

Kaskaden von lokalen Grammatiken

9. Juli 2008

Vorlesung Syntaktische Analyse mit lokalen Grammatiken

Lukas Bulwahn
bulwahn@in.tum.de

Überblick

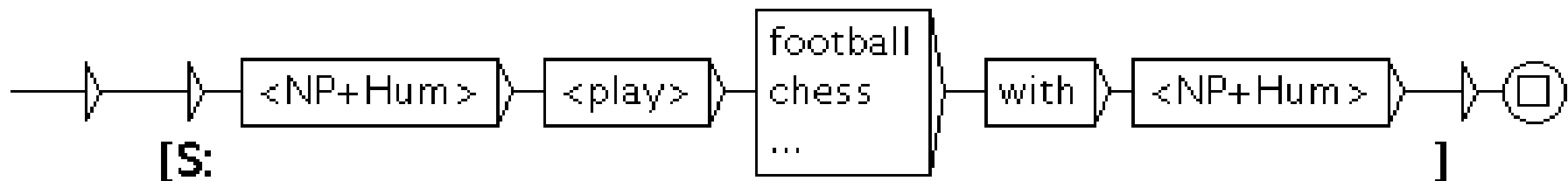
- Motivation und Begriffserklärung
- Vorteile und technische Umsetzung
- Fallstudien mit Transduktor-Kaskaden
 - Eigennamen-Extraktion
 - FASTUS System
 - FSA-Parsing

Was ist eine Kaskade?

- Eine Kaskade ist eine Hintereinanderausführung von mehreren lokalen Grammatiken mit Ausgabe (Transduktoren)
- Dabei wird die Ausgabe eines Transduktors als Eingabe eines weiteren Transduktors verwendet.

Einfaches Beispiel einer Kaskade

- Beispielsatz:
Tom plays football with his younger brother Jerry.
- Anwenden der NP-Grammatik ergibt:
[NP+PR+Hum:Tom] plays football with
[NP+PR+Hum:his younger brother Jerry].
- Danach Anwenden der PAS-Grammatik:



ExamplePlayGamewith.grf

Tue Jul 08 09:44:24 GMT 2008

Historische Ansätze bei Grammatikformalismen

- 1960: Chomsky: Betrachtung der Ausdruckskraft von Grammatikformalismus (Chomsky-Hierarchie)
- Reguläre Grammatiken (lokale Grammatik) sind nicht ausdrucksstark genug um natürliche Sprache zu analysieren.
- Daher Verwendung von anderen Formalismen wie z.B. kontextfreie Grammatiken (PCFG,...) und Transformationsgrammatiken

Historische Ansätze bei Grammatikformalismen

- Problem: Regeln sind nicht allgemeingültig, sondern lexikalisch (Gross, 1975; Boons and Leclere, 1976)

2 Notwendigkeiten für einen Grammatikformalismus:

- **ausdrucksstark genug**: Alle Spracheigenschaften müssen durch den Formalismus ausgedrückt werden können.
- **unvollständig funktionsfähig, modular, und erweiterbar**: Eine vollständige Grammatik ist zu schwierig zu erzeugen.

Transduktor-Kaskaden erfüllen diese Eigenschaft.

Wieso verwendet man lokale Grammatiken?

einfach

- Der Grammatikformalismus der lokalen Grammatik ist einfach und verständlich.

präzise

- Lokale Grammatiken sind modular und können sehr genaue Erkennungen erstellen.

effizient

- Lokale Grammatiken können in endliche Automaten (FSA) kompiliert werden. Diese können Text sehr schnell verarbeiten.

Vorteil der Kaskaden

- Komplexe Sätze müssen nicht vollständig erkannt werden. (**Bottom-Up-Verfahren**)
- Durch Kaskaden kann man zunächst einfache sichere Transduktoren verwenden. (**islands of certainty**)
- Aus den bereits korrekten Erkennungen können weitere größere Einheiten zusammengesetzt werden.

Technische Umsetzung von Kaskaden

In Unitex gibt es keine Unterstützung für Kaskaden:

Daher verwendet man folgendes Verfahren:

- Anwendung der 1. lokalen Grammatik und Extraktion aller Treffer
- Erstellen eines Lexikons aus den Treffern, wobei Transducer-Output Lexikonkategorien werden.
- Anwendung der 2. lokalen Grammatik mit erstelltem Lexikon

Technische Umsetzung von Kaskaden

In Outilex ist die Unterstützung von Kaskaden bereits implementiert:

- Transduktoren transformieren Satzautomaten in neue Satzautomaten.
- Daher können Transduktoren auf natürliche Art und Weise kaskadiert werden.

Beispiele für Verwendung von Kaskaden

3 Fallstudien für die Verwendung von Kaskaden:

- Extraktion von Eigennamen (Friburger und Maurel, 2002)
- Das FASTUS System (Hobbs et al., 1993)
- Parsing mit FSA Transducern (Roche, 1996)

Extraktion von Eigennamen

- Ziel der Arbeit:
Extraktion von Eigennamen anhand linker und rechter Kontexte aus Zeitungsartikeln im Französischen
- Sehr genaue Extraktion möglich:
96,9% recall und 99,1% precision bei Erkennung
- Verwendung der Kaskaden, um Transducer mit längeren Kontexte vor kürzeren Kontexten anzuwenden (**Longest-Match-Verhalten**)

Beispiel für EN-Extraktion

- Beispielsequenz:
Monsieur Jean Dupont
- Zwei passende Muster:
A: Monsieur gefolgt von Nachname
B: Monsieur gefolgt von Vorname und Nachname
- Durch Anwenden von B vor A wird die richtige Lesart erkannt; die falsche dadurch verhindert.

