A Feasibility Study on Low Level Techniques for Improving Parsing Accuracy for Spanish Using Maltparser

Miguel Ballesteros[1], Jesús Herrera[1], Virginia Francisco[2], and Pablo Gervás[2]

[1] Departamento de Ingeniería del Software e Inteligencia Artificial
[2] Instituto de Tecnología del Conocimiento
Universidad Complutense de Madrid
C/ Profesor José García Santesmases, s/n
E–28040 Madrid, Spain
{miballes,jesus.herrera,virginia}@fdi.ucm.es, pgervas@sip.ucm.es

Abstract. In the last years dependency parsing has been accomplished by machine learning–based systems showing great accuracy but usually under 90% for Labelled Attachment Score (LAS). Maltparser is one of such systems. Machine learning allows to obtain parsers for every language having an adequate training corpus. Since generally such systems can not be modified the following question arises: Can we beat this 90% LAS by using better training corpora? Some previous work points that high level techniques are not sufficient for building more accurate training corpora. Thus, by analyzing the words that are more frequently incorrectly attached or labelled, we study the feasibility of some low level techniques, based on n–version parsing models, in order to obtain better parsing accuracy.

1 Introduction

In the 10th edition of the Conference of Computational Natural Language Learning (CoNLL) a first Shared Task on Multilingual Dependency Parsing was accomplished [1]. Thirteen different languages including Spanish were involved. Participants should implement a parsing system that could be trained for all these languages. Maltparser achieved great results in this task, in which Spanish was proposed for parsing.

The goal of the present work was to study the feasibility of low level techniques to obtain a better parsing performance when the parsing system (based on machine learning) can not be modified. 90% Labelled Attachment Score seems to be a *de facto* limit for contemporary dependency parsers. Some previous works [2] have been developed on how to improve dependency parsing by applying high level tecnhiques to obtain better training corpora. The conclusions of these works are that overall accuracy can not be enhanced by modifying training corpus' size or its sentences' lengths. In adition local accuracy is important too, but it has not been solved yet. N–version parsers could be the way to obtain better overall

S. Konstantopoulos et al. (Eds.): SETN 2010, LNAI 6040, pp. 39–48, 2010.

accuracies by obtaining better local accuracies. N–version parsers consist of n specifically trained models, each one able to parse one kind or a small range of kinds of sentences. Thus, a n–version parser should select the specific model that would better parse the sentence that is used as input. Each specific model would improve parsing accuracy of the sentences for which is specialized, producing a better overall parsing acurracy.

After selecting a small number of words that are more frequently incorrectly attached or labelled, we started a thorough analysis of the parsings that contained those words. We selected the two most frequently incorrectly attached or labelled words, i.e., the conjunction *and* ("y" or "e" in Spanish) and the preposition *to* ("a" in Spanish.). These words led us to develop preliminary works on low level techniques useful to reach better parsing accuracy by improving attachment and labelling.

Maltparser 0.4 is the public available software of the system presented by Nivre's group to the CoNLL–X Shared Task. Since Spanish was the language for which we decided to develop the present work and we have already developed some previous work on dependency parsing using Maltparser [3,4,5], we used Maltparser 0.4 to carry out our experiments.

The paper is organized as follows: Section 2 describes the CoNLL–X Shared Task focusing on Spanish participation; also we show our results when replicating the participation of Nivre's group. Section 3 shows our consideration about local parsing accuracy. Section 4 shows two cases study in which the conjunction and preposition "a" are used to evaluate the feasibility of low level techniques oriented to obtaining better local parsing results. Finally, Section 5 shows the conclusions of the presented work and suggests some future work.

2 The CoNLL–X Shared Task

Each year the Conference of Computational Natural Language Learning (CoNLL) features a shared task, the 10th CoNLL Shared Task was Multilingual dependency parsing [1]. The goal of this Shared Task was to label dependency structures by means of a fully automatic dependency parser. This task provided a benchmark for evaluating the parsers presented to it accross 13 languages among which is Spanish. Systems were scored by computing their Labelled Attachment Score (LAS), i.e. the percentage of "scoring" tokens for which the system had predicted the correct head and dependency label [6]. Also Unlabelled Attachment Score (UAS) and Label Accuracy (LA). UAS is the percentage of "scoring" tokens for which the system had predicted the correct head [7]. LA is the percentage of "scoring" tokens for which the system had predicted the correct dependency label [8].

