Hungarian Data-Driven Syntactic Parsing in 2014

Zsolt Szántó[1], Richárd Farkas[1], Anders Björkelund[2], Özlem Çetinoğlu[2], Agnieszka Faleńska[2,3], Thomas Müller[4], Wolfgang Seeker[2]

[1] Szegedi Tudományegyetem, TTIK, Informatikai Tanszékcsoport, Szeged Árpád tér 2.
[2] Institute for Natural Language Processing, University of Stuttgart, Germany,
[3] Institute of Computer Science, University of Wrocław, Poland,
[4] Center for Information and Language Processing, University of Munich, Germany,
e-mail:{szantozs,rfarkas}@inf.u-szeged.hu,
{anders,ozlem,muellets,seeker}@ims.uni-stuttgart.de,
agnieszka.falenska@cs.uni.wroc.pl

1 Introduction

In prior work on data-driven syntactic parsing of Hungarian [1–4], it has been shown that parsers developed for English [5] struggle with the complexity introduced by morphologically rich languages (MRL). The Statistical Parsing of Morphologically Rich Languages (SPMRL) workshop series aims to foster the development of parsing techniques dedicated to morphologically rich languages.

In this year, the workshop hosted the SPMRL 2014 Shared Task, which was the second shared task on parsing morphologically rich languages. The challenge involves parsing both dependency and constituency representations of nine languages.

We present the contribution of the team IMS-Wrocław-Szeged-CIS, which was a joint effort of four universities. Our team achieved the best scores on all languages in the dependency track and on all languages (except for Polish) in the constituency track. In this paper, we introduce these dependency and constituency parsing systems and give extra analysis and discussions on the Hungarian treebanks.

2 The SPMRL Shared Tasks

In 2013, the organizers made the first shared task on parsing morphologically rich languages, which contains challenges in the two most commonly used syntactic frameworks (dependency and constituency) on nine morphologically rich languages (Arabic, Basque, French, German, Hebrew, Hungarian, Korean, Polish, and Swedish).

This year's shared task was an extension of the first challenge, where every annotated corpus from last year was extended with a large unlabeled data set

[3]. The system established by our team is also an extended version of the IMS-SZEGED-CIS [6] team's system, which managed to get the highest scores in every category last year.

The newspaper sub-corpora of the Szeged Treebank [7] and the Szeged Dependency Treebank [8] were employed as the Hungarian treebanks of the shared task as the organizers collected treebanks only from the newspaper domain for each language. The unlabeled data also contains newspaper articles, and came from the Hungarian National Corpus [9], which contains 1747239 sentences. We provided automatic POS-tagging and dependency parsing using magyarlanc [10] for the unlabeled data to the shared task organizers.

3 Preprocessing

The dependency parsers require POS/morphological tagging. To predict the data we use the language independent tool MarMoT[5] [11]. In Hungarian, we analyzed the word forms with the language-specific morphological analyzer of magyarlanc [10] and we use these information as features in MarMoT. We achieved 98.49 POS and 97.45 full morphological description accuracy on the development set.

4 Constituency Parsing

Our constituency parsing architecture consists of two steps. First, we deal with lexical sparsity and exploit product grammars. Second, we apply a reranker where we investigate new feature templates. In the following sections we focus on the methods to alleviate lexical sparsity and features we use in the reranker.

4.1 Lexical Sparsity

The out-of-vocabulary issue is a crucial problem in morphologically rich languages, as a word can have many different forms depending on its syntactic and semantic context. Last year, we replaced rare words by their morphological analysis produced by MarMoT [6] (similar to the strategy of backing off rare words to their POS tag in the CCG literature [12]). We call this strategy *Replace*.

This year, we experimented with an alternative approach, which exploits the available unlabeled data [2]. We followed [13] and enhanced a lexicon model trained on the treebank training data with frequency information about the possible morphological analyses of tokens (*ExtendLex*).

We note that the two strategies lead to fundamentally different representations. In the *Replace* version the output parses contain morphological descriptions instead of tokens and only main POS tags are used as preterminal labels while in the *ExtendLex* approach the tokens at the terminal level remain unchanged morphological analyses are employed as preterminal labels.

[5] `https://code.google.com/p/cistern/`

Table 1 shows the results achieved by the two strategies on the development sets. As our baselines we use the *Berkeley parser* [5] by removing morphological annotations and leaving only POS tags in preterminals (**mainPOS**), and by using full morphological descriptions (**fullMorph**). We adopt the products of respective grammars [14] as well (*ExtendLex Product* and *Replace Product*).

