Nbest Dependency Parsing with linguistically rich models

Xiaodong Shi
Institute of Artificial Intelligence
Department of Computer Science
Xiamen University, Xiamen 361005
mandel@xmu.edu.cn

Yidong Chen
Institute of Artificial Intelligence
Department of Computer Science
Xiamen University, Xiamen 361005
ydchen@xmu.edu.cn

Abstract

We try to improve the classifier-based deterministic dependency parsing in two ways: by introducing a better search method based on a non-deterministic nbest algorithm and by devising a series of linguistically richer models. It is experimentally shown on a ConLL 2007 shared task that this results in a system with higher performance while still keeping it simple enough for an efficient implementation.

1 Introduction

This work tries to improve the deterministic dependency parsing paradigm introduced in (Covington 2001, Nivre 2003, Nivre and Hall, 2005) where parsing is performed incrementally in a strict left-to-right order and a machine learned classifier is used to predict deterministically the next parser action. Although this approach is very simple, it achieved the state-of-art parsing accuracy. However, there are still some problems that leave further room for improvement:

(1) A greedy algorithm without backtracking cannot ensure to find the optimal solution. In the course of left-to-right parsing, when further context is seen, the previous decisions may be wrong but a deterministic parser cannot correct it. The usual way of preventing early error "commitment" is to enable a k-best or beam-search strategy (Huang and Chiang 2005, Sagae and Lavie 2006).

(2) A classifier based approach (e.g. using SVM or memory based learning) is usually linguistically naïve, to make it applicable to multiple languages. However, a few studies (Collins 1999, Charniak et al 2003, Galley et al 2006) have shown that linguistically sophisticated models can have a better accuracy at parsing, language modeling, and machine translation, among others.

In this paper we explore ways to improve on the above-mentioned deterministic parsing model to overcome the two problems. The rest of the paper is organized as follows. Section 2 argues for a search strategy better at finding the optimal solution. In section 3 we built a series of linguistically richer models and show experimental results demonstrating their practical consequences. Finally we draw our conclusions and point out areas to be explored further.

2 Dependency Parsing Enhancements

In the classifier-based approach as in Nivre (2003) a parse tree is produced by a series of actions similar to a left-to-right shift-reduce parser. The main source of errors in this method is the irrevocability of the parsing action and a wrong decision can therefore lead to further inaccuracies in later stages. So it cannot usually handle garden-path sentences. Moreover, each action is usually predicted using only the local features of the words in a limited window, although dynamic features of the local context can be exploited (Carreras 2006).

To remedy this situation, we just add a scoring function and a priority queue which records nbest partial parses. The scoring function is defined on the parsing actions and the features of a partial parse. It can be decomposed into two subfunctions:

score(a,y)=parsing_cost(a,y) + lm(y)

where **a** is parsing actions and **y** is partial parses, and parsing cost (*parsing_cost*) is used to implement certain parsing preferences while the lingustic model score (*lm*) is usually modeled in the linguistic (in our case, dependency model) framework.

Proceedings of the 10th Conference on Parsing Technologies, pages 80–82,
Prague, Czech Republic, June 2007. ©2007 Association for Computational Linguistics

In the usual nbest or beam-search implementation (e.g. Huang and Chiang 2005, Sagae and Lavie 2006), only *lm* is present.

We give justification of the first term as follows: Many probability functions need to know the dependency label and relative distance between the dependent and the head. However, during parsing sometimes this head-binding can be very late. This means a right-headed word may need to wait very long for its *right* head, and so a big partial-parse queue is needed, while psychological evidence suggests that there is some overhead involved in processing every word and a word tends to attach locally. By modeling parsing cost we can first use a coarse probability model to guide the nbest partial results in order not to defer the probability calculation. As parsing progresses, more information becomes available; we can have a better estimation of our linguistic probability model to rectify the inaccuracy.

This use of a coarse scoring mechanism to guide the early parsing for possible later rectification of the decision is a novel feature of our parsing framework and enables better searching of the solution space. To implement it, we just remember the exact score of the every major decision (wait, add a dependent or attach a head) in parsing, and re-score when more context is available. Compared with (Charniak 2005), our parsing process requires only one pass.

Thus, we can strike a balance between accuracy, memory and speed. With a moderately-sized *n* (best partial results), we can reduce memory use and get higher speed to get a same accuracy. An added advantage is that this idea is also useful in other bottom-up parsing paradigms (not only in a dependency framework).

In a word, our main innovation is the use of a parsing cost to influence the search paths, and the use of an evolving *lm* function to enable progressively better modeling. The nbest framework is general enough to make this a very simple modification to the basic algorithm of Nivre (2003).

3 Better Linguistic Modeling

In our modeling we combine different linguistic models by using many probability functions:

$$lm(y) = \Sigma \log P(w_i, w_j, x, y) = \Sigma \mathbf{W} * \log \mathbf{P}$$

where **w** are the trained weight vector and **P** is a vector of probability functions. In our system we considered the following functions:

P1: function measuring the probability of a head and a dependent. This is the base function in most dependency parsing framework.

P2: function calculating the subcategorization frame probability;

P3: function calculating the semantic frame using a Chinese FrameNet (Liu 2006).

P4: function measuring the semantic affinity between a head and a dependent using resources such as Hownet (Dong 2003).

