Recognizing NPs with an Ensemble of Classifiers

Erik Tjong Kim Sang
Center for Dutch Language and Speech
University of Antwerp
erikt@uia.ua.ac.be

Goal

Finding a better learning method for recognizing noun phrases (NPs).

Inspiration

The project Learning Computational Grammars in which seven European sites apply machine learning methods for NP recognition.

[HZD98] which shows that POS tagger performance can be improved by combining the output of different systems.

Recognizing NPs

There are two variants of this task. The first consists of recognizing non-recursive NPs (baseNPs):

In [early trading] in [Hong Kong]
[Monday] , [gold] was quoted at
[$ 366.50] [an ounce] .

The second consists of recognizing arbitrary NPs:

In [early trading] in [Hong Kong]
[Monday] , [gold] was quoted at
[[$ 366.50] [an ounce]] .

A recognizer for the second variant can work bottom-up.

Classifier combination

Suppose we use five learning algorithms for predicting whether an NP starts at a certain position or not.

	c_1	c_2	c_3	c_4	c_5	correct
word_1	.	.	.	.	.	.
word_2	[	[	[	[	[	[
word_3	.	.	.	.	.	.
word_4	[	.	[	[	[	[
word_5	.	.	[	.	.	.
word_6	[	[	[	[	.	[
word_7	[	.	.	.	.	.
word_8	[	[	[	.	[	[

We can combine the results with majority voting: choose the result that has been predicted most often.

Obtaining different classifiers

How do we obtain results for different classifiers?

1. Use different learning algorithms.

   Disadvantage: most algorithms require a lot of tuning in order to get a reasonable performance.

2. Use one learning algorithm with different parameters or with different versions of the data.

   Disadvantage: the errors made by these systems are more related and therefore there is fewer improvement to gain.

We have chosen the second method and have applied one learning algorithm to five different NP representations.

Different NP representations

	IOB1	IOB2	IOE1	IOE2	O	C
In	O	O	O	O	.	.
early	I	B	I	I	[	.
trading	I	I	I	E	.	]
in	O	O	O	O	.	.
Hong	I	B	I	I	[	.
Kong	I	I	E	E	.	]
Monday	B	B	I	E	[	]
,	O	O	O	O	.	.
gold	I	B	I	E	[	]
was	O	O	O	O	.	.
quoted	O	O	O	O	.	.
at	O	O	O	O	.	.
$	I	B	I	I	[	.
366.50	I	I	E	E	.	]
an	B	B	I	I	[	.
ounce	I	I	I	E	.	]
.	O	O	O	O	.	.

Memory-Based Learning

What classification should items 7 and 8 get based on the classifications of items 1 to 6?

	F_1	F_2	F_3	F_4	
1.	by	Arizona	banking	NNP	I
2.	Colorado	,	Arizona	,	I
3.	its	Arizona	real	NNP	I
4.	lenders	across	the	IN	O
5.	lenders	got	hit	VBD	O
6.	the	Arizona	real	NNP	I
7.	lenders	against	Arizona	IN	?
8.	against	Arizona	real	NNP	?

F_1: word left to focus word
F_2: focus word
F_3: word right to focus word
F_4: POS tag for focus word

Erik Tjong Kim Sang 10/12/1999

Combination methods

We compare five different voting methods and four versions of stacked classifiers:

Majority voting
Each classifier gets one vote. The majority wins.

TotPrecision, TagPrecision, TagFB1
The weight of each classifier is determined by its performance on some held-out part of the training data.

TagPair
Uses weights for results which are associated with results of classifier pairs.

Stacked classifiers
A second classifier processes the results and determines the most probable classification.

Results

section 21	O	C	$F_{\beta=1}$
Representation			
IOB1	97.81	97.97	91.68
IOB2	97.63	97.96	91.79
IOE1	97.80	97.92	91.54
IOE2	97.72	97.94	92.06
O+C	97.72	98.04	92.03
Simple Voting			
Majority	98.04	98.19	92.82
TotPrecision	98.04	98.19	92.82
TagPrecision	98.04	98.19	92.82
TagFB1	98.04	98.19	92.82
Pairwise Voting			
TagPair	98.04	98.19	92.82
Memory-Based			
Tags	97.90	98.16	92.67
Tags + POS	97.71	98.15	92.29
Decision Trees			
Tags	97.90	98.16	92.67
Tags + POS	97.73	98.15	92.32

Results standard baseNP data sets

section 20	precision	recall	$F_{\beta=1}$
Majority	93.63%	92.89%	93.26
[MPRZ99]	92.4%	93.1%	92.8
[TV99]	92.50%	92.25%	92.37
[RM95]	91.80%	92.27%	92.03
[ADK98]	91.6%	91.6%	91.6

section 00	precision	recall	$F_{\beta=1}$
Majority	95.04%	94.75%	94.90
[TV99]	93.71%	93.90%	93.81
[RM95]	93.1%	93.5%	93.3

Results standard NP data set

section 20	precision	recall	$F_{\beta=1}$
Majority	90.00%	78.38%	83.79
[CoNLL99]	91.28%	76.06%	82.98

Error analysis

Estimated cause of errors in the held-out training data of the first baseNP experiment (false positives and false negatives):

FPos	FNeg	
28 %	29 %	POS error
16 %	18 %	Problem with conjunction
15 %	12 %	Punctuation mark attachment
11 %	12 %	Split NPs or combined neighbors
5 %	4 %	Adverb attachment
3 %	3 %	NPs containing *to*
3 %	1 %	Error in NP segmentation
0 %	2 %	NP consisting of *that*
19 %	19 %	Unknown

Major error causes are the preprocessing stage and hard cases.

Concluding remarks

1. Combining classifiers improves performance for NP recognizers.
2. In our experiment setup the error reduction is smaller (8%) than in the POS tagger experiment of [HZD98] (19%).

Future work

Repeat this work while using different learning algorithms.