Table 1. PARSEVAL scores on the development set for the predicted setting.

	Hungarian
Berkeley Parser $_{mainPOS}$	83.84
Berkeley Parser $_{fullMorph}$	87.18
ExtendLex	88.99
Replace	89.59
ExtendLex Product	90.43
Replace Product	90.72

4.2 Reranker Features

The second step of our constituency pipeline is discriminative reranking. We conduct ranking experiments on the 50-best outputs of the product grammars. Like last year, we use a slightly modified version of the Mallet toolkit [15], where the reranker is trained for the maximum entropy objective function of [16] and uses the standard feature set from [16] and [17] (**dflt**). This year we investigated new feature templates exploiting automatic dependency parses of the sentence in question [18]; Brown clusters [19]; and atomic morphological feature values [2]. Our purpose here is to investigate the efficiency of these feature templates in Hungarian. For these studies we used the product grammar configuration.

The results of these feature template are shown in Table 2.

We create features from the full morphological description by using each morphological feature separately (**morph**). This approach allows us to combine a word with its morphological features (kutya-N-Cas=n). New features are established using constituency labels and morphological features of the word's head, as well as morphological features of the head and its dependent. As we only use the main POS tags in the case of the *Replace* method, these new features could only be applicable to *ExtendLex*. These new features yield a slight improvement over the *dflt* feature set (0.22 percentage points).

We also created features based on automatic dependency parsing (**dep**). These features are made from heads of constituents and their dependency relations. We used features describing relations between the same head-dependent pairs in both the constituency and dependency parses. The frequency of these relations was also used. These features are especially interesting for Hungarian because we have two manually annotated corpora in both representations as opposed to the other SPMRL languages. The results reveal that in spite of the

annotation differences, this feature template has a considerable added value. For *Replace*, the improvement is moderate, while for *ExtendLex* the result increases from 91.09 to 91.89.

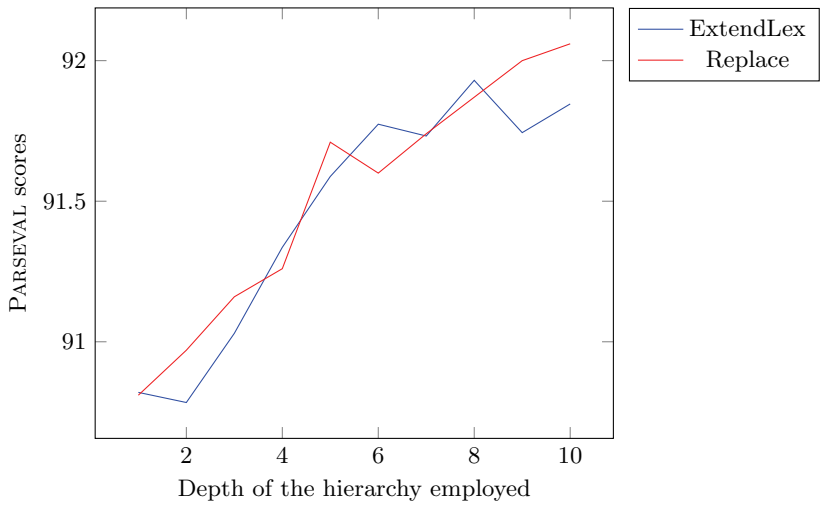

Fig. 1. Result of Brown cluster based feature templates.

We defined Brown cluster-based features (**Brown**). Brown clustering is a context-based hierarchical clustering over words. Utilizing these clusters we duplicate every other feature containing words and we replace words with their Brown clusterID (to a pre-set depth). The Brown cluster features improve our results in both representations. In the case of Brown clusters we investigated the effect of different levels of the hierarchical tree. The results achieved with *ExtendLex* are depicted on Figure 1.

Table 2 shows the final results of reranker on the development set. In the $ExtendLexReranked_{dflt+morph+Brown+dep}$ configuration we used the five level deep Brown clusters, because there was not enough time to calibrate this parameter. Reranking with default features improves the scores over product grammars both for *ExtendLex* and *Replace*. In the case of both representations the combination of feature templates slightly increases our scores.

5 Dependency Parsing

Like last year our dependency parsing system is based on two main steps. The first step is the parsing which creates the list of potential trees for each sentence and similar to constituency, the final step is reranking, which selects one of the possible trees. The full system is shown in Figure 2.

