Building a Large-Scale Annotated Chinese Corpus

Nianwen Xue
IRCS, University of Pennsylvania
Suite 400A, 3401 Walnut Street
Philadelphia, PA 19104, USA
xueniwen@linc.cis.upenn.edu

Fu-Dong Chiou and Martha Palmer
CIS, University of Pennsylvania
200 S 33rd Street
Philadelphia, PA 19104, USA
{chioufd,mpalmer}@linc.cis.upenn.edu

Abstract

In this paper we address issues related to building a large-scale Chinese corpus. We try to answer four questions: (i) how to speed up annotation, (ii) how to maintain high annotation quality, (iii) for what purposes is the corpus applicable, and finally (iv) what future work we anticipate.

Introduction

The Penn Chinese Treebank (CTB) is an ongoing project, with its objective being to create a segmented Chinese corpus annotated with POS tags and syntactic brackets. The first installment of the project (CTB-I) consists of Xinhua newswire between the years 1994 and 1998, totaling 100,000 words, fully segmented, POS-tagged and syntactically bracketed and it has been released to the public via the Penn Linguistic Data Consortium (LDC). The preliminary results of this phase of the project have been reported in Xia *et al* (2000). Currently the second installment of the project, the 400,000-word CTB-II is being developed and is expected to be completed early in the year 2003. CTB-II will follow the standards set up in the segmentation (Xia 2000b), POS tagging (Xia 2000a) and bracketing guidelines (Xue and Xia 2000) and it will use articles from Peoples' Daily, Hong Kong newswire and material translated into Chinese from other languages in addition to the Xinhua newswire used in CTB-I in an effort to diversify the sources.

The availability of CTB-I changed our approach to CTB-II considerably. Due to the existence of CTB-I, we were able to train new automatic Chinese language processing (CLP) tools, which crucially use annotated corpora as training material. These tools are then used for preprocessing in the development of the CTB-II. We also developed tools to control the quality of the corpus. In this paper, we will address three issues in the development of the Chinese Treebank: annotation speed, annotation accuracy and usability of the corpus. Specifically, we attempt to answer four questions: (i) how do we speed up the annotation process, (ii) how do we maintain high quality, i.e. annotation accuracy and inter-annotator consistency during the annotation process, and (iii) for what purposes is the corpus applicable, and (iv) what are our future plans? Although we will touch upon linguistic problems that are specific to Chinese, we believe these issues are general enough for the development of any single language corpus.

1 Annotation Speed

There are three main factors that affect the annotation speed : annotators' background, guideline design and more importantly, the availability of preprocessing tools. We will discuss how each of these three factors affects annotation speed.

1.1 Annotator Background

Even with the best sets of guidelines, it is important that annotators have received considerable training in linguistics, particularly in syntax. In both the segmentation/POS tagging phase and the syntactic bracketing phase, understanding the structure of the sentences is essential for correct annotation with reasonable speed. For example, 的/de is assigned two part-of-speech tags, depending on where it occurs in the sentence. It is tagged as DEC when it marks the end of the preceding modifying clause and DEG when it follows a nominal phrase. This

distinction is useful in that it marks different relations : between the nominal phrase and the noun head, and between the clause and the noun head respectively.

1.a. 负责人/NN 的/**DEG** 责任/NN
leader DE responsibility
'leader's responsibility'
b. 最近/NT 举行/VV 的/**DEC** 示威/NN 活动/NN
recently hold DE demonstration
'recently held demonstration'

During the bracketing phase, the modifying clause is further divided into relative clauses and complement (appositive) clauses. The structures of these two types of clauses are different, as illustrated in 2:

2.a. (NP (CP (WHNP-1 *OP*)
(CP (IP (NP-SBJ (-NONE- *T*-1))
(VP (NP-TMP 最近/NT) recently
(VP 举行/hold))) hold
的/**DEC**))
(NP 示威/NN 活动/NN)) demonstration
'recently held demonstration'

b. (NP (CP-APP (IP (NP-SBJ (-NONE- *pro*))
(VP (PP 对/P to
(NP 国家/NN)) nation
(VP 负责/VV))) responsible
的/**DEC**)
(NP 态度/NN)) attitude
'the attitude that one is responsible to the nation'

The annotator needs to make his/her own judgment as to whether the preceding constituent is a phrase or a clause. If it is a clause, he then needs to decide whether it is a complement clause or a relative clause. That is just one of the numerous places where he would have to draw upon his training in syntax in order to annotate the sentence correctly and efficiently. Although it is hard to quantify how the annotator's background can affect the annotation speed, it is safe to assume that basic training in syntax is very important for his performance.

