

# Projektion von gesprochener Sprache auf eine Handlungsrepräsentation

Bachelorarbeit  
von

**Markus Kocybik**

An der Fakultät für Informatik  
Institut für Programmstrukturen  
und Datenorganisation (IPD)

Erstgutachter:	Prof. Dr. Walter F. Tichy
Zweitgutachter:	Prof. Dr. Ralf Reussner
Betreuender Mitarbeiter:	Dipl. Inform. Sebastian Weigelt

Bearbeitungszeit: 01.04.2015 – 31.07.2015



---

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Die Regeln zur Sicherung guter wissenschaftlicher Praxis im Karlsruher Institut für Technologie (KIT) habe ich befolgt.

**Karlsruhe, 31.07.2015**

.....  
(Markus Kocybik)



## **Kurzfassung**

In dieser Bachelorarbeit wird ein Werkzeug entworfen und implementiert, das einen Text auf eine initiale Handlungsrepräsentation für natürliche Sprache abbildet. Bei den Eingabetexten handelt es sich um gesprochene Anweisungen an den humanoiden Haushaltsroboter ARMAR-III, die mithilfe eines automatischen Spracherkenners in Text überführt werden. Die Herausforderung hierbei ist, dass die Ausgabe des automatischen Spracherkenners keine Punktierung, Wortfehler, Wiederholungen und Füllwörter enthalten. Um dieses Ziel zu erreichen, werden Gesamtsysteme untersucht, die verwandte Problemstellungen adressieren. Nach der Betrachtung dieser Gesamtsysteme entsteht der Ansatz, aktuelle Standard-Sprachverarbeitungswerkzeuge, die auf geschriebener Sprache optimiert wurden, zu verwenden, um Informationen aus dem Eingabetext zu extrahieren, welche anschließend in der initialen Handlungsrepräsentation dargestellt werden können. Um diesen Ansatz zu untersuchen, werden elf aktuelle Sprachverarbeitungswerkzeuge mit einem 468 Wörter umfassenden Korpus analysiert. Bei dieser Analyse wird festgestellt, dass die Wort- und Phrasenerkennung mithilfe aktueller Standard-Sprachverarbeitungswerkzeuge, die auf geschriebener Sprache optimiert wurden, verwertbare Ergebnisse für gesprochene Sprache liefern. Die hier durchgeführte Analyse führt aber auch zum Ergebnis, dass Standard-Sprachverarbeitungswerkzeuge ungeeignet sind, um syntaktische und semantische Informationen zu extrahieren. Dies kann auf die fehlende Punktierung der Eingabesätze zurückgeführt werden. Um dennoch syntaktische und semantische Informationen zu extrahieren, wird eine Heuristik vorgestellt, die aus einem Eingabetext, der aus einer Folge von Anweisungen ohne Punktierung besteht, einzelne Anweisungen erkennt. Dazu werden Annahmen über die Satzstruktur des Eingabetextes getroffen und jedem Wort eine Befehlsnummer zugewiesen. Das hier erstellte Werkzeug ist in der Lage, jedem Wort des Eingabetextes eine Wortart (engl. POS Tag), phrasale Markierung (engl. Chunk) und Befehlsnummer zuzuweisen und diese in einer initialen Handlungsrepräsentation darzustellen. Bei der Evaluation des Werkzeuges kann eine Genauigkeit (engl. Accuracy) von 94 Prozent in der Wortartenerkennung, 93 Prozent in der Phrasenerkennung und 99 Prozent in der Zuweisung von Befehlsnummern erzielt werden.



# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>1</b>
1.1. Projekt „PARSE“ . . . . .	1
1.2. Zielsetzung . . . . .	2
<b>2. Grundlagen</b>	<b>3</b>
2.1. Automatische Spracherkenner und die Herausforderungen von gesprochener Sprache . . . . .	4
2.2. NLP-Aufgaben . . . . .	4
2.2.1. Tokenisierung . . . . .	4
2.2.2. Wortartenerkennung . . . . .	5
2.2.3. Lemmatisierung . . . . .	6
2.2.4. Syntaxanalyse . . . . .	6
2.2.5. Phrasenerkennung durch Chunking . . . . .	8
2.2.6. Eigennamenerkennung . . . . .	9
2.2.7. Zuweisung von semantischen Rollen . . . . .	9
2.2.8. Koreferenzanalyse . . . . .	10
2.3. Offene und abgeschlossene NLP-Werkzeuge . . . . .	11
<b>3. Verwandte Arbeiten</b>	<b>13</b>
3.1. Sprachverarbeitungswerkzeuge . . . . .	13
3.1.1. SENNA . . . . .	13
3.1.2. Memory-Based Shallow Parser (MBSP) . . . . .	14
3.1.3. BIOS . . . . .	15
3.1.4. Illinois NLP Curator . . . . .	16
3.1.5. Apache OpenNLP . . . . .	16
3.1.6. ClearNLP . . . . .	17
3.1.7. Werkzeuge der Stanford NLP Group . . . . .	18
3.1.8. Berkeley, Stanford, Collins und BLLIP/Charniak Parser . . . . .	18
3.2. Gesamtsysteme . . . . .	19
3.2.1. Semantic Mapping Between Natural Language Questions and SQL Queries via Syntactic Pairing . . . . .	19
3.2.2. Precise-System . . . . .	20
3.2.3. The KIT Translation Systems for IWSLT 2013 . . . . .	20
3.2.4. Pegasus . . . . .	21
3.2.5. Metafor . . . . .	22
3.2.6. Shallow Semantic Parsing for Spoken Language Understanding . . . . .	22
3.2.7. CALO-Projekt . . . . .	23
3.2.8. Tagging Spoken Language Using Written Language Statistics . . . . .	23
<b>4. Analyse</b>	<b>25</b>
4.1. Analysekorpus . . . . .	26
4.2. Wortartenerkennung . . . . .	27
4.3. Phrasenerkennung . . . . .	32

4.4.	Syntaxanalyse . . . . .	33
4.5.	Zuweisung von semantischen Rollen . . . . .	35
4.6.	Fazit und Validität . . . . .	35
<b>5.</b>	<b>Entwurf und Implementierung</b>	<b>37</b>
5.1.	Entwurf . . . . .	37
5.2.	Die verwendeten Sprachverarbeitungswerkzeuge . . . . .	38
5.3.	Heuristik zum Erkennen von Befehls Grenzen . . . . .	39
5.4.	Interne Repräsentation . . . . .	42
5.5.	Verwendung des erstellten Werkzeuges . . . . .	42
<b>6.</b>	<b>Evaluation</b>	<b>45</b>
6.1.	Evaluationskorpus und Vergleichsvorgang . . . . .	45
6.1.1.	Evaluationsvorgang: Transkriptionen . . . . .	46
6.1.2.	Evaluationsvorgang: ASR-Texte . . . . .	47
6.2.	Evaluationsvarianten . . . . .	48
6.3.	Evaluationsergebnisse . . . . .	49
6.3.1.	Wortartenerkennung . . . . .	50
6.3.2.	Befehlsnummern . . . . .	50
6.3.3.	Phrasenerkennung . . . . .	51
6.3.4.	Bewertung . . . . .	53
<b>7.</b>	<b>Zusammenfassung und Ausblick</b>	<b>55</b>
	<b>Literaturverzeichnis</b>	<b>59</b>
	<b>Anhang</b>	<b>63</b>
A.	Penn Treebank part-of-speech tags . . . . .	63
B.	Chunk tags . . . . .	64
C.	Einige grammatikalische Relationen aus [MCBCD06] . . . . .	64
D.	Analysekorpus . . . . .	64
E.	Musterlösungen Wortartenerkennung . . . . .	67
F.	Musterlösungen Phrasenerkennung . . . . .	69
G.	Ausgaben der Sprachverarbeitungswerkzeuge . . . . .	71
G.1.	Stanford POS Tagger . . . . .	71
G.2.	OpenNLP Wortartenerkennung . . . . .	74
G.3.	Illinois POS Tagger . . . . .	78
G.4.	ClearNLP . . . . .	81
G.5.	MBSP . . . . .	85
G.6.	SENNA . . . . .	89
G.7.	BIOS . . . . .	94
G.8.	OpenNLP Phrasenerkennung . . . . .	98
H.	Syntaxbäume . . . . .	102
H.1.	SENNA . . . . .	102
H.2.	Charniak . . . . .	104
H.3.	Stanford . . . . .	106
H.4.	Berkeley . . . . .	109
I.	Evaluation . . . . .	111
I.1.	Transkriptionen . . . . .	111
I.2.	Ausgaben der Google Speech To Text API . . . . .	113
I.3.	Musterlösung zur Evaluation . . . . .	114
I.4.	Ausgabe mit SENNA . . . . .	117
I.5.	Ausgabe mit SENNA und Stanford . . . . .	120

I.6.	Ausgabe mit SENNA, Stanford und Optimierungsliste . . . . .	123
I.7.	Ausgaben für die ASR Texte (SENNA und Stanford) . . . . .	126
J.	Heuristik zur Zuweisung von Befehlsnummern . . . . .	129



# Abbildungsverzeichnis

1.1. Einordnung der hier erstellten Anwendung . . . . .	2
2.1. Ausgabe eines typisierten Dependenzparsers . . . . .	7
2.2. Ausgabe eines Konstituentenparser . . . . .	8
2.3. Die <b>Pipeline</b> der NLP-Aufgaben . . . . .	11
3.1. Links: Syntaxbaum der Frage Rechts: Syntaxbaum der entsprechenden SQL-Abfrage (SELECT state_name FROM border_db WHERE border='Texas') . . . . .	19
3.2. Ideensprache des Pegasus-Projektes [RM06] . . . . .	21
3.3. Auszug eines Eintrages für die Idee <i>smallness</i> . . . . .	22
4.1. Die zwei Konstituentenbäume für „Go to the fridge. Open the door.“ . . . . .	34
4.2. Ausgabe des Stanford Parsers grafisch dargestellt . . . . .	34
5.1. Entwurf zur Kombination der verwendeten Sprachverarbeitungswerkzeuge . . . . .	38
5.2. Angepasster Entwurf . . . . .	41
5.3. Exemplarischer AGG-Graph . . . . .	43



