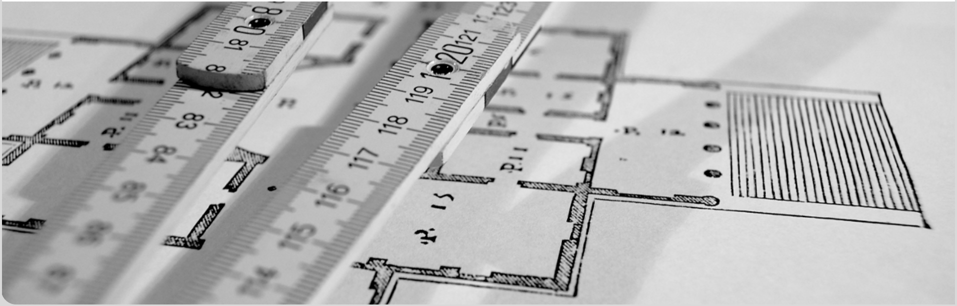
Karlsruhe Institute of Technology

Rule-Based Reordering on Multiple Syntactic Levels in SMT

Ge Wu | September 3, 2014

INSTITUTE FOR ANTHROPOMATICS AND ROBOTICS (IAR)



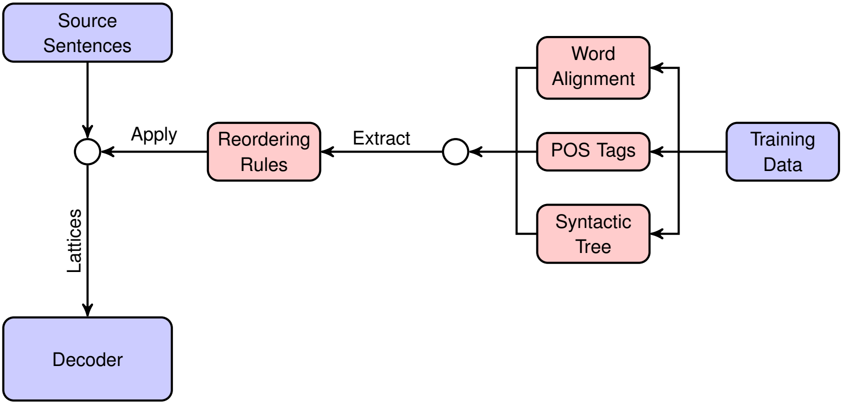
Outline

1. Introduction
2. Multi-Level-Tree (MLT) Reordering
   - Extension of tree rule based reordering to multiple syntactic levels
3. Evaluation
   - English to Chinese: **1.61** Improvement of BLEU score
   - Chinese to English: **2.16** Improvement of BLEU score
4. Conclusion

Introduction

- Rule-based pre-ordering approaches [Rottmann and Vogel 2007; Niehues and Kolss 2009; Herrmann et al. 2013]
- Hierarchical phrase-based model [Chiang 2007]
- More adaptive pre-ordering approach for Chinese based on syntactic structures

Pre-ordering System



Rreordering Rules

- **Short rules**

```
after the accident -> the accident after (0.5)

WRB MD DT -> DT WRB DT (0.3)
```

Rreordering Rules

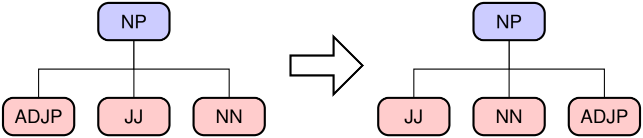
- Short rules
- **Long rules**

```
NN * MD -> * MD NN (0.14)
```