Das FASTUS System

- Das FASTUS System ist ein Informationsextraktionssystem.
- Drei wichtige Eigenschaften von Information Extraktion:
 - Nur Teile des Textes sind relevant.
 - Information wird in einer einfach strukturierten Datenbanken gespeichert.
 - Subtile Bedeutungsunterschiede spielen keine Rolle.

Evaluation des FASTUS Systems

- Das FASTUS System wurde anhand der Extraktion von Informationen über Terroranschlägen evaluiert.
- Ergebnis:
 - 44 % recall; 55 % precision
 - schnelle domänen-spezifische Entwicklung
 - effiziente Bearbeitung (2000 Wörter pro Minute)

Beispielsatz für FASTUS

- Salvadoran President-elect Alfredo Cristiani condemned the terrorist **killing** of **Attorney General Roberto Garcia Alvarado** and accused the **Farabundo Marti National Liberation Front (FMLN)** of the crime.
- Perpetrator Org: **FMLN**
Confidence: **suspected or accused by authorities**
Human Target: **Roberto Garcia Alvarado**
Description: **Attorney General: Roberto Garcia Alvarado**
Effect: **Death: Roberto Garcia Alvarado**

Phasen bei FASTUS System

- Das FASTUS System besteht aus 4 Phasen:
 - Triggering: Anhand von Trigger Wörtern/Phrasen werden die relevanten Sätze ausgewählt.
 - Phrasenerkennung
 - Mustererkennung (PAS-Erkennung)
 - Zusammenführen von Ereignissen:
Ergebnisse aus verschiedene Sätzen werden zu einem Eintrag zusammengefügt.
Dabei wird semantisch konsistent gearbeitet.

Phrasen- und Mustererkennung

- In der ersten Phase werden Nominalphrasen, Verbalphrasen, und einige Adverbiale erkannt.
- 96,4 % korrekte Erkennungen
- In der zweiten Phase werden Muster wie folgt erkannt:
killing of <HumanTarget>
<GovtOfficial> accused <PerpOrg>
bomb was placed by <Perp> on <PhysicalTarget>
- Insgesamt 95 solcher Muster

Weitere Pseudo-Syntax

- Um eine vollständige Syntax zu simulieren, wurden weitere einfache Transducer mit eingebaut:
- Subject VerbGroup1 {NounGroup|Other}*
Conjunction VerbGroup2
wird zu "Subject VerbGroup2".
- Ähnliches für andere Inserts und Relativsätze

Parsing with Finite State Transducers (Roche, 1996)

- Die Arbeit von Roche behandelt wie bestimmte Konzepte der Sprache mit Transduktoren behandelt werden können.
- Dies ergänzt die Arbeiten von Gross (Es wird bereits angenommen, eine präzise lexikalische Grammatik existiere.)

Ziel der Arbeit von Roche

- Ziel der Arbeit:
Skizze, wie eine vollständige Grammatik mit Transduktoren erstellt werden könnte
- Dabei werden folgende Konstruktionen betrachtet:
 - Umgang mit Modalverben, Inserts
 - Umgang mit Satzkomplementen
 - Stützverbkonstruktionen, Idiome
 - Stützverbrekonstruktion aus Nominalphrasen

Beispiel: Satzkomplemente

- John expected Mary to come.
- Durch Regel: N expected N to Vinf W
→ N expected N (S N Vinf W)
- wird der Satz verarbeitet zu:
X N(Mary) come. wobei X=[John expected]
- Jetzt kann dadurch die Satzgrammatik von
come verwendet werden:
- Mary come → [S: Mary come]

Beispiel: Stützverbrekonstruktion aus Nominalphrasen

John's concessions to his friends were unexpected.

- der elementare Satz soll erkannt werden:
John makes concessions to his friends
- [N A's concessions to B N] → [N [S A V^{sup} ?
concessions to B S] N]
John's concessions to his friends →
John V^{sup}? concessions to his friends.
- Erkennung durch die make concessions-
Struktur, z.B. [S A make concessions to B S]

Zusammenfassung

- Kaskaden lokaler Grammatiken können für viele Zwecke verwendet werden (Extraktion von lexikalischen Einheiten, Informationsextraktion, vollständige Syntaxanalyse)
- Durch die Hintereinanderschaltung bleiben die einzelnen Grammatiken einfach und präzise.
- Korpora können dabei sehr schnell durchsucht und verarbeitet werden.

Literatur

- Nathalie Friburger und Denis Maurel. Finite-State Transducer Cascade to Extract Proper Names in Texts. 2002.
- Jerry R. Hobbs, John Bear, David Israel, Mabry Tyson. FASTUS: A Finite-state Processor for Information Extraction from Real-world Text. 1993.
- Emmanuel Roche. Parsing with Finite-State Transducers. 1996.