

Optimierung von POS-Tagger-Ergebnissen

Präsentation zur Bachelorarbeit - betreut von Sebastian Weigelt und Mathias Landhäußer

Viktor Kiesel | 25. April 2016

IPD TICHY



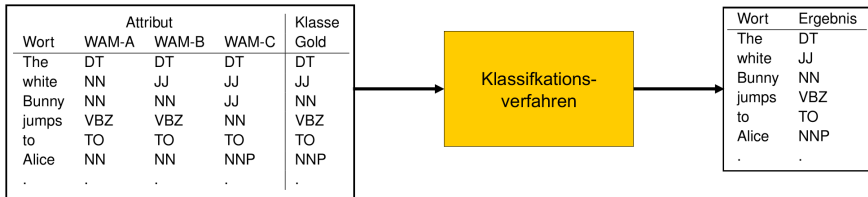
- Ziel: Optimierung von Wortartmarkiererergebnissen
- Motivation:
 - Wortartmarkierung meist erster Verarbeitungsschritt
 - Fehler führen zu Folgefehler, die Weiterverarbeitung verhindern
 - Wichtig gute Wortartmarkierung durchzuführen
- Lösungsvorschläge:
 - Training auf anwendungszecknahem Textsammlung/Korpus
 - Verbesserung, aber nicht optimal
 - Standardmarkierer: Von 96,17% auf 98,27% Genauigkeit
 - Kombination mehrerer Markierer zu einem Ensemble

- Ali und Pazzani[AP96] zeigen Kombination von Klassifikatoren möglich
 - Schwach korrelierte Klassifikatoren besser
- Van Halteren[vZD98] kombiniert Wortartmarkierer
 - Einfache Mehrheitswahl

Wort	Wortartmarkierer			Wortart korrekt
	A	B	C	
The	DT	DT	DT	DT
white	<u>NN</u>	JJ	JJ	JJ
Bunny	NN	NN	<u>JJ</u>	NN
jumps	VBZ	VBZ	<u>NN</u>	VBZ
to	TO	TO	TO	TO
Alice	<u>NN</u>	<u>NN</u>	NNP	NNP
.

- Paarweises Wahlverfahren am besten (97,92% Genauigkeit)

- Klassifikationsverfahren zur Kombination
- Ergebnisse (Wortartmarkierungen) der Wortartmarkierer als Attribut
 - Korrekte Markierung ist Klasse



- Wort optional
- Nur ein einzelnes Wort wird betrachtet

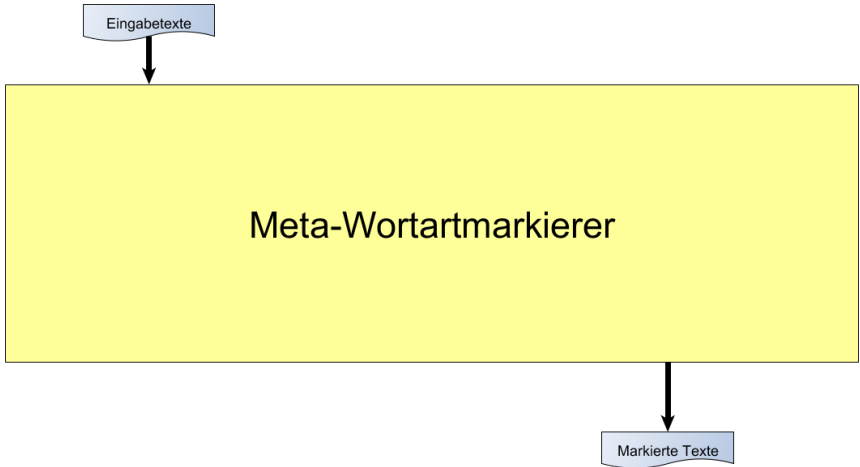
- “Wallstreet Journal”-Korpus (WSJ) ist Standardkorpus
- NLCI- und PARSE-Korpus wurden halbautomatisch markiert
 - Mit verschiedenen Wortartmarkierern vormarkiert
 - Ergebnisvergleich als Hilfe
 - Handkorrektur durch Erfahrung

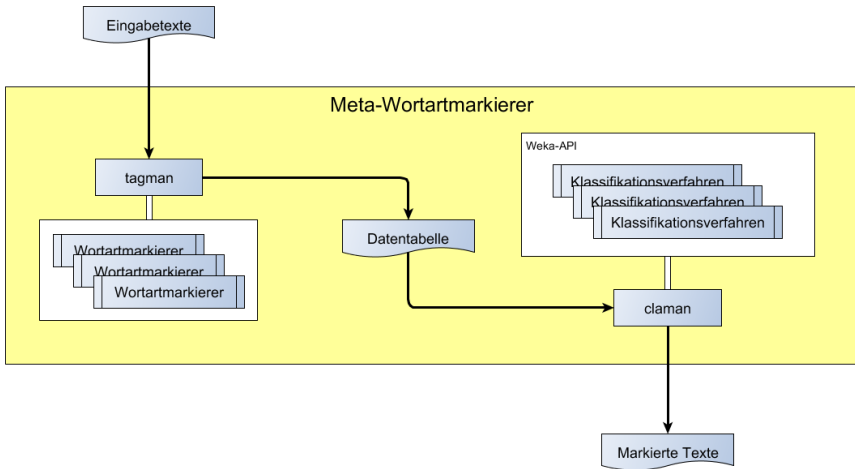
Korpuseigenschaften

Korpus	Bereich	Worte	Stil
WSJ	Wirtschaft	94.085	Fachzeitschrift
NLCI	Erzählung	18.124	Skript
PARSE	Alltag	2.174	Aufgabenerteilung

