In: *Proceedings of CoNLL-2000 and LLL-2000*, pages 148–150, Lisbon, Portugal, 2000.

Improving Chunking by Means of Lexical-Contextual Information in Statistical Language Models

Ferran Pla and **Antonio Molina** and **Natividad Prieto**
Universitat Politècnica de València
Camí de Vera s/n
46020 València (Spain)
{fpla, amolina, nprieto}@dsic.upv.es

1 Introduction

In this work, we present a stochastic approach to shallow parsing. Most of the current approaches to shallow parsing have a common characteristic: they take the sequence of lexical tags proposed by a POS tagger as input for the chunking process. Our system produces tagging and chunking in a single process using an Integrated Language Model (ILM) formalized as Markov Models. This model integrates several knowledge sources: lexical probabilities, a contextual Language Model (LM) for every chunk, and a contextual LM for the sentences. We have extended the ILM by adding lexical information to the contextual LMs. We have applied this approach to the CoNLL-2000 shared task improving the performance of the chunker.

2 Overview of the system

The baseline system described in (Pla et al., 2000a) uses bigrams, formalized as finite-state automata. It is a transducer composed of two levels (see Figure 1). The upper one (Figure 1a) represents the contextual LM for the sentences. The symbols associated to the states are POS tags (C_i) and chunk descriptors (S_i). The lower one modelizes the different chunks considered (Figure 1b). In this case, the symbols are the POS tags (C_i) that belong to the corresponding chunk (S_i). Next, a regular substitution of the lower models into the upper level is made (Figure 1c). In this way, we get a single Integrated LM which shows the possible concatenations of lexical tags and chunks. Also, each state is relabeled with a tuple (C_i, S_j) where $C_i \in \mathcal{C}$ and $S_j \in \mathcal{S}$. $\mathcal{C}$ is the POS tag set used and $\mathcal{S} = \{[S_i,\ S_i], S_i, S_0\}$ is the chunk set defined. $[S_i$ and $S_i]$ stand for the initial and the final state of chunk whose descriptor is S_i. The label S_i is assigned to those states which are inside S_i chunk, and S_0 is assigned to those states which are outside of any chunk. All the LMs involved have been smoothed by using a back-off technique (Katz, 1987). We have not specified lexical probabilities in every state of the different contextual models. We assumed that $P(W_j|(C_i, S_i)) = P(W_j|C_i)$ for every $S_i \in \mathcal{S}$.

Once the integrated transducer has been made, the tagging and shallow parsing process consists of finding the sequence of states of maximum probability on it for an input sentence. Therefore, this sequence must be compatible with the contextual, syntactical and lexical constraints. This process can be carried out by dynamic programming using the Viterbi algorithm (Viterbi, 1967), which has been appropriately modified to use our models. From the dynamic programming trellis, we can obtain the maximum probability path for the input sentence through the model, and thus the best sequence of lexical tags and the best segmentation in chunks, in a single process.

3 Specialized Contextual Language Models

The contextual model for the sentences and the models for chunks (and, therefore, the ILM) can be modified taking into account certain words in the context where they appear. This specialization us allows to set certain contextual constraints which modify the contextual LMs and improve the performance of the chunker (as shown below). This set of words can be defined using some heuristics such as: the most frequent words in the training corpus, the words with a higher tagging error rate, the words that belong to closed classes (prepositions, pronouns, etc.), or whatever word chosen following some linguis-

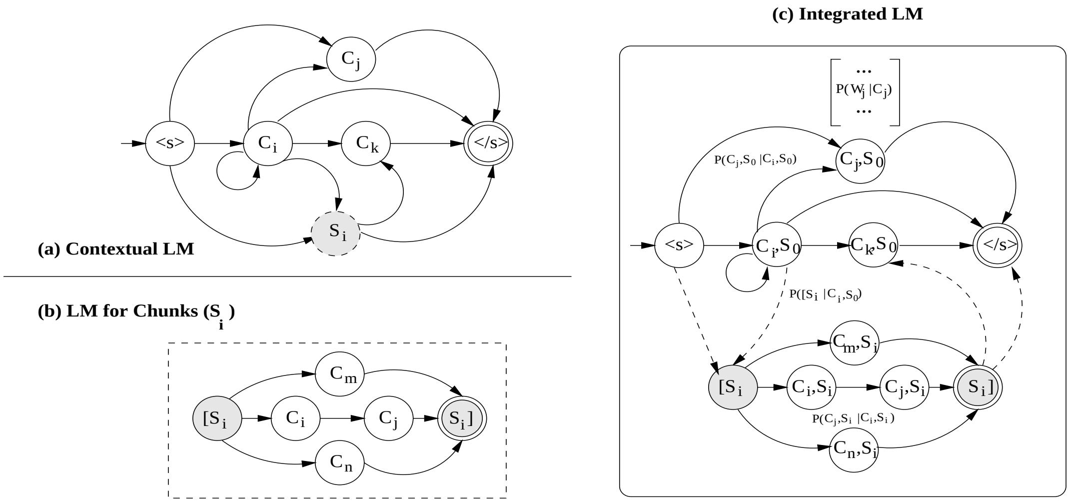

Figure 1: Integrated Language Model for Tagging and Chunking.

tic criterion.

To do this, we added to the POS tag set the set of structural tags (W_i, C_j) for each specialized word W_i in all of their possible categories C_j. Then, we relabelled the training corpus: if a word W_i was labelled with the POS tag C_j, we changed C_j for the pair (W_i, C_j). The learning process of the bigram LMs was carried out from this new training data set.