1.2 How Guideline Design can Affect Annotation Speed

In addition to the annotator's background, the way the guidelines are designed also affects the annotation speed and accuracy. It is important to factor in how a particular decision in guideline design can affect the speed of the annotation. In general, the more complex a construction is, the more difficult and error-prone its annotation is. In contemporary theoretical linguistics the structure of a sentence can be very elaborate. The example in 3 shows how complicated the structure of a simple sentence "they seem to understand" can be. The pronoun "they" cyclically moves up in the hierarchy in three steps.

3. (TP (DP-1 they)
(T' (T-2 seem)
(VP (DP-3 *-1)
(V' (V *-2)
(TP (DP-4 *-3)
(T' (T to)
(VP (DP *-4)
(V' understand))))))))

However, such a representation is infeasible for annotation guidelines. Wherever possible, we try to simplify structures without loss of information. For example, in a raising construction, instead of introducing a trace in the subject position of the complement clause of the verb, we allow the verb to take another verb phrase as its complement. Information is not lost because raising verbs are the only verbs that take a verb phrase as their complement. The structure can be automatically expanded to the "linguistically correct" structure if necessary:

4.a. before simplification
(IP (NP-SBJ 负责人) leader
(VP 应该 should
(IP-OBJ (NP-SBJ *-1)
(VP 负责)))) responsible
b. after simplification
(IP (NP-SBJ 负责人) leader
(VP 应该 should
(VP 负责))) responsible
'Leaders should be responsible.'

In some cases, we have to leave some structures flat in order not to slow down our annotation speed. One such example is the annotation of noun phrases. It is very useful to mark which noun modifies which, but sometimes it is hard to decide because there is too much ambiguity. We decided against annotating the internal structure of noun phrases where they consist of a string of nouns:

5. (NP 工程/project 施工/construction 招投标/bidding 管理/management 办法/method)
'project construction bidding management method'

We believe decisions like these make our guidelines simple and easy to follow, without compromising the requirement to annotate the most important grammatical relations.

1.3 Speeding up Annotation with Automatic Tools

The availability of CTB-I makes it possible to train an increasingly more accurate set of CLP tools. When used as preprocessors, these tools substantially, and sometimes greatly, accelerated our annotation. We will briefly describe how we trained segmenters, taggers and parsers for use as preprocessors.

1.3.1 Machine Learning Approaches to Chinese Word Segmentation

Using the data from CTB-I, we trained an automatic word segmenter, using the maximum entropy approach. In general, machine learning approaches to Chinese word segmentation crucially hinge on the observation that word components (here we loosely define word components to be Chinese characters) can occur on the left, in the middle or on the right within a word. It would be a trivial exercise if a given character always occurs in one of these positions across all words, but in actuality, it can be ambiguous with regard to its position within a word. This ambiguity can be resolved by looking at the context, specifically the neighboring characters and the distribution of the previous characters (left, middle, or right). So the word segmentation problem can be modeled as an ambiguity resolution problem that readily lends itself to machine learning approaches. It should be pointed out that the ambiguity cannot be completely resolved just by looking at neighboring words. Sometimes syntactic context is also needed (Xue 2001). As a preliminary step, we just looked at the immediate contexts in our experiments.

In training our maximum entropy segmenter, we reformulated the segmentation problem as a tagging problem. Specifically, we tagged the characters as LL (left), RR (right), MM (middle) and LR (single-character word), based on their distribution within words. A character can have multiple tags if it occurs in different positions within different words.

6. 产/LL e.g. 产生 'to come up with'
产/LR e.g. 产 小麦 'to grow wheat'
产/MM e.g. 生产线 'assembly line'
产/RR e.g. 生产 'to produce'

The training data can be trivially derived from a manually segmented corpus.

7. a. 中国 科学家 发现 十 枚 鸟类 化石
b. 中/LL 国/RR 科/LL 学/MM 家/RR 发/LL 现/RR 十/LR 枚/LR 鸟/MM 类/RR 化/LL 石/RR
'Chinese scientists discovered ten pieces of bird fossil.'