Our research is focused on Spanish parsing. For this language results across the 19 participants ranged from 47.0% to 82.3% LAS, with an average of 73.5%. The Spanish treebank used was AnCora [9], [10], a 95,028 wordforms corpus containing open–domain texts annotated with their dependency analyses. AnCora

was developed by the Clic group at Barcelona University. The results of Spanish parsing were in the average. The two participant groups with the highest total score for Spanish were McDonald's group [11] (82.3% LAS) and Nivre's group [12] (81.3% LAS). We are specially interested in Nivre's group research because we used their system (Maltparser 0.4) for the experiments presented in this paper. Other participants that used the Nivre algorithm in the CoNLL–X Shared Task were Johansson's group [13] and Wu's group [14]. Their scores on Spanish parsing were 78.2% (7th place) and 73.2% (13th place), respectively. The evaluation shows that the approach given by Nivre gives competitive parsing accuracy for the languages studied. More specifically Spanish parsing scored 81.3% LAS, only 1 point under the best one [11], which did not use the Nivre algorithm but a Eisner's bottom–up span algorithm in order to compute maximum spanning trees.

In our work, the first step was to replicate the participation of Nivre's group in the CoNLL–X Shared Task for Spanish. We trained Maltparser 0.4 with the section of AnCora that was provided as training corpus in the CoNLL–X Shared Task (89,334 wordforms) and the system was set as referred by Nivre's group in [12]. Once a model was obtained, we used it to parse the section of AnCora that was provided as test set in the CoNLL–X Shared Task (5,694 wordforms). We obtained the same results as the Nivre's group in the Shared Task, i.e., LAS = 81.30%, UAS = 84.67% and LA = 90.06%. These results serve us as a baseline for our work which is presented in the following sections.

3 Local Parsing Accuracy

Considering the baseline experiment described in Section 2, despite a high overall parsing accuracy only 358 wordforms of the test corpus obtain a 100% LAS, UAS and LA in all parsed sentences, i.e., only 6.3% of the wordforms. If considering sentences, only 38 sentences of the test corpus (18.4% of them) were parsed without errors. An end user should usually expect a high local parsing accuracy (at the sentence level) rather than a high overall parsing accuracy. But nowadays a remarkable percentage of sentences in Spanish shows almost one error when parsed by Maltparser. Our hypothesis is that by enhancing local accuracy, not only overall accuracy should be enhanced, but end user satisfaction should be increased.

We found that there is a small set of words that show an incorrect attachment, labelling or both. These words are the prepositions "a" (*to*), "de" (*of*), " en" (*in*), "con" (*with*), "por" (*for*), the conjunction *and*, which has two wordings: "y" or "e", and the nexus "que" (*that*). All these words sometimes cause error in the dependency, in the head tag, or in both tags. For instance there are only 20 sentences (340 wordforms), in the test corpus presented in Section 2, with only one error after parsing. That is 9.7% of the corpus' sentences and 5.98% of its wordforms. We found that in 10 of these 20 sentences the only failure is caused by one of the words listed above.

4 Cases Study: The Conjunction and the Preposition "a"

The conjunction and the preposition "a" are the words that caused a parsing error more frequently. This is why we selected them as cases to study to determine if low level techniques are feasible to increase parsing accuracy. We started experimenting these tecnhiques with the conjunction. The study of the obtained errors when parsing conjunctions, began with a manual analysis of AnCora. Thus, we extracted from AnCora every sentence containing a conjunction ("y" or "e"). There are 1.586 sentences with at least one conjunction in the whole AnCora. We inspected these sentences to find labelling patterns and in doing so we obtained a list of patterns that depend on conjunction's action. For instance, a pattern is given when conjunction acts as nexus in a coordinated copulative sentence and another pattern is given when it acts as the last nexus in a list of nouns. For example, the following sentence match these two patterns: *Los activos en divisas en poder del Banco Central* **y** *el Ministerio de Finanzas se calculan en dólares estadounidenses* **y** *su valor depende del cambio oficial rublo–dólar que establece el Banco Central* (The foreign exchange assets held by the Central Bank **and** the Ministry of Finance are calculated in U.S. dollars **and** its value depends on the ruble-dollar official exchange rate established by the Central Bank). In this example the first *y* is a nexus between the proper nouns *Banco Central* and *Ministerio de Finanzas* and the second *y* acts as a coordinated copulative nexus. These patterns guided the experiments described below.

