Discontinuous Statistical Machine Translation with Target-Side Dependency Syntax

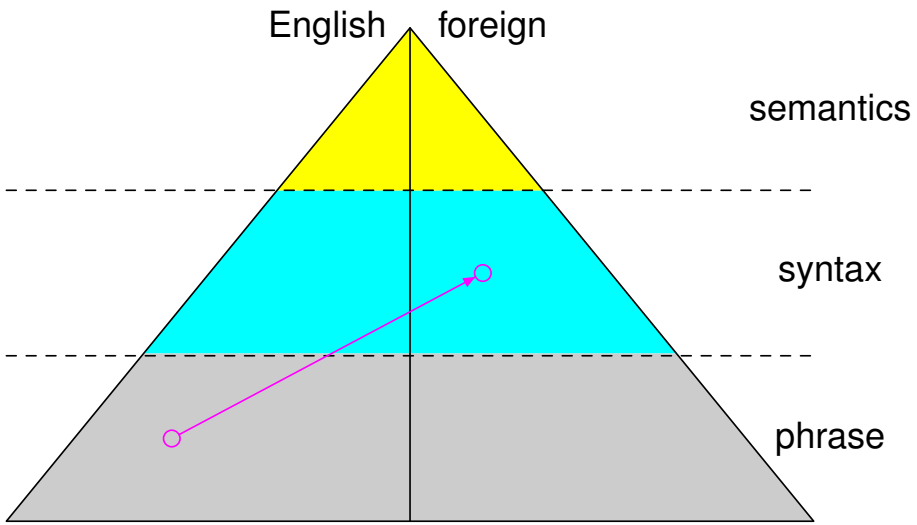
Nina Seemann Andreas Maletti

University of Stuttgart
– Institute for Natural Language Processing –
Pfaffenwaldring 5b
70569 Stuttgart

September 17, 2015

Outline

Syntax-based Machine Translation



- Source language side is a string
- Target language side requires syntactic annotations

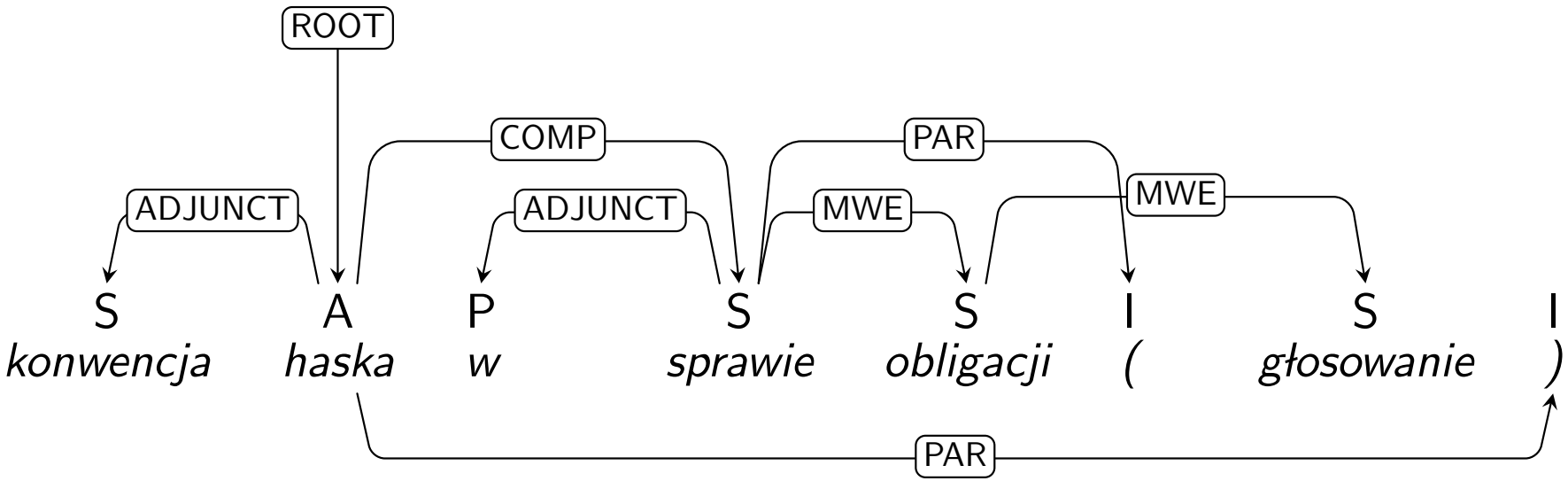
Discontinuous Target Languages

We want to translate from English to Russian and Polish:

- morphologically rich
- free word order languages
- grammatically agreeing parts spread out over whole sentence
- syntax difficult to express in terms of constituency structure
- not parseable by constituency parser
- but by dependency parsers