Using 80,000 words from CTB-I as training data and the remaining 20,000 words as testing data, the maximum entropy segmenter achieved an accuracy of 91%, calculated by the F-measure, which combines precision and recall[1]. Compared with 'industrial strength' segmenters that have reported segmentation accuracy in the upper 90% range (Wu and Jiang 2000), this accuracy may seem to be relatively low. There are two reasons for this. First, the 'industrial strength' segmenters usually go through several steps (name identification, number identification, to name a few), which we did not do. Second,

[1] F-measure = (precision * recall * 2) / (precision + recall).

CTB-I is a relatively small corpus and we believe as we have more data available, we will be able to retrain our segmenters on more data and get increasingly more accurate segmenters. The more accurate segmenters in turn help speed up our annotation.

1.3.2 Training a POS Tagger

Unlike segmenters, a POS tagger is a standard tool for the processing of Indo-European languages where words are trivially identified by white spaces in text form. Once the sentences are segmented into words, Chinese POS taggers can be trained in a similar fashion as POS taggers for English. The contexts that are used to predict the part-of-speech tag are roughly the same in both Chinese and English. These are the surrounding words, the previous tags and word components. One notable difference is that Chinese words lack the rich prefix and suffix morphology in Indo-European languages that are generally good predictors for the part-of-speech of a word. Another difference is that words in Chinese are not as long as English words in terms of the number of characters or letters they have. Still, some characters are useful predictors for the part-of-speech of the words they are components of.

Our POS tagger is essentially the maximum entropy tagger by Ratnaparkhi (1996) retrained on the CTB-I data. We used the same 80,000 words chunk that was used to train the segmenter and used the remaining 20,000 words for testing. Our results show that the accuracy of this tagger is about 93% when tested on Chinese data. Considering the fact that our corpus is relatively small, this result is very promising. We expect that better accuracy will be achieved as more data become available.

The training and development of Chinese segmenters and taggers speeds up our annotation, and at the same time as more data are annotated we are able to train more accurate preprocessing tools. This is a bootstrapping cycle that helps both the annotation and the tools. The value of preprocessing in segmentation and POS tagging is substantial and these automatic tools turn annotation into an error-correction activity rather than annotation from scratch. From our estimate, correcting the output of a segmenter and a POS-tagger is nearly twice as fast as annotating the same data from scratch in the segmentation and POS-tagging phase.

The value of a parser as a preprocessing tool is less obvious, since when an error is made, the human annotator has to do considerable backtracking and undo some of the incorrect parses produced by the automatic parser. So we conducted an experiment and our results show that even with the apparent drawback of having to backtrack from the parses produced by the parser, the parser is still a useful preprocessing tool that helps annotation substantially. We will discuss this result in the next subsection.

1.3.3 Training a Statistical Parser

In order to determine the usefulness of the parser as a preprocessing tool, we used Chiang's parser (Chiang 2000), originally developed for English, which was retrained on data from CTB-I. We used 80,000 words of fully bracketed data for training and 10,000 words for testing. The parser obtains 73.9% labeled precision and 72.2% labeled recall. We then conducted an experiment to determine whether the use of a parser as a preprocessor improves annotation speed. We randomly selected a 13,469-word chunk of data form the corpus. This chunk was blindly divided into 2 portions of equal size (6,731 words for portion 1, 6,738 words for portion 2). The first portion was annotated from scratch. The second portion was first preprocessed by this parser and then an annotator corrected its output. The throughput rate was carefully recorded. In both cases, another annotator made a final pass over the first annotator's annotation, and discussed discrepancies with the first annotator. The adjudicated data was designated as the Gold Standard. This allows us to measure the "quality" of each portion in addition to the throughput rate. The experimental results are tabulated in 8:

8. Experimental results

Portion	Precision	Recall	Time	Accuracy
1	N/A	N/A	28h:01m	99.84%

2	76.73%	75.36%	16h:21m	99.76%

The results clearly show that using the parser as a preprocessor greatly reduces the time needed for the annotation (specifically, 42%), compared with the time spent on annotation from scratch. This suggests that even in the bracketing phrase, despite the need to backtrack sometimes, preprocessing can greatly benefit treebank annotation. In addition, the results show that the annotation accuracy remains roughly constant.

