A corrigendum to Sun and Jurafsky (2004) “Shallow Semantic Parsing of Chinese”

TR-CSLR-2005-01

Ying Chen, University of Colorado

Honglin Sun, Brandeis University

Dan Jurafsky, Stanford University

June 2005

Abstract: Sun and Jurafsky (2004) “Shallow Semantic Parsing of Chinese” reported inflated numbers for Chinese syntactic and semantic parsing because of accidentally using hand-corrected part-of-speech tags instead of automatic tags. This technical report reruns the experiments reported in that paper, reporting the correct, non-inflated numbers. The corrected results show a syntactic parsing F1-score of 81.15 on the 100K words Chinese Treebank 1.0 test set, rather than 85.9 as reported in Sun and Jurafsky (2004). The semantic parsing performance F-score is 72.46 rather than 76.7. The main results of S+J (2004), which discuss the interesting difference between semantic parsing in English and Chinese, and the role of Chinese strict ordering of adverbials, remain unaffected by these changes. But an implication of our results (one which confused us mightily at the time) that the Collins parser worked much better than the Bikel implementation of the Collins algorithm, has now been shown to be a mistake.

1. Syntactic parsing

Sun and Jurafsky (2004) reported on Chinese semantic parsing experiments that involved as a component a port of the Collins syntactic parser (Collins 1999) to Chinese. The experiments reported in that paper accidentally used the gold-standard part-of-speech (POS) tags from the test set, rather than using the output of a part-of-speech tagger. Because the POS tags were taken from the gold-standard tagging of the test-set itself, the performance scores reported were inflated, and the comparison with other parsers was invalid. The goal of the experiments in this paper is to repeat the Collins’ parser experiments from Sun and Jurafsky (2004) with more realistic parts-of-speech.

1.1 An SVM-based part-of-speech tagger for Chinese.

Because the errors in Sun and Jurafsky (2004) were based on using gold-standard rather than automatic part-of-speech tagging, we begin by describing the part-of-speech tagger we used to replace their incorrectly-used gold standard tags with automatic ones. We use a standard

SVM-based tagger that we have used for other purposes in our lab.

In the Collins parsing model, words are considered ‘unknown’ if they were never seen in the training data, or if they occurred in the training data less often than a threshold. For our experiments in Chinese, for example, we set this threshold to 3. Thus for the purposes of this paper, we refer to words that never occur in the training data as *truly unknown*, words that occur in the training data with count < 3 as *low frequency*. All other words we refer to as *high frequency*. (For details on the Collins model, see Collins (1999) and Bikel (2004).)

For this paper a POS tagger was used based on one-versus-all SVM classifiers based on Yamcha (Kudo and Matusomoto 2001). For each example to be classified, the SVM uses the following features: contains-date, contains-number, contains-ordinal, contains-proper-name and the preceding and following two characters. All of the “contains” features work by determining if a word contains a particular character. For example, the date feature assigns a 1 value if the word includes the characters in the date character list, such as 年(year), 月(month), 日(day), and a 0 value otherwise. The other “contains” features work similarly. The word features include the text of the given word. The tagger was trained on a randomly sampled 45% of the Penn Chinese Treebank Release 4, and tested on a disjoint 10% of this data. The resulting accuracy on this dataset was 92%.

The model presented here handles POS tagging by first running this SVM tagger and then applying this information in one of three ways, depending on the word frequency:

High frequency words: The SVM tagger output is ignored and the first POS in the lexicon created from the training data is assigned to the word (this is just a dummy, since in this case the Collins parser will retag the word).

Low frequency words: We use the parser to choose among the possible tags, by creating multiple copies of the test sentence, one with each POS each word had in the training data, and allowing the parser to choose. Consider as a pedagogical example the English sentence:

(1) Ben watched the sky for a while.

Assume that all words but *watch* are high frequency words and have only one POS, and that that *watch* is a low frequency word that has been seen with two POS tags in the training data: VV and NN. The sentences produced would then be:

(2) Ben NR watched VV the DT sky NN for PP a DT while NN
(3) Ben NR watched NN the DT sky NN for PP a DT while NN

Both these sentences would be passed to the parser, and the POS sequence associated with the tree assigned a higher probability by the parser is chosen as the final result.

Truly unknown words: The POS assigned by the SVM tagger is assigned to the word.

The original Collins' parser code varied in its assignment of the low frequency or high frequency label depending on the order of the words in the training data. If a word with multiple parts-of-speech had one high-frequency POS and one low-frequency POS, the word was labeled high-frequency only if the high-frequency type appeared last in the training data. For example, if the word *watch* were added to the lexicon first as a noun and then as a verb (with the noun form being low-frequency), e.g.

watch NN LOW_FREQUENCY
watch VV HIGH_FREQUENCY

then *watch* would be considered a high-frequency word. If the "watch" examples were added in the opposite order, however, e.g.,

watch VV HIGH_FREQUENCY
watch NN LOW_FREQUENCY

then *watch* would be treated as a low-frequency word. For the experiments of this paper, the parser was modified so that any occurrence of a word with the high-frequency label would cause that word to be considered high frequency. So in both of the examples above, *watch* would now be considered a high-frequency word.

Sun and Jurafsky (2004) and the current paper both use Collins' original evaluation algorithm.

1.2 Re-running the Sun and Jurafsky (2004) results, using the SVM automatic tags

This section reports the comparison between the performance of the Collins' parser using gold-standard POS tags, reported by S+J (2004), and the new correct performance using automatic tags (produced as described in section 1.1 above), in other words without using gold-standard POS tags.