# Tabellenverzeichnis

2.1.	Auszug des Penn Treebank-Standards . . . . .	6
2.2.	Übersetzter Auszug des PropBank-Eintrages für das Verb „sell“ . . . . .	10
4.1.	Analyse der Wortartenerkennung . . . . .	28
4.2.	SENNA: Fehler bei der Tokenisierung . . . . .	29
4.3.	Analyse der Wortartenerkennung (Verbform irrelevant) . . . . .	29
4.4.	Fehlerhaft klassifizierte Nomen und Verben . . . . .	31
4.5.	Analyse der Phrasenerkennung . . . . .	32
5.1.	Ausgabe für die Wortarten- und Phrasenerkennung . . . . .	40
5.2.	Ausgabe für die Wortarten- und Phrasenerkennung . . . . .	41
6.1.	Genauigkeitsmaß hinsichtlich der Befehlsnummern . . . . .	47
6.2.	Beispiel zum Vergleich der Wortfolgen . . . . .	48
6.3.	Evaluationsergebnisse . . . . .	49
6.4.	Evaluationsergebnisse hinsichtlich der Wortartenerkennung . . . . .	50
6.5.	Evaluationsergebnisse hinsichtlich der Zuweisung von Befehlsnummern . . . . .	51
6.6.	Evaluationsergebnisse hinsichtlich der Phrasenerkennung . . . . .	52
6.7.	Musterlösung für <i>please open the fridge</i> (Phrasenerkennung) . . . . .	52
6.8.	Ausgabe für <i>please open the fridge</i> (3. Variante) . . . . .	52
6.9.	Ausgabe für <i>please open the fridge</i> (1. und 2.Variante) . . . . .	53
6.10.	Evaluationsergebnisse hinsichtlich der Phrasenerkennung . . . . .	53



# 1. Einleitung

Im Alltag werden humanoide Haushaltsroboter eingesetzt, um Aufgaben in der Küche zu erledigen und somit den Menschen zu entlasten. Ein aktuelles Beispiel ist der am KIT entwickelte ARMAR-III [Tam06]. Wenn man nun gesprochene Sprache als Kommunikationsmöglichkeit zwischen Mensch und Maschine betrachtet, könnte eine exemplarische Instruktion an einen humanoiden Haushaltsroboter folgendermaßen lauten:

*„Put the dirty plates into the dishwasher.  
After that turn the dishwasher on.“*

Spricht man nun diese Instruktion in einen automatischen Spracherkenner (ASR), wird man feststellen, dass mit gesprochener Sprache mehrere Herausforderungen verbunden sind: Einerseits treten bei gesprochener Sprache Wiederholungen und Füllwörter auf, andererseits sind heutige Spracherkenner noch nicht in der Lage jedes gesprochene Wort korrekt zu erfassen. Zusätzlich können die meisten Spracherkenner keine Satztrennzeichen, wie Kommata oder Punkte, erkennen. Die Ausgabe eines automatischen Spracherkenners könnte für die obige Instruktion beispielsweise folgendermaßen aussehen:

*„Put eh the thirty plates into the dishwasher after turn the dishwasher on.“*

Damit ein humanoider Haushaltsroboter diese Ausgabe verstehen kann, ist ein robuster Parser, welcher Informationen extrahiert und eine Repräsentation bereitstellt die vom System verarbeitet und verstanden werden kann, von essentieller Bedeutung. Da Standard-Parser nur in der Lage sind grammatikalisch korrekte Texte zu verarbeiten, muss hier ein robuster Parser entwickelt werden, welcher eventuell weniger Informationen aus dem Eingabetext extrahieren kann, sich dafür aber nicht auf fehlerlose Eingabetexte verlassen muss. Ziel dieser Bachelorarbeit ist, im Rahmen des Gesamtprojektes „PARSE“, die Entwicklung solch eines robusten Parsers.

## 1.1. Projekt „PARSE“

Ziel des Gesamtprojektes „PARSE“ ist es ARMAR-III neue Anweisungen mithilfe von gesprochener Sprache beizubringen. Dabei ist das Projekt in mehrere Teilprojekte unterteilt. Zunächst wird ein Sprachkorpus mithilfe von Sprachaufnahmen erstellt. Mithilfe

dieses Sprachkorpus kann der verwendete Spracherkenner evaluiert und eventuell modifiziert werden. Dieser Teil des Gesamtprojektes beschäftigt sich mit der initialen Abbildung der Sprachdaten auf eine Handlungsrepräsentation, welche in einem weiteren Teilprojekt entworfen wurde. Anschließend sollen Agenten, die in diesem Projekt erstellte, initiale Repräsentation modifizieren. Diese aufgearbeitete Repräsentation soll dann auf das Zielsystem abgebildet werden.

## 1.2. Zielsetzung

Das Ziel dieser Bachelorarbeit ist die Implementierung eines Werkzeuges, welches es ermöglicht, einen Text auf eine Handlungsrepräsentation für natürliche Sprache abzubilden. Bei den Texten handelt es sich um gesprochene Anweisungen, die mithilfe eines automatischen Spracherkenners in Text überführt wurden. Die Herausforderung hierbei ist, dass die Ausgabe des automatischen Spracherkenners keine Punktierung, Wortfehler, Wiederholungen und Füllwörter enthält. Mithilfe der bereitgestellten initialen Handlungsrepräsentation und der anschließenden Modifikation durch Agenten, soll es im Anschluss möglich sein, ARMAR-III neue Anweisungen beizubringen.

Dieser Sachverhalt soll anhand der Abbildung unter 1.1 verdeutlicht werden.

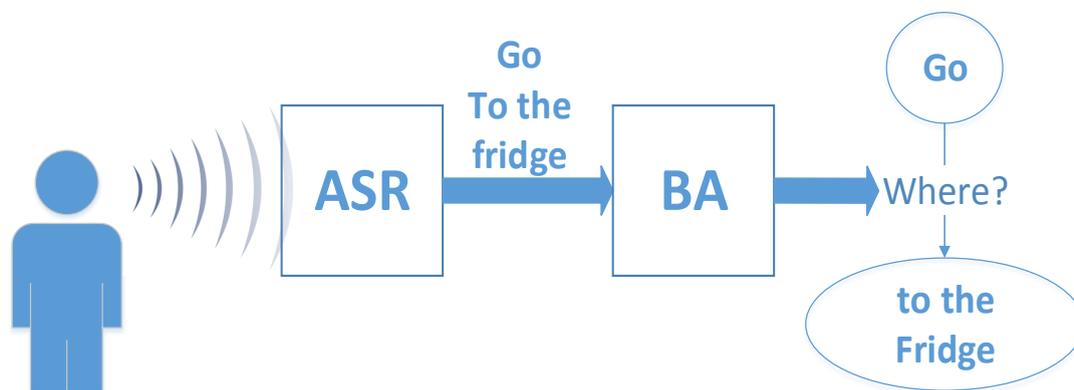


Abbildung 1.1.: Einordnung der hier erstellten Anwendung

Auf der linken Seite sieht man eine Person, die mithilfe von gesprochener Sprache mit ARMAR-III kommuniziert. Das dadurch erzeugte akustische Signal wird mithilfe eines automatischen Spracherkenners (ASR) in Text überführt. Die Ausgabe des automatischen Spracherkenners dient als Eingabe für die in dieser Bachelorarbeit zu erstellende Anwendung, hier als „BA“ bezeichnet. Diese Anwendung soll dann einen Graphen erzeugen, welcher als initiale Handlungsrepräsentation dient. Dass die initiale Handlungsrepräsentation letztendlich durch einen Graphen dargestellt wird, wurde in einer weiteren Bachelorarbeit unter [Sch15] diskutiert. Dabei wurde festgestellt, dass eine Graphenrepräsentation ausreicht, um jegliche Informationen bezüglich der Sprachverarbeitung darzustellen.

Diese Ausarbeitung stellt zunächst die benötigten Grundlagen der Sprachverarbeitung kompakt vor. Anschließend werden verwandte Arbeiten diskutiert und auf ihre Anwendbarkeit bezüglich der in dieser Ausarbeitung verbundenen Herausforderungen untersucht. Danach wird anhand einer Analyse ein Entwurf aufgestellt, welcher als Grundlage für die Implementierung dient. Abschließend wird eine Evaluation durchgeführt, um das Ergebnis dieser Ausarbeitung bewerten zu können.

## 2. Grundlagen

Dieses Kapitel stellt die Grundlagen der Sprachverarbeitung vor, die bezüglich dieser Ausarbeitung von Bedeutung sind. Welche Themen innerhalb der Sprachverarbeitung relevant sind, soll erneut mithilfe der Abbildung unter 1.1 verdeutlicht werden. Das hier zu erstellende Werkzeug nimmt als Eingabe die Ausgabe eines automatischen Spracherkenners entgegen. Da es sich beim automatischen Spracherkenner um ein separates Modul handelt, wird in diesem Kapitel die grundlegende Funktionsweise eines automatischen Spracherkenners lediglich kompakt vorgestellt. Jedoch sollte ausführlich untersucht werden, welche potenziellen Fehler ein automatischer Spracherkenner in der Ausgabe produziert, da diese Ausgabe als Eingabe des hier erstellten Werkzeuges dient. Der Schwerpunkt des Kapitels liegt auf der Informationsextraktion. Dabei werden Informationen vorgestellt, die mithilfe von Rechnerlinguistik-Werkzeugen extrahiert werden können. Im Rahmen der Sprachverarbeitung wird das Extrahieren von Informationen in einzelne zu bewerkstelligende Aufgaben unterteilt, die auch als **NLP-Aufgaben** (engl. **N**atural **L**anguage **P**rocessing) bezeichnet werden. Die gewonnenen Informationen können innerhalb einer initialen Handlungsrepräsentation dargestellt werden. Zu den NLP-Aufgaben zählen unter anderem die

- Tokenisierung (engl. Tokenization),
- Wortartenerkennung (engl. Part of Speech Tagging - POS Tagging),
- Lemmatisierung (engl. Lemmatization),
- Phrasenerkennung (engl. Chunking/ Shallow Parsing),
- Syntaxanalyse (engl. Syntactic Parsing - PSG),
- Eigennamenerkennung (engl. Name Entity Recognition - NER),
- Zuweisung von semantischen Rollen (engl. Semantic Role Labeling - SRL),
- und Koreferenzanalyse (engl. Coreference Resolution).

Aktuelle Sprachverarbeitungswerkzeuge, die diese Aufgaben adressieren, werden in Abschnitt 3.1 diskutiert. Die Untersuchung und Bewertung der Nutzbarkeit aktueller Sprachverarbeitungswerkzeuge, hinsichtlich der Herausforderungen des hier zu erstellenden Werkzeuges, wird im Analysekapitel unter 4.2 vorgenommen.

