Treelet Probabilities for HPSG Parsing and Error Correction

Angelina Ivanova♠, Gertjan van Noord♡

♠University of Oslo, Department of Informatics
angelii@ifi.uio.no
♡University of Groningen, Center for Language and Cognition
g.j.m.van.noord@rug.nl

Abstract

Most state-of-the-art parsers aim to produce an analysis for any input despite errors. However, small grammatical mistakes in a sentence often cause a parser to fail to build a correct syntactic tree. Applications that can identify and correct mistakes during parsing are particularly interesting for processing user-generated noisy content. Such systems potentially could take advantage of the linguistic depth of broad-coverage precision grammars. In order to choose the best correction for an utterance, probabilities of parse trees of different sentences should be comparable which is not supported by discriminative methods underlying parsing software for processing deep grammars. In the present work we assess the treelet model for determining generative probabilities for HPSG parsing with error correction. In the first experiment the treelet model is applied to the parse selection task and shows superior exact match accuracy than the baseline and PCFG. In the second experiment it is tested for the ability to score the parse tree of the correct sentence higher than the constituency tree of the original version of the sentence containing grammatical error.

Keywords: treelet model, parsing with error correction, English Resource Grammar

1. Introduction and motivation

Sentences which contain small mistakes such as a spelling error or an unintended repeated word typically cause severe problems for syntactic parsers. In such cases, a parser should produce an analysis for the intended sentence, rather than for the given literal sentence. In order to construct a model which is capable of parsing with such error correction, we need access to (statistical) models of potential errors, as well as models of the a priori likelihood of particular sentence/parse combinations.

The linguistic depth of broad-coverage grammars such as LFG (Kaplan and Bresnan, 1982), HPSG (Pollard and Sag, 1987) and CCG (Steedman, 2000) could be potentially exploited in parsing with error correction. However the most successful wide-coverage grammar-based parsers do not allow cross-sentence comparison of parse tree probabilities because they use log-linear discriminative modeling for estimating model parameters (Abney, 1997; Johnson et al., 1999).

In this project we therefore evaluate the treelet model (Pauls and Klein, 2012) for estimating generative probabilities for HPSG parsing with error correction. We carry out an empirical study of the model's performance on HPSG parse trees to test its ability to choose the tree of the error-corrected version of the sentence. We compare the treelet model results with plain probabilistic context-free grammar (PCFG) and trigram models.

2. Related work

Research interest in the area of grammatical error correction has recently been attracted by several shared tasks: Helping Our Own (HOO) 2011 and 2012 (Dale and Kilgarriff, 2011; Dale et al., 2012) and CoNLL 2013 (Ng et al., 2013). The problem has also been explored in the field of psycholinguistics, e.g. (Levy, 2008).

Parsing with error correction is important on domains of user-generated content such as Twitter, Facebook and others. Small typos that a reader might not even notice can significantly affect the accuracy of the parser. Wagner and Foster (2009) showed that grammatical errors negatively affect the probability assigned to the best parse.

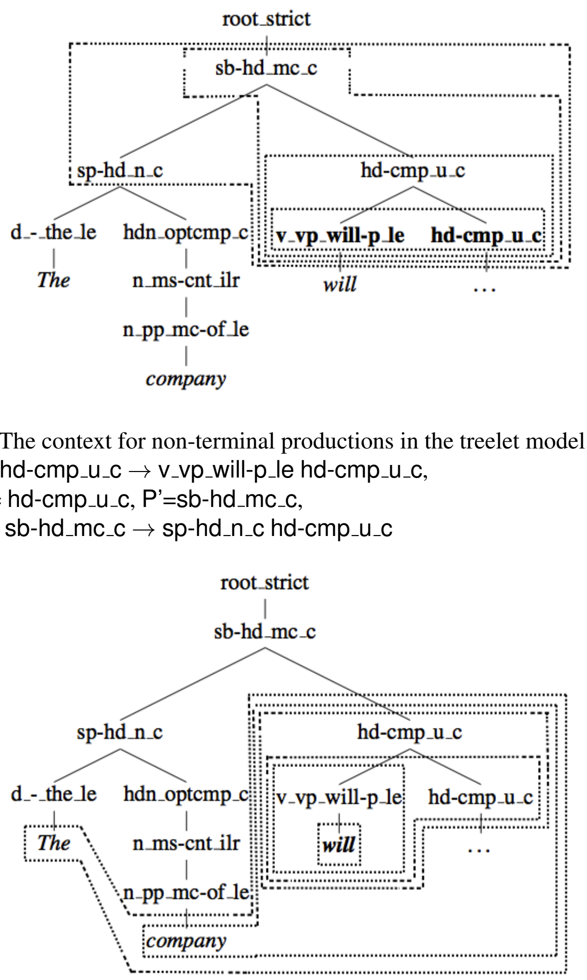
The treelet model that we implement for our task was proposed in Pauls and Klein (2012). The idea of the model is that distribution over rule yields in a constituency tree can be computed using the formula $P(T) = \Pi_{r \in T} p(C_1^d | h)$ where T is a constituency tree consisting of context-free rules of the form $r = P \rightarrow C_1 \dots C_d$, P is the parent symbol of the rule r, $C_1^d = C_1 \dots C_d$ are its children and h is the context. The context for non-terminal productions includes parent node P, grandparent node P' and the rule r' that generated P (see Figure 1a), while for terminal (lexical) productions the context covers P, the sibling R immediately to the right of P, parent rule r' and the previous two words w_{-1} and w_{-2} (see Figure 1b).