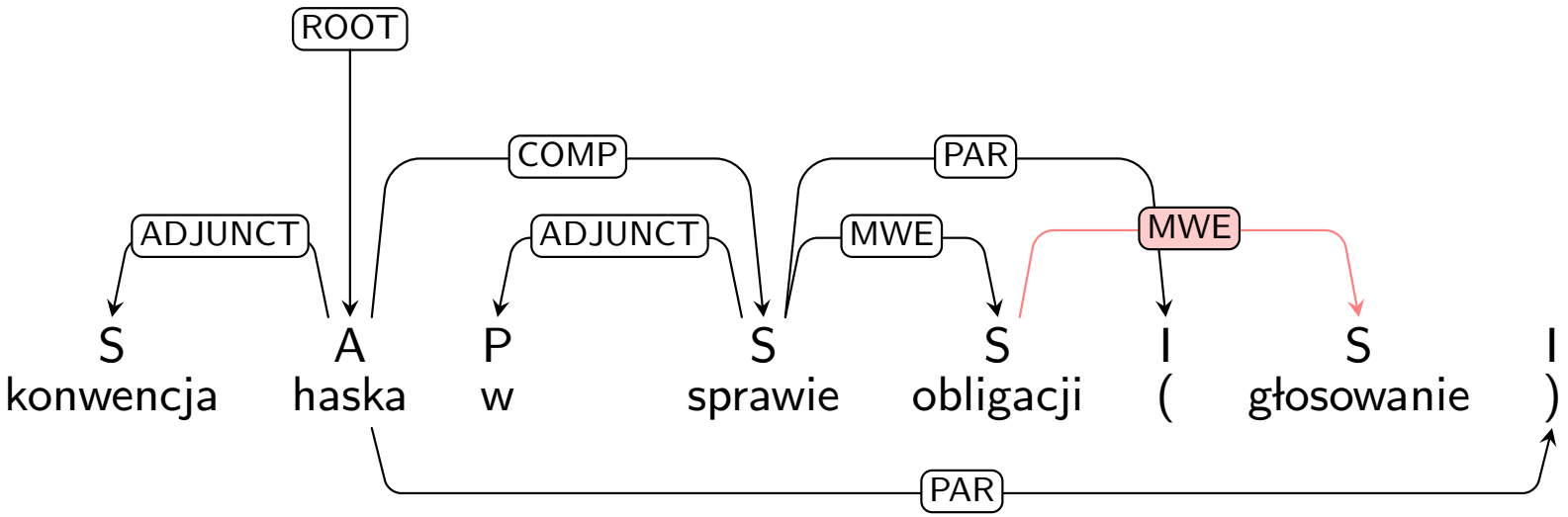
Outline

Dependency Parsing

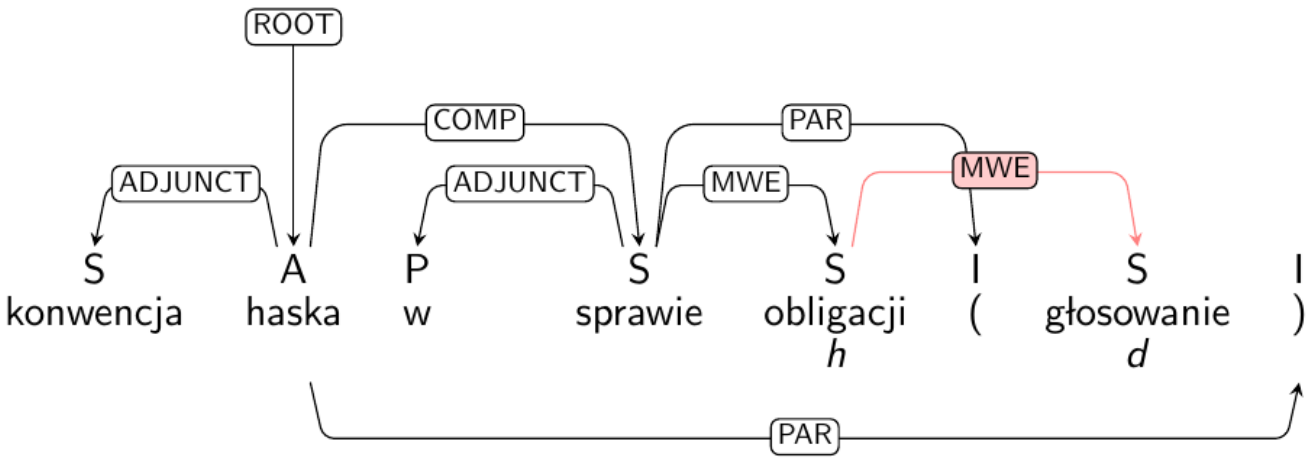


Non-projective Dependency Parse

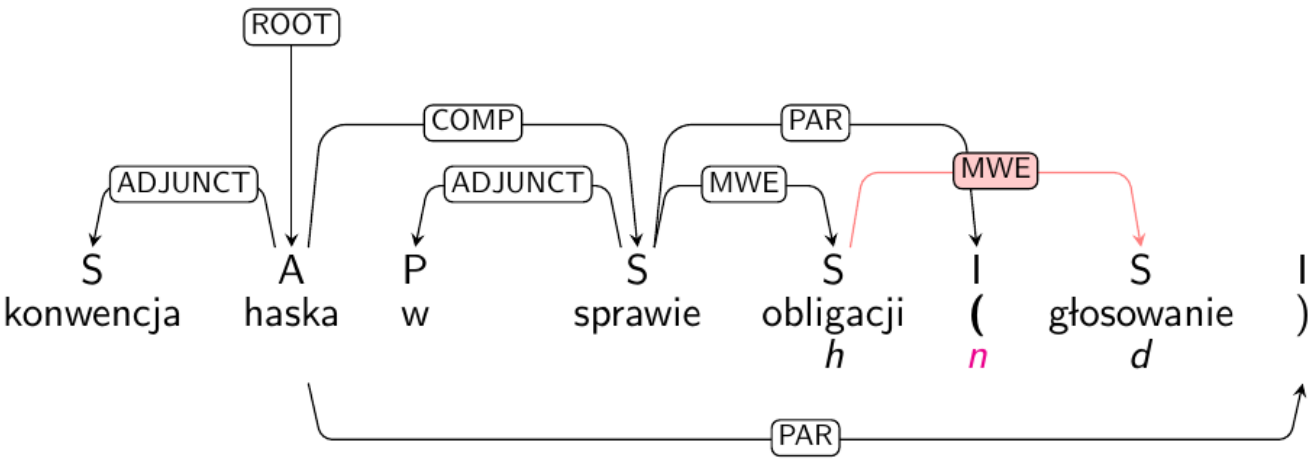
- $h \rightarrow d$ is *projective* iff h dominates all nodes in the linear span between h and d.
- Dependency parse is projective iff all its edges are projective.