## 2.1. Automatische Spracherkennung und die Herausforderungen von gesprochener Sprache

Unter der automatischen Spracherkennung (engl. **Automatic Speech Recognition** - ASR) wird die maschinelle Umsetzung von gesprochener Sprache in geschriebenen Text verstanden. Dabei wird die automatische Spracherkennung häufig als Prozess des Musterabgleichs charakterisiert. Diese Einschätzung beruht auf dem vorherrschenden Verfahren, mit dem der Spracherkennung ein unbekanntes akustisches Signal als Muster interpretiert und mit einer Reihe von gespeicherten Referenzmustern vergleicht. Die Referenzmuster werden zuvor in einer Trainingsphase erzeugt. Die Entscheidung, welche sprachliche Äußerung vorliegt, trifft der Spracherkennung anhand eines Ähnlichkeitsmaßes, welches dem Vergleich zwischen dem aktuellen Muster und den Referenzmustern zugrunde liegt. Dabei verwendet die automatische Spracherkennung überwiegend Techniken der statistischen und strukturellen Mustererkennung, wie dem **Hidden-Markov Modell** [KUCC<sup>+</sup>10, S.216].

Da die Herausforderungen von gesprochener Sprache bereits unter 1.2 motiviert wurden, sollen die Schwierigkeiten in der Verarbeitung der Ausgabe eines automatischen Spracherkenners mithilfe einer beispielhaften Ausgabe erläutert werden:

*„Put eh the the dirty plates into the dishwasher after that burn the dishwasher on“* [2]

Zunächst einmal ist hervorzuheben, dass es einen Unterschied zwischen geschriebener und gesprochener Sprache gibt. In diesem Beispiel findet man das **Füllwort** „ehh“ und die **Wortwiederholung** „the the“. In einer weiteren Bachelorarbeit [Gue15] wurde ein Sprachkorpus mithilfe von Sprachaufnahmen erstellt, die Anweisungen an ARMAR-III simulieren. Bei der Analyse dieses Sprachkorpus ist aufgefallen, dass Wortwiederholungen und Füllwörter die am häufigsten auftretenden Phänomene von gesprochener Sprache darstellen. Es gibt noch weitere Besonderheiten von gesprochener Sprache, wie Ellipsen und das häufige Verwenden von Pronomen. Da diese Besonderheiten jedoch selten im Sprachkorpus auftraten, soll an dieser Stelle nicht genauer darauf eingegangen werden. Neben den Herausforderungen, die mit gesprochener Sprache verbunden sind, gibt es auch Erschwernisse, die auf den automatischen Spracherkennung zurückzuführen sind. Zu diesen Erschwernissen gehört vor allem das Auftreten von **Wortfehlern**. Unter [2] wurde beispielsweise das Wort „burn“ offensichtlich fehlerhaft vom automatischen Spracherkennung erkannt. **Wortfehler** treten im Sprachkorpus häufig auf und müssen dementsprechend gesondert betrachtet werden. Zusätzlich beinhaltet dieses Beispiel keine Satztrennungszeichen. Es existieren zwar automatische Spracherkennung, die Satztrennungszeichen anhand von Pausen und der Grundfrequenz der Schallwellen ableiten können, jedoch wurde im Gesamtprojekt „PARSE“ bewusst auf diese Funktion verzichtet. Im Analysekapitel unter 4 wird diese Entscheidung genauer diskutiert.

## 2.2. NLP-Aufgaben

In diesem Abschnitt werden die NLP-Aufgaben der Sprachverarbeitung vorgestellt. Dabei ist das Ziel Informationen aus einem Eingabetext zu extrahieren. Diese Informationen können anschließend innerhalb einer initialen Handlungsrepräsentation dargestellt werden. Die im Folgenden vorgestellten Inhalte wurden überwiegend aus den Werken „*Speech and Language Processing*“ von Daniel Jurafsky [DJH09] und „*Computerlinguistik und Sprachtechnologie*“ von Kai-Uwe Carstensen [KUCC<sup>+</sup>10] entnommen.

### 2.2.1. Tokenisierung

Vor der eigentlichen Extraktion von Informationen, muss der Eingabetext in linguistische Einheiten, wie Wörter oder Sätze, segmentiert werden. Dieser Segmentierungsvorgang wird

als **Tokenisierung** bezeichnet. Ansätze zur Tokenisierung lassen sich in symbolische und statistische Verfahren unterteilen. Ersteres beruht hauptsächlich auf regulären Ausdrücken, die sich auf Heuristiken zurückführen lassen, wohingegen statistische Verfahren die Erkennung von Satzgrenzen als Klassifikationsproblem auffassen, welches sich durch die Analyse morphosyntaktischer Merkmale eines Wortes am potentiellen Satzende und dessen Kontext lösen lässt [KUCC<sup>+</sup>10, S.270].

Im Englischen, als auch im Deutschen, werden **segmentierte Schriftsysteme** verwendet, wohingegen Chinesisch und Japanisch Beispiele für **nicht-segmentierte Schriftsysteme** darstellen. Die Verwendung eines segmentierten Schriftsystems bedeutet, dass einzelne Wörter durch ein Leerraumzeichen voneinander getrennt werden. Jedoch gibt es einige Ausnahmen, wie beispielsweise Kontraktionen, Abkürzungen oder das Genitiv-S. Im Bereich der Wortsegmentierung wird ein einzelnes Wort auch als **Token** bezeichnet. Neben der Wortsegmentierung kann auch auf Satzebene segmentiert werden. Die Herausforderungen bezüglich der Satzsegmentierung soll folgendes Beispiel verdeutlichen:

*He stopped at Meadows Dr. White Falcon was still open.*

Da im Englischen, als auch im Deutschen, einzelne Sätze durch einen Punkt voneinander getrennt werden, scheint die Satzsegmentierung zunächst unproblematisch. Betrachtet man jedoch das obige Beispiel, fällt auf, dass das Wort „Dr.“ einen Punkt enthält, welcher kein Satzende repräsentiert, sondern für die Abkürzung *Drive* steht. Dementsprechend wird der Punkt nicht ausschließlich zur Satzsegmentierung verwendet, sondern beispielsweise auch um Abkürzungen, Datumsangaben oder Trennzeichen in Zahlen zu kennzeichnen.

Bei der Betrachtung des Sprachkorpus unter [Gue15] ist aufgefallen, dass eine triviale Tokenisierung auf Wortebene anhand der Leerraumzeichen ausreicht. Die Tokenisierung auf Satzebene kann hingegen nicht durchgeführt werden, weil die Ausgaben des automatischen Spracherkenners, wie in 2.1 beschrieben, keine Satztrennungszeichen beinhalten. Im Analysekapitel unter 4 wird die damit verbundene Problematik genauer diskutiert.

### 2.2.2. Wortartenerkennung

Nach der Tokenisierung des Eingabetextes folgt die Anreicherung um grammatikalische Informationen. Dabei wird zunächst jedes Token gemäß seiner Wortart klassifiziert [KUCC<sup>+</sup>10, S.271]. Ein System, das solch eine Analyse und Generierung leistet, wird als **Wortartenerkennung** (engl. **Part-of-Speech Tagger/ POS-Tagger**) bezeichnet.

Ein wichtiges Kriterium für die Performanz eines Wortartenerkenners ist das sogenannte **Tagset**. Das **Tagset** stellt die Auswahl des Inventars an Wortarten dar. Alle Sprachverarbeitungswerkzeuge im Abschnitt 3.1 verwenden den **Penn Treebank-Standard**, welcher sich im Anhang unter A befindet.

Verfahren zur Wortartenerkennung lassen sich in regelbasierte und stochastische Verfahren unterteilen. Während stochastische Verfahren über Wahrscheinlichkeiten, dass ein Wort im betrachteten Kontext einer bestimmten Wortart angehört, operieren, ordnen regelbasierte Wortartenerkennung dem Token in einem ersten Schritt diejenige Wortart zu, mit dem es im Trainingskorpus am häufigsten annotiert wurde [KUCC<sup>+</sup>10, S.271]. Es existieren auch Verfahren, die sowohl regelbasierte als auch stochastische Ansätze miteinander kombinieren. Der **Bill-Tagger** [DJH09, S.185] ist solch ein Beispiel.

Da in der späteren Analyse der Schwerpunkt auf der Wortartenerkennung liegt, weil diese die Basis für alle weiteren NLP-Aufgaben darstellt, soll eine beispielhafte Ausgabe verdeutlichen, wie Sprachverarbeitungswerkzeuge ein Token einer Wortart zuordnen:

*Go*/*VB* *to*/*TO* *the*/*DT* *fridge*/*NN* ./.

Hier wird das Wort „Go“ durch die Markierung „VB“ als Verb gekennzeichnet. In dieser Ausgabe wird das Token und die Wortart mithilfe eines Schrägstriches voneinander getrennt. Die Verwendung eines Schrägstriches entspricht keiner Norm. Manche Sprachverarbeitungswerkzeuge verwenden stattdessen einen Unterstrich oder eine tabellarische Darstellung. Einen für das Beispiel passenden Auszug des **Penn Treebank-Standards** wurde unter Tabelle 2.1 bereitgestellt.

Markierung	Beschreibung
VB	Verb Basisform
TO	to
DT	Artikel
NN	Nomen im Singular oder Massennomen

Tabelle 2.1.: Auszug des Penn Treebank-Standards

### 2.2.3. Lemmatisierung

Bei der Lemmatisierung (engl. Lemmatization) wird jedem Wort eines laufenden Textes seine Grundform zugeordnet. Die Grundform eines Wortes wird auch als **Lemma** bezeichnet. Ein Lemma steht stellvertretend für eine Menge von Wörtern, die den gleichen Wortstamm, die gleiche Wortklasse und einen ähnlichen Wortsinn besitzen [DJH09, S.120]. Um das Lemma eines Tokens zuverlässig zu bestimmen, ist eine vollständige morphologische Analyse des Wortes notwendig. Beispielsweise repräsentiert „sing“ das Lemma für die Wörter „sing“, „sang“ und „sung“.

Wenige der unter 3.1 aufgelisteten Sprachverarbeitungswerkzeuge bieten die Möglichkeit einer expliziten Lemmatisierung. Jedoch muss beachtet werden, dass häufig vor einer semantischen Analyse des Eingabetextes, wie beim Zuweisen von semantischen Rollen, das unter 2.2.7 vorgestellt wird, eine Lemmatisierung durchgeführt wird. Durch die vorherige Lemmatisierung kann, im Rahmen der unter 2.2.7 vorgestellten Zuweisung von semantischen Rollen, darauf verzichtet werden für jede Konjugation einen separaten PropBank-Eintrag zu hinterlegen.