The Contextual LMs obtained has some specific states which are related to the specialized words. In the basic Language Model (ILM), a state was labelled by (C_i, S_j). In the specialized ILM, a state was specified for a certain word W_k (only if the W_k word belongs to the category C_i). In this way, the state is relabelled with the tuple (W_k, C_i, S_j) and only the word W_k can be emitted with a probability equal to 1.

4 Experimental Work

We applied both approaches (ILM and specialized ILM) using the training and test data of the CoNLL-2000 shared task (*http://lcg-www.uia.ac.be/conll2000*). We also evaluated how the performance of the chunker varies when we modify the specialized word set. Nevertheless, the use of our approach on other corpora (including different languages), other lexical tag sets or other kinds of chunks can be done in a direct way.

Although our system is able to carry out tagging and chunking in a single process, we will not present tagging results for this task, as the POS tags of the data set used are not supervised and, therefore, a comparison is not possible.

We would like to point out that we have simulated a morphological analyzer for English. We have constructed a tag dictionary with the lexicon of the training set and the test set used. This dictionary gave us the possible lexical tags for each word from the corpus. In no case, was the test used to estimate the lexical probabilities.

As stated above, several criterion can be chosen to define the set of specialized words. We have selected the most frequent words in the training data set. We have not taken into account certain words such as punctuation symbols, proper nouns, numbers, etc. This fact did not decrease the performance of the chunker and also reduced the number of states of the contextual LMs. Figure 2 shows how the performance of the chunker ($F_{\beta=1}$) improves as a function of the size of the specialized word set. The best results were obtained with the set of words whose frequency in the training corpus was larger than 80 (about 470 words). We obtained similar results when only considering the words of the training set belonging to closed classes (*that, about, as, if, out, while, whether, for, to, ...*).

In Table 1 we present the results of chunking with the specialized ILM. When comparing these results with the results obtained using the basic ILM, we observed that, in general, the F-score was improved for each chunk. The best improvement was observed for SBAR (from 0.37 to 79.46), PP (from 88.94 to 95.51) and PRT (38.82 to 66.67).

5 Conclusions

In this paper, we have presented a system for Tagging and Chunking based on an Integrated Language Model that uses a homogeneous formalism (finite-state machine) to combine different knowledge sources. It is feasible both in terms of performance and also in terms of computational efficiency.

All the models involved are learnt automatically from data, so the system is very flexible with changes in the reference language, changes in POS tags or changes in the definition of chunks.

Our approach allows us to use any regular model which has been previously defined or learnt. In previous works, we have used bigrams (Pla et al., 2000a), and we have combined them with other more complex models which had been learnt using grammatical inference techniques (Pla et al., 2000b). In this work, we used only bigram models improved with lexical-contextual information.

The F_β score obtained increased from 86.64 to 90.14 when we used the specialized ILM. Never-

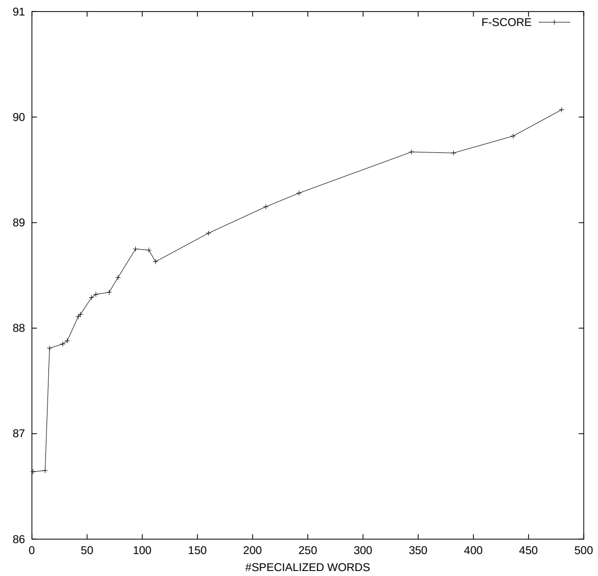

Figure 2: F-score as a function of the number of specialized words in the ILM

test data	precision	recall	$F_{\beta=1}$
ADJP	72.89 %	66.89 %	69.76
ADVP	79.65%	74.13%	76.79
CONJP	40.00%	66.67%	50.00
INTJ	100.00%	100.00%	100.00
LST	0.00%	0.00%	0.00
NP	90.28%	89.41%	89.84
PP	95.89%	95.14%	95.51
PRT	60.31%	74.53%	66.67
SBAR	82.07%	77.01%	79.46
VP	91.53%	91.58%	91.55
all	90.63%	89.65%	90.14

Table 1: Chunking results using specialized ILM (Accuracy= 93.79%)

theless, we believe that the models could be improved with a more detailed study of the words whose contextual information is really relevant to tagging and chunking.

6 Acknowledgments

This work has been partially supported by the Spanish Research Project CICYT (TIC97-0671-C02-01/02).

References

S. M. Katz. 1987. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35.

F. Pla, A. Molina, and N. Prieto. 2000a. Tagging and Chunking with Bigrams. In *Proceedings of the COLING–2000*, Saarbrücken, Germany, August.

F. Pla, A. Molina, and N. Prieto. 2000b. An Integrated Statistical Model for Tagging and Chunking Unrestricted Text. In *Proceedings of the Text, Speech and Dialogue 2000*, Brno, Czech Republic, September.

A. J. Viterbi. 1967. Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm. *IEEE Transactions on Information Theory*, pages 260–269, April.