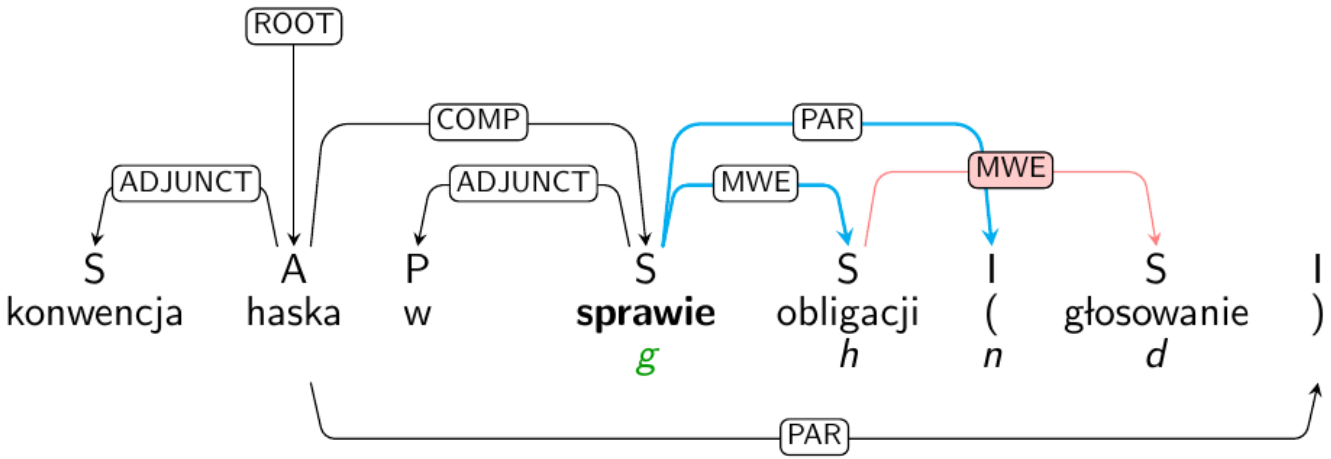
Lifting [Kahane et al., 1998]



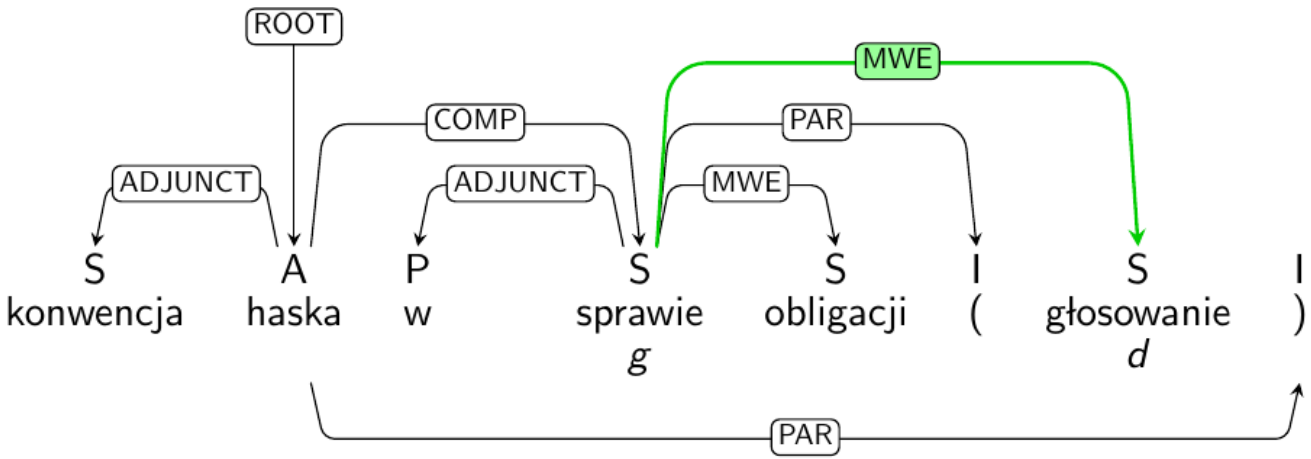
Lifting [Kahane et al., 1998]



Lifting [Kahane et al., 1998]

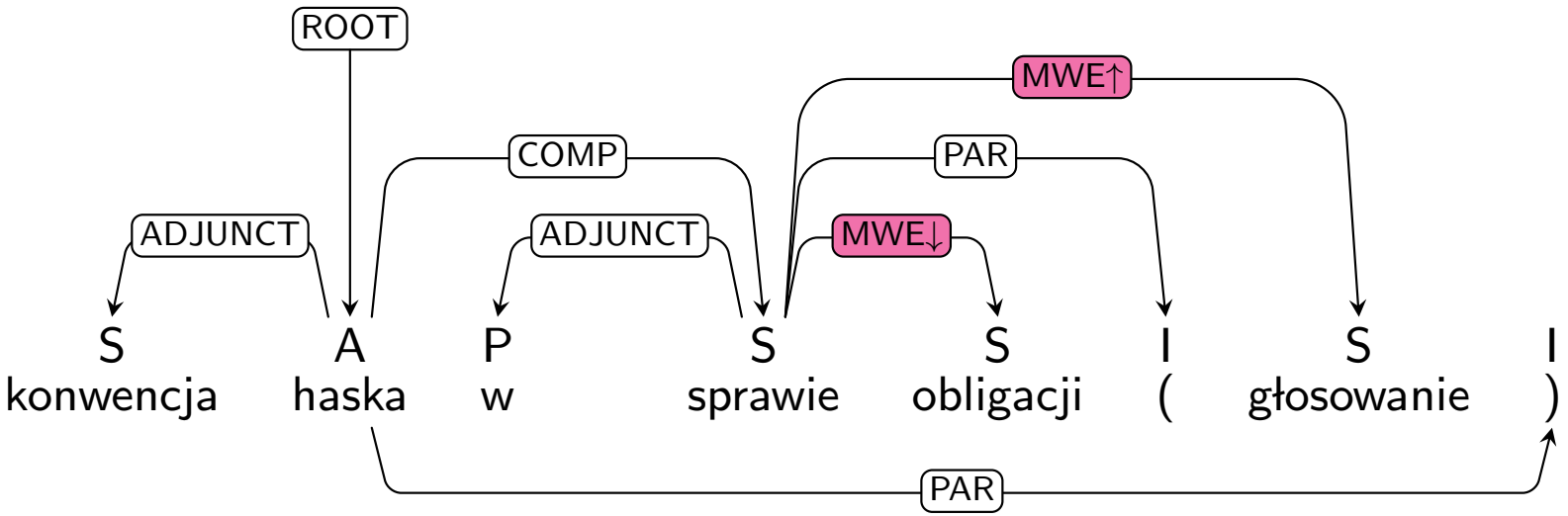


Lifting [Kahane et al., 1998]