### 2.2.4. Syntaxanalyse

In der linguistischen Tradition lassen sich hinsichtlich der Frage, wie syntaktische Strukturen dargestellt werden können, zwei Ansätze unterscheiden. Der erste Ansatz fasst syntaktische Strukturen als Relation zwischen Wörtern auf. Es handelt sich hierbei um die **Dependenzsyntax** oder **Abhängigkeitssyntax**. Im zweiten Ansatz, die **Konstituentenstruktursyntax**, werden neben Wörtern auch komplexere Einheiten, die sogenannten **Konstituenten** definiert. In Abgrenzung zum ersten Ansatz werden die Relationen zwischen den Konstituenten aufgebaut. In beiden Ansätzen können Strukturgraphen zur Notation verwendet werden.

In der Dependenzgrammatik wird meist angenommen, dass das Verb die Struktur eines Satzes in entscheidendem Maße festlegt. Alle anderen Wörter sind unmittelbar oder mittelbar vom Verb abhängig [KUCC<sup>+</sup>10, S.282]. Zwischen den Knoten, welche die Wörter des Eingabesatzes repräsentieren, gibt es typisierte Kanten, die aus einem fixen Inventar von grammatikalischen Relationen bestehen. Ein Beispiel für solch ein Inventar befindet sich unter [MCBCD06]. Eine Teilmenge des Inventars wurde im Anhang unter C bereitgestellt. Unter der Abbildung 2.1 befindet sich die beispielhafte Ausgabe eines typisierten **Dependenzparsers** für den Eingabesatz „*They hid the letter on the shelf*“.

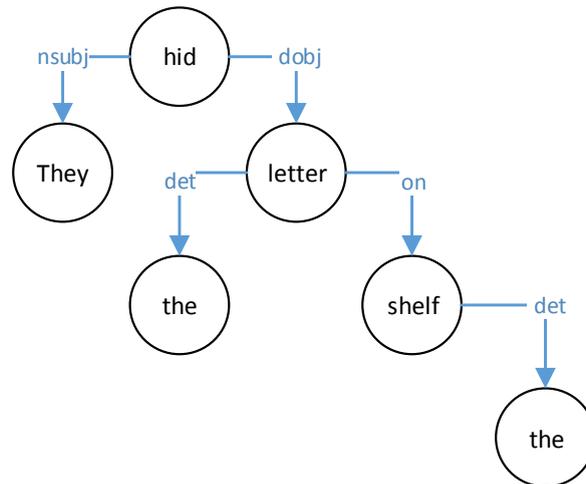


Abbildung 2.1.: Ausgabe eines typisierten Dependenzparsers

Die Konstituentenparser basieren auf der Idee, dass sich die Syntax von Sprachen, wie Englisch und Deutsch, mithilfe von kontextfreien Grammatiken beschreiben lassen. Es ist nicht bewiesen, ob sich die gesamte englische Syntax vollständig durch eine kontextfreie Grammatik beschreiben lässt, jedoch wird dies vermutet [DJH09, S.576]. Als Beispiel, aus [KUCC<sup>+</sup>10, S.284] entnommen, soll ein Fragment einer kontextfreien Grammatik im Deutschen dienen:

$$\begin{aligned}
 G = & \left( \left\{ S, NP, VP, DET, N, V \right\}, \right. \\
 & \left\{ der, Hund, bellt, sieht, die, Katze \right\}, \\
 & \left\{ \begin{array}{ll} S & \rightarrow NP VP, \\ NP & \rightarrow DET N, \\ VP & \rightarrow V, \\ VP & \rightarrow V NP, \\ DET & \rightarrow der, \\ DET & \rightarrow die, \\ N & \rightarrow Hund, \\ N & \rightarrow Katze, \\ V & \rightarrow bellt, \\ V & \rightarrow sieht \end{array} \right\}, \\
 & S)
 \end{aligned}$$

Mithilfe der kontextfreien Grammatik  $G$  lässt sich der Satz „Der Hund bellt“ als Baumgraph, wie unter Abbildung 2.2, beschreiben. Eine alternative Notation für die Ausgabe des Syntaxbaumes stellt die Klammerdarstellung dar. Für die Abbildung 2.2 würde die Klammerdarstellung folgendermaßen lauten:

$$(S (VP (NP (DET Der) (N Hund)) (VP (V bellt))))$$

Die Nichtterminalsymbole repräsentieren die Konstituenten. So steht das Nichtterminalsymbol  $NP$  für eine Nominalphrase, auch nominale Konstituente. Die verwendeten Konstituenten sind unter den Sprachverarbeitungswerkzeugen am **Penn Treebank-Projekt** [MAB03] orientiert.

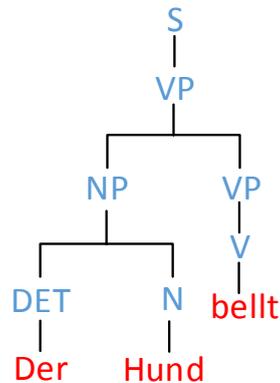


Abbildung 2.2.: Ausgabe eines Konstituentenparser

Aktuelle Konstituentenparser erhalten als Eingabe neben dem tokenisierten Eingabetext die entsprechenden Wortarten zu jedem Token und geben einen Syntaxbaum, wie unter Abbildung 2.2, aus. Dabei bilden der CKY - und Early-Algorithmus [DJH09, S.461] die Basis für die Generierung der Aufgabe.

### 2.2.5. Phrasenerkennung durch Chunking

Anders als bei traditionellen Parsern eines Satzes, wie im Unterabschnitt 2.2.4 vorgestellt, liefert die **Phrasenerkennung durch Chunking** keine vollständigen Syntaxstrukturen, die hierarchische Beziehungen zwischen Konstituenten repräsentieren, sondern identifiziert nebengeordnete und nicht überlappende Teilstrukturen (engl. *chunks*), denen phrasale Markierungen zugeordnet werden. Der signifikanteste Unterschied zwischen der Phrasenerkennung und der Syntaxanalyse liegt darin, dass Phrasenerkennung keine Relationen zwischen den erkannten Teilstrukturen aufbauen. Durch dieses Verhalten bietet die Phrasenerkennung eine Möglichkeit, syntaktische Informationen für Eingabetexte ohne Satztrennungszeichen zu erhalten. Die Anwendbarkeit der Phrasenerkennung im Vergleich zur Syntaxanalyse, bezüglich dieser Ausarbeitung, wird im Analysekapitel unter 4 diskutiert. Phrasenerkennung bauen entweder auf regelbasierten Ansätzen, bei denen endliche Zustandsautomaten zum Einsatz kommen oder maschinellen Lernverfahren auf [DJH09, S.486].

Für die Phrasenerkennung wird ein Eingabetext erwartet, welcher bereits in Token segmentiert wurde. Zusätzlich wird vorausgesetzt, dass jedes Token einer Wortart zugeordnet wurde. Eine beispielhafte Ausgabe einer Phrasenerkennung, soll folgendes Beispiel verdeutlichen:

[<sub>NP</sub>The morning flight] [<sub>PP</sub> from] [<sub>NP</sub> Denver][<sub>VP</sub> has arrived.]

Der Satz „*The morning flight from Denver has arrived*“ ist hier in vier nicht überlappenden Phrasen unterteilt. Beispielsweise wurde „*The morning flight*“ mithilfe der phrasalen Markierung „*NP*“ als Nominalphrase klassifiziert. Die einzelnen phrasalen Markierungen sind, einschließlich ihrer Beschreibung, im Anhang unter B zu finden. Neben der obigen Darstellung des Beispiels, die ausschließlich zur Veranschaulichung dient, werden folgende zwei Ausgabeformate bei den Sprachverarbeitungswerkzeugen im Abschnitt 3.1 verwendet:

Token	IOB - Format	IOBES -Format
The	B-NP	B-NP
morning	I-NP	I-NP
flight	I-NP	E-NP
from	B-PP	S-PP
Denver	B-NP	S-NP
has	B-VP	B-VP
arrived	I-VP	E-VP
.	O	O

Die erste Spalte dieser Tabelle stellt die einzelnen Token dar. In der zweiten Spalte werden die Phrasen mithilfe des IOB - Formats (engl. **I**nside-**O**utside-**B**egin-Format) gekennzeichnet. Dabei markiert „B“ den Beginn und „I“ das Innere einer Phrase. Das Ende einer Phrase kann durch den Beginn einer neuen Phrase impliziert werden. Jedes Token, das keiner Phrase zugeordnet werden konnte, wird durch „O“ gekennzeichnet. Die dritte Spalte stellt das IOBES -Format (engl. **I**nside-**O**utside-**B**egin-**E**nd-**S**ingle-Format) dar. Dieses Format entspricht dem IOB-Format, es besteht lediglich zusätzlich die Möglichkeit, das Ende einer Phrase durch ein „E“ und Phrasen, bestehend aus einem einzelnen Token, mit „S“ zu kennzeichnen.

Bezüglich der Forschung und Entwicklung von Phrasenerkennern muss noch die **CoNLL-Konferenz** (engl. **C**onference **o**n **N**atural **L**anguage **L**earning) genannt werden. Bei dieser Konferenz werden Aufgaben bezüglich der Sprachverarbeitung formuliert, die dazu dienen, Sprachverarbeitungswerkzeuge miteinander und den aktuellen Forschungsstand untereinander zu vergleichen. Im Jahr 2000 hat die CoNLL-Konferenz eine zu lösende Aufgabe bezüglich der Phrasenerkennung veröffentlicht. Dabei wurde das IOB - Format als Standard festgesetzt und stellt dadurch bis heute das gängigste Format für Phrasenerkennung dar.

### 2.2.6. Eigennamenerkennung

Bei der Eigennamenerkennung (engl. **N**amed **e**ntity **r**ecognition - NER), wird der Eingabetext in vordefinierte Kategorien, wie Personen, Standorte, Organisationen, Zeiten und Quantitäten klassifiziert [EFT02]. Dabei werden Daten-gesteuerte Verfahren wie überwachte und semi-überwachte Strategien angewendet. Da die von der Eigennamenerkennung extrahierten Informationen keine Bedeutung für das Projekt „PARSE“ besitzen, aber einige Sprachverarbeitungswerkzeuge unter 3.1 diese Funktionalität anbieten, soll ausschließlich eine exemplarische Ausgabe die Eigennamenerkennung verdeutlichen:

[PER Wolf], currently a journalist in [LOC Argentina], played with [PER Del Bosque] in the final years of the seventies in [ORG Real Madrid] .