Wortartmarkierer (Modell)	Genauigkeiten in %		
	WSJ	NLCI	PARSE
Einfache Mehrheitswahl	98,43	97,65	93,01
Stanford (bidirectional)	98,02	96,17	90,66
Berkeley	96,94	95,86	86,61
SENNA	97,72	95,85	89,70
OpenNLP (MaxEnt)	96,71	95,17	90,11
MATE	95,38	94,91	89,60
Stanford (3 words left)	97,83	94,45	89,56
clearNLP	95,75	93,94	89,47
TreeTagger	92,58	92,58	80,04
OpenNLP (Perceptron)	95,98	92,41	85,33
Jitar (Selbsttrainiert)	98,54	91,71	85,37
Illinois	91,27	86,36	83,53

Art	Verfahren
Ausschlussverfahren:	VFI HyperPipes
Regeln:	Ridor JRip NNge PART OneR
Nächste Nachbarn:	KStar IBk
Stützvektormaschine :	LibSVM
Entscheidungsbaum:	REPTree J48
Bayes:	NaiveBayes BayesNet

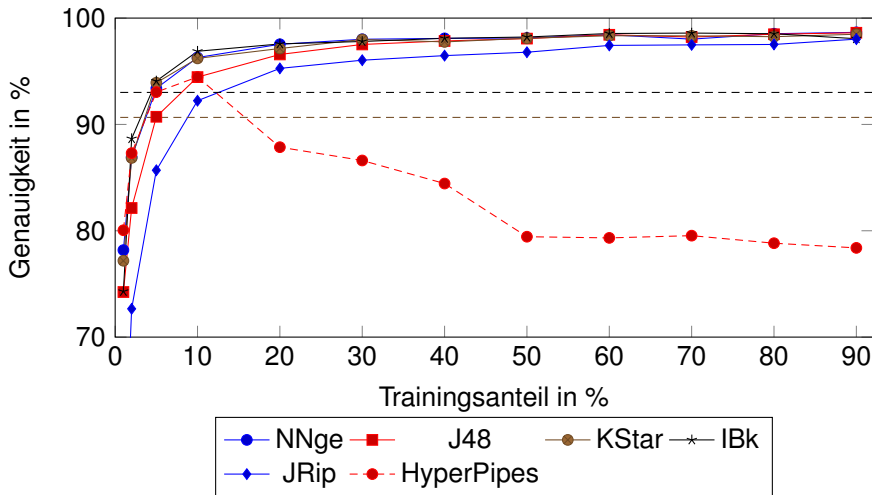




- Genauigkeit als Metrik: Qualität des Verfahrens
- Zwei Validierungsverfahren:
 - 10-fache Kreuzvalidierung: Qualität der Verfahren
 - Monte-Carlo-Kreuzvalidierung: Lernkurven abhängig von Trainingsmenge
- Auswirkung von Worten als Attribut

Korpus	Genauigkeit der besten 3 Verfahren		Vergleich Name (Genauigkeit)
	Mit Worten	Ohne Worte	
WSJ	J48 (98,85%)	J48 (98,73%)	Jitar (98,54%) Wahl (98,43%)
	PART (98,70%)	REPTree (98,70%)	
	REPTree (98,69%)	PART (98,64%)	
NLCI	IBk (99,11%)	IBk (98,95%)	Wahl (97,65%) Stanford-bidir (96,17%)
	KStar (99,05%)	REPTree (98,86%)	
	NNge (98,97%)	J48 (98,80%)	
PARSE	IBk (98,57%)	NNge (98,66%)	Wahl (93,01%) Stanford-bidir (90,66%)
	PART (98,48%)	J48 (98,62%)	
	KStar (98,34%)	KStar (98,48%)	

Lernkurve auf PARSE-Korpus ohne Worte



- Wortartmarkierer mit Klassifikationsverfahren erfolgreich kombiniert
 - Tagman: Wortartmarkierer ausgewählt und koordiniert
 - Claman: Ausgabe mit Klassifikationsverfahren kombiniert
- Wahl des Verfahrens wichtig: Kein absolut bestes Verfahren
 - Je nach Korpus siegen unterschiedliche Verfahren
 - J48 erreicht immer höhere Genauigkeiten als Vergleichsmarkierer
- Beobachtung bei Wortnutzung
 - Regelbasiert ohne Wörtern besser
 - Instanzbasierte schlechter
- Lernkurven: Genauigkeit steigt meist mit mehr Trainingsdaten
 - Hyperpipes und VFI sinkt Genauigkeit
 - Training muss überwacht werden

- Anzahl der Wortartmarkierer und Klassifikationsverfahren erhöhen
 - Markierer bereits ausgewählt
 - Verfahren bereits implementiert
- Weitere Attribute
 - Wortkontext
 - Wortmerkmale
- Übertragbarkeit überprüfen
 - Verfahren auf einem Korpus trainieren
 - Auf den anderen Korpora testen

Vielen Dank fürs Zuhören!
Gibt es Fragen?

- Ali, Kamal M. und Michael J. Pazzani (1996). “Error reduction through learning multiple descriptions”. en. In: *Machine Learning* 24.3, S. 173–202. ISSN: 0885-6125, 1573-0565. DOI: 10.1007/BF00058611.
- van Halteren, Hans, Jakub Zavrel und Walter Daelemans (1998). “Improving Data Driven Wordclass Tagging by System Combination”. In: *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*. ACL '98. Stroudsburg, PA, USA: Association for Computational Linguistics, S. 491–497. DOI: 10.3115/980845.980928.