P5: Other Chinese specific probability functions defined on the features of the head, the dependents, the partial parse and the input.

Model P2 is a probability function on *pseudo* subcategorization frames (as a concatenation of all the dependents' labels) as we don't know the distinction of arguments and adjuncts in the dependency Treebank. We used a Markovian subcategorization scheme with left and right STOP delimiters to ease the data sparseness. And as a first approximation, we also experimented with a model where each label can only be used a certain times in a direction. This model is called P2' in Table 4.

Other functions (P3-P5) are also very useful with its different linguistic content. Model P5 actually contains a lot of Chinese-specific functions, e.g. between a sentence-final particle and a verb.

We designed a series of experiments to show to effectiveness of each model. We use the Chinese training data of the ConLL 2007 shared task. We divided the training data by a 9:1 split. Table 1 shows the statistics.

	Training	testing
sentences	51777	5180
Words	302943	34232

Table 1. Experimental data

In the baseline model, we train a simple probability function between a head and a dependent using deleted interpolation. For nbest=1, we have a deterministic model.

	LAS	UAS	time
Deterministic	41.64 %	44.11 %	8s
nbest = 50	71.30 %	76.34 %	72s
nbest = 500	71.90 %	76.99 %	827s

Table 2. baseline systems

It can be seen (Table 3) that combing different linguistic information can lead to significant increase of the accuracy. However, different models have different contributions. Our experiments confirm with Collins's result in that subcategorization carries very important linguistic content.

	LAS	UAS	time
P1	71.90 %	76.99 %	827s
P1 + P2'	73.45 %	78.44 %	832s
P1 + P2' + P2	77.92 %	82.42 %	855s
P1 + P2 + P3	79.13%	83.57%	1003s
P1-4	81.21%	85.78%	1597s
P1-5	83.12%	87.03%	2100s
Verb valency	85.32 %	89.12 %	-
DE refinement	85.98%	90.20%	-

Table 3. systems with different linguistic models

3.1 Relabeling of the parse treebank

Sometimes the information needed in the modeling is not in the data explicitly. Implicit information can be made explicit and accessible to the parser.

In the Chinese Treebank the relation label is often determined by the head word's semantic type. We tried the relabeling of coarse POS info of the verb in a effort to detect its valency; and refinement of the auxiliary word 的 DE (as error analysis shows it is the where the most errors occur). Results are in Table 3.

We also tried refinement of the relation label by using the two connected words. However, this does not improve the result. Automatic linguistic modeling using latent label (Matsuzaki 2005) can also be attempted but is not yet done.

4 Conclusions

In this paper we showed that simple classifier-based deterministic dependency parsing can be improved using a more flexible search strategy over an nbest parsing framework and a variety of linguistically richer models. By incorporating different linguistic knowledge, the parsing model can be made more accurate and thus achieves better results.

Further work to be done includes ways to combine machine learning based on the automatic feature selection with manual linguistic modeling: an interactive approach for better synergistic modeling (where the machine proposes and the human guides). Various a priori models can be tried by the machine and patterns inherent in the data can be revealed to the human who can then explore more complex models.

References

Xavier Carreras, Mihai Surdeanu, and Lluís Màrquez. 2006. *Projective Dependency Parsing with Perceptron*. In Proceedings of CoNLL-X. 181-185.

Liang Huang and David Chiang. 2005. Better *k*-best parsing. In *Proceedings of IWPT*.

Eugene Charniak; K. Knight, and K.Yamada. 2003. *Syntax-based language models for statistical machine translation*. In MT Summit IX. Intl. Assoc. for Machine Translation.

Eugene Charniak and Mark Johnson. 2005. *Coarse-tofine n-best parsing and maxent discriminative reranking*. In Proceedings of ACL.

Michael Collins. 1999. *Head-Driven Statistical Models-for Natural Language Parsing*. PhD Dissertation, University of Pennsylvania.

Michael Collins. 2004. Parameter Estimation for Statistical Parsing Models: Theory and Practice of Distribution-Free Methods. In Harry Bunt el al, *New Developments in Parsing Technology*, Kluwer.

Michael A. Covington. 2001. *A fundamental algorithm for dependency parsing*. Proceedings of the 39th Anual ACM Southeast Conference, pp. 95-102.

Zhendong Dong and Qiang Dong. 2003. *HowNet - a hybrid language and knowledge resource*. In Proceeding of Natural Language Processing and Knowledge Engineering.

M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer. 2006. *Scalable Inference and Training of Context-Rich Syntactic Models*. In Proc. ACL-COLING.

Kaiying Liu. 2006. *Building a Chinese FrameNet*. In Proceeding of 25th anniversary of Chinese Information Processing Society of China.

Takuya Matsuzaki, Yusuke Miyao, Jun'ichi Tsujii. 2005. *Probabilistic CFG with latent annotations*. In Proceedings of ACL-2005.

Joakim Nivre. 2003. *An efficient algorithm for projective dependency parsing*. In Proceedings of IWPT. 149-160.

Joakim Nivre and Johan Hall. 2005. *MaltParser: A Language-Independent System for Data-Driven Dependency Parsing*. In Proceedings of the Fourth Workshop on Treebanks and Linguistic. Theories, Barcelona, 9-10 December 2005. 137-148.

Sagae, K. and Lavie, A. 2006 *A best-first probabilistic shift-reduce parser*. In Proceedings of ACL.