Yoshimoto et al. (2013) applied the treelet model for automatic correction of verb form and subject-verb agreement errors. Their system is able to correct errors in which the target verb is distant from its subject.

The ERG and the PET parser were exploited for error detection and correction in the 2013 CoNLL Shared Task (Flickinger and Yu, 2013). The grammar was enhanced with mal-rules to permit certain types of grammatical errors during parsing, and the post-processing script identified and corrected errors using the derivation tree of the most probable analysis produced by the parser.

3. Parsing with the ERG

The LinGO English Resource Grammar (ERG) (Flickinger, 2000) is a stochastic unification-based HPSG grammar that has been developed over the last 20 years. In this formalism, the syntactic level is expressed in a derivation tree that



(a) The context for non-terminal productions in the treelet model.
r = hd-cmp_u_c → v_vp_will-p_le hd-cmp_u_c,
P = hd-cmp_u_c, P'=sb-hd_mc_c,
r' = sb-hd_mc_c → sp-hd_n_c hd-cmp_u_c

(b) The context for terminal productions in the treelet model.
P = v_vp_will-p_le, R = hd-cmp_u_c,
r' = hd-cmp_u_c → v_vp_will-p_le hd-cmp_u_c,
w_{-1} = *company*, w_{-2} = *The*

Figure 1: Conditioning contexts for the treelet model

records how an analysis for an input sentence is derived. In this study the treelet model is computed over derivation trees simplified to a phrase structure tree representation.

The parser that is commonly used with ERG is called PET (Callmeier, 2000). PET exploits log-linear model for parse disambiguation (Toutanova et al., 2005; Zhang et al., 2007; Dridan, 2009). For each sentence we can produce a ranked n-best list of candidate analyses.

Should the ranking scores be used as probabilities, normalization over the first 500 analyses is required because the ranking model has been trained over the 500 candidate analyses. However, these probabilities are discriminative and permit only comparison of parse trees of the same sentence. We apply the treelet model in order to overcome the limitations of discriminative log-linear models and obtain generative probabilities that allow us not only to compare the likelihood of various analyses of one sentence but also of different sentences.

We've chosen to work with HPSG rather than PCFG, to benefit from the linguistically motivated precision grammar ERG. Parsing with a large-scale broad-coverage grammar exhibit sufficient accuracy, efficiency and coverage for real application tasks (Schäfer and Kiefer, 2011). In HPSG parsing precision is favored over recall as is common in rule-based approaches. In the settings of our error-correction task high precision is more important than high recall since the treelet model relies on the syntactic analysis which involves the risks of introducing noise and error propagation. Another fact that influenced our choice is that in the head-to-head comparison of the HPSG PET parser and the PCFG Berkeley parser, the former yields superior in- and cross-domain accuracy (Ivanova et al., 2013). And finally, ERG has already been successfully used in one of the systems for grammatical error correction in the 2013 CoNLL Shared Task on Grammatical Error Correction (Flickinger and Yu, 2013).

