and QP from the rule, and then the rule is modified as follows:

(NP (r NN NT NR) (r))

Additionally, the different bigrams formed by combining the first word (or POS) and last word (or POS) located in two adjacent chunks can also capture some correlations between adjacent chunks, and templates 17-22 are designed to express this kind of bigram information.

ID	*Feature template*
1	$wlabel(w)\ t_0$ for all w in c_0
2	$bigram\ (w)\ label(w) t_0$ for all w in c_0
3	$biPOS(w)\ label(w) t_0$ for all w in c_0
4	$w_{-1} w_1 label(w_0)\ t_0$, where $len(c_0)$=1
5	$start\_word(c_0) t_0$
6	$start\_POS(c_0) t_0$
7	$end\_word(c_0) t_0$
8	$end\_POS(c_0) t_0$
9	$wend\_word\ (c_0)\ t_0$ where $w \in c_0$ and $w \neq end\_word(c_0)$
10	$pend\_POS\ (c_0)\ t_0$ where $p \in c_0$ and $p \neq end\_POS(c_0)$
11	$internalPOSs(c_0)\ t_0$
12	$internalWords(c_0)\ t_0$
13	$specitermMatch(c_0)$
14	$t_{-1} t_0$
15	$head(c_{-1}) t_{-1} head(c_0) t_0$
16	$headPOS(c_{-1}) t_{-1} headPOS(c_0) t_0$
17	$end\_word(c_{-1}) t_{-1} start\_word(c_0) t_0$
18	$end\_POS(c_{-1}) t_{-1} start\_POS(c_0) t_0$
19	$end\_word(c_{-1}) t_{-1} end\_word(c_0) t_0$
20	$end\_POS(c_{-1}) t_{-1} end\_POS(c_0) t_0$
21	$start\_word(c_{-1}) t_{-1} start\_word(c_0) t_0$
22	$start\_POS(c_{-1}) t_{-1} start\_POS(c_0) t_0$
23	$end\_word(c_{-1}) t_0$
24	$end\_POS(c_{-1}) t_0$
25	$t_{-1} t_0 start\_word(c_0)$
26	$t_{-1} t_0 start\_POS(c_0)$
27	$internalWords(c_{-1})\ t_{-1}\ internalWords(c_0)\ t_0$
28	$internalPOSs(c_{-1})\ t_{-1}\ internalPOSs(c_0)\ t_0$

Table 1: Feature templates.

6 Experiments

6.1 Data Sets and Evaluation

Following previous studies on Chinese chunking in (Chen et al., 2006), our experiments were performed on the CTB4 dataset. The dataset consists of 838 files. In the experiments, we used the first 728 files (FID from chtb 001.fid to chtb 899.fid) as training data, and the other 110 files (FID from chtb 900.fid to chtb 1078.fid) as testing data. The training set consists of 9878 sentences, and the test set consists of 5920 sentences. The standard evaluation metrics for this task are precision p (the fraction of output chunks matching the reference chunks), recall r (the fraction of reference chunks returned), and the F-measure given by $F = 2pr/(p + r)$.

Our model has two tunable parameters: the number of training iterations N; the number of top *k-best* outputs. Since we were interested in finding an effective feature representation at chunk-level for phrase chunking, we fixed $N = 10$ and $k = 5$ for all experiments. In the following experiments, our model has roughly comparable training time to the sequence labeling approach based on CRFs.

6.2 Chinese NP chunking

NP is the most important phrase in Chinese chunking and about 47% phrases in the CTB4 Corpus are NPs. In this section, we present the results of our approach to NP recognition.

Table 2 shows the results of the two systems using the same feature representations as defined in Table 1, but using different loss functions for learning. As shown, learning with F1 loss can improve the F-score by 0.34% over learning with 0-1 loss. It is reasonable that the model optimized with respect to evaluation metrics directly can achieve higher performance.

Loss Function	Precision	Recall	F1
0-1 loss	91.39	90.93	91.16
F1 loss	**92.03**	**90.98**	**91.50**

Table 2: Experimental results on Chinese NP chunking.

6.3 Chinese Text Chunking

There are 12 different types of phrases in the chunking corpus. Table 3 shows the results from