Rreordering Rules

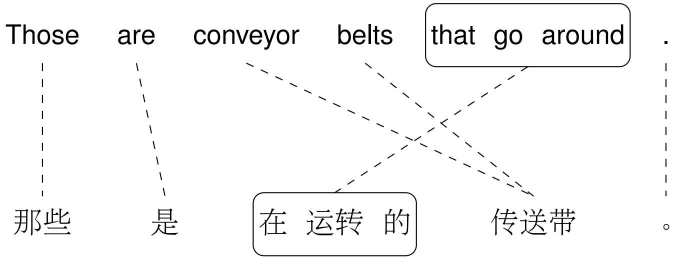
- Short rules
- Long rules
- **Tree rules**

```
NP ( ADJP JJ NN ) -> JJ NN ADJP (0.16)
```



Chinese Word Orders

- **Premodifier instead of postmodifier**
  - Adverbials
  - Relative clauses
  - Preposition phrases



Chinese Word Orders

- Premodifier instead of postmodifier
- **Questions**

How will you cook the chicken ?

You will how cook chicken ?

你 会 怎么 做 鸡肉 呢 ?

Chinese Word Orders

- Premodifier instead of postmodifier
- Questions
- **Special sentence constructions**

There aren't many people around that are really involved with architecture as clients.

Never would India have thought on this scale before.

Chinese Word Orders

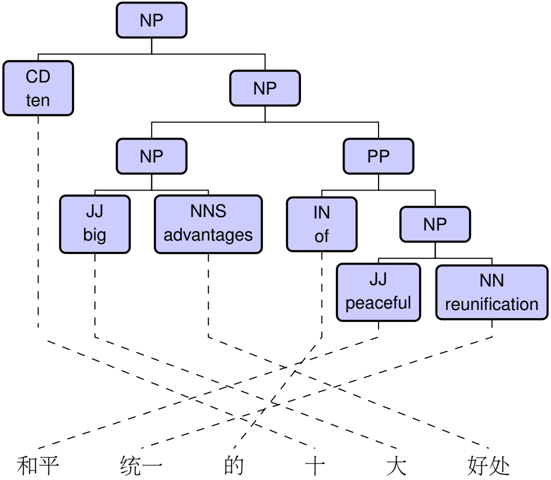
- Premodifier instead of postmodifier
- Questions
- Special sentence constructions
- **Long distance position change**

I find this very much disturbing when we are talking about what is going on right and wrong with democracy these days.