Table 2. PARSEVAL scores of the reranker on the development set for the predicted setting.

	Hungarian
ExtendLex Reranked $_{dflt}$	91.06
ExtendLex Reranked $_{dflt+morph}$	91.27
ExtendLex Reranked $_{dflt+dep}$	91.88
ExtendLex Reranked $_{dflt+Brown}$	91.93
ExtendLex Reranked $_{dflt+morph+Brown+dep}$	92.05
Replace Reranked $_{dflt}$	91.09
Replace Reranked $_{dflt+dep}$	91.89
Replace Reranked $_{dflt+brown}$	92.06
Replace Reranked $_{dflt+dep+brown}$	92.40

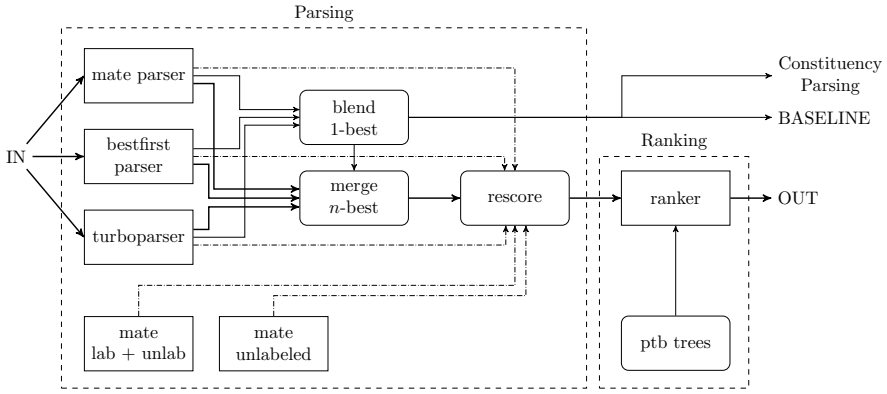

Fig. 2. The architecture of the 2014 dependency parsing system.

The parsing step is based on three different dependency parsers and a blender [20] which combines the results of the parsers. As baseline parsers with use the mate parser[6] [21], TurboParser[7] [22] and an internal implementation of the Easy-First parser [23].

In this year we used supertags which encode more syntactic information than standard POS tags. We used supertags following Ouchi et al. [24]. We made features from supertags for the mate parser and the TurboParser.

To make use of the unlabeled data we trained two self-trained models [25, 26], which are based on the mate parser. The first model was trained on unlabeled data only (**ulbl**) and in the second model we used also labeled and unlabeled data (**lbl+ulbl**).

[6] `https://code.google.com/p/mate-tools`
[7] `http://www.ark.cs.cmu.edu/TurboParser`

Table 3. UAS/LAS on Hungarian development set.

	UAS	LAS
mate	88.12	84.47
bestfirst	83.30	75.52
turbo	87.44	83.39
blend	88.09	84.24
mate_{ulbl}	86.17	82.26
$\text{mate}_{lbl+ulbl}$	88.07	84.38

Table 3 shows the result of the parsers, the last two lines show the result of the self-trained models. In Hungarian the mate parser got slightly better scores than the blending system while at other languages the blender gets great improvement compared to the standalone parsers (in Swedish more than 1%). The reason for this is the relatively bad performance of the Easy-First parser on Hungarian – in contrast to other languages.

The last step is the reranking, which chooses the best tree for each sentence from the output of the parsing step. In this step we optimized the feature sets for each language individually.

Table 4. UAS/LAS of the ranker on Hungarian development set for the predicted setting. Baseline denotes the blended trees.

	UAS	LAS
Baseline	88.09	84.24
Ranked_{dflt}	88.12	84.34
$\text{Ranked}_{no\text{-}ulbl}$	88.67	84.99
Ranked_{opt}	88.72	85.08
Oracle	91.91	8.37

Table 4 contains the results of the reranking system with different feature sets: default feature set ($Ranked_{dflt}$), optimized feature set ($Ranked_{opt}$), and optimized feature set but without features that are based on unlabeled data ($Ranked_{no\text{-}ulbl}$). The feature set optimization yields improvements, while the usage of unlabeled data only leads to minor improvements

To better understand what is special about Hungarian, we statistically analysed the output of the dependency parsing system on the development set.

From Table 5, it is striking that the labels which describe virtual nodes (containing VAN or ELL) get very low F-measures. These low scores might have two reasons, on the one hand, these relationships are relatively rare, so the parser cannot learn enough about them. On the other hand, these elements are not present in the surface structure, but they are present syntactically.