Im Eingabetext „*Wolf, currently a journalist in Argentina, played with Del Bosque in the final years of the seventies in Real Madrid.*“ wurden vier Entitäten erkannt. Dabei wurden „*Wolf*“ und „*Del Bosque*“ durch *PER* als Person, „*Argentina*“ durch *LOC* als Standort und „*Real Madrid*“ durch *ORG* als Organisation gekennzeichnet.

### 2.2.7. Zuweisung von semantischen Rollen

Bei der Zuweisung von semantischen Rollen (engl. **S**emantic **r**ole **l**abeling - SRL) handelt es sich um eine Aufgabe innerhalb der Sprachverarbeitung, die semantische Prädikat-Argument-Strukturen erkennt. Wird beispielsweise der Satz „*Markus sold the book to*

*Christina.*“ betrachtet, ist die Aufgabe der Zuweisung von semantischen Rollen, das Verb „sell“ als Prädikat, „Markus“ als Verkäufer, „book“ als die verkaufte Ware und „Christina“ als Empfängerin zu kennzeichnen. Zur Kennzeichnung der semantischen Rollen haben sich zwei Standards entwickelt, nämlich die **Proposition Bank/ PropBank**- und die **FrameNet**-Annotation. Diese beiden Annotationen wurden in ihren gleichnamigen Projekten, dem PropBank-Projekt [MDK05] bzw. dem FrameNet-Projekt [CFCJL98] entwickelt.

Da alle Sprachverarbeitungswerkzeuge unter 3.1 die PropBank-Annotation verwenden, soll hierauf der Fokus liegen. Die grundlegende Idee der PropBank-Annotation ist, dass jedes Verb spezifische Mengen an semantischen Rollen besitzt. Dabei werden die einzelnen Rollen eines Verbes durch Nummern anstatt repräsentativer Namen gekennzeichnet (**Arg0**, **Arg1**, **Arg2** usw.). Im Allgemeinen, repräsentiert **Arg0** den *Handelnden* (engl. Agent) und **Arg1** den *Behandelten* (engl. Patient). Ab dem zweiten Argument (**Arg2**) ist die semantische Rolle vom zugehörigen Verb abhängig. Für den obigen Beispielsatz „*Markus sold the book to Christina*“ könnte die Ausgabe eines Sprachverarbeitungswerkzeuges für die Zuweisung von semantischen Rollen folgendermaßen aussehen:

[Arg0 Markus] sold [Arg1 the book] [Arg1 to Christina.]

Diese Ausgabe ist nun mithilfe des PropBank-Eintrages für das Verb „sell“, welcher sich unter 3.3 befindet, interpretierbar. Unter [Pro] sind alle PropBank-Einträge abrufbar.

Arg0:	Verkäufer
Arg1:	verkaufter Gegenstand
Arg2:	Käufer
Arg3:	bezahlter Preis
Arg4:	Benefaktiv

Tabelle 2.2.: Übersetzter Auszug des PropBank-Eintrages für das Verb „sell“

Für die Zuweisung von semantischen Rollen ist ein tokenisierter Eingabetext erforderlich, welcher mit Wortarten annotiert und mithilfe der Syntaxanalyse strukturiert wurde. Die Zuweisung von semantischen Rollen baut somit meist direkt auf den Syntaxbaum auf. Jedoch können unterschiedliche Syntaxbäume zu selben semantischen Rollen führen. Beispielsweise enthält der Satz „*The book was sold by Markus to Christina.*“ dieselben semantischen Rollen wie im obigen Beispiel, obwohl sich die syntaktische Struktur unterscheidet. Die Umsetzung von der Zuweisung von semantischen Rollen stellt sich als eine komplexe Aufgabe dar. Grundsätzlich sind eine Prädikatenlogik und maschinelle Lernverfahren notwendig. Für weitere Informationen sei auf [DJH09, S.579-714] verwiesen.

### 2.2.8. Koreferenzanalyse

Die Koreferenzanalyse stellt eine weitere Aufgabe innerhalb der Sprachverarbeitung dar. **Koreferenz** liegt vor, wenn in einer Äußerung mit zwei unterschiedlichen sprachlichen Ausdrücken dasselbe bezeichnet wird [Kor]. Zur Verdeutlichung sollen folgende zwei Sätze betrachtet werden:

- (1) *Bob* sagte, *er* schreibe seine Bachelorarbeit.
- (2) *Er* ist ein begabter amerikanischer Schauspieler. Die Rede ist von *Bob*.

In beiden Beispielen sind *Bob* und *er* koreferent. Im ersten Beispiel stellt *er* eine sogenannte **Anapher** dar. Eine **Anapher** stellt den Verweis eines Satzteiles auf einen anderen, vor ihm stehenden, Satzteil auf. Im zweiten Beispiel verweist *er* auf eine nachfolgende sprachliche Einheit und wird als **Katapher** bezeichnet.

Die Koreferenzanalyse ist im Rahmen des Gesamtprojektes „PARSE“ essentiell, da die Verwendung von Pronomen, die auf andere sprachliche Einheiten verweisen, ein bekanntes Phänomen von gesprochener Sprache darstellt. Jedoch wird diese Ausarbeitung nicht die Implementierung der Koreferenzanalyse umsetzen, da dies innerhalb der unter 1.1 vorgestellten Agenten, in Verbindung mit einer *Diskursanalyse*, realisiert wird.

### 2.3. Offene und abgeschlossene NLP-Werkzeuge

Nachdem die Aufgaben der Sprachverarbeitung (NLP-Aufgaben) erläutert wurden, soll dieser Abschnitt die gegenseitige Abhängigkeit der einzelnen Aufgaben verdeutlichen. Hierbei soll die Abbildung unter 2.3 zur Unterstützung dienen.

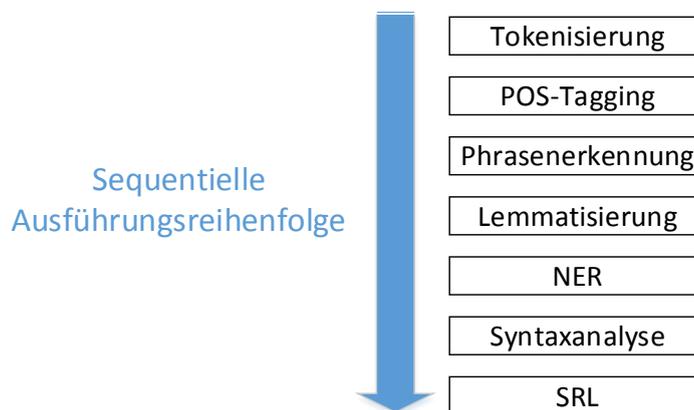


Abbildung 2.3.: Die **Pipeline** der NLP-Aufgaben

Verallgemeinert lässt sich feststellen, dass die einzelnen NLP-Aufgaben sequentiell voneinander abhängig sind (Pipeline). Dabei wird der Eingabetext zunächst mithilfe der Tokenisierung segmentiert, anschließend in Wortarten klassifiziert, dann durch die Phrasenerkennung analysiert usw.

Eine grundlegende Charakteristik von aktuellen Sprachverarbeitungswerkzeugen ist die Möglichkeit, zwischen den einzelnen Aufgaben Einfluss zu nehmen. Verfügt ein Sprachverarbeitungswerkzeug über die Möglichkeit, nicht nur den Eingabetext, sondern auch weitere Informationen entgegenzunehmen, um die Ausführungsreihenfolge unter Abbildung 2.3 fortzusetzen, so wird das Sprachverarbeitungswerkzeug in dieser Ausarbeitung als **offenes Sprachverarbeitungswerkzeug** bezeichnet. Bietet ein Werkzeug lediglich die Möglichkeit den Eingabetext ohne jegliche Zusatzinformationen entgegenzunehmen, wird es als **abgeschlossenes Sprachverarbeitungswerkzeug** kategorisiert. Zur Verdeutlichung soll ein Beispiel dienen: Abhängig vom Ausgangsszenario kann es möglich sein, dass neben dem Eingabetext die korrekten Wortarten bereits vorliegen. In solch einem Szenario wäre es wünschenswert, einem Sprachverarbeitungswerkzeug neben dem Eingabetext die korrekten Wortarten als Eingabe zu übergeben, sodass die darauf aufbauenden Aufgaben, wie die Phrasenerkennung, mit den korrekten Wortarten ausgeführt werden. Falls ein Sprachverarbeitungswerkzeug diese Möglichkeit anbietet, wird es als *offen* charakterisiert, andernfalls als *geschlossen*.



## 3. Verwandte Arbeiten

Um die Zielsetzung unter 1.2 zu erfüllen, wurden zu Beginn dieser Ausarbeitung folgende zwei Fragen betrachtet:

1. Mit welchen Werkzeugen können die Informationen, die in einer initialen Handlungsrepräsentation vorhanden sein sollen, siehe Abschnitt 2.2, extrahiert werden?
2. Welche Gesamtsysteme wurden bereits realisiert, die mit dieser Ausarbeitung und dem Gesamtprojekt „PARSE“ vergleichbar sind und wie lassen sich die dort verwendeten Ansätze für diese Ausarbeitung verwenden?

Entsprechend dieser Fragen wurde dieses Kapitel strukturiert. Zunächst sollen aktuelle Sprachverarbeitungswerkzeuge vorgestellt werden. Anschließend werden komplette Systeme diskutiert, die mit dieser Ausarbeitung und dem Gesamtsystem „PARSE“ vergleichbar sind.

### 3.1. Sprachverarbeitungswerkzeuge

Im Grundlagenkapitel, unter Abschnitt 2.2, wurden die zu bewerkstellenden NLP-Aufgaben der Sprachverarbeitung vorgestellt, die als Ergebnis extrahierte Informationen bereitstellen, welche in einer initialen Handlungsrepräsentation dargestellt werden können. Zur Bewerkstelligung der NLP-Aufgaben wurden Sprachverarbeitungswerkzeuge entwickelt. Dabei befasst sich ein Sprachverarbeitungswerkzeug meist mit mehreren NLP-Aufgaben. Dieser Abschnitt stellt aktuelle Sprachverarbeitungswerkzeuge vor, wobei die Bewertung und Anwendbarkeit der einzelnen Sprachverarbeitungswerkzeuge im Analysekapitel unter 4 vorgenommen wird.