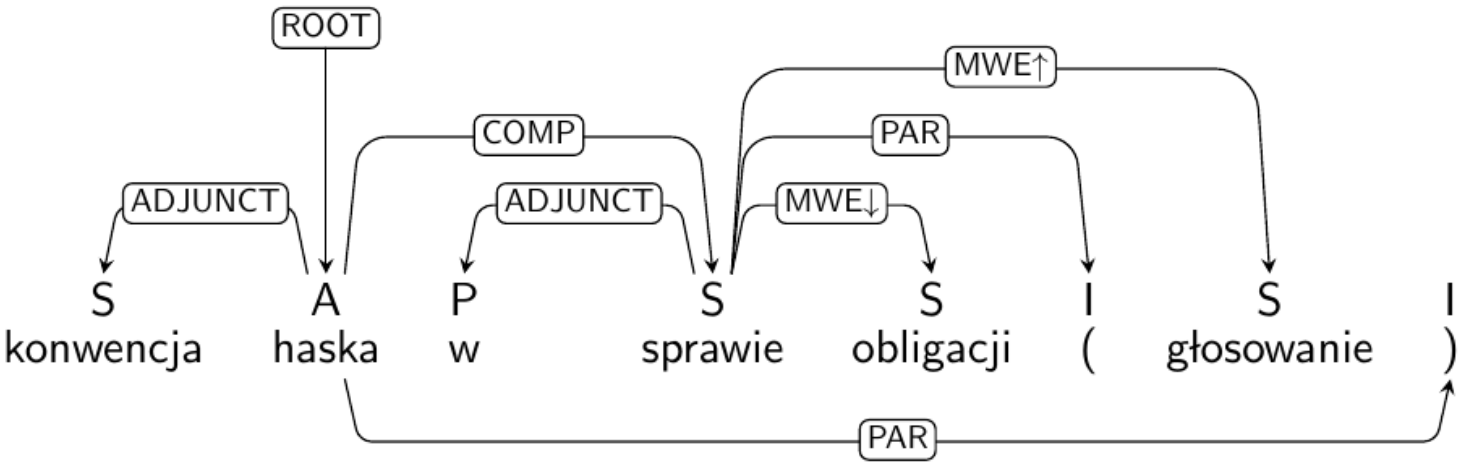


Lifting [Nivre and Nilsson, 2005]

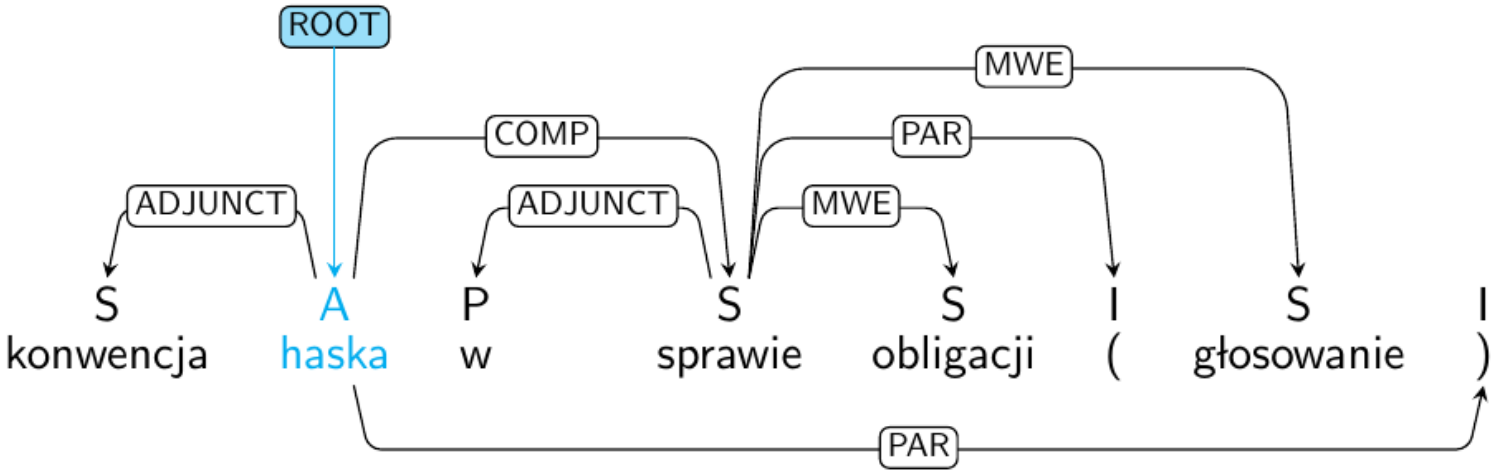
Refined the lifting process by performing the same operation but document the lifting in the labels ⇒ path