4. Data

4.1. ERG gold data

For the experiments we use ERG treebanks:

- 36,918 sentences from DeepBank (Flickinger et al., 2012) version 1.0;
- 50,997 sentences collected from the WeSearch treebank (Read et al., 2012), Verbmobil and other resources.

We exported data into the format of the derivation trees, extracted a context-free backbone from them and randomly split the result collection of trees into train, development and test sets, as summarized in Table 1.

Set	# of sentences	# of tokens	# of types
Train	63,298	957,980	75,034
Dev	7,034	105,553	20,659
Test	17,583	265,384	36,401

Table 1: Sizes of train, development and test set of the ERG data

The development set was used for modeling unknown words and rules distributions and estimation of the back-off parameters for the treelet and PCFG models.

4.2. NUS Corpus of Learner English and Wikipedia

For evaluation of the treelet model we used the NUS Corpus of Learner English (Dahlmeier et al., 2013) and Wikipedia snapshots of 2012 and 2013.

The NUS corpus consists of about 1,400 essays written by university students at the National University of Singapore on a wide range of topics, such as environmental pollution, healthcare, etc. All annotations were prepared by professional English instructors at the NUS Centre for English Language Communication. We selected only non-overlapping corrections that are in the scope of one paragraph from this corpus and thus obtained **2,181** pairs of erroneous and corrected sentences.

In order to prepare the Wikipedia dataset[1], we collected a number of documents present in both Wikipedia 2012 and 2013, sentence-split the paragraphs with jmx_mxterminator, aligned them with Microsoft Aligner and collected pairs of sentences that differ only by one word with Damerau-Levenshtein distance less or equal to three. We parsed these sentences with PET and obtained pairs of phrase-structure trees from the HPSG derivation trees. With the Wikipedia test set we work under the assumption that the version of a sentence extracted from Wikipedia 2013 provides grammatical error correction for the version from Wikipedia 2012. The size of this data collection is **4,604** pairs of sentences.

5. Experiments

We apply the treelet model to two tasks: i) parse selection; and ii) scoring parse trees of erroneous and corrected sentences. The aim of the first experiment is to measure how well the treelet model copes with the parse selection problem compared to the PCFG model. In the second experiment we evaluate the treelet's ability to assign to the corrected sentence a higher probability than to the erroneous one in comparison to the PCFG and SRILM trigram models.

5.1. Treelet model for parse selection

In the experiment below we contrast the treelet and PCFG models for the parse selection task. Parsing the raw text of the ERG data test set with PET, we retrieved up to 500 analyses for each sentence. We compute how many times either model chooses the gold-standard analysis among PET options. Our upper-bound is the number of sentences for which the PET output contained the gold-standard analysis. We define as a baseline the number of sentences for which a random choice model correctly selects the gold standard. For the evaluation of the models we took 12,311 sentences for which the PET parser produced at least 2 analyses and the gold-standard was among them.

Upper-bound	12,311 sent.	100%
Treelet	4,487 sent.	**36.45%**
PCFG	2,905 sent.	23.60%
Random	621 sent.	5.04%

Table 2: Number and percentage of sentences for which treelet, PCFG and random choice models scored the gold-standard parse tree higher than other analyses produced by the PET parser

Table 2 provides a summary of results and shows that the treelet model outperforms random and PCFG models.

We can relate our result of 36.45% of exact match for the treelet model with the work of (Packard, 2012) who obtained roughly 41% exact match accuracy on the WeSearch treebank with a log-linear parser for the ERG grammar. However the setups are significantly different: i) the test set in (Packard, 2012) is much more homogeneous as it covers only two domains: NLP and Linux (authors used the WeSearch treebank for training and testing); ii) the size of the data set is much bigger in our experiment (we used 63,298 sentences for training and 12,311 for testing vs. around 9,100 for training and testing in (Packard, 2012). The performance of the treelet model for the parse selection task is thus reasonable.

In order to analyze the strengths of the treelet model over PCFG, we collected the two sets of sentences:

1) 2905 sentences for which the PCFG model selected the gold standard analysis;

2) 2227 sentences for which the treelet model selected the gold standard analysis while the PCFG model selected a non-gold standard analysis.