#### 3.1.1. SENNA

Bei *SENNA* [RJL<sup>+</sup>11] handelt es sich um eine in C geschriebene Sprachverarbeitungssoftware, welche Vorhersagen über die Tokenisierung, Wortartenerkennung, Phrasenerkennung, Eigennamenerkennung, Syntaxanalyse und der Zuweisung von semantischen Rollen zu einem gegebenen Eingabetext treffen kann. Die Funktionsweise von *SENNA* wird durch ein mehrschichtiges neuronales Netzwerk realisiert. Die genaue Architektur und Implementierung wird unter [RJL<sup>+</sup>11] beschrieben. Entsprechend der Kategorisierung im Abschnitt 2.3 handelt es sich bei [RJL<sup>+</sup>11] um ein *abgeschlossenes Sprachverarbeitungswerkzeug*. Um eine Vorhersage zu erhalten, muss *SENNA* lediglich ein Text als Eingabe übergeben werden. Zur Verdeutlichung folgt eine exemplarische Ausgabe von *SENNA* für den Eingabetext „Go there and take the popcorn.“:

Go	VB	S-VP	O	Go	S-V	O	(S1(S(S(VP(VP*
there	RB	S-ADVP	O	-	S-AM-DIR	O	(ADVP*))
and	CC	O	O	-	O	O	*
take	VB	S-VP	O	take	O	S-V	(VP*
the	DT	B-NP	O	-	B-A1	B-A1	(NP*
popcorn	NN	E-NP	O	-	E-A1	E-A1	*)
.	.	O	O	-	O	O	*)

In der ersten Spalte steht der Eingabetext, welcher auf Wortebene zeilenweise segmentiert wurde (Tokenisierung). In der zweiten Spalte findet man zu jedem Wort seine entsprechende Wortart. Hier wird beispielsweise das Wort „Go“ korrekterweise durch „VB“ als Verb gekennzeichnet. Die Phrasenerkennung findet man in der dritten Spalte. Dabei werden die Phrasen gemäß des IOBES-Formats, siehe Abschnitt 2.2.5, gekennzeichnet. In der vierten Spalte werden die erkannten Eigennamen aufgelistet, was in diesem Beispiel zu keinem Ergebnis führt, da im Eingabetext keine Eigennamen vorhanden sind. Die fünfte Spalte kennzeichnet alle erkannten Verben. Die sechste und siebte Spalte geben die Zuweisung der semantischen Rollen entsprechend der PropBank-Annotation an. Jedes erkannte Verb wird einschließlich seiner Argumente spaltenweise dargestellt. Somit ergeben sich für diesen Eingabetext mit zwei Verben, zwei Spalten. Zuletzt wird der Syntaxbaum mithilfe der Klammerdarstellung angegeben.

Bei *Senna* ist eine Besonderheit in den Ausgaben der Phrasenerkennung und der darauf aufbauenden NLP-Aufgaben aufgefallen. In der üblichen Architektur von Sprachverarbeitungswerkzeugen ist vor der Wortartenerkennung eine Tokenisierung erforderlich, die Phrasenerkennung und Syntaxanalyse bauen wiederum auf der Wortartenerkennung auf und zuletzt wird die Zuweisung der semantischen Rollen aufgrund der Erkenntnisse der Phrasenerkennung und Syntaxanalyse durchgeführt. Aus diesem theoretischen Grundverständnis kann angenommen werden, dass Fehler, die beispielsweise in der Syntaxanalyse auftreten, zu entsprechenden Folgefehlern in der Zuweisung der semantischen Rollen führen. Aufgrund des bei *SENNA* angewendeten neuronalen Netzwerkes, ist das Auftreten von solchen Folgefehlern nicht zwingend, da die einzelnen Schichten des neuronalen Netzwerkes, die jeweils eine NLP-Aufgabe repräsentieren, nicht isoliert voneinander verbunden sind.

### 3.1.2. Memory-Based Shallow Parser (MBSP)

Bei *MBSP* [TVW05] handelt es sich um ein als Pythonmodul realisiertes Sprachverarbeitungswerkzeug, welches Vorhersagen über die Tokenisierung, Wortartenerkennung, Phrasenerkennung, Lemmatisierung und Erkennung von präpositionalen Nominalphrasen tätigt. Die Erkennung von präpositionalen Nominalphrasen ist keine der grundlegenden NLP-Aufgaben und stellt somit ein Alleinstellungsmerkmal gegenüber anderen Sprachverarbeitungswerkzeugen dar. Zusätzlich weicht die Phrasenerkennung von *MBSP* von der unter Abschnitt 2.2.5 vorgestellten Zielsetzung ab. *MBSP* versucht neben der Klassifikation von Phrasen, diese gegenseitig in Beziehung zu setzen. Somit werden beispielsweise einzelne Phrasen als Subjekt oder Objekt eines Satzes klassifiziert.

Wie bei *SENNA* soll die Funktionsweise anhand der Ausgabe eines Beispieltextes erläutert werden:

WORD	TAG	CHUNK	ROLE	ID	PNP	ANCHOR	LEMMA
Okay	UH	INTJ	-	-	-	-	okay
ARMAR	NNP	NP	SBJ	1	-	-	ARMAR
,	,	-	-	-	-	-	,
look	VBP	VP	-	1	-	A1	look
for	IN	PP	CLR	1	PNP	P1	for
the	DT	NP	CLR	1	PNP	P1	the
refrigerator	NN	NP^	CLR	1	PNP	P1	refrigerator
.	.	-	-	-	-	-	.

Der Eingabetext lautet hier „*Okay ARMAR, look for the refrigerator.*“. Anfangs ist zu erkennen, dass der tabellarischen Ausgabe eine Wortsegmentierung vorausgeht. Die Inhalte der ersten drei Spalten sind unter *MBSP* und *SENNA* identisch. Lediglich das Format der Phrasenerkennung unterscheidet sich. *MBSP* markiert Phrasen mithilfe eines Zirkumflexes (^). Dabei besteht die Markierung des ersten Wortes ausschließlich aus der Abkürzung der entsprechenden Phrase. Jedes darauffolgende Wort, welches dieser Phrase angehört, wird mit der gleichen Abkürzung und einem darauffolgenden Zirkumflex gekennzeichnet. So kennzeichnet beispielsweise *NP^* bei „*refrigerator*“, dass dieses Wort mit dem vorherigen Artikel „*the*“ eine Nominalphrase bildet. Die vierte und fünfte Spalte kennzeichnen Beziehungen zwischen Phrasen. Phrasen, die untereinander in Beziehung stehen, besitzen eine identische ID. In diesem Beispiel wird „*ARMAR*“ durch „*SBJ*“ als Subjekt des Satzes markiert. Das zugehörige Verb ist anhand eines fehlenden Eintrages in der vierten Spalte erkennbar und besitzt einen Eintrag unter *ID*. „*CLR*“ entspricht einer Phrase, die in enger Beziehung zum Verb steht. Die Beschreibung aller Tags von *MBSP* kann unter [MBS] gefunden werden. Die sechste Spalte stellt die erkannten präpositionalen Nominalphrasen dar. „*P1*“ kennzeichnet die präpositionale Nominalphrase, die das Verb „*look*“, welches durch „*A1*“ markiert ist, modifiziert. In der letzten Spalte wird das Lemma des entsprechenden Wortes angegeben.

*MBSP* ist in einer Server-Client-Architektur umgesetzt, die auf instanzbasierten Lernalgorithmen (engl. Memory Based Learning) aufbaut. Die Serverseite von *MBSP* unterteilt sich in vier Komponenten:

Modulname	Adresse	Aufgabe der Serverkomponente
MBSP.CHUNK	localhost:6061	Wortarten und Phrasenerkennung
MBSP.LEMMA	localhost:6062	Lemmatisierung
MBSP.PREPOSITION	localhost:6063	präpositionale Nominalphrasen
MBSP.RELATION	localhost:6064	Auffinden der Beziehungen zwischen Phrasen

Durch diese Architektur stellt *MBSP* ein *offenes* Sprachverarbeitungswerkzeug dar. Beispielsweise benötigt die Komponente MBSP.RELATION als Eingabe neben dem Eingabetext die entsprechenden Phrasen, um Beziehungen zwischen ihnen aufzufinden. Jedoch müssen die Phrasen nicht zwangsläufig von MBSP.CHUNK geliefert werden. Es ist aber nicht möglich, neben dem Eingabetext Wortarten anzugeben, um die Wortarten der Phrasenerkennung vorzugeben, da die Komponente MBSP.CHUNK die Klassifikation von Wortarten und Phrasen in einer gemeinsamen abgeschlossenen Komponente kapselt.

### 3.1.3. BIOS

Bei *BIOS* [Sur] handelt es sich um eine Software, welche die Aufgaben der Tokenisierung, Wortartenerkennung, Phrasenerkennung und Eigennamenerkennung erfüllt. Dabei muss beachtet werden, dass es sich bei der Wortartenerkennung nur um eine Hüllklasse des

*TnT taggers* [Bra00] handelt, welcher nicht mehr zu den aktuellsten Wortartenerkennern gehört. *BIOS* ist in Java implementiert und stellt eine übersichtliche Java-Schnittstelle zur Verfügung.

Jede der vier Aufgaben wird unabhängig voneinander ausgeführt. Somit stellt *BIOS* ein *offenes* Sprachverarbeitungswerkzeug dar. Möchte man beispielsweise die Phrasenerkennung mit vorhandenen Wortarten durchführen, wobei die Wortarten nicht durch den *TnT taggers* vorgegeben wurden, ist dies problemlos möglich. Vor der Verwendung von *BIOS* müssen Modelle zur Phrasen- und Eigennamenerkennung trainiert werden. Diese Modelle dienen im Allgemeinen, neben dem Eingabetext, als zusätzliche Eingabe und enthalten die durch ein maschinelles Lernverfahren, erlernten Wahrscheinlichkeiten und Heuristiken, welche der Klassifikator für die Klassifikation benötigt. Sprachverarbeitungswerkzeuge, die solche Modelle als zusätzliche Eingabe erwarten, besitzen den Vorteil, dass das zu verwendende maschinelle Lernverfahren nicht fest vorgegeben ist. Bei *BIOS* muss das Modell mithilfe der *Mil-Bibliothek* [Yan08] trainiert werden. Die *Mil-Bibliothek* bietet unter anderem die Maximum-Entropie-Methode und das Konzept der Perzeptrons an.

