(Re)ranking Meets Morphosyntax

Anders Björkelund, Özlem Çetinoğlu, Richárd Farkas,
Thomas Müller, Wolfgang Seeker

Institute for Natural Language Processing, Stuttgart
Szeged University, Szeged
Center for Information and Language Processing, Munich

Section 1

Preprocessing

Preprocessing

Attempt to get best possible POS and feature predictions.

Preprocessing

Attempt to get best possible POS and feature predictions.

- State-of-the-art machine learning

Preprocessing

Attempt to get best possible POS and feature predictions.

- State-of-the-art machine learning
- Additional lexical information provided by morphl. analyzers

Tagging for MRLs

- MRLs have big tagsets (100 - 2000 tags)
- Tag lattices are thus typically pruned using analyzers:
  - Requires analyzers with a tagset comparable to the treebank
  - Which were not available for all ST languages

MarMoT [Müller et al., 2013]

- Approximate CRF tagger

MarMoT [Müller et al., 2013]

- Approximate CRF tagger
- Uses a pruning strategy to deal with big tag sets

MarMoT [Müller et al., 2013]

- Approximate CRF tagger
- Uses a pruning strategy to deal with big tag sets
- Allows for fast second- and third-order tagging

MarMoT [Müller et al., 2013]

- Approximate CRF tagger
- Uses a pruning strategy to deal with big tag sets
- Allows for fast second- and third-order tagging
- Can use the output of analyzers as additional features

MarMoT [Müller et al., 2013]

- Approximate CRF tagger
- Uses a pruning strategy to deal with big tag sets
- Allows for fast second- and third-order tagging
- Can use the output of analyzers as additional features

Want to know more about MarMot?

Talk on Saturday: Efficient Higher-Order CRFs for Morphological Tagging

Morphological Analyzers

- We extracted dictionaries using freely available analyzers

Arabic	AraMorph	[Buckwalter, 2002]
Basque	Apertium	[Forcada et al., 2011]
French	IMS internal tool	[Zhou, 2007]
Hungarian	Magyarlanc	[Zsibrita et al., 2013]
Korean	HanNanum	[Park et al., 2010]
Polish	Morfeusz	[Woliński, 2006]
Swedish	Granska	[Domeij et al., 2000]

Morphological Analyzers

- We extracted dictionaries using freely available analyzers

Arabic	AraMorph	[Buckwalter, 2002]
Basque	Apertium	[Forcada et al., 2011]
French	IMS internal tool	[Zhou, 2007]
Hungarian	Magyarlanc	[Zsibrita et al., 2013]
Korean	HanNanum	[Park et al., 2010]
Polish	Morfeusz	[Woliński, 2006]
Swedish	Granska	[Domeij et al., 2000]

- The external lexical information improves predictions substantially

	Basque	French	German	Hebrew	Hungarian	Korean	Polish	Swedish
Plain	94.77	97.30	97.87	95.43	97.32	81.81	96.71	96.14
MA	95.29	97.50	98.07	96.89	98.24	86.78	97.88	97.13
Δ	0.52	0.20	0.20	1.46	0.92	4.97	1.17	0.99

Comparison with Provided Predictions

- Compare the mate Parsers' [Bohnet, 2010] LAS for provided predictions and predictions by MarMoT

Comparison with Provided Predictions

- Compare the mate Parsers' [Bohnet, 2010] LAS for provided predictions and predictions by MarMoT
- Good preprocessing gives substantial improvements for many languages

	Basque	French	German	Hebrew	Hungarian	Korean	Polish	Swedish
ST	83.50	84.49	90.85	75.89	82.84	82.39	85.81	77.16
MarMoT	84.43	84.84	91.46	79.37	84.41	85.76	86.30	77.05
Δ	0.93	0.35	0.61	3.48	1.57	3.37	0.49	-0.11

Section 2

Phrase-Structure Parsing

Phrase-Structure Parsing

A simple pipeline of parsing methods, known to work well.

Phrase-Structure Parsing

A simple pipeline of parsing methods, known to work well.