We compared these two sets according to the following criteria: (a) length of sentences; (b) frequency of "and" coordination; (c) amount of out-of-vocabulary (OOV) words (words that are not in the vocabulary that was used to train the treelet and PCFG models). With respect to the first criteria, there is a difference between an average sentence length in the two collections: 9 tokens for the first set and 13 tokens for the second set (inclusion and exclusion of the outliers does not influence these numbers). We identified how many times "and" coordination occurred in the set by searching for the lexical type c_xp_and_le in the gold standard analyses of the sentences. There are 419 "and" coordinations in the collection for which PCFG makes the correct parse choice, while there are 548 "and" coordinations in the set for which the treelet model makes the correct choice but the PCFG fails to select the gold standard parse tree. The difference in the number of OOV is insignificant: 1179 tokens (1163 types) in the first set and 1136 tokens (1117 types) in the second set which constitutes 3.9% and 4.5% of the collections' sizes correspondingly. To sum up, the sentences on which the treelet model outperformed the PCFG model are on average longer and contain slightly more coordination structures, while most of the vocabulary has been seen by the model during training.

5.2. Treelet model for scoring parse trees of erroneous and corrected sentences

To test the strength of the model for the error correction task, we apply it to the sentence pairs that consist of a sentence with a mistake and its corrected version. We parsed both sentences in each pair with PET and computed probabilities of the 1-best parse tree for each sentence using the treelet and PCFG models. We test the ability of the treelet model to score the parse tree of the corrected version of the sentence higher than the parse tree of the original version of the sentence.

In addition we score probabilities of both sentences in each pair with the SRILM trigram model. Even though the individual scores of the treelet and the trigram models are of course not compatible, in this task we can compare the number of times each model prioritized the corrected sentence.

Table 3 shows that all the models beat the baseline (random choice model), and the treelet model appears to perform best on the NUS corpus data.

[1] The parallel sentences from Wikipedia 2012 and Wikipedia 2013 are available at `http://heim.ifi.uio.no/~angelii/wiki_pairs.html`

Model	NUS corpus			Wikipedia		
	Corrected	Equal	Original	Corrected	Equal	Original
Oracle	2,223		0	4,604		0
Treelet	1,449	11	763	1,884	994	1,726
PCFG	1,304	11	908	1,835	996	1,773
Trigram	1,249	80	894	1,732	1,294	1,578
Random	1,112		1,111	2,302		2,302

Table 3: Number of sentences for which treelet, PCFG and random choice models 1) scored the parse tree of original version of the sentence higher (column "Original"), 2) scored the parse tree of corrected version of the sentence higher (column "Corrected"), 3) scored parse trees of original and corrected versions equally (column "Equal")

On the Wikipedia data the results for the treelet model are to some degree better than for the other models (see Table 3), however the statistical significance tests - binomial test, population proportions and analysis of variance, - show that differences in performance of models are not significant.

For the error analysis, we automatically collected a sample of 100 pairs of Wikipedia sentences for which the treelet model scored the original version of a sentence higher than the corrected one, and we manually ranked which version is preferable for a human reader. Since the sentences are evaluated without context, we have to ignore context-sensitive corrections. We rank the versions of a sentence equally if modification results in a valid sentence and affects only the tense of the originally proper sentence ("he also enjoys fishing" vs. "he also enjoyed fishing"). We also neglect stylistic alterations ("hanged" vs. "hung") because we do not know which pattern should be favored in a larger context. In addition, we disregard spelling corrections for named entities transliterated from non-latin scripts (Cyrillic, Arabic, Chinese and others), e.g. both "Showashinzan" and "Shōwa- shinzan" are acceptable transliterations. The results of the manual evaluation suggest that 56 sentences should have been ranked equally. For these cases the treelet model scored the original version of the sentence higher based on the statistics collected from the training corpus, e.g. "is" is about 4 times more frequent than "was" and "has" is 2 times more frequent than "had" in the ERG data. For the 2 sentences the original version should have been preferred therefore the treelet model performed correctly but was penalized due to the imperfect evaluation heuristic that the revision version is always more accurate. For the rest 42 sentences automatic evaluation was fair, but only one correction concerned a grammatical error unrelated to named entities, while the rest involved typos in proper nouns and stylistic, semantic and factual types of errors. This analysis shows that only 1% of the randomly selected sample examples for which the treelet model is automatically classified to make the wrong choice, concern the type of errors that model is targeted to tackle.