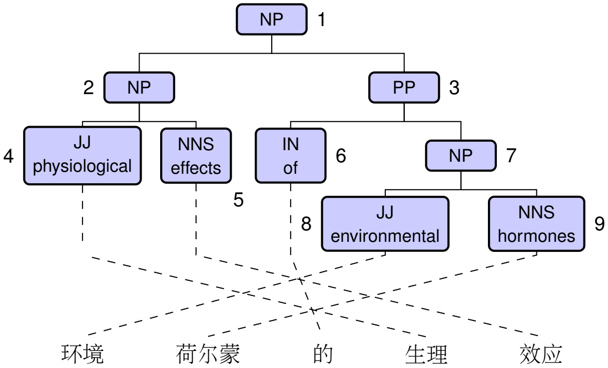
现在，每当我跟别人讨论我们的民主什么是对的，什么是错的我都为此觉得很无力。

Reordering on multiple syntactic levels

Extension of tree rule based reordering to multiple syntactic levels

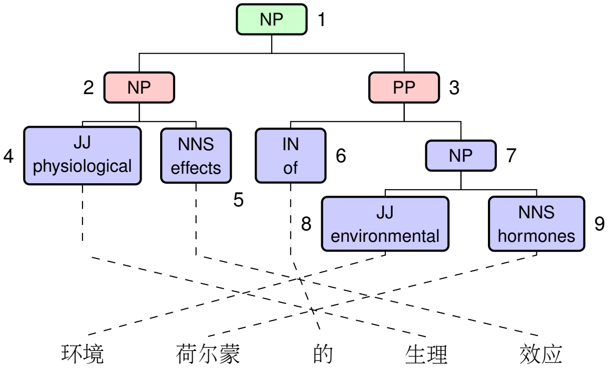


Reordering Patterns



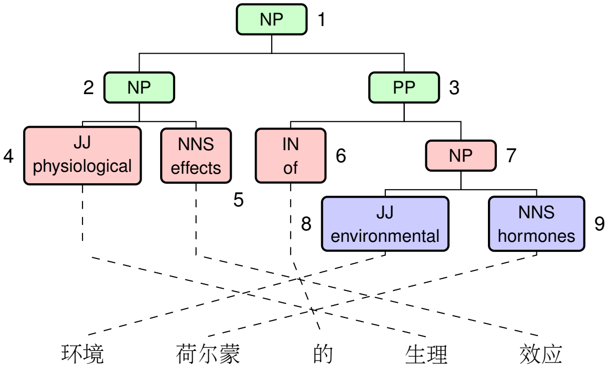
`Root Depth Pattern`

Reordering Patterns



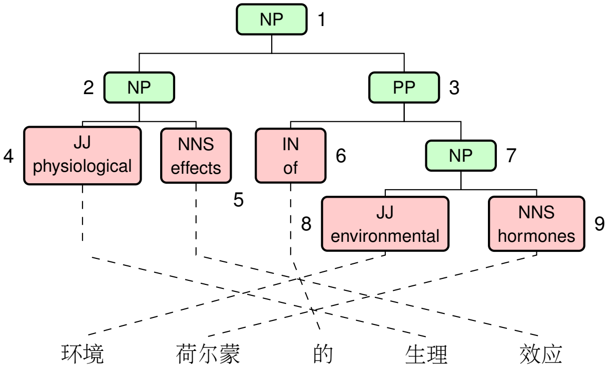

```
Root  Depth  Pattern
1     1      NP ( NP0 PP1 ) -> 1 0
```

Reordering Patterns



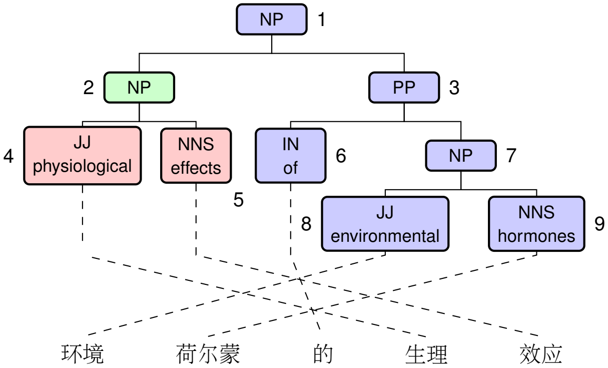
Root	Depth	Pattern
1	1	NP ($\color{red}{NP_0}$ $\color{red}{PP_1}$) -> 1 0
1	2	NP (NP ($\color{red}{JJ_0}$ $\color{red}{NNS_1}$) PP ($\color{red}{IN_2}$ $\color{red}{NP_3}$)) -> 3 2 0 1

Reordering Patterns



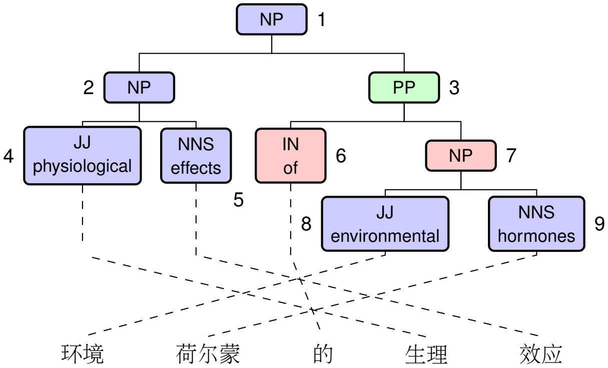
Root	Depth	Pattern
1	1	NP (NP_0 PP_1) -> 1 0
1	2	NP (NP (JJ_0 NNS_1) PP (IN_2 NP_3)) -> 3 2 0 1
1	3	NP (NP (JJ_0 NNS_1) PP (IN_2 NP (JJ_3 NNS_4))) -> 3 4 2 0 1

