

Beschreibung der Programmierprojekte

Projektumfang

Abzugeben sind

1. Quellcode, kompilier- und lauffähig,
2. Dokumentation, bestehend aus
 - a) Beschreibung des eingesetzten Verfahrens, ggf. mit Literaturverweisen,
 - b) Beschreibung von benötigter externer Software oder Daten,
 - c) Systemvoraussetzungen,
 - d) Funktionsumfang und Grenzen der Software,
 - e) Evaluierungsergebnis und
 - f) Bedienungsanleitung.

Zeitplan

17.12.2008	Kurzpräsentation des geplanten Systems (10 Minuten), Abgabe der (vorläufigen) Verfahrensbeschreibung (Umfang etwa 3 Seiten)
4.2.2009	Präsentation der Systeme, soweit fertiggestellt; erste Evaluierungsergebnisse; Erfahrungsbericht, Verbesserungs-/Erweiterungsvorschläge
31.3.2009	Abgabe von Code und Dokumentation

Evaluierung

1. TE-Erkennen

Als Evaluierungsdatenset verwenden Sie bitte die Testdaten aus RTE-3, die Sie unter http://www.nist.gov/tac/tracks/2008/rte/past_data/RTE3-TEST.tar.gz finden. Verwenden Sie dieses Datenset nicht zum Trainieren Ihres Klassifizierers! Als Evaluierungsmaßstab verwenden wir die Genauigkeit (*accuracy*), die angibt, in wieviel Prozent der Fälle die Antwort Ihres Systems mit der Antwort im Goldstandard übereinstimmt.

Wenn Sie ein System für eine andere Sprache als Englisch entwickeln, müssen Sie ein eigenes Datenset erstellen, und zwar mindestens 50 Text-Hypothesen-Paare Trainingsdaten und mindestens 50 T-H-Paare Testdaten (25 mal „folgt“, 25 mal „folgt nicht“). Halten Sie sich bei der Erstellung an die Methode der Datengewinnung, die im PASCAL RTE Challenge angewandt wurde.

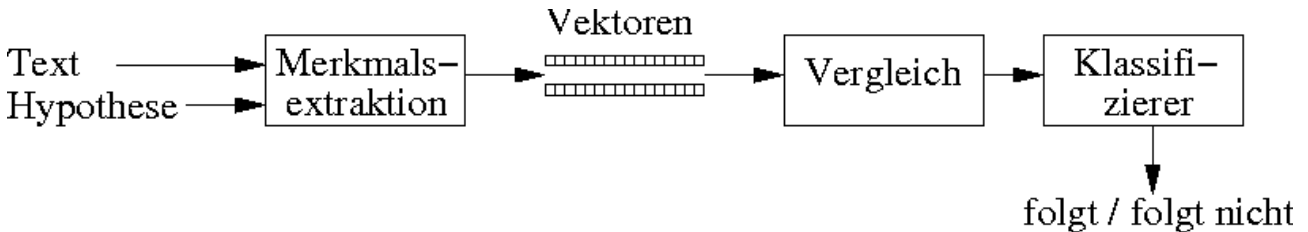
2. Paraphrasenextraktion

Zur Bestimmung der Genauigkeit ist aus den gefundenen Paraphrasen eine repräsentative Zufallsstichprobe zu ziehen, und diese manuell auf Korrektheit zu bewerten. Außerdem Bestimmung des Konfidenzintervalls.

Projektthemen

1. Baseline TE-Erkenner (*Rebel, Lenz, Dittrich, Dione*)

Es erfolgt keine syntaktische Analyse der Eingabe und es wird keine formal-logische Repräsentation des Inhalts aufgebaut.



Zur Verbesserung der Leistungsfähigkeit sollen aber einige der folgenden Wissensquellen oder Werkzeuge genutzt werden:

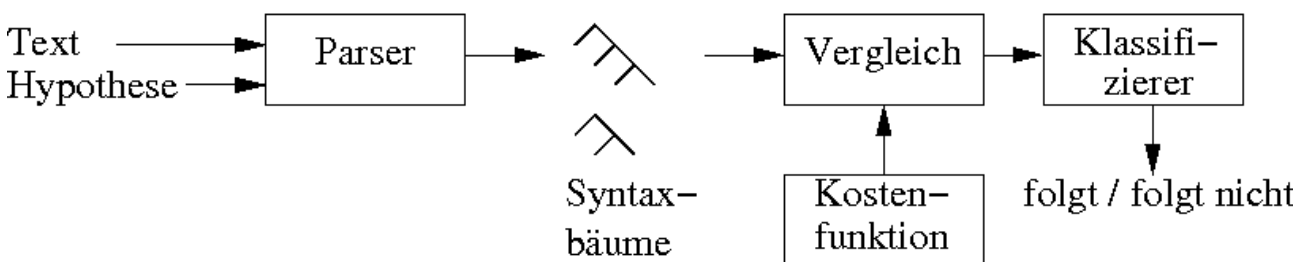
- Stemming, Lemmatisierung (z.B. Tree-Tagger)
- Eigennamenerkennung (z.B. <http://nlp.stanford.edu/software/CRF-NER.shtml>)
- Mehrwortausdrücke und Eigennamen (mit Varianten): [HeiNER](#)
- WordNet und WordNet::Similarity
- korpusbasierte Maße für die semantische Ähnlichkeit von Wörtern (z.B. LSA, [SemanticVectors](#), [DISCO](#))
- Paraphrasen
- Wikipedia-Redirects
- ...

Wichtig ist, jeweils zu evaluieren, ob durch den Einsatz einer dieser Erweiterungen eine Leistungsverbesserung erzielt wird.

Literatur: Jijkoun & de Rijke (2005)

2. Syntaxbasierter TE-Erkenner (*Oltmann*)

Dieses TE-System arbeitet auf der Grundlage von Syntaxbäumen. Mit Hilfe von Editieroperationen wird versucht, den Syntaxbaum des Textes auf den Hypothesenbaum abzubilden. Editieroperationen verursachen Kosten, die auf unterschiedliche Art gemessen werden können.



Literatur: Kouylekov & Magnini (2005)

3. Paraphrasenextraktion (*Rosner, Bablitz*)

Entwicklung einer Lösung zur automatischen Generierung einer deutsch- oder englischsprachigen Datenbank aus Paraphrasen auf Satz-, Phrasen- oder Wortebene.

Überlegungen zur Nutzung der extrahierten Paraphrasen innerhalb eines TE-Erkenners.

Literatur: Barzilay & Lee (2003), Brockett & Dolan (2005), Szpektor et al. (2004)