5.3. Discussion

The treelet model demonstrates better performance than PCFG and random models for parse selection. Subsequently, on the NUS corpus the treelet model scores the parse tree of the corrected version of the sentence higher than its original version with grammatical errors more often than PCFG, trigram and SRILM trigram models. However on the Wikipedia dataset, performances of the models are not significantly better than chance and the pairwise difference between the models on this set are not significant either.

The main reason for the fact that advantages of the treelet model are verifiable on the NUS corpus but appear to be statistically insignificant on the Wikipedia data could be the difference between the two domains. The NUS corpus is a collection of essays of university students manually corrected by professional English language instructors. It has been professionally annotated for the error correction task and released specifically for research purposes. On the contrary, Wikipedia articles could be created and edited by anyone, therefore it is less homogeneous in style and contains more noise.

We manually analyzed a set of randomly selected 100 sentences from the Wikipedia data in order to shed some light on error corrections that are intuitively hard for our system. Many of the small corrections that we observe in the Wikipedia data occur in proper nouns rather than in dictionary words, e.g. "Stuebing" - "Stübing". This leads to the fact that many words with mistakes were not seen during training, therefore they are assigned to the unknown word class on the test set. For this reason there is no difference for the models if the error was corrected or not: in the original and corrected versions of the sentence it is analyzed as unknown word. Another important fact is that the extracted sentences from Wikipedia 2013 do not always provide grammatical error corrections to the sentences from Wikipedia 2012 since some of the edits concern the semantic level, e.g. "most" - "many", and others are stylistic, e.g. "you" - "one". Finally, some errors can only be corrected on the discourse level, e.g. "his daughter" - "their daughter", or are simply factual: "The blood of snakes, dogs and cats are poisonous to them" - "The blood of snakes, rats and cats are poisonous to them".

6. Conclusions

In this paper we discussed the potential of the treelet model in application to parse selection and error correction tasks. In the first experiment of parse selection the treelet model outperformed PCFG and random choice systems. In the second experiment we evaluated the ability of the treelet system to rank the parse tree of the corrected version of the sentence higher than the original version of the erroneous sentence. The results on the NUS corpus are in favor of the treelet model in contrast to PCFG and trigram models.

However, the advantages of the treelet model are not obvious in the sentence pairs that we extracted from Wikipedia 2012 and Wikipedia 2013 under the hypothesis that sentences from the latter provide the corrections for the sentences of the former resource. Our analysis suggests that the reasons for this are the noise in the Wikipedia and error types that cannot be tackled solely with the treelet model. The future work directions include enhancement of the current simple implementation of the treelet model with the transformations suggested in (Pauls and Klein, 2012) and supplying the system with a large dictionary of named entities. The goal is to build a system for HPSG-parsing with error correction that would consist of a module for generation of the possible sentence corrections and a parsing module (with the PET parser and the treelet model for estimation of generative probabilities) and would select the final parse tree based on the joint probability of the scores provided by these two modules.

7. Acknowledgements

We are grateful to Lilja Øvrelid and Stephan Oepen for helpful feedback and discussions, as well as to anonymous reviewers for their comments. The first author is funded by the Norwegian Research Council through its WeSearch project. Experimentation is made possible through access to the ABEL high-performance computing facilities at the University of Oslo, and we are grateful to the Scientific Computing staff at UiO, as well as to the Norwegian Metacenter for Computational Science, and the Norwegian tax payer.

8. References

Abney, S. P. (1997). Stochastic attribute-value grammars. *Computational Linguistics*, 23:597 – 618.

Callmeier, U. (2000). PET. A platform for experimentation with efficient HPSG processing techniques. *Natural Language Engineering*, 6(1):99 – 108.

Dahlmeier, D., Ng, H. T., and Wu, S. M. (2013). Building a Large Annotated Corpus of Learner English: The NUS Corpus of Learner English. In *Proceedings of the 8th Workshop on Innovative Use of NLP for Building Educational Applications*, page 22 – 31, Atlanta, Georgia.

Dale, R. and Kilgarriff, A. (2011). Helping our own: the HOO 2011 pilot shared task. In *Proceedings of the 13th European Workshop on Natural Language Generation*, page 242 – 249, Stroudsburg, PA, USA.

Dale, R., Anisimoff, I., and Narroway, G. (2012). HOO 2012: a report on the preposition and determiner error correction shared task. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, page 54 – 62, Stroudsburg, PA, USA.

Dridan, R. (2009). *Using lexical statistics to improve HPSG parsing*. PhD thesis, Department of Computational Linguistics, Saarland University.