Reordering Patterns



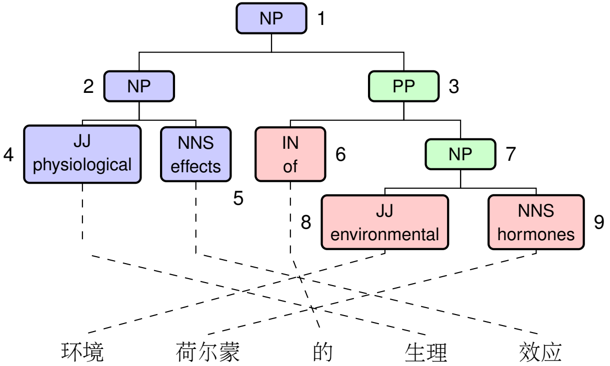
Root	Depth	Pattern
1	1	NP (NP_0 PP_1) -> 1 0
1	2	NP (NP (JJ_0 NNS_1) PP (IN_2 NP_3)) -> 3 2 0 1
1	3	NP (NP (JJ_0 NNS_1) PP (IN_2 NP (JJ_3 NNS_4))) -> 3 4 2 0 1

Reordering Patterns



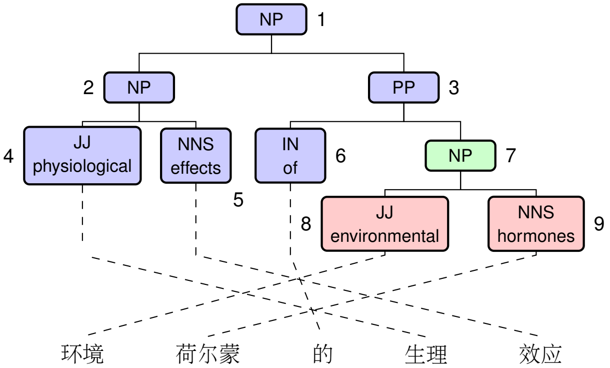
Root	Depth	Pattern
1	1	NP (NP_0 PP_1) -> 1 0
1	2	NP (NP (JJ_0 NNS_1) PP (IN_2 NP_3)) -> 3 2 0 1
1	3	NP (NP (JJ_0 NNS_1) PP (IN_2 NP (JJ_3 NNS_4))) -> 3 4 2 0 1
3	1	PP (IN_0 NP_1) -> 1 0

Reordering Patterns



Root	Depth	Pattern
1	1	NP (NP_0 PP_1) -> 1 0
1	2	NP (NP (JJ_0 NNS_1) PP (IN_2 NP_3)) -> 3 2 0 1
1	3	NP (NP (JJ_0 NNS_1) PP (IN_2 NP (JJ_3 NNS_4))) -> 3 4 2 0 1
3	1	PP (IN_0 NP_1) -> 1 0
3	2	PP (IN_0 NP (JJ_1 NNS_2)) -> 1 2 0

Reordering Patterns



Root	Depth	Pattern
1	1	NP (NP_0 PP_1) -> 1 0
1	2	NP (NP (JJ_0 NNS_1) PP (IN_2 NP_3)) -> 3 2 0 1
1	3	NP (NP (JJ_0 NNS_1) PP (IN_2 NP (JJ_3 NNS_4))) -> 3 4 2 0 1
3	1	PP (IN_0 NP_1) -> 1 0
3	2	PP (IN_0 NP (JJ_1 NNS_2)) -> 1 2 0

Rule Extraction

- **Search from all nodes with all possible depths**
- Rule probability

```
ADJP ( JJ , JJ ) -> 0 1 2    (0.84239130 = 310 / 368)
ADJP ( JJ , JJ ) -> 0 2 1    (0.02989130 =  11 / 368)
ADJP ( JJ , JJ ) -> 1 0 2    (0.07880435 =  29 / 368)
ADJP ( JJ , JJ ) -> 1 2 0    (0.02173913 =   8 / 368)
ADJP ( JJ , JJ ) -> 2 0 1    (0.02717391 =  10 / 368)
```

