Memory-based semantic role labeling: Optimizing features, algorithm, and output

Antal van den Bosch, Sander Canisius, Iris Hendrickx,
Walter Daelemans and Erik Tjong Kim Sang

Tilburg University, The Netherlands
University of Antwerp, Belgium

Approach

We applied our standard NLP tools to the shared task: memory-based learning in combination with wrappers for feature selection and parameter optimization.

Additionally we evaluated classifier stacking for obtaining features representing predicted classes for neighbors and trigram output classes for inhibiting impossible output sequences like `A1 A0 A1`.

Suspected systematic errors (duplicate A0-A5 phrases inside one sentence and missing V tags) were removed from the system output.

Performances on the development data

	Prec.	Recall	$F_{\beta=1}$	method
a	51.6%	51.9%	51.8	feature selection
b	57.3%	52.7%	54.9	parameter optimization
c	58.8%	54.2%	56.4	feature selection
d	59.5%	53.9%	56.5	parameter optimization
e	64.3%	54.2%	58.8	classifier stacking
f	66.3%	56.3%	60.9	parameter optimization
g	66.5%	56.3%	60.9	feature selection
h	68.1%	56.8%	61.9	classifier stacking
i	68.3%	57.5%	62.4	feature selection
j	68.9%	57.8%	62.9	classifier stacking
k	69.1%	57.8%	63.0	classifier stacking
	50.6%	30.3%	37.9	baseline

Features

Features	a-b	c-d	e-f	g-h	i-k
words	-1–0	-2–1	-2–1	-2–1	-2–1
PoS tags	0–1	0–1	0–1	-1–1	-1–1
chunk tags	0	0–2	0–2	-1–1	-1–1
NE tags	–	–	–	–	–
output classes	NA	NA	-3–3	-3–3	-3–3
distances	cNV	cNVw	cNVw	Vw	cNV
main verb	+	+	+	+	+
role pattern	+	+	+	+	+
voice	+	+	+	+	+
current clause	+	+	+	+	+
previous prep.	–	+	+	+	–
Total	12	18	24	23	24

Parameters

Parameters	a	b-c	d-e	f-k
algorithm	IB1	IB1	IB1	IB1
distance metric	O	M	J	O
switching threshold	NA	2	2	NA
feature weighting	nw	nw	nw	nw
neighborhood size	1	15	19	1
class weights	Z	ED1	ED1	Z

We have evaluated different feature weighting schemes and consistently found that uniform feature weights performed best.

Concluding remarks

Our system finished seventh in a field of ten participating systems (test set performance: $F_{\beta=1}$=60.1).

In post-deadline work, classifier stacking has been integrated in the learner and additional features (chunk paths and clause paths) have been included. This has only led to small performance improvements.