2 Quality Control

If the preprocessing tools give a substantial boost in our annotation speed, the use of evaluation tools, especially in the bracketing phase, helps us monitor the annotation accuracy and inter-annotator consistency, and thus the overall quality of the corpus. From our experience, we have learned that despite the best effort of human annotators, they are bound to make errors, especially mechanical errors due to oversight or fatigue. These mechanical errors often happen to be the errors automatic tools are good at detecting. In this section, we will describe how we monitor our annotation quality and the tools we used to detect errors.

2.1 Double Annotation and Evaluation

To monitor our annotation accuracy and inter-annotator consistency, we randomly selected 20% of the files to double annotate. That is, for these files, each annotator annotates them independently. The annotators meet weekly to compare those double annotated files. This is done in three steps: first, an evaluation tool[2] is run on each double annotated file to determine the inter-annotator consistency. Second, the annotators examine the results of the comparison and the inconsistencies detected by the evaluation tool. These inconsistencies are generally in the form of crossed brackets, extra brackets, wrong labels, etc.. The annotators examine the errors and decide on the correct annotation. Most of the errors are obvious and the annotators can agree on the correct annotation. In rare occasions, the errors can be due to misinterpretation of the guidelines, which is possible given the complexity of the syntactic constructions encountered in the corpus. Therefore the comparison is also an opportunity of continuing the training process for the annotators. After the inconsistencies are corrected or adjudicated, the corrected and adjudicated file are designated as the Gold Standard. The final step is to compare the Gold Standard against each annotator's annotation and determine each annotator's accuracy. Our results show that both measures are in the high 90% range, which is a very satisfactory result.

2.2 Post-annotation Checking with Automatic Tools

As a final quality control step, we run LexTract (Xia 2001) and a corpus search tool developed by Beth Randall[3]. These tools are generally very good at picking up mechanical errors made by the human annotator. For example, the tools detect errors such as missing brackets, wrong phrasal labels and wrong POS tags. If a phrasal label is not found in the bracketing guidelines, the tools will be able to detect it. The annotators will then manually fix the error. Using these tools allows us to fix the mechanical errors and get the data ready for the final release.

3 Usability

As we have discussed earlier, in order to finish this project in a reasonable time frame, some decisions have been made to simplify this phase of the project. In this section, we will briefly describe what has been achieved. We then try to anticipate future work on top of the current phase of the project

3.1 Current Annotation

As we have briefly mentioned in previous sections, the bracketing phase of this project focuses on the syntactic relationships between constituents. In our guidelines, we selected three

[2] The tool was written by Satoshi Sekine and Mike Collins. More information can be found at <www.cs.nyu.edu/cs/projects/proteus/evalb>

[3] <www.cis.upenn.edu/~brandall>

grammatical relations as the most basic, namely, complementation, adjunction and coordination. Each of these three grammatical relations is assigned a unique structure represented schematically as follows:

9. Hierarchical Representational Schemes

a. Complementation

head-initial	head-final
(XP X	(XP (YP)
(YP)	(ZP)
(ZP)	...
...)	X)

b. Adjunction:

Left adjunction	Right adjunction
(XP (YP)	(XP (XP)
(ZP)	...
...	(YP)
(XP))	(ZP))

c. Co-ordination:

(XP {CONJ}
(XP)
{CONJ}
(XP)
...)

Besides the hierarchical representations, functional tags are used to mark additional information. These functional tags can be regarded as secondary and are used to complement the hierarchical representations. For example, in Chinese, multiple noun phrases (labeled NP in the Chinese Treebank) can occur before the verb within a clause (or above the verb if seen hierarchically). Structurally, they are all above the verb. Therefore, they are further differentiated by secondary functional tags. Generally, an NP marked -SBJ (subject) is required. There can optionally be topics (marked by -TPC) and adjuncts (marked by -ADV, -TMP, etc.).

10. (IP (NP-PN-TPC 海尔	Haier
集团)	group
(NP-TMP 九十年代)	1980s
(PP-LOC 在	in
(NP 国内外))	country inside outside
(NP-SBJ 知名度)	recognition level
(VP (ADVP 很)	very
(VP 高)))	high

'In the 1990s, Haier Group is highly recognized both domestically and overseas. '

Similarly, multiple NPs can also occur after the verb and they can be marked as -OBJ (for object) or -EXT (basically a cover term for all other phrases that are not marked -OBJ). This representational scheme allows the identification of such basic grammatical relations as subject, object and adjuncts in the corpus, which can be used to train syntactic parsers. However, as we will discuss in the next section, it is not enough for other CLP tasks that require deeper annotation.