- Rule pruning
- Rule number doubles in comparison with tree rules

Rule Extraction

- Search from all nodes with all possible depths
- **Rule probability**

```
ADJP ( JJ , JJ ) -> 0 1 2    (0.84239130 = 310 / 368)
ADJP ( JJ , JJ ) -> 0 2 1    (0.02989130 =  11 / 368)
ADJP ( JJ , JJ ) -> 1 0 2    (0.07880435 =  29 / 368)
ADJP ( JJ , JJ ) -> 1 2 0    (0.02173913 =   8 / 368)
ADJP ( JJ , JJ ) -> 2 0 1    (0.02717391 =  10 / 368)
```

- Rule pruning
- Rule number doubles in comparison with tree rules

Rule Extraction

- Search from all nodes with all possible depths
- Rule probability

```
ADJP ( JJ , JJ ) -> 0 1 2    (0.84239130 = 310 / 368)
ADJP ( JJ , JJ ) -> 0 2 1    (0.02989130 =  11 / 368)
ADJP ( JJ , JJ ) -> 1 0 2    (0.07880435 =  29 / 368)
ADJP ( JJ , JJ ) -> 1 2 0    (0.02173913 =   8 / 368)
ADJP ( JJ , JJ ) -> 2 0 1    (0.02717391 =  10 / 368)
```

- **Rule pruning**
- Rule number doubles in comparison with tree rules

Rule Extraction

- Search from all nodes with all possible depths
- Rule probability

```
ADJP ( JJ , JJ ) -> 0 1 2    (0.84239130 = 310 / 368)
ADJP ( JJ , JJ ) -> 0 2 1    (0.02989130 =  11 / 368)
ADJP ( JJ , JJ ) -> 1 0 2    (0.07880435 =  29 / 368)
ADJP ( JJ , JJ ) -> 1 2 0    (0.02173913 =   8 / 368)
ADJP ( JJ , JJ ) -> 2 0 1    (0.02717391 =  10 / 368)
```

- Rule pruning
- **Rule number doubles in comparison with tree rules**

Rule Application

- **Search from all nodes with all possible depths**
- Search depth decreases to avoid duplicate applications

```
PP ( IN0 NP1 ) -> 1 0
PP ( IN0 NP ( JJ1 NNS2 ) ) -> 1 2 0
```

- Reorderings as paths in word lattices (size doubles approx.)
- Threshold for adding a path

Rule Application

- Search from all nodes with all possible depths
- **Search depth decreases to avoid duplicate applications**

$\texttt{PP (} \texttt{IN}_0 \texttt{ NP}_1 \texttt{) -> 1 0}$

$\texttt{PP (} \texttt{IN}_0 \texttt{ NP (JJ}_1 \texttt{ NNS}_2 \texttt{)) -> 1 2 0}$

- Reorderings as paths in word lattices (size doubles approx.)
- Threshold for adding a path

Rule Application

- Search from all nodes with all possible depths
- Search depth decreases to avoid duplicate applications

PP (IN_0 NP_1) -> 1 0
PP (IN_0 NP (JJ_1 NNS_2)) -> 1 2 0

- **Reorderings as paths in word lattices (size doubles approx.)**
- Threshold for adding a path

Rule Application

- Search from all nodes with all possible depths
- Search depth decreases to avoid duplicate applications

```
PP ( IN0 NP1 ) -> 1 0
PP ( IN0 NP ( JJ1 NNS2 ) ) -> 1 2 0
```

- Reorderings as paths in word lattices (size doubles approx.)
- **Threshold for adding a path**

Results: English -> Chinese

Data: LDC & TED, 1 reference
Train: 75MB / 454K sentences
Dev: 164KB / 919 sentences
Test: 263KB / 1663 sentences