The accuracy of the parser is also poor on the FROM, TO, LOCY, TTO labels. These relations are also not too frequent and these labels contain not

Table 5. Precision, recall and F-measure of dependency relations.

Label	Recall	Prec.	F	Label	Recall	Prec.	F
APPEND	64.23	79.90	71.21	NE	92.54	90.07	91.29
ATT	93.32	94.07	93.69	NEG	94.88	93.41	94.14
ATT-VAN-CONJ	31.25	66.67	42.55	NUM	98.38	98.38	98.38
ATT-VAN-MODE	24.14	41.18	30.44	OBJ	97.64	96.13	96.88
ATT-VAN-OBL	25.93	63.64	36.85	OBL	94.91	92.07	93.47
ATT-VAN-PRED	51.35	62.30	56.30	PRED	62.61	80.90	70.59
ATT-VAN-PUNCT	37.29	61.11	46.32	PREVERB	98.44	97.83	98.13
ATT-VAN-SUBJ	44.62	53.70	48.74	PUNCT	99.03	96.24	97.62
AUX	100.00	100.00	100.00	QUE	93.33	87.50	90.32
CONJ	93.64	93.99	93.81	ROOT	85.00	88.24	86.59
COORD	75.17	79.75	77.39	ROOT-ELL-PUNCT	16.67	50.00	25.00
COORD-ELL-OBL	0.00	NaN	NaN	ROOT-VAN-ATT	16.13	35.71	22.22
COORD-ELL-PUNCT	21.05	50.00	29.63	ROOT-VAN-CONJ	76.85	74.11	75.46
COORD-VAN-CONJ	33.33	33.33	33.33	ROOT-VAN-COORD	36.36	26.67	30.77
COORD-VAN-MODE	18.75	23.08	20.69	ROOT-VAN-MODE	42.86	48.00	45.28
COORD-VAN-OBL	37.50	50.00	42.86	ROOT-VAN-NEG	60.00	46.15	52.17
COORD-VAN-PRED	46.15	48.65	47.37	ROOT-VAN-OBL	43.75	46.67	45.16
COORD-VAN-PUNCT	31.58	42.86	36.37	ROOT-VAN-PRED	69.91	63.20	66.39
COORD-VAN-SUBJ	59.46	57.89	58.66	ROOT-VAN-PUNCT	71.43	77.67	74.42
DAT	81.82	83.15	82.48	ROOT-VAN-SUBJ	60.44	56.70	58.51
DET	99.53	97.99	98.75	SUBJ	91.72	88.16	89.90
FROM	47.83	68.75	56.41	TFROM	72.73	80.00	76.19
INF	96.71	94.50	95.59	TLOCY	91.10	80.20	85.30
LOCY	50.62	73.21	59.85	TO	51.11	79.31	62.16
MODE	85.97	84.10	85.02	TTO	50.00	44.83	47.27

only syntactic but semantic information (e.g. they denote temporal or spatial dimensions) as well. Many of the phrases that get these labels are ambiguous between marking time or space, moreover, the tridirectionality in the Hungarian adverbial system may also lead to ambiguity, which makes it difficult for the parser to select the appropriate label.

Among the frequent labels ($freq > 1000$), the worst results were seen at *COORD* because coordination is a problematic phenomenon for dependency grammars in general.

Another interesting problem shows up on the POS tag level. Hungarian nouns in dative and genitive case have the same surface form, which makes POS tagging of these words difficult. The dative case usually marks an indirect object of verb, while the genitive case marks a possessive relation, and these syntactic roles are coded in different labels. The tokens with genitive case achieved an LAS of 77.08, with dative case an LAS of 78.21 while an LAS of 86,00 in case of all nouns. This example reveals a direct error propagation from the POS tagger to the dependency parser.

6 Summary

In this paper, we introduced the current state of the Hungarian data-driven syntactic parsing in dependency and constituency representations as well. We introduced the systems of the team IMS-Wrocław-Szeged-CIS, which achieved the highest scores in the SPMRL 2014 Shared Task. We also presented results on novel approaches for handling lexical sparsity in constituency parsers and we reported the added value of features in a constituency reranking framework. At the dependency parsing side, we presented a short error analysis in dependency results and highlighted Hungarian-specific challenges.

