.24	18.05
Gemischt-gleich	**57.87**	67.12	**62.15**	**18.90**
Gemischt-alle	57.21	**67.68**	62.01	18.80

Table: Experimentelle Ergebnisse der Anwendung von Delexikalisierungsparsing auf FNHD.

- Wir haben verschiedene Kombinationen von Trainingdatensatz ausprobiert:
  - **Nur MD**: Enthält ausschließlich delexikalisierte MD-Bäume (18977 MD-Bäume).
  - **Nur FNHD**: Enthält ausschließlich delexikalisierte FNHD-Bäume (18977 FNHD-Bäume).
  - **Gemischt-gleich**: Verwendet eine gleiche Anzahl gemischter MD- und FNHD-Bäume (18977 MD-Bäume + 18977 FNHD-Bäume).
  - **Gemischt-alle**: Kombiniert alle MD- und FNHD-Bäume(47474 MD-Bäume + 18977 FNHD-Bäume).

Verwendung des Tetra-Taggings auf FNHD

model	Recall	Precision	FScore	CompleteMatch	TaggingInternalAcc.	TaggingLeafAcc.
dbmdz/convbert-base-german-europeana-cased	70.91	73.08	71.98	27.90	82.61	95.57
benjamin/roberta-base-wechsel-german	**76.58**	**77.44**	**77.01**	**33.90**	**85.26**	**96.50**
dbmdz/bert-base-german-europeana-cased	75.19	76.26	75.72	32.30	84.79	96.38
redewiedergabe/bert-base-historical-german-rw-cased	71.73	73.80	72.75	29.25	83.16	95.71
bert-base-german-cased	70.52	72.81	71.64	26.95	82.37	95.56
dbmdz/distilbert-base-german-europeana-cased	66.29	69.25	67.74	24.75	81.10	95.23
dbmdz/electra-base-german-europeana-cased-generator	67.21	69.97	68.56	24.55	80.99	95.26
dbmdz/bert-base-historic-multilingual-cased	74.46	76.06	75.25	31.20	84.12	96.19

Wir haben die **Tetra-Tagging-Parsing-Methode** mit mehreren vortrainierten Sprachmodellen kombiniert und die Modelle auf FNHD-Baumbanken trainiert.

Verwendung des Benepar auf FNHD

	enhg_only	mix_equal	mix_all
model-convbert-german-europeana	**83.22**	82.99	83.12
model-roberta-wechsel-german	83.01	**83.05**	82.95
model-bert-german-europeana	82.24	**82.75**	82.35
model-historical-german	80.44	80.67	**80.68**
model-bert-base-german-cased	79.80	79.83	**79.91**
model-distilbert-german-europeana	78.38	**78.41**	78.39
model-electra-german-europeana	78.15	**78.18**	78.03
model-bert-historic-multilingual	**81.37**	81.26	81.15

Wir haben den **Berkeley Neural Parser** mit mehreren vortrainierten Sprachmodellen kombiniert und die Modelle auf verschiedene Kombinationen von Baumbanken trainiert. (F1-Score wird angegeben.)

Vielen Dank
für Ihre Aufmerksamkeit!

References

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2020. Tetra-tagging: Word-synchronous parsing with linear-time inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6255–6261, Online. Association for Computational Linguistics.

Helmut Schmid. 2019. Deep learning-based morphological taggers and lemmatizers for annotating historical texts. In *Proceedings of the 3rd international conference on digital access to textual cultural heritage*, pages 133–137.

George Smith. 2003. A brief introduction to the tiger treebank, version 1. Technical report, Universität Potsdam.