	BLEU Score	Improvement	TER
Baseline	12.07		72.15
+Short Rules	12.50	0.43	71.41
+Long Rules	12.99	0.92	70.71
+Tree Rules	13.38	1.31	68.27
+MLT Rules	13.81	1.74	68.20
Oracle Reordering	**18.58**	**6.51**	**62.13**
Long Rules	12.31	0.24	71.81
Tree Rules	13.30	1.23	70.42
MLT Rules	13.68	1.61	70.25

Results: Chinese -> English

Data: LDC, 3 references

Train: 47MB / 303K sentences

Dev: 142KB / 919 sentences

Test: 220KB / 1663 sentences

	BLEU Score	Improvement	TER
Baseline	21.80		62.09
+Short Rules	22.90	1.10	61.64
+Long Rules	23.13	1.33	61.43
+Tree Rules	23.84	2.04	60.95
+MLT Rules	24.14	2.34	60.79
Oracle Reordering	**26.80**	**5.00**	**56.97**
Long Rules	22.10	0.30	62.21
Tree Rules	23.35	1.55	61.52
MLT Rules	23.96	2.16	60.83

Conclusion

- **Better translation quality**
  - English to Chinese: **1.61** Improvement of BLEU score
  - Chinese to English: **2.16** Improvement of BLEU score
- Better syntactic structure
  - More possible reorderings
  - Improvement for more complicated reorderings
- Space for further improvement

Conclusion

- Better translation quality
  - English to Chinese: **1.61** Improvement of BLEU score
  - Chinese to English: **2.16** Improvement of BLEU score
- **Better syntactic structure**
  - More possible reorderings
  - Improvement for more complicated reorderings
- Space for further improvement

Conclusion

- Better translation quality
  - English to Chinese: **1.61** Improvement of BLEU score
  - Chinese to English: **2.16** Improvement of BLEU score
- Better syntactic structure
  - More possible reorderings
  - Improvement for more complicated reorderings
- **Space for further improvement**

Outlook

- **Better reordering approaches**
- Vector presentation instead of POS tags as features
- Reordering with less information

Outlook

- Better reordering approaches
- **Vector presentation instead of POS tags as features**
- Reordering with less information

Outlook

Karlsruhe Institute of Technology

- Better reordering approaches
- Vector presentation instead of POS tags as features
- **Reordering with less information**

Thank you for your attention

Data Size

English -> Chinese

Data Set		#Sentence	#Word		Size (Byte)	
			English	Chinese	English	Chinese
Training Data	LDC	303K	10.96M	8.56M	60.88M	47.27M
	TED	151K	2.58M	2.86M	14.24M	15.63K
Development Data		919	30K	25K	164K	142K
Test Data		1663	47K	38K	263K	220K

Chinese -> English

Data Set	#Sentence	#Word		Size (Byte)	
		Chinese	English	Chinese	English
Training Data	303K	8.56M	10.96M	47.27M	60.88M
Development Data	919	25K	30K	142K	164K
Test Data	1663	38K	47K	220K	263K

Lattice Size

System: English –> Chinese
Data set: test data

	Numer of Rules	Size of Lattices
Short Rules	362873	13M
Long Rules	106081	6.8M
Tree Rules	5067	7.3M
MLT Rules	10312	12M

References I

Alexandra Birch. „Reordering Metrics for Statistical Machine Translation“. In: (2011).

Alexandra Birch, Miles Osborne, and Phil Blunsom. „Metrics for MT Evaluation: Evaluating Reordering“. In: *Machine Translation* 24.1 (Mar. 2010). ISSN: 0922-6567. DOI: 10.1007/s10590-009-9066-5. URL: http://dx.doi.org/10.1007/s10590-009-9066-5.

Phil Blunsom, Edward Grefenstette, Nal Kalchbrenner, et al. „A Convolutional Neural Network for Modelling Sentences“. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. 2014.