#### 3.1.4. Illinois NLP Curator

Die *Cognitive Computation Group* der Universität von Illinois stellt eine Vielzahl von Sprachverarbeitungswerkzeugen zur Verfügung. Darunter zählen unter anderem

- das *Illinois NLP Tool* zur Tokenisierung,
- der *Illinois POS Tagger* zur Wortartenerkennung,
- der *Illinois Named Entity Recognizer* zur Erkennung von Eigennamen,
- das *Illinois SRL-Werkzeug* zur Zuweisung von semantischen Rollen,
- der *Illinois Wikifier*, um eine Verbindung zwischen Entitäten des Eingabetextes mit vorhandenen Wikipediaeinträgen herzustellen

und viele mehr. Alle Sprachverarbeitungswerkzeuge der *Cognitive Computation Group* sind unter [Ill] aufgelistet. Zur produktiven Verwendung dieser Werkzeuge muss eine zentrale Serverkomponente, der *Illinois NLP Curator*, installiert werden, die die einzelnen Sprachverarbeitungswerkzeuge von Illinois untereinander koordiniert und zentral verwaltet. Die einzelnen Sprachverarbeitungswerkzeuge können abhängig von der Ressourcenverfügbarkeit auf mehrere Server verteilt werden. Wird der Curator einschließlich aller verwendbaren Sprachverarbeitungswerkzeuge auf einem Server installiert, empfiehlt die *Cognitive Computation Group* 32GB RAM. Der Curator und die einzelnen Sprachverarbeitungswerkzeuge können ausschließlich unter Linux installiert werden. Die Kommunikation zwischen dem Curator und den Clients wird über *Apache Thrift* [MSK] realisiert. Da die Installation im Allgemeinen etwas Zeit in Anspruch nimmt, hat die *Cognitive Computation Group* zu allen verfügbaren Sprachverarbeitungswerkzeugen Online-Demos zur Verfügung gestellt, die auch im Rahmen der hier stattgefundenen Analyse unter Kapitel 4 verwendet wurden.

#### 3.1.5. Apache OpenNLP

*Apache OpenNLP* [Apa] ist ein *offenes* Sprachverarbeitungswerkzeug, dass, wie alle hier erwähnten Sprachverarbeitungswerkzeuge, auf maschinellem Lernen basiert. Es unterstützt

- die Wortsegmentierung,
- die Eigennamenerkennung,
- das Auffinden von vordefinierten Kategorien innerhalb eines Eingabetextes,

- die Wortartenerkennung,
- die Phrasenerkennung,
- die Syntaxanalyse
- und die Koreferenzanalyse.

*Apache OpenNLP* ist in Java implementiert und bietet zu jeder einzelnen Aufgabe ein Werkzeug, welches entweder über die Kommandozeile oder über eine Java-Schnittstelle verwendet werden kann. Jedes dieser Werkzeuge kann unabhängig voneinander verwendet werden. Dabei ist als Eingabe ein vortrainiertes Modell notwendig. Hierbei kann entweder selbstständig ein Modell mithilfe eines maschinellen Lernverfahrens und eines entsprechenden Korpus, durch die von *OpenNLP* bereitgestellten Werkzeuge, erstellt oder ein vortrainiertes Modell aus [Ope] verwendet werden.

*OpenNLP* bietet zwei Alleinstellungsmerkmale. Bei der Wortarten- und Phrasenerkennung ist *OpenNLP* in der Lage, Konfidenzen neben der vorgeschlagenen Klassifikation anzugeben. Zusätzlich ist es möglich, nicht nur die beste Klassifikation, sondern die n-besten Klassifikationen ausgeben zu lassen. Zur Verdeutlichung folgt nun ein Beispiel zur Wortartenerkennung inklusive Konfidenzen. Dabei wurden die besten drei Sequenzen der Wortartenerkennung angegeben:

Go	VB	0.939945837924	NNP	0.0299122238557	VB	0.939945837924
to	TO	0.971664151775	TO	0.943862972131	TO	0.971664151775
the	DT	0.941945961758	DT	0.944923324163	DT	0.941945961758
fridge	NN	0.955769870855	NN	0.955769870855	NNS	0.0204677242168
Score:		-0.195723568959917		-3.66915194653927		-4.0393915151896

In diesem Beispiel wurde als Eingabetext „Go to the fridge“ übergeben, wie man an der ersten Spalte erkennen kann. Die nächste Spalte enthält die erkannte Wortart einschließlich der Konfidenz. Somit wurde „Go“ durch „VB“ mit der Konfidenz 0.939945837924 als Verb erkannt. Damit ergibt sich als erste Sequenz „VB, TO, DT, NN“. In der letzten Zeile kann man die Bewertung für die entsprechenden Sequenzen ablesen. Für die eben genannte Sequenz ergibt sich die höchste Bewertung von -0.195723568959917 und stellt damit die vom verwendeten Modell wahrscheinlichste Wortartenklassifikation dar. Folglich wird die zweite Sequenz von der dritten und die dritte Sequenz von der vierten Spalte repräsentiert.

### 3.1.6. ClearNLP

*ClearNLP* [JD03] ist ein *offenes* Sprachverarbeitungswerkzeug, welches folgende Aufgaben übernehmen kann:

- Tokenisierung
- Wortartenerkennung
- Phrasenerkennung
- Syntaxanalyse durch eine Dependenzsyntax
- Analyse der Morphologie
- Eigennamenerkennung
- Zuweisung von semantischen Rollen

Jede Aufgabe benötigt, wie *BIOS* und *OpenNLP*, neben dem Eingabetext ein vortrainiertes Modell, welches entweder selbst generiert oder aus einer vorhandenen Auswahl von Modellen heruntergeladen werden kann. *ClearNLP* wurde mit Java realisiert und befindet sich noch in der Entwicklungsphase.

### 3.1.7. Werkzeuge der Stanford NLP Group

Die *Stanford NLP Group* hat eine Vielzahl von unterschiedlichen Sprachverarbeitungswerkzeugen entwickelt. Darunter befindet sich der *Stanford Parser*, welcher bereits im vorherigen Abschnitt vorgestellt wurde. Bis auf die Zuweisung von semantischen Rollen und der Phrasenerkennung, findet man zu allen in Abschnitt 2.2 vorgestellten Sprachverarbeitungsaufgaben entsprechende Werkzeuge. Unter [Sta] befindet sich eine Auflistung aller Werkzeuge der *Stanford NLP Group*. Besonders hervorzuheben sind der *Stanford POS Tagger* und *Stanford CoreNLP*.

Beim *Stanford POS Tagger* [KDCDY03] handelt es sich um einen Maximum-Entropie-Klassifikator zur Wortartenerkennung, der neben dem Eingabetext ein vortrainiertes Modell entgegennimmt. Dabei werden vortrainierte Modelle angeboten. Es ist aber auch möglich, dass Training der Modelle selbstständig durchzuführen.

*Stanford CoreNLP* stellt ein *offenes* Sprachverarbeitungswerkzeug dar, das die einzelnen Sprachverarbeitungswerkzeuge von Stanford zusammenführt. Hierbei gilt, wie beim Wortartenerkennung: Jedem Werkzeug muss als Eingabe ein Modell übergeben werden. Auch hier werden vortrainierte Modelle bereitgestellt.

### 3.1.8. Berkeley, Stanford, Collins und BLLIP/Charniak Parser

Zum Aufbau von Konstituentengrammatiken wurden im Rahmen dieser Bachelorarbeit, neben *SENNA*, folgende Parser betrachtet:

- Berkeley Parser [SLRD06]
- Stanford Parser [DCD03]
- Collins Parser [Col99]
- Charniak Parser [Cha00]

Diese vier Parser verwenden Erweiterungen der in Unterabschnitt 2.2.4 vorgestellten kontextfreien Grammatiken, um die Syntax der englischen Sprache zu modellieren. Die in Abschnitt 4.4 untersuchten Stanford und Berkeley Parser verwenden beispielsweise kontextfreie Grammatiken, die jeder Produktionsregel eine Auftrittswahrscheinlichkeit zuordnen. Solch eine Grammatik wird auch als **probabilistische kontextfreie Grammatik** (engl. **Probabilistic Context Free Grammars - PCFG**) bezeichnet. Der Collins und Charniak Parser verwenden wiederum eine Erweiterung der probabilistischen kontextfreien Grammatik, nämlich den sogenannten **kopflexikalisierten PCFGs**. In diesen **kopflexikalisierten PCFGs** wird angenommen, dass jede Produktionsregel einen *lexikalischen Kopf* besitzt. Der *lexikalischen Kopf* stellt innerhalb einer Phrase das grammatikalisch wichtigste Wort dar [DJH09, S.443]. So stellt beispielsweise ein Nomen innerhalb einer Nominalphrase oder ein Verb innerhalb einer Verbphrase den *lexikalischen Kopf* dar. Eine entsprechende Produktionsregel könnte beispielsweise folgendermaßen aussehen:

$$NP(\text{fridge}) \rightarrow DET(\text{the}) NP(\text{fridge}).$$

Eine detailliertere Beschreibung über die einzelnen Erweiterungen der kontextfreien Grammatik und welche Vorteile beziehungsweise Nachteile damit verbunden sind, kann unter [DJH09, S.419] gefunden werden.

## 3.2. Gesamtsysteme

Im vorherigen Abschnitt wurden Sprachverarbeitungswerkzeuge vorgestellt, die zur Bewerkstelligung von NLP-Aufgaben entworfen wurden. Dieser Abschnitt stellt hingegen Gesamtsysteme vor, die entweder mit dieser Ausarbeitung, also der Erstellung einer Handlungsrepräsentation aus einem Eingabetext, oder mit dem Gesamtprojekt „PARSE“ vergleichbar sind. Dabei wurden bei den jeweiligen Systemen ebenfalls einzelne NLP-Aufgaben gelöst. Dieser Abschnitt soll die allgemeine Herangehensweise der Projekte vorstellen und dieser Ausarbeitung beziehungsweise dem Gesamtprojekt „PARSE“ gegenüberstellen.

### 3.2.1. Semantic Mapping Between Natural Language Questions and SQL Queries via Syntactic Pairing

Die wissenschaftliche Publikation „*Semantic Mapping Between Natural Language Questions and SQL Queries via Syntactic Pairing*“ [AA10] stellt einen Ansatz vor, um natürliche Sprache auf SQL-Abfragen abzubilden. Das Forschungsgebiet, in das sich diese Arbeit eingliedert, wird als **NLIDB** (engl. **N**atural **L**anguage **I**nterface to a **D**atabase) bezeichnet. Der hier diskutierte Ansatz basiert auf der Analyse von Syntaxbäumen. Dabei wird angenommen, dass sowohl die in natürlicher Sprache vorhandene Frage, als auch die zu projizierende SQL-Abfrage mithilfe von Syntaxbäumen dargestellt werden können. Abbildung 3.1 veranschaulicht diesen Sachverhalt.