- PCFGs with Latent Annotations [Petrov et al., 2006]

Phrase-Structure Parsing

A simple pipeline of parsing methods, known to work well.

- PCFGs with Latent Annotations [Petrov et al., 2006]
- Product Grammars [Petrov, 2010]

Phrase-Structure Parsing

A simple pipeline of parsing methods, known to work well.

- PCFGs with Latent Annotations [Petrov et al., 2006]
- Product Grammars [Petrov, 2010]
- Reranking [Collins, 2000] and [Charniak and Johnson, 2005]

Grammars with Latent Annotations [Petrov et al., 2006]

- Start with a simple binary grammar

Grammars with Latent Annotations [Petrov et al., 2006]

- Start with a simple binary grammar
- Incrementally, split symbols and fit to the data using EM

Grammars with Latent Annotations [Petrov et al., 2006]

- Start with a simple binary grammar
- Incrementally, split symbols and fit to the data using EM
- Merge symbols that only yield small improvements in likelihood

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50

Grammars with Latent Annotations [Petrov et al., 2006]

- Start with a simple binary grammar
- Incrementally, split symbols and fit to the data using EM
- Merge symbols that only yield small improvements in likelihood
- Signature-based unknown word model seems to be to simple for productive languages

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50

Grammars with Latent Annotations [Petrov et al., 2006]

- Start with a simple binary grammar
- Incrementally, split symbols and fit to the data using EM
- Merge symbols that only yield small improvements in likelihood
- Signature-based unknown word model seems to be to simple for productive languages
- Results for Polish are suboptimal because of a unary chain limit of the Berkeley parser

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50

Replacing

- Replace rare words by the morphological tag assigned by MarMoT

Replacing

- Replace rare words by the morphological tag assigned by MarMoT
- Morphological tag contains most of the syntactic information (exceptions: e.g. PP-attachment)

Replacing

- Replace rare words by the morphological tag assigned by MarMoT
- Morphological tag contains most of the syntactic information (exceptions: e.g. PP-attachment)
- MarMoT takes more lexical features into account (e.g., suffixes and lexical context)

Replacing II

- Replacing words with frequency < 20

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Vocab.	3506	418	2096	3300	653	707	1462	219	381
Repl.	0.19	0.50	0.18	0.25	0.33	0.46	0.54	0.55	0.40

Replacing II

- Replacing words with frequency < 20
- Languages with biggest improvements are agglutinative

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Vocab.	3506	418	2096	3300	653	707	1462	219	381
Repl.	0.19	0.50	0.18	0.25	0.33	0.46	0.54	0.55	0.40
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50
Replaced	78.70	84.33	79.68	82.74	89.55	89.08	82.84	87.12	75.52
Δ	0.46	15.16	-0.06	1.00	1.72	5.18	11.87	3.01	1.02

Product Grammars [Petrov, 2010]

- Train 8 PCFG-LAs with different seed values

Product Grammars [Petrov, 2010]

- Train 8 PCFG-LAs with different seed values
- Extract k-best lists and score them with the product model (Tree-Level inference)

Product Grammars [Petrov, 2010]

- Train 8 PCFG-LAs with different seed values
- Extract k-best lists and score them with the product model (Tree-Level inference)

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50
Replaced	78.70	84.33	79.68	82.74	89.55	89.08	82.84	87.12	75.52
Product	80.30	86.21	81.42	84.56	90.49	89.80	84.15	88.32	79.25
Δ	1.60	1.88	1.74	1.82	0.94	0.72	1.31	1.20	3.73

Reranking

- Maximum entropy reranker of [Charniak and Johnson, 2005]

Reranking

- Maximum entropy reranker of [Charniak and Johnson, 2005]
- Languages independent features of [Collins, 2000] and [Charniak and Johnson, 2005]

Reranking

- Maximum entropy reranker of [Charniak and Johnson, 2005]
- Languages independent features of [Collins, 2000] and [Charniak and Johnson, 2005]
- No tuning of the feature set