4.1 The Conjunction

In this subsection we present two different approaches we have applied to the conjuction.

First Approach to the Conjuction. The first approach that we studied was an n–version parsing model. Our idea was to determine if some kind of "difficult" sentences could be succesfully parsed by specific parsers while a general parser would parse the non–troubled sentences. The first specific parser that we tried to obtain was supposed to accurately parse quoted sentence sections containing conjunctions. This situation is quite commonly given and corresponds to one of the labbeling patterns that we have identified as problematic. This way we trained a parsing model with Maltparser 0.4 for sentences that contain conjuctions. The system was set as in Nivre's group participation in the CoNLL–X Shared Task. The training corpus consisted of only quoted sentence sections containing conjunctions. These sentence sections were obtained from the section of AnCora provided as training corpus for Spanish in the CoNLL–X Shared Task. It consisted of 22 sentence sections starting and finishing by a quotation mark and containing conjunctions. The test corpus was obtained in a similar way from the section of AnCora provided as test corpus for Spanish in the CoNLL–X Shared Task. This test corpus contained 7 sentences. To analyse this approach, we incrementally built a training corpus and we evaluated the

parsing performance for every trained model. The method we followed to build this corpus is described below:

- First of all, we selected the longest sentence of the training subcorpus of quoted sentence sections and this was the first subcorpus added to the incremental training corpus.
- Then we iterated until every sentence section was added to the incremental training corpus. In each iteration we did the following:
  - Malparser 0.4 was trained with the incremental corpus.
  - The trained model was tested by parsing the test subcorpus with it.
  - The remaining longest sentence section was added to the incremental corpus.

The results of this experiment are showed in Figure 1, in which we plotted LAS, UAS and LA for every iteration. In the x axis we represented the number of sentences contained in the incremental training corpus in every iteration and in the y axis the values for LAS, UAS and LA.

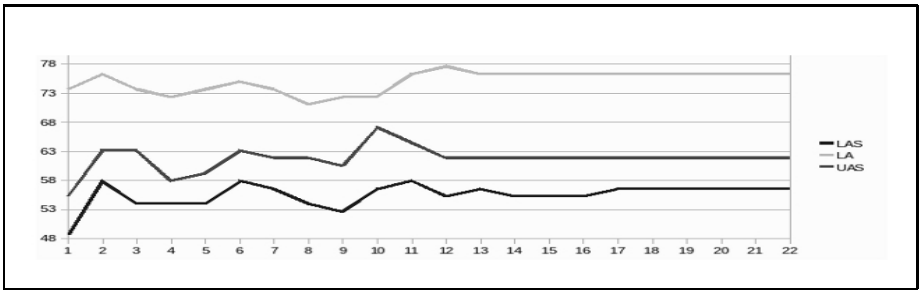

Fig. 1. LAS, UAS and LA when training a parsing model incrementally with quoted sentence sections containing conjunctions from AnCora

If we take only conjunction parsing into consideration the results are quite good. In the first iteration 3 conjunctions were incorrectly parsed, but in the second and the other iterations only 1 conjunction was incorrectly parsed. But as seen in Figure 1 the overall results did worse than those obtained by the general parser. Therefore, despite the improvement in local accuracy this approach does not seem to be realistic. This is because the number of available samples is not sufficient to train a specific model. This model should not only be able to obtain good results for parsing conjunctions but also for all the words of the whole quoted sentence. This led us to investigate another approach which is explained in the next section.