We first summarize the original incorrect results from S+J (2004). Table 1 repeats S+J's (2004) Table 7. In this experiment, the test set is the same 113 sentence set as used for the semantic parsing experiments in S+J (2004), and the training data is the 250K word Penn Chinese Treebank (CTB) Release 2, excluding the test set. Table 2 repeats S+J's (2004) Table 8. In this experiment, articles 1-270 are used for training and articles 271-300 are used for testing.

	LP	LR	F
All sentences	81.6	82.1	81.0
Len <= 40 words	86.1	85.5	86.7

Table 1. Repeat of Sun and Jurafsky (2004) Table 7 ("Results for syntactic parsing, trained on CTB Release 2, tested on the test set used in semantic parsing")

We first note that S+J's (2004) Table 7 had typos in it; Table 1b below corrects these typos.

	LP	LR	F
All sentences	82.1	81.1	81.6
Len <= 40 words	85.5	86.7	86.1

Table 1b. Correction of typos in Sun and Jurafsky (2004) Table 7 ("Results for syntactic parsing, trained on CTB Release 2, tested on the test set used in semantic parsing")

	LP	LR	F
Len <= 40 words	86.4	85.5	85.9

Table 2. Repeat of final line of Sun and Jurafsky (2004) Table 8: "Comparison with other parser: TEST2"

Tables 3 and 4 then show the replacement for tables 1b and 2 when we correctly use the SVM POS tagger rather than the gold standard tags.

	LP	LR	F
All sentences	78.85	78.88	78.86
Len <= 40 words	83.84	83.66	83.75

Table 3. Rerunning of Sun and Jurafsky (2004) Table 7 ("Results for syntactic parsing, trained on CTB Release 2, tested on the test set used in semantic parsing") using the SVM POS tagger as described in section 1.1 rather than the gold standard POS tags.

	LP	LR	F
All sentences	79.17	76.30	77.71
Len <= 40 words	81.74	81.03	81.15

Table 4. Rerunning of final line of Sun and Jurafsky (2004) Table 8: "Comparison with other parser: TEST2" using the SVM POS tagger as described in section 1.1 rather than the gold standard POS tags.

Note that syntactic parsing performance drops from an F-score of 81.6 to 78.86 on the entire test set as described in Table 3, and from an F-score of to 85.9 to 81.15 on the sentences with length <= 40 in the test set described in Table 4.

In summary, Table 4b shows the complete corrected version of Sun and Jurafsky (2004) Table 8, giving the comparison to other parsers.

	LP	LR	F
Bikel and Chiang 2000	77.2	76.2	76.7
Chiang and Bikel 2002	81.1	78.8	79.9
Levy & Manning 2003	78.4	79.2	78.8
Collins parser	81.74	81.03	81.15

Table 4b. Replacing Sun and Jurafsky (2004) Table 8: "Comparison with other parser: TEST2" for sentences with Len <= 40 words, using the SVM POS tagger as described in section 1.1 rather than the gold standard POS tags.

2. Semantic parsing

Since the ported Collins parser described in section 1 was used as an input to the semantic parsing algorithm in S+J (2004), the semantic parsing results described in that paper were also inflated. This section describes our corrected results, in which we replace the "cheating" syntactic parse trees that used the gold-standard-pos with the corrected syntactic parse trees from the non-cheating algorithm described in section 1.

In the 113 sentences in the test set, 4 sentences are labeled with incorrect POS tags. Since the semantic parsing algorithm begins by finding a target verb, these 4 sentences would completely fail in semantic parsing. We therefore separately report performance on sentences with correct POS tags. The first row in Table 6 presents the results on the 109 sentences in which the target verb was assigned correct POS, and the second row presents the result on all 113 sentences, considering the 4 sentences with incorrect POS tags for target verbs as wrong.

	LP	LR	F
110 sentences	86.0	70.8	77.6
113 sentences	86.0	69.2	76.7

Table 5. Repeat of S+J (2004) Table 9: "Result for semantic parsing using automatic syntactic parses"

	LP	LR	F
109 sentences	80.66	67.86	73.71
113 sentences	80.66	65.77	72.46

Table 6. Rerunning S+J (2004) Table 9: "Result for semantic parsing using automatic syntactic parses" using the corrected syntactic parser described in section 1 that did not use the gold standard POS tags.

3. Conclusion

Sun and Jurafsky (2004) “Shallow Semantic Parsing of Chinese” reported inflated numbers for Chinese syntactic and semantic parsing because of accidentally using hand-corrected part-of-speech tags instead of automatic tags. We reran the experiments reported in that paper, and have reported the correct, non-inflated numbers. The corrected results show a syntactic parsing F1-score of 81.15 on the 100K words Chinese Treebank 1.0 test set, rather than 85.9 as reported in Sun and Jurafsky (2004). The semantic parsing performance F-score is 72.46 rather than 76.7. The main results of S+J (2004), which discuss the interesting difference between semantic parsing in English and Chinese, and the role of Chinese strict ordering of adverbials, remain unaffected by these changes. But an implication of our results (one which confused us mightily at the time) that the Collins parser worked much better than the Bikel implementation of the Collins algorithm, has now been shown to be a mistake.

4. References

Bikel, Daniel M. (2004). Intricacies of Collins’ Parsing Model. Computational Linguistics 30:4, 479-512

Collins, Michael (1999). Head-driven Statistical Models for Natural Language Parsing. Unpublished PhD. Dissertation, University of Pennsylvania.

Taku Kudo, Yuji Matsumoto (2001). Chunking with Support Vector Machine. Proceedings of NAACL 2001.

Sun and Jurafsky (2004). Shallow Semantic Parsing of Chinese. Proceedings of NAACL 2004.