References II

David Chiang. „Hierarchical Phrase-Based Translation". In: *computational linguistics* 33.2 (2007), pp. 201–228.

Michael Collins, Philipp Koehn, and Ivona Kučerová. „Clause Restructuring for Statistical Machine Translation". In: *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2005, pp. 531–540.

Josep M Crego and Nizar Habash. „Using Shallow Syntax Information to Improve Word Alignment and Reordering for SMT". In: *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics. 2008, pp. 53–61.

References III

Marie-Catherine De Marneffe, Bill MacCartney, Christopher D Manning, et al. „Generating Typed Dependency Parses from Phrase Structure Parses“. In: *Proceedings of LREC*. Vol. 6. 2006, pp. 449–454.

Nizar Habash. „Syntactic Preprocessing for Statistical Machine Translation“. In: *MT Summit XI* (2007), pp. 215–222.

Teresa Herrmann, Jan Niehues, and Alex Waibel. „Combining Word Reordering Methods on Different Linguistic Abstraction Levels for Statistical Machine Translation“. In: *Proceedings of the Seventh Workshop on Syntax, Semantics and Structure in Statistical Translation*. Atlanta, Georgia: Association for Computational Linguistics, June 2013, pp. 39–47. URL: http://www.aclweb.org/anthology/W13-0805.

References IV

- Teresa Herrmann et al. *Analyzing the Potential of Source Sentence Reordering in Statistical Machine Translation*. 2013.
- Philipp Koehn. *Statistical Machine Translation*. 1st. New York, NY, USA: Cambridge University Press, 2010. ISBN: 0521874157, 9780521874151.
- Philipp Koehn et al. „Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation“. In: *IWSLT*. 2005, pp. 68–75.
- Uri Lerner and Slav Petrov. „Source-Side Classifier Preordering for Machine Translation“. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '13)*. 2013.
- Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. „Building a Large Annotated Corpus of English: The Penn Treebank“. In: *Computational linguistics* 19.2 (1993), pp. 313–330.

References V

- Tomas Mikolov et al. „Efficient Estimation of Word Representations in Vector Space“. In: *arXiv preprint arXiv:1301.3781* (2013).
- Jan Niehues and Muntsin Kolss. „A POS-Based Model for Long-Range Reorderings in SMT“. In: *Proceedings of the Fourth Workshop on Statistical Machine Translation*. Association for Computational Linguistics. Athens, Greece, 2009, pp. 206–214.
- Kishore Papineni et al. „BLEU: a Method for Automatic Evaluation of Machine Translation“. In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2002, pp. 311–318.
- Maja Popovic and Hermann Ney. „POS-Based Word Reorderings for Statistical Machine Translation“. In: *International Conference on Language Resources and Evaluation*. 2006, pp. 1278–1283.

References VI

- Kay Rottmann and Stephan Vogel. *Word Reordering in Statistical Machine Translation with a POS-Based Distortion Model*. 2007.
- Beatrice Santorini. „Part-of-Speech Tagging Guidelines for the Penn Treebank Project (3rd revision)“. In: (1990).
- Christoph Tillmann. „A Unigram Orientation Model for Statistical Machine Translation“. In: *Proceedings of HLT-NAACL 2004: Short Papers*. Association for Computational Linguistics. 2004, pp. 101–104.
- Chao Wang, Michael Collins, and Philipp Koehn. „Chinese Syntactic Reordering for Statistical Machine Translation“. In: *EMNLP-CoNLL*. Citeseer. 2007, pp. 737–745.

References VII

Yuqi Zhang, Richard Zens, and Hermann Ney. „Chunk-Level Reordering of Source Language Sentences with Automatically Learned Rules for Statistical Machine Translation“. In: *Proceedings of the NAACL-HLT 2007/AMTA Workshop on Syntax and Structure in Statistical Translation*. Association for Computational Linguistics. 2007, pp. 1–8.