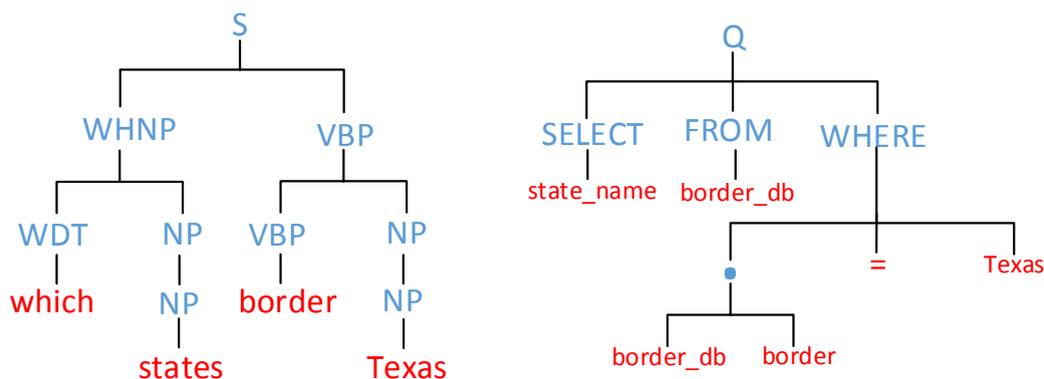


Abbildung 3.1.: Links: Syntaxbaum der Frage

Rechts: Syntaxbaum der entsprechenden SQL-Abfrage

(SELECT state\_name FROM border\_db WHERE border='Texas')

Die Abbildung einer Frage auf die entsprechende Abfrage wird mithilfe von maschinellen Lernverfahren realisiert. Zunächst wird ein Trainingskorpus von korrekten Frage/Abfrage-Paaren benötigt. Anschließend wird ein Klassifizierer (**S**upport **V**ector **M**achine - SVM) mithilfe des Korpus trainiert. Um nun neue Fragen auf die entsprechende SQL-Abfrage abzubilden, werden mithilfe des Klassifizierers jeder vorhandenen SQL-Abfrage aus dem Trainingskorpus eine Wahrscheinlichkeiten zugewiesen, wodurch eine Rangliste entsteht. Die SQL-Abfrage mit der höchsten Wahrscheinlichkeit stellt die vom Klassifizierer ausgewählte Vorhersage dar.

Für diesen Ansatz sollte die syntaktische Struktur der abzubildenden Fragen möglichst bekannt sein, da der Klassifizierer durch den Korpus aus Frage/Abfrage-Paaren und einer entsprechenden Trainingsphase die Abbildungsmuster erlernt. Für Fragen die jedoch syntaktische Strukturen besitzen, die nicht im Trainingskorpus vorhanden sind, kann kein

verwertbares Ergebnis erwartet werden. Versucht man also den Ansatz auf diese Ausarbeitung anzuwenden, müsste die syntaktische Struktur der Eingabetexte, welche Anweisungen an ARMAR-III darstellen, bekannt sein und ein entsprechender Trainingskorpus erstellt werden. Da die in dieser Bachelorarbeit betrachteten Eingabetexte aus einer beliebig langen Folge von Anweisungen bestehen, ist die Erstellung solch eines Trainingskorpus nicht umsetzbar. Somit stellt der in dieser wissenschaftlichen Publikation vorgestellte Ansatz keine Alternative für diese Ausarbeitung dar.

### 3.2.2. Precise-System

Bei der von Ana-Maria Popescu veröffentlichten Publikation [AMOH03] wird das *Precise-System* vorgestellt. Dabei wird, wie bei Abschnitt 3.2.1, die Problemstellung natürlichsprachliche Fragen auf eine Datenbank abzubilden betrachtet. Das Precise-System unterscheidet sich von anderen NLIDB-Systeme insofern, als dass vor der Abbildung der Frage auf eine SQL-Abfrage, eine Klassifikation der Frage durchgeführt wird, um zu entscheiden, ob die Frage korrekt verarbeitet werden kann. Eine verarbeitbare Frage wird in dieser Publikation als „semantically tractable question“ bezeichnet. Erst nachdem eine Frage als „semantically tractable question“ klassifiziert wurde, wird die entsprechende SQL-Abfrage durchgeführt. Andernfalls kann das System dem Benutzer eine Rückmeldung liefern. Um diese Klassifikation zu realisieren, wurde ein theoretisches Modell entworfen. Dabei wird in einem ersten Schritt die Frage in einzelne Token segmentiert und von semantisch irrelevanten Token befreit. Beispielsweise wird aus der Frage „*What are the HP jobs on a Unix system?*“ die Tokenaufzählung (*what, HP, job, Unix, system*) generiert. Anschließend versucht das Precise-System die Tokenaufzählung auf vorhandene Datenbankelemente, wie Relationen, Attribute oder Werte, abzubilden. Das Problem, nämlich eine Tokenaufzählung auf eine Menge von Datenbankelemente abzubilden, sodass das theoretische Modell eingehalten wird, kann auf ein Graphenproblem reduziert werden. Dabei gibt das theoretische Modell anhand von drei Regeln vor, wann genau eine gültige Abbildung zwischen einer Tokenaufzählung und Datenbankelementen vorliegt. Ist solch eine Abbildung vorhanden, wird eine entsprechende SQL-Abfrage ausgeführt.

Das Precise-System stellt eine interessante Herangehensweise der NLIDB-Problemstellung dar. Ob dieser Ansatz auch für gesprochene Sprache mit fehlenden Satztrennungszeichen anzuwenden ist, müsste untersucht werden. Da jedoch die grundlegende Herangehensweise von der des Gesamtprojektes „PARSE“ abweicht, wird das Precise-System in dieser Ausarbeitung nicht näher betrachtet.

### 3.2.3. The KIT Translation Systems for IWSLT 2013

Bei der wissenschaftlichen Publikation „*The KIT Translation Systems for IWSLT 2013*“ [TLHW13] wurde, im Rahmen der *TED-Aufgabe* des IWSLT (engl. International Workshop on Spoken Language Translation), ein System zur maschinellen Übersetzung konzipiert. Dabei wurde ein Korpus aus transkribierten Sprachaufnahmen der TED-Konferenzen zur Verfügung gestellt. Eine besondere Herausforderung hierbei war, dass die transkribierten Sprachaufnahmen keine Satztrennungszeichen enthalten. Diese Problematik ist mit der dieser Ausarbeitung identisch. Die in dieser wissenschaftlichen Publikation vorgestellte Lösung für den Umgang mit fehlenden Satztrennungszeichen ist die Verwendung eines Systems, welches als Eingabe die Ausgabe eines automatischen Spracherkenners erhält und die fehlenden Satztrennungszeichen mithilfe von maschinellen Lernverfahren vorhersagt. Dabei ist zunächst ein Trainingskorpus notwendig, der einen Text sowohl mit, als auch ohne Satztrennungszeichen beinhaltet, um darauf maschinelle Lernverfahren anzuwenden.

Im Rahmen dieser Ausarbeitung wurde der Ansatz, Satztrennungszeichen mithilfe von maschinellen Lernverfahren vorherzusagen, in Betracht gezogen. Da aber falsch vorhergesagte

Satztrennungszeichen, genauer falsch positive (engl. *false positive*) Satztrennungszeichen, zu nicht verarbeitbaren Eingabetexten führen, wurde dieser Ansatz nicht weiter betrachtet. Im Analysekapitel unter 4 wird diese Entscheidung hinsichtlich der falsch positiven Satztrennungszeichen detaillierter beschrieben.

### 3.2.4. Pegasus

Bei *Pegasus* [RM06] handelt es sich um ein natürlichsprachliches Programmiersystem. Dabei soll das Programmieren mithilfe von geschriebener Sprache ermöglicht werden. Der *Pegasus*-Prototyp unterstützt momentan die Sprachen Englisch und Deutsch. Die Funktionsweise von *Pegasus* soll anhand folgender Anweisung illustriert werden:

„If the first element of the second row of the matrix is smaller than 3 then write „I can understand you!““

Der erste und essentielle Schritt besteht in der Überführung der geschriebenen Anweisung in eine Ideensprache (engl. Idea-Notation), die in Abbildung 3.2 dargestellt ist.

```
((
  ((
    (be, predicate),
    ((element, first, (row, second, (matrix))),subject),
    (smallness, comparative),
    ((three, than), object)
  ), statement),
  ((
    (writing, predicate),
    (you, subject),
    (string, object)
  ), command)
), condition)
```

Abbildung 3.2.: Ideensprache des Pegasus-Projektes [RM06]

Um den Ausdruck der Ideensprache zu konstruieren, wird die Syntax des Eingabetextes mithilfe einer einfachen kontextfreien Grammatik analysiert. Die Einschränkung der Syntax durch eine knappe kontextfreie Grammatik, wie unter [RM06] zugegeben, beschränkt die Komplexität der erlaubten Eingabesätze. Semantische Informationen, wie Prädikate, Subjekte und Objekte, werden direkt aus der kontextfreien Grammatik abgeleitet. Folglich sind keine semantischen Informationen extrahierbar, falls der Eingabesatz nicht von der kontextfreien Grammatik erfasst werden konnte. Das notwendige lexikalische Wissen zur Überführung in die Ideensprache wird aus einem hinterlegten Wörterbuch bezogen. Ein Auszug des Eintrages für den Begriff *smallness* befindet sich in der Abbildung 3.3. Liegt der Ausdruck in der Ideensprache vor, wie in Abbildung 3.2, wird diese von außen nach innen aufgelöst und mit den in einer internen Bibliothek hinterlegten Bedeutungen verglichen. Wie das *Pegasus*-Projekt Bedeutungen definiert und wie diese in einer Bibliothek hinterlegt werden, ist in der wissenschaftlichen Publikation unter [RM06] zu finden.

Im Allgemeinen ist die Problemstellung und Herangehensweise des Pegasus-Projektes mit der des Gesamtprojektes „PARSE“ vergleichbar. Über eine Zwischenrepräsentation soll eine Abbildung von natürlicher Sprache in ein ausführbares Programm ermöglicht werden. Durch die Verwendung von geschriebener Sprache und der vereinfachten kontextfreien Grammatik, welche die Komplexität der Eingabetexte beschränkt, bleiben Herausforderungen unbeachtet, die das Gesamtprojekt „PARSE“ bewusst adressiert. Zusätzlich ist

ENTITY	ACTION	PROPERTY
smallness	to downsize	smaller
	downsized	smallest
	downsized	smallest
	downsize	
	1 object, direct	

Abbildung 3.3.: Auszug eines Eintrages für die Idee *smallness*

es ein Ziel dieser Ausarbeitung eine initiale Handlungsrepräsentation zu erstellen, welche unabhängig von der anschließenden Verwendung, ARMAR-III neue Anweisungen beizubringen, ist. Somit wird gewährleistet, dass