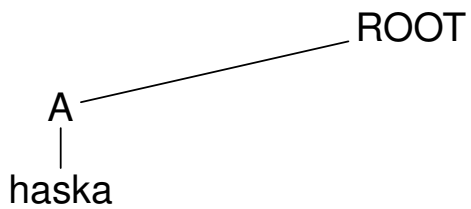


Conversion from dependency to constituency tree

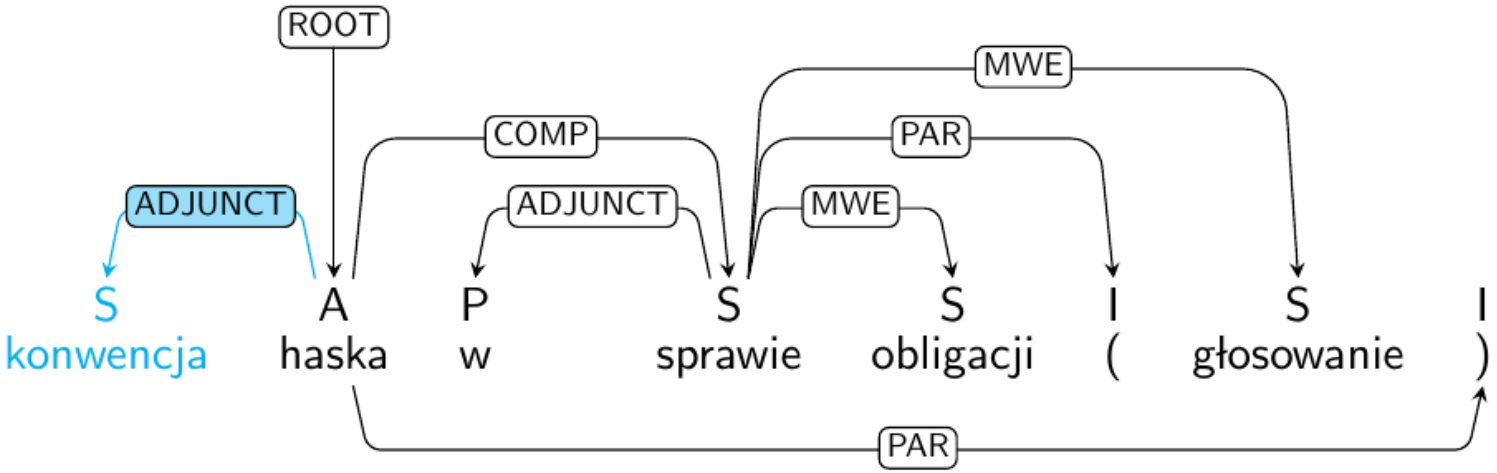


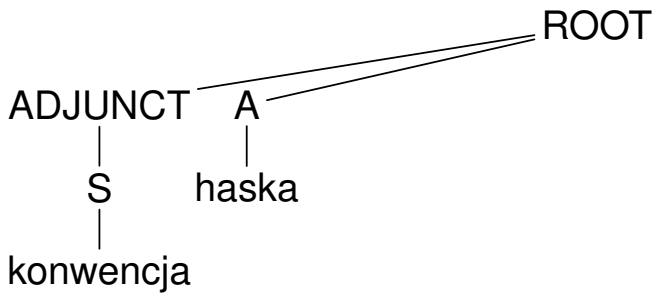
Conversion from dependency to constituency tree



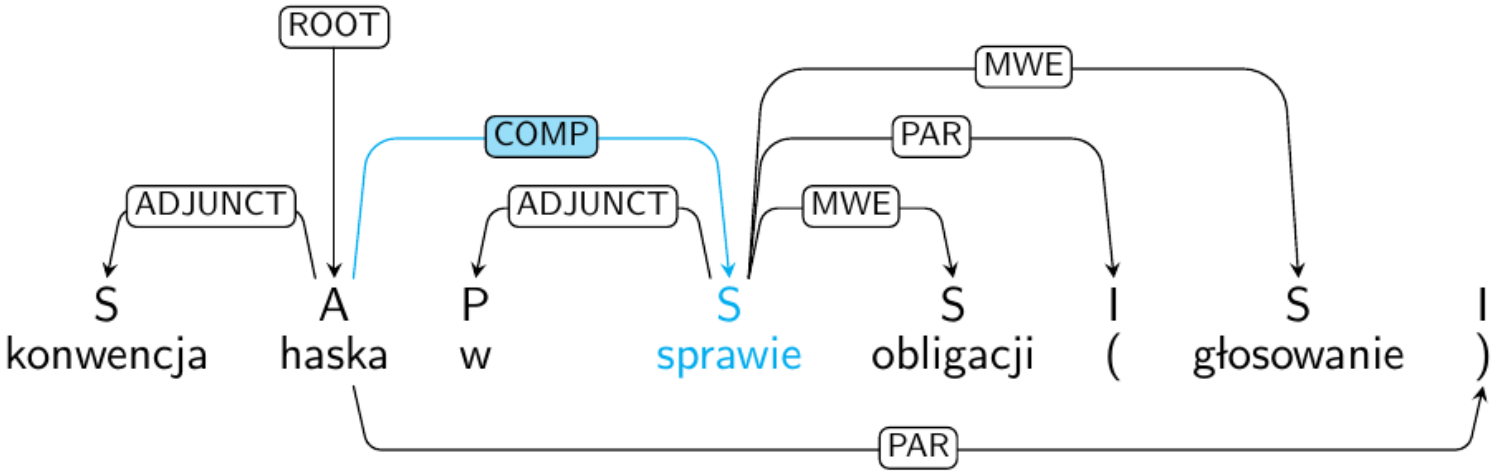


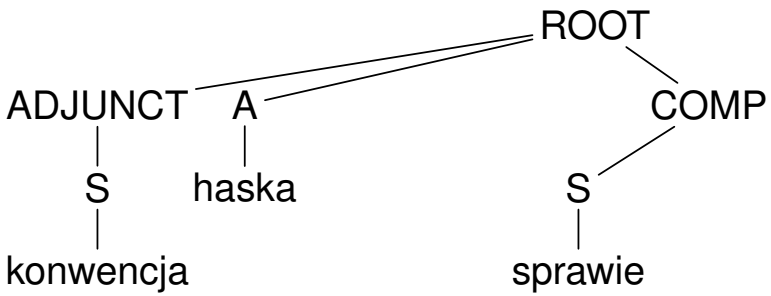
Conversion from dependency to constituency tree