Acknowledgements

This work was supported in part by the European Union and the European Social Fund through the project FuturICT.hu (grant no.: TÁMOP-4.2.2.C-11/1/KONV-2012-0013).

References

1. Farkas, R., Vincze, V., Schmid, H.: Dependency Parsing of Hungarian: Baseline Results and Challenges. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. (2012) 55–65
2. Szántó, Zs., Farkas, R.: Special techniques for constituent parsing of morphologically rich languages. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden (2014) 135–144
3. Björkelund, A., Özlem Çetinoğlu, Faleńska, A., Farkas, R., Müller, T., Seeker, W., Szántó, Zs.: The IMS-Wrocław-Szeged-CIS entry at the SPMRL 2014 Shared Task: Reranking and Morphosyntax meet Unlabeled Data. In: Notes of the SPMRL 2014 Shared Task on Parsing Morphologically-Rich Languages, Dublin, Ireland (2014)
4. Bohnet, B., Nivre, J., Boguslavsky, I., Farkas, R., Ginter, F., Hajic, J.: Joint morphological and syntactic analysis for richly inflected languages. Transactions of the Association of Computational Linguistics **1** (2013) 415–428
5. Petrov, S., Barrett, L., Thibaux, R., Klein, D.: 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. (2006) 433–440
6. Björkelund, A., Çetinoğlu, O., Farkas, R., Müller, T., Seeker, W.: (re)ranking meets morphosyntax: State-of-the-art results from the SPMRL 2013 shared task. In: Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages, Seattle, Washington, USA (2013) 135–145
7. Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged Treebank. In Matoušek, V., Mautner, P., Pavelka, T., eds.: Text, Speech and Dialogue: Proceedings of TSD 2005. Springer (2005)
8. Vincze, V., Szauter, D., Almási, A., Móra, Gy., Alexin, Z., Csirik, J.: Hungarian dependency treebank. In: LREC. (2010)

9. Váradi, T.: The Hungarian National Corpus. In: Proceedings of the Second International Conference on Language Resources and Evaluation. (2002) 385–389
10. Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A Tool for Morphological and Dependency Parsing of Hungarian. In: Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013. (2013) 763–771
11. Müller, T., Schmid, H., Schütze, H.: Efficient Higher-Order CRFs for Morphological Tagging. In: Proceedings of EMNLP. (2013)
12. Clark, S., Curran, J.R.: Wide-coverage efficient statistical parsing with ccg and log-linear models. Computational Linguistics **33** (2007)
13. Goldberg, Y., Elhadad, M.: Word Segmentation, Unknown-word Resolution, and Morphological Agreement in a Hebrew Parsing System. Computational Linguistics **39**(1) (2013) 121–160
14. Petrov, S.: 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, Los Angeles, California (2010) 19–27
15. McCallum, A.K.: MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu (2002)
16. Charniak, E., Johnson, M.: Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. ACL '05 (2005) 173–180
17. Collins, M.: Discriminative Reranking for Natural Language Parsing. In: Proceedings of the Seventeenth International Conference on Machine Learning. ICML '00 (2000) 175–182
18. Farkas, R., Bohnet, B.: Stacking of dependency and phrase structure parsers. In: Proceedings of COLING 2012, Mumbai, India (2012) 849–866
19. Brown, P.F., Della Pietra, V.J., deSouza, P.V., Lai, J.C., Mercer, R.L.: Class-based n-gram models of natural language. Computational Linguistics **18**(4) (1992) 467–479
20. Sagae, K., Lavie, A.: Parser combination by reparsing. In: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, New York City, USA (2006) 129–132
21. Bohnet, B.: Top Accuracy and Fast Dependency Parsing is not a Contradiction. In: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Beijing, China (2010) 89–97
22. Martins, A., Smith, N., Xing, E., Aguiar, P., Figueiredo, M.: Turbo Parsers: Dependency Parsing by Approximate Variational Inference. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA (2010) 34–44
23. Goldberg, Y., Elhadad, M.: 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, Los Angeles, California (2010) 742–750
24. Ouchi, H., Duh, K., Matsumoto, Y.: Improving dependency parsers with supertags. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers, Gothenburg, Sweden (2014) 154–158
25. Charniak, E.: Statistical parsing with a context-free grammar and word statistics. In: Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence. AAAI'97/IAAI'97 (1997) 598–603

26. McClosky, D., Charniak, E., Johnson, M.: Effective self-training for parsing. In: Proceedings of the Human Language Technology Conference of the NAACL, Main Conference, New York City, USA (2006) 152–159