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50
Replaced	78.70	84.33	79.68	82.74	89.55	89.08	82.84	87.12	75.52
Product	80.30	86.21	81.42	84.56	90.49	89.80	84.15	88.32	79.25
Reranked	81.24	87.35	82.49	85.01	90.49	91.07	84.63	88.40	79.53
Δ	0.94	1.14	1.07	0.45	0.00	1.27	0.48	0.08	0.28

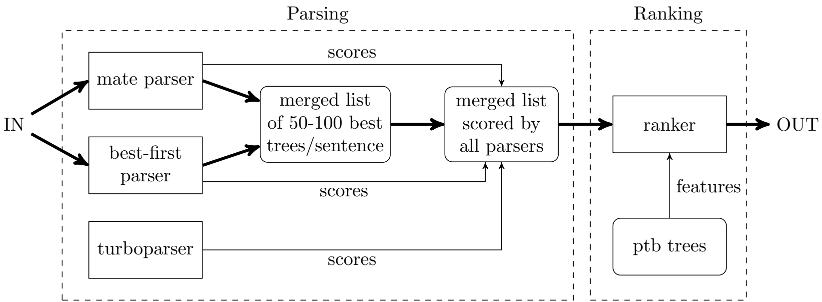
Conclusion

	Arabic	Basque	French	German	Hebrew	Hung.	Korean	Polish	Swedish
Berkeley	78.24	69.17	79.74	81.74	87.83	83.90	70.97	84.11	74.50
Reranked	81.24	87.35	82.49	85.01	90.49	91.07	84.63	88.40	79.53
Δ	3.00	18.18	2.75	3.27	2.66	7.17	13.66	4.29	5.03

Section 3

Dependency Parsing

Architecture of the Ranking System



- Three different parsers: mate [Bohnet, 2010], Best-First (in-house EasyFirst [Goldberg and Elhadad, 2010]), Turboparser [Martins et al., 2010]
- Merged list is scored by all three parsers [Zhang et al., 2009]
- Scored list is ranked to find the optimal parse

NBest Generation and Scoring

- We modified mate's non-projective approximation algorithm [McDonald and Pereira, 2006] to produce n-best lists
- Best-first parser uses beam search, which naturally produces n-best lists

- Rescoring with all three parsers is important since the scores are the most important features for the ranker
- We modified all parsers to extract scores from a given tree

Performance of All Three Parsers (Dev Set)

	Arabic	Basque	French	German	Hebrew	Hungarian	Korean	Polish	Swedish
Baseline results for individual parsers									
mate'		83.50	84.49	90.85	75.89	82.84	82.39	85.81	77.16
mate	85.42	84.43	84.84	91.46	79.37	84.41	85.76	86.30	77.05
bf	85.32	75.90	83.92	91.10	79.57	75.94	83.97	83.75	75.36
turbo	85.35	83.84	84.57	91.54	78.95	82.80	86.23	85.55	76.15

- mate' does not use MarMoT preprocessing,
  all others use MarMoT
- mate best, then Turboparser, then best-first
- submitted mate as baseline run,
  came out as **overall second best** (off-the-shelf + MarMoT)

Oracle Scores

	Arabic	Basque	French	German	Hebrew	Hungarian	Korean	Polish	Swedish
	Oracle scores for n-best lists								
mate	88.74	89.85	87.81	95.84	83.03	88.19	92.96	91.67	81.66
bf	89.46	86.46	88.68	96.60	85.67	81.79	92.94	93.74	82.46
merged	90.71	91.91	90.43	97.44	87.18	88.76	94.65	95.29	84.96

- Oracle scores improve
  when combining n-best lists of mate and best-first
- Best-first parser often shows slightly better oracle scores
- Parsers provide different sets of n-best lists

Ranking

- Same ranking model as for constituencies
- Trained with 5-fold jackknifing on training data
- Feature sets for each language were optimized manually via cross-validation on training data

Ranking – Features that Helped

- **Scores** of each parser – most important features
- A binary feature that marks the **highest score** in the list
- Products and normalized **products of scores**
- Projectivity features [Hall et al., 2007]
  (number of non-projective edges, ill-nestedness)
- **Paths in the constituency tree** between head and dependent
- **Case agreement** marks whether head and dependent
  have the same case value
- **Function label uniqueness** marks whether a label occurs
  more than once that never does in the training data