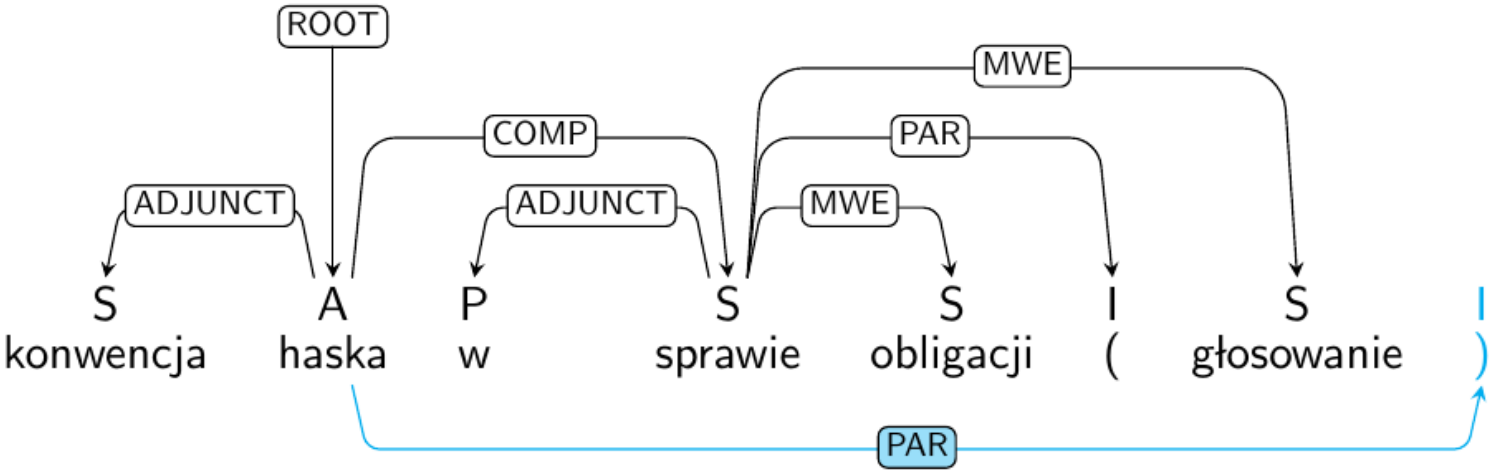


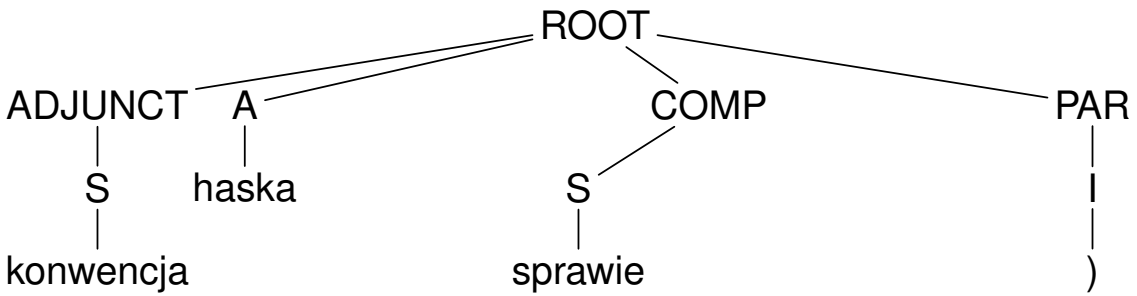
Conversion from dependency to constituency tree