A more Complex Approach. In this section we study the feasibility of a more complex n–version parsing model. As seen in Section 4.1, specific models can be

trainied to obtain high accurate parsings for a specific word, but these models cannot deal with the whole sentence in which the specific word is contained. This is what inspired this new approach. The idea is to obtain several specific models, each one able to accurately parse a single word into a specific context. Thus, the word would be one of the words that are more frequently incorrectly parsed and the context would be one of the labelling patterns referred in the beginning of Section 4. For instance, one of these words is the conjunction "y" and one of the contexts in which it can be found is the one presented in Subsection 4.1, i.e., quoted sentence sections. This way, after parsing a sentence with a general model (such as the one presented in Section 2) a program should decide if the parsed sentence contains a word that must be parsed by a specific model. In that case the program should choose the appropriated specific model for this word in the context in which it is. Once the sentence is parsed with the specific model, the result for the "problematic" word is replaced in the result obtained by the general model. This way the best of both parsings can be made. In the case of the conjunction, the labelling given to it by the specific parser is cut from this parsing and pasted into the parsing given by the general model, by replacing the labelling given to the conjunction by the general parser. This easy solution is posible because the conjunction is always a leaf of the parsing tree and its labellings can be changed without affecting the rest of the parsing. To study if this n–version parsing model could be useful to get more accurate parsings we developed the experiments described below.

For the first experiment we trained a specific model for coordinated copulative sentences. We built a specific training corpus with the set of unambiguous coordinated copulative sentences contained in the section of AnCora that was provided as training corpus in the CoNLL–X Shared Task. This specific training corpus contains 361 sentences (10,561 wordforms). Then we parsed all the coordinated copulative sentences contained in the section of AnCora that was provided as test corpus in the CoNLL–X Shared Task (16 sentences, 549 wordforms).

MaltParser uses history–based feature models for predicting the next action in the deterministic derivation of a dependency structure, which means that it uses features of the partially built dependency structure together with features of the (tagged) input string. More precisely, features are defined in terms of the wordform (LEX), part–of–speech (POS) or dependency type (DEP) of a token defined relative to one of the data structures STACK, INPUT and CONTEXT. A feature model is defined in an external feature specification[1]. We set the experiments described above with the same feature model that Nivre's group used in its participation in the CoNLL–X Shared Task. We also used this feature model in the present experiment and we obtained that the conjuntion was incorrectly parsed 8 times (in a test set containing 16 conjunctions). This fact led us to investigate other feature models. After a few failed attempts we found a feature model where 12 of the 16 conjunctions were parsed correctly. This feature model is shown in Figure 2.

[1] An in–depth description of these feature models can be found in `http://w3.msi.vxu.se/~nivre/research/MaltParser.html#features`

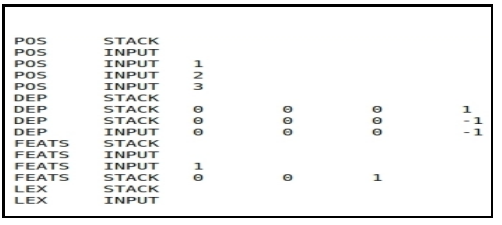

Fig. 2. Feature model for coordinated copulative sentences

Despite that the results were enhanced by using the new feature model, the general parsing model (obtained in Section 2) parses correctly 13 of these 16 conjunctions. It could mean that specific models are not feasible for our objectives. Since the accuracies reached by both models were very similar, we developed some other experiments to confirm or reject this hypothesis. Thus, we tried new specific parsers for other combinations conjunction–context.

For the second experiment we developed a specific parser for conjunctions acting as a nexus in a list of proper nouns. We built a specific training corpus with the set of unambiguous sentences containing conjunctions acting as a nexus in lists of proper nouns, from the section of AnCora that was provided as training corpus in the CoNLL–X Shared Task. This specific training corpus contains 59 sentences (1,741 wordforms). After the training we parsed all the sentences containing conjunctions acting as a nexus in the lists of proper nouns, from the section of AnCora that was provided as test corpus in the CoNLL–X Shared Task (5 sentences, 121 wordforms). We set this training with the same feature model that Nivre's group used in its participation in the CoNLL–X Shared Task. This specific model parsed all 5 conjunctions of the test set successfully, while the general model parsed only 4 of these conjunctions successfully.

We developed a third experiment to evaluate a specific model for parsing conjunctions acting as a nexus in the lists of common nouns. We built a specific training corpus with the set of unambiguous sentences containing conjunctions acting as a nexus in the lists of common nouns, from the section of AnCora that was provided as training corpus in the CoNLL–X Shared Task. This specific training corpus contains 266 sentences (8,327 wordforms). After the training we parsed all the sentences containing conjunctions acting as a nexus in the lists of proper nouns, from the section of AnCora that was provided as test corpus in the CoNLL–X Shared Task (15 sentences, 480 wordforms). Once again the best feature model was the one that Nivre's group used in its participation in the CoNLL–X Shared Task. This specific model parsed 12 of the 15 conjunctions of the test set successfully, while the general model parsed only 10 of these conjunctions successfully.