Ranking – Features that Didn't Work

- Lexical and PoS-based features over edges
- "Subcat" features (chains of PoS of daughters of a node)
- For German, we tested features on agreement violations or verb complexes

Ranking Results – Dev Sets

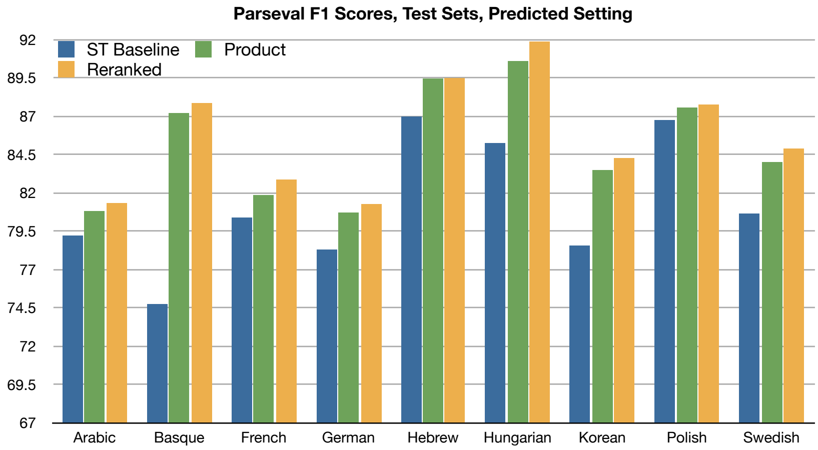
	Arabic	Basque	French	German	Hebrew	Hungarian	Korean	Polish	Swedish
Baseline	85.42	84.43	84.84	91.46	79.37	84.41	85.76	86.30	77.05
Ranked	**86.74**	**85.61**	**85.96**	**92.68**	**81.02**	**84.77**	**87.12**	**87.69**	**78.57**
Δ	1.32	1.18	1.12	1.22	1.65	0.36	1.36	1.39	1.52
Oracle	90.71	91.91	90.43	97.44	87.18	88.76	94.65	95.29	84.96

- Ranking improves results for all languages
- Improvements are between 1.1% (French) and 1.6% (Hebrew)
- Except Hungarian, but it seems to be the dev set (test set results improve more)

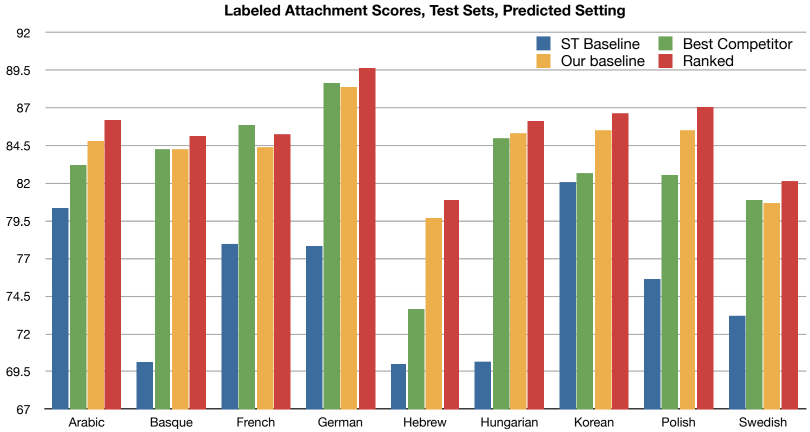
Section 4

Conclusion

Test Set Results – Constituency Parsing



Test Set Results – Dependency Parsing



Test Set Results – Predicted Tokenization

	Arabic	Hebrew
Best Competitor	90.75	88.33
Dep. Baseline	91.13	89.27
Dep. Ranked	91.74	89.47
Constituency	92.06	89.30

- Unlabeled TedEval scores
- Used tokenization provided by the organizers, preprocessing by MarMoT
- No clear advantage for either dependencies or constituencies

Conclusions – What We Learned

- Good preprocessing gives good parsing scores
- Standard techniques for constituency parsing work well on many languages (not just English)
- Replacing rare words with morphology helps, especially for agglutinative languages
- Ranking mate parser alone is very difficult, but having scores from different parsers makes it work well
- Off-the-shelf MarMoT and Off-the-shelf mate give good results already

Thank you for your attention!
Questions?