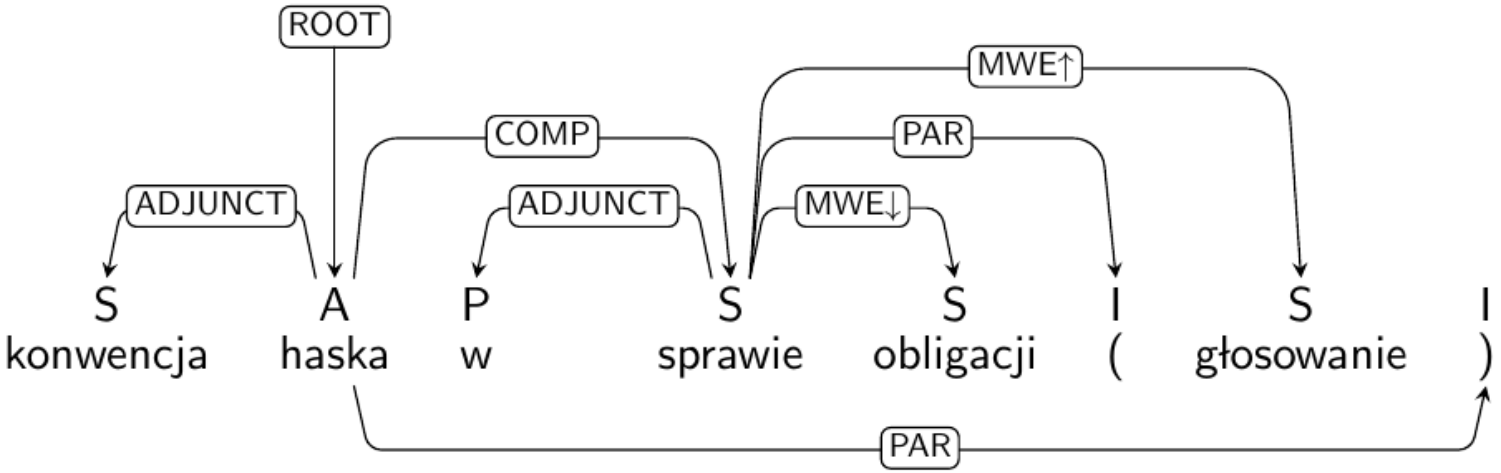


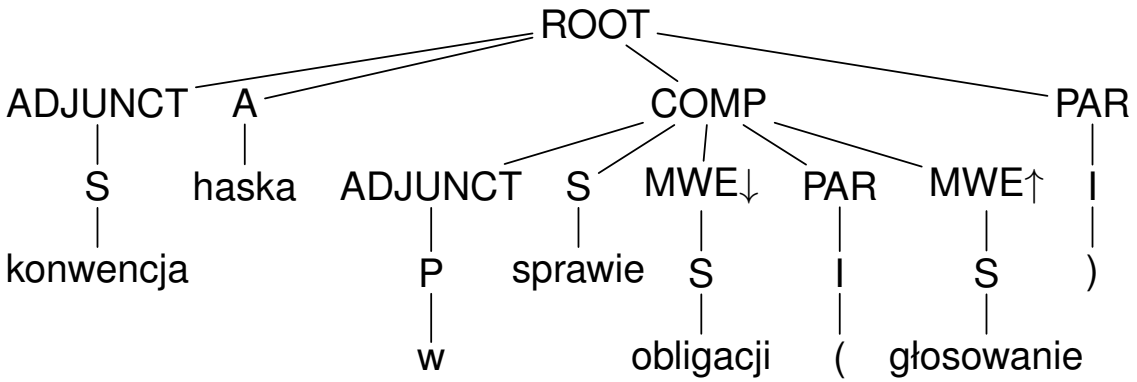
Conversion from dependency to constituency tree





Conversion from dependency to constituency tree



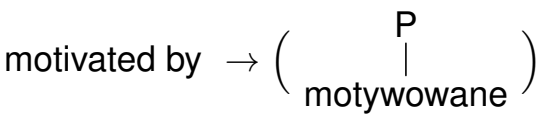


Preserves discontinuities!

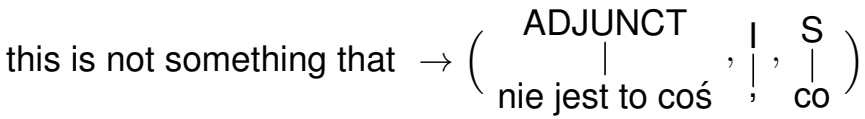
Outline

String-to-Tree Multi Bottom-up Tree Transducer

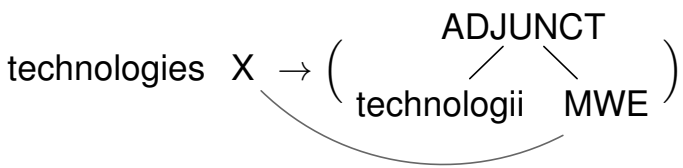
lexical continuous rule:



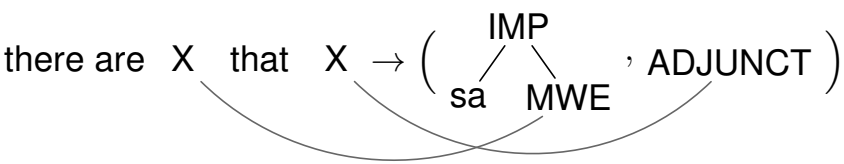
lexical discontinuous rule:

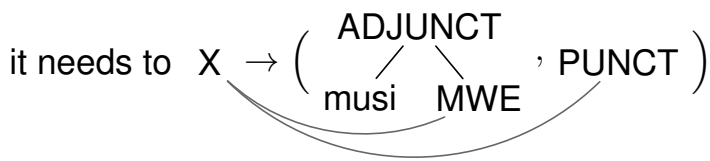


structural continuous rule:



structural discontinuous rules:





Translation Model

Standard log-linear model with the following 8 features:

- ...
- *gap penalty* 100^{1-c} (c is the number of target tree fragments)

We use the MBOT-Moses decoder [Braune et al. 2013]:

- standard MOSES syntax-based decoder
- extended to handle target side discontinuities

Outline

Setup

	English to Polish	**English to Russian**
training data	7th EuroParl corpus	WMT 2014
language model	5-gram SRILM	
tuning data	cut from EuroParl ($\approx$ 3k)	WMT 2014
test data	cut from EuroParl($\approx$ 3k)	WMT 2014

Training Pipeline

Target side:

- TreeTagger [Schmid 1996]
- MaltParser [Nivre et al. 2006, Sharoff & Nivre 2011]
- Path-Lifting
- Conversion into constituency tree

Parallel Data:

- tokenized and lowercased
- length-ratio filtered up to length 80
- word alignments by GIZA++ [Och & Ney 2003] with *grow-diag-final-and*