Flickinger, D. and Yu, J. (2013). Toward more precision in correction of grammatical errors. In *Proceedings of the 17th Conference on Natural Language Learning*, page 68 – 73, Sofia, Bulgaria.

Flickinger, D., Zhang, Y., and Kordoni, V. (2012). DeepBank. A dynamically annotated treebank of the Wall Street Journal. In *Proceedings of the 11th International Workshop on Treebanks and Linguistic Theories*, page 85 – 96, Lisbon, Portugal.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6 (1):15 – 28.

Ivanova, A., Oepen, S., Dridan, R., Flickinger, D., and Øvrelid, L. (2013). On different approaches to syntactic analysis into bi-lexical dependencies. An empirical comparison of direct, PCFG-based, and HPSG-based parsers. In *Proceedings of the 13th International Conference on Parsing Technologies*, page 63 – 72, Nara, Japan, November.

Johnson, M., Geman, S., Canon, S., Chi, Z., and Riezler, S. (1999). Estimators for stochastic 'unification-based' grammars. In *Proceedings of the 37th Meeting of the Association for Computational Linguistics*, page 535 – 541, College Park, USA.

Kaplan, R. and Bresnan, J. (1982). Lexical Functional Grammar. A formal system for grammatical representation. In Bresnan, J., editor, *The Mental Representation of Grammatical Relations*, page 173 – 281. MIT Press, Cambridge, MA, USA.

Levy, R. (2008). A noisy-channel model of rational human sentence comprehension under uncertain input. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, page 234 – 243, Honolulu, USA.

Ng, H. T., Wu, S. M., Wu, Y., Hadiwinoto, C., and Tetreault, J. (2013). The CoNLL-2013 Shared Task on Grammatical Error Correction. In *Proceedings of the 17th Conference on Natural Language Learning: Shared Task*, page 1 – 12, Sofia, Bulgaria.

Packard, W. (2012). Choosing an evaluation metric for parser design. In *Proceedings of Human Language Technologies: The 2012 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, page 29 – 34, Montréal, Canada.

Pauls, A. and Klein, D. (2012). Large-scale syntactic language modeling with treelets. In *Proceedings of the 50th Meeting of the Association for Computational Linguistics*, page 959 – 968, Jeju, Republic of Korea.

Pollard, C. and Sag, I. A. (1987). *Information-Based Syntax and Semantics. Volume 1: Fundamentals*. CSLI Lecture Notes # 13. CSLI Publications, Stanford, USA.

Read, J., Flickinger, D., Dridan, R., Oepen, S., and Øvrelid, L. (2012). The WeSearch Corpus, Treebank, and Treecache. A comprehensive sample of user-generated content. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, page 1829 – 1835, Istanbul, Turkey.

Schäfer, U. and Kiefer, B. (2011). Advances in Deep Parsing of Scholarly Paper Content. In Bernardi, R., Chambers, S., Gottfried, B., Segond, F., and Zaihrayeu, I., editors, *Advanced Language Technologies for Digital Libraries*, volume 6699 of *Lecture Notes in Computer Science*, pages 135–153. Springer Berlin Heidelberg.

Steedman, M. (2000). *The Syntactic Process*. MIT Press, Cambridge, MA, USA.

Toutanova, K., Manning, C. D., Flickinger, D., and Oepen, S. (2005). Stochastic HPSG Parse Disambiguation using the Redwoods Corpus. *Research on Language and Computation*, 3:83 – 105.

Wagner, J. and Foster, J. (2009). The effect of correcting grammatical errors on parse probabilities. In *Proceedings of the 11th International Conference on Parsing Technologies*, page 176 – 179, Paris, France.

Yoshimoto, I., Kose, T., Mitsuzawa, K., Sakaguchi, K., Mizumoto, T., Hayashibe, Y., Komachi, M., and Matsumoto, Y. (2013). NAIST at 2013 CoNLL grammatical error correction shared task. In *Proceedings of the 17th Conference on Natural Language Learning: Shared Task*, page 26 – 33, Sofia, Bulgaria.

Zhang, Y., Oepen, S., and Carroll, J. (2007). Efficiency in unification-based n-best parsing. In *Proceedings of the 10th International Conference on Parsing Technologies*, page 48 – 59, Prague, Czech Republic.