A last experiment was accomplished to find more evidences for the feasability of this n–version parsing model. Doing this we developed a specific model for parsing conjunctions acting as a nexus in the lists of adjectives or constructions acting as adjectives. We built a specific training corpus with the set of unambiguous

sentences containing conjunctions acting as nexus in lists of adjectives or constructions acting as adjectives, from the section of AnCora that was provided as training corpus in the CoNLL–X Shared Task. This specific training corpus contains 59 sentences (3,155 wordforms). After the training we parsed all the sentences containing conjunctions acting as a nexus in the lists of adjectives, from the section of AnCora that was provided as test corpus in the CoNLL–X Shared Task (5 sentences, 113 wordforms). The feature model that Nivre's group used in its participation in the CoNLL–X Shared Task gave the best results again. This specific model parsed all the 5 conjunctions of the test set successfully, while the general model parsed 4 of these conjunctions successfully.

The parsings given by the general parsing model to the conjunctions involved in the previous four experiments were replaced by the parsings given by the specific models. This way we combined both parsings as seen above in this section. Then, we recomputed LAS, UAS and LA for this combined parsing, obtaining the following values: LAS = 81.92%, UAS = 85.31% and LA = 90.06%. The results show a slight enhancement with respect to the results given by the general parsing model presented in Section 2. In addition, in the combined parsing the conjunction does not belong to the set of words that are more frequently incorrectly parsed. This improvement seems to indicate that this n–version parsing model is feasible and overall accuracy could be improved via local accuracy improvement.

4.2 The Preposition "a"

Once we found the promising approach presented in Section 4.1 we applied it to the following word in the list of more frequently wrong parsed words. This way we followed the steps stated previously. We started looking for the different ways in which the preposition "a" is attached and labelled. Six cases were found, as shown in Table 1.

A specific parser was trained for each case using Maltparser 0.4 set as in the CoNLL-X Shared Task. We used the feature model proposed in the Shared Task, except for case number 1 for which we used a *m3.par* model. This model was chosen empirically because the one proposed in the Shared Task was not suitable for tackling case number 1. In all the cases, except for case number 5, the quality of the labelling and the attachment of the word "a" were clearly improved, as shown in Table 1. Case number 5 is very challenging because we had only 8

Table 1. Attachment and labelling of the preposition "a" in AnCora. Found cases and LAS only for the preposition "a", before and after the application of our method.

Case	#1	#2	#3	#4	#5	#6
Label	CD	CI	CC	CREG	_	_
Attached to a	verb					noun
LAS$_a$ before	62.5%	42.9%	60.0%	25.0%	0.0%	50.0%
LAS$_a$ after	87.5%	100%	100%	75.0%	0.0%	100%

sentences containing it in the training set and 1 sentence in the test set. Perhaps the problem is in the small number of sentences used for training. Since case number 5 is not frequently given we did not make any particular efforts to solve it in such a preliminary work. Nevertheless, it remains as a very interesting case study for future work.

Once again the improvement in local accuracy is beneficial to the overall accuracy. When aplying the labellings and attachments given by all the specific parsers presented in Sections 4.1 and 4.2, we obtain the following new overal values for the test set: LAS = 82.17%, UAS = 85.51% and LA = 90.32%.

5 Conclusions and Future Work

Previous work shows that high level techniques, such as controlling training corpus size or its sentences' lengths, are not sufficient for improving parsing accuracy when using machine learning–based systems that can not be modified. This led us to investigate low level techniques, based on the detailed study of the words that are more frequently incorrectly parsed. In this work we study the feasibility of these low level techniques to reach better parsing accuracy. The idea presented in this paper is to develop n–version parsing models. Each parsing model is trained to accurately parse a specific kind of word in a specific context. This way, local accuracy is enhanced by avoiding errors given by general parsers, i.e., by enhacing local accuracy. Therefore, if a sentence contains one of the words that are more frequently incorrectly parsed by general parsers, it is simultaneously sent to a specific parser and to a general parser. After this, both parsings are combined in order to make the best of them.