Download MarMoT:
`http://code.google.com/p/cistern/wiki/marmot`
Download mate tools:
`http://code.google.com/p/mate-tools`

Bohnet, B. (2010).
Top Accuracy and Fast Dependency Parsing is not a Contradiction.
In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China. Coling 2010 Organizing Committee.

Buckwalter, T. (2002).
Buckwalter Arabic Morphological Analyzer Version 1.0.
Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Catalog No.: LDC2002L49.

Charniak, E. and Johnson, M. (2005).
Coarse-to-fine n-best parsing and MaxEnt discriminative reranking.
In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 173–180.

Collins, M. (2000).
Discriminative Reranking for Natural Language Parsing.

In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 175–182.

Domeij, R., Knutsson, O., Carlberger, J., and Kann, V. (2000).
Granska-an efficient hybrid system for Swedish grammar checking.
In *In Proceedings of the 12th Nordic Conference in Computational Linguistics*.

Forcada, M. L., Ginestí-Rosell, M., Nordfalk, J., O'Regan, J., Ortiz-Rojas, S., Pérez-Ortiz, J. A., Sánchez-Martínez, F., Ramírez-Sánchez, G., and Tyers, F. M. (2011).
Apertium: A free/open-source platform for rule-based machine translation.
Machine Translation.

Goldberg, Y. and Elhadad, M. (2010).
An Efficient Algorithm for Easy-First Non-Directional Dependency Parsing.
In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association*

for Computational Linguistics, pages 742–750, Los Angeles, California. Association for Computational Linguistics.

Hall, K., Havelka, J., and Smith, D. A. (2007).
Log-Linear Models of Non-Projective Trees, k-best MST Parsing and Tree-Ranking.
In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 962–966, Prague, Czech Republic. Association for Computational Linguistics.

Martins, A., Smith, N., Xing, E., Aguiar, P., and Figueiredo, M. (2010).
Turbo Parsers: Dependency Parsing by Approximate Variational Inference.
In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 34–44, Cambridge, MA. Association for Computational Linguistics.

McDonald, R. and Pereira, F. (2006).
Online learning of approximate dependency parsing algorithms.

In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 81–88, Trento, Italy. Association for Computational Linguistics.

Müller, T., Schmid, H., and Schütze, H. (2013).
Efficient Higher-Order CRFs for Morphological Tagging.
In *In Proceedings of EMNLP*.

Park, S., Choi, D., Kim, E.-k., and Choi, K. (2010).
A plug-in component-based Korean morphological analyzer.
In *Proceedings of HCLT2010*.

Petrov, S. (2010).
Products of Random Latent Variable Grammars.
In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, Los Angeles, California. Association for Computational Linguistics.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006).
Learning accurate, compact, and interpretable tree annotation.

In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 433–440. Association for Computational Linguistics.

Woliński, M. (2006).
Morfeusz - A practical tool for the morphological analysis of Polish.
In *Intelligent information processing and web mining*, pages 511–520. Springer.

Zhang, H., Zhang, M., Tan, C. L., and Li, H. (2009).
K-Best Combination of Syntactic Parsers.
In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1552–1560, Singapore. Association for Computational Linguistics.

Zhou, Z. (2007).
Entwicklung einer französischen Finite-State-Morphologie.
Diplomarbeit, Institute for Natural Language Processing, University of Stuttgart.

Zsibrita, J., Vincze, V., and Farkas, R. (2013).
Magyarlanc 2.0: Szintaktikai elemzés és felgyorsított szófaji egyértelműsítés.
In *IX. Magyar Számítógépes Nyelvészeti Konferencia*.