3.2 Future Annotation

The annotations provided during the bracketing phase may be enough for training syntactic parsers, but they are not sufficient for other CLP tools and applications. Among other things, there are at least two areas in which the Chinese treebank can be enhanced, that is, more fine-grained predicate-argument structure annotation and coreference annotation.

As we have discussed above, one pre-verb noun phrase is marked as subject with the -SBJ tag and one post-verb noun phrase can be marked as -OBJ. However, the subject and object in the Chinese Treebank are defined primarily in structural terms. The semantic relation between the subject and the verb is not uniform across all verbs, or even for different instances of the same verb. The same is true for the relation between the object and the verb. For some verbs, there are systematic alternations between the subject and the verb, with the same NP occurring in the subject position in one sentence but in the object position in another, with the thematic role it assumes remaining constant.

11. a. (IP (NP-SBJ 新年	New Year
招待会)	reception
(VP (NP-TMP 今天)	today
(PP-LOC 在	at
(NP-PN 钓鱼台	Diaoyutai
国宾馆))	Hotel
(VP 举行))	hold

'New Year reception was held in Diaoyutai Hotel today.'

b. (IP (NP-PN-SBJ 唐家璇) Tang Jiaxuan
(VP (NP-TMP 今晚) tonight
(PP-LOC 在 at
(NP-PN 钓鱼台 Diaoyutai
国宾馆)) hotel
(VP 举行 hold
(NP-OBJ 新年 New Year
招待会)))) reception
'Tang Jiaxuan held a New Year reception at Diaoyutai Hotel tonight.'

In 11, 新年 招待会 ("New Year reception") is the subject in 11a while it is the object in 11b. However, in both cases, it is the theme. This may be problematic for some tools and applications. For an information extraction task, for example, if one wants to find all events held at a hotel, it is not enough to just look for the object in the parse tree, one also needs to know what thematic role the noun phrase assumes. One might also want to extract information from sentences with pronouns. We believe predicate-argument structure annotation and coreference annotation will be useful enhancements to this corpus and we will explore these possibilities.

Summary

In this paper we have shown that the use of annotation tools, not only for segmentation and POS tagging, but also for syntactic bracketing, can speed up the annotation process. We have also discussed how to ensure the quality of the corpus. We believe these methods are generalizable to the development of copora in other languages.

Acknowledgements

This research was funded by DOD MDA902-97-C-0307 and DARPA N66001-00-1-8915. We gratefully acknowledge comments from Susan Converse and two anonymous reviewers.

References

David Chiang. 2000. Statistical parsing with an automatically-extracted tree adjoining grammar. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 456-463, Hong Kong, 2000

Fu-Dong Chiou, David Chiang, and Martha Palmer. 2001. Facilitating Treebank Annotation with a Statistical Parser. In *Proc. of the Human Language Technology Conference* (HLT-2001), San Diego, CA.

Adwait Ratnaparkhi. A Maximum Entropy Part-Of-Speech Tagger. 1996. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*, May 17-18, 1996. University of Pennsylvania.

Andi Wu and Zixin Jiang. 2000. Statistically Enhanced New Word Identification in a Rule-Based Chinese System. In *Proceedings of the Second Chinese Language Processing Workshop (in conjunction with ACL)*, HKUST, Hong Kong, p46-51.

Fei Xia. 2000a. *The Part-of-speech Guidelines for the Penn Chinese Treebank Project*. Technical Report IRCS 00-06, University of Pennsylvania.

Fei Xia. 2000b. *The Segmentation Guidelines for the Penn Chinese Treebank Project*. Technical Report IRCS 00-07, University of Pennsylvania.

Fei Xia. 2001. *Automatic Grammar Generation from Two Different Perspectives*. PhD dissertation, University of Pennsylvania.

Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Okurowski, John Kovarik, Fu-Dong Chiou, Shizhe Huang, Tony Kroch, Mitch Marcus. 2000. Developing Guidelines and Ensuring Consistency for Chinese Text Annotation. In *Proc. of the 2nd International Conference on Language Resources and Evaluation (LREC-2000)*, Athens, Greece.

Nianwen Xue and Fei Xia. 2000. *The Bracketing Guidelines for the Penn Chinese Treebank* Project. Technical Report IRCS 00-08, University of Pennsylvania.

Nianwen Xue. 2001. *Defining and Automatically Identifying Words in Chinese*. PhD Dissertation, University of Delaware, 2001.