This work relies on two cases study: the conjunction and the preposition "a", because these are the parts of speech most frequently incorrectly parsed. These preliminary experiments show that these kinds of low level techniques are promising for improving parsing accuracy under the circumstances described in this paper. A lot of promising future work remains encouraged by the present one. This future work includes not only similar studies on the rest of the words that are more frequently incorrectly parsed, but on the development of programs that must accurately send each sentence to adecuate specific parsers, when necessary. Also, some effects that could be given in this kind of work, such as overfitting, should be studied.

This work focused on Maltparser 0.4 and Spanish, but similar analyses could be accomplished to study other languages and/or parsers, complementing the present one.

Acknowledgments

This work has been partially funded by *Banco Santander Central Hispano* and *Universidad Complutense de Madrid* under the *Creación y Consolidación de Grupos de Investigación* program, Ref. 921332–953.

References

1. Buchholz, S., Marsi, E.: CoNLL–X Shared Task on Multilingual Dependency Parsing. In: Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL–X), pp. 149–164 (2006)
2. Ballesteros, M., Herrera, J., Francisco, V., Gervás, P.: Improving Parsing Accuracy for Spanish using Maltparser. Journal of the Spanish Society for Natural Language Processing (SEPLN) 44 (in press, 2010)
3. Herrera, J., Gervás, P.: Towards a Dependency Parser for Greek Using a Small Training Data Set. Journal of the Spanish Society for Natural Language Processing (SEPLN) 41, 29–36 (2008)
4. Herrera, J., Gervás, P., Moriano, P.J., Moreno, A., Romero, L.: Building Corpora for the Development of a Dependency Parser for Spanish Using Maltparser. Journal of the Spanish Society for Natural Language Processing (SEPLN) 39, 181–186 (2007)
5. Herrera, J., Gervás, P., Moriano, P.J., Moreno, A., Romero, L.: JBeaver: un Analizador de Dependencias para el Español Basado en Aprendizaje. In: Borrajo, D., Castillo, L., Corchado, J.M. (eds.) CAEPIA 2007. LNCS (LNAI), vol. 4788, pp. 211–220. Springer, Heidelberg (2007)
6. Nivre, J., Hall, J., Nilsson, J.: Memory–based Dependency Parsing. In: Proceedings of CoNLL–2004, Boston, MA, USA, pp. 49–56 (2004)
7. Eisner, J.: Three New Probabilistic Models for Dependency Parsing: An Exploration. In: Proceedings of the 16th International Conference on Computational Linguistics (COLING 1996), Copenhagen, pp. 340–345 (1996)
8. Yamada, H., Matsumoto, Y.: Statistical Dependency Analysis with Support Vector Machines. In: Proceedings of International Workshop of Parsing Technologies (IWPT 2003), pp. 195–206 (2003)
9. Palomar, M., Civit, M., Díaz, A., Moreno, L., Bisbal, E., Aranzabe, M., Ageno, A., Martí, M.A., Navarro, B.: 3LB: Construcción de una Base de Datos de Árboles Sintáctico–Semánticos para el Catalán, Euskera y Español. In: Proceedings of the XX Conference of the Spanish Society for Natural Language Processing (SEPLN), Sociedad Española para el Procesamiento del Lenguaje Natural, pp. 81–88 (2004)
10. Taulé, M., Martí, M., Recasens, M.: AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In: Proceedings of 6th International Conference on Language Resources and Evaluation (2008)
11. McDonald, R., Lerman, K., Pereira, F.: Multilingual Dependency Analysis with a Two-Stage Discriminative Parser. In: Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL–X), pp. 216–220 (2006)
12. Nivre, J., Hall, J., Nilsson, J., Eryiğit, G., Marinov, S.: Labeled Pseudo–Projective Dependency Parsing with Support Vector Machines. In: Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL–X), pp. 221–225 (2006)
13. Johansson, R., Nugues, P.: Investigating Multilingual Dependency Parsing. In: Proceedings of the Conference on Computational Natural Language Learning, CoNLL–X (2006)
14. Wu, Y., Lee, Y., Yang, J.: The Exploration of Deterministic and Efficient Dependency Parsing. In: Proceedings of the Conference on Computational Natural Language Learning, CoNLL–X (2006)