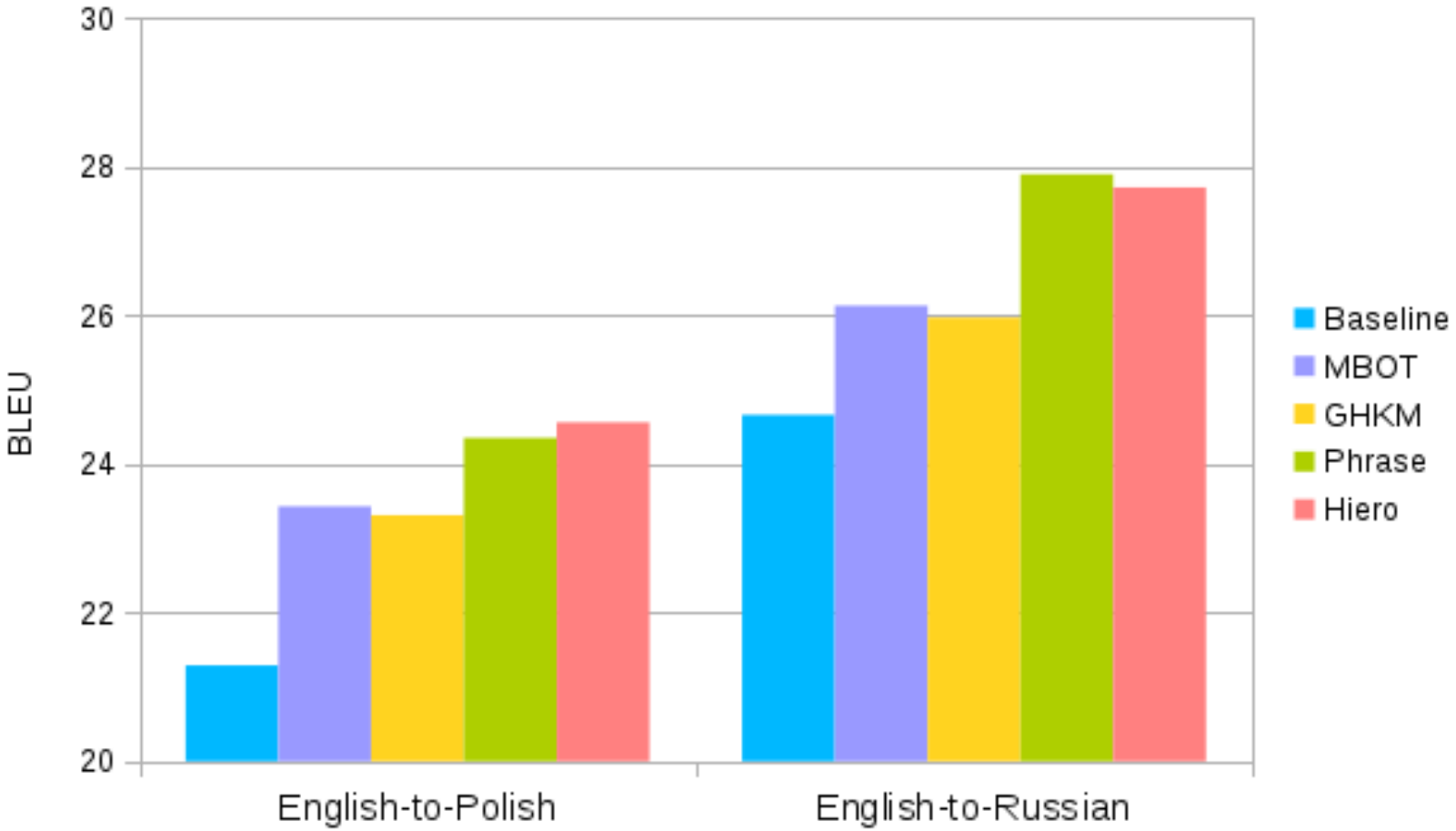
Tuning:

Minimum error rate training [Och 2003]

Experimental Results

Translation task	System	BLEU
English-to-Polish	Baseline	21.29
	MBOT	**23.43**
	GHKM	23.31
	Phrase-based	24.35
	Hiero	24.56
English-to-Russian	Baseline	24.66
	MBOT	**26.13**
	GHKM	25.97
	Phrase-based	27.90
	Hiero	27.72

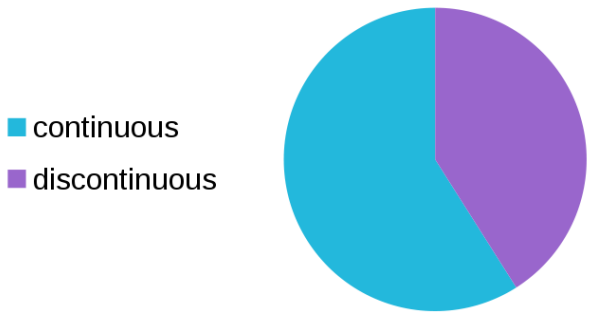
Losses across the systems



Analysis of rules used during decoding

English-to-Polish

All rules:



Structural rules:

English-to-Russian

All rules:

Structural rules:

Outline

Recap

- Translation into free word order languages
- Discontinuous constituents
- Dependency parsers producing non-projective parses:
  1. Projectivize by lifting technique documenting process
  2. Transform projective dependency trees into constituent-like trees
- String-to-tree local multi bottom-up tree transducers
- Discontinuous translation model

Conclusion

- MBOT avoids large quality drop between (hierarchical) phrase-based system and continuous string-to-tree one
- Discontinuous tree fragments yield significant improvements
- Overall performance similar to (hierarchical) phrase-based systems
- But, outscoring (hierarchical) phrase-based remains a challenge
- Can syntactic information actually help the translation quality in those translation tasks?

Thank you!

Questions?!?

Related Work

Xie et al., 2011:

- dependency-to-string model with head-dependent rules
- custom-made decoder

Li et al., 2014:

- transform dependency trees into (a kind of) constituency trees
- use the conventional syntax-based models of Moses

Sennrich et al., 2015:

- transform (non-projective) dependency trees into constituency trees
- using the syntactic functions provided by the parser
- string-to-tree GHKM model of Moses

References

Eisner: Learning Non-Isomorphic Tree Mappings for Machine Translation. ACL 2003.
Kahane et al.: Pseudo-Projectivity: A Polynomially Parsable Non-Projective Dependency Grammar. ACL 1998.
Li et al.: Transformation and Decomposition for Efficiently Implementing and Improving Dependency-to-String Model In Moses. SSST 2014.
Nivre and Nilsson: Pseudo-projective Dependency Parsing. ACL 2005.
Nivre et al.: MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. LREC 2006.
Schmid: Probabilistic Part-of-Speech Tagging Using Decision Trees. New Methods in Language Processing 1994.
Sennrich et al.: A tree does not make a well-formed sentence: Improving syntactic string-to-tree statistical machine translation with more linguistic knowledge. Computer Speech & Language 32, 2015.
Sharoff and Nivre: The proper place of men and machines in language technology Processing Russian without any linguistic knowledge. Dialogue 2011.
Sun et al.: A Non-Contigous Tree Sequence Alignment-based Model for Statistical Machine Translation. ACL 2009.
Wróblewska and Przepiórkowski: Induction of Dependency Structures Based on Weighted Projection. ICCCI 2012.
Xie et al.: A Novel Dependency-to-string Model for Statistical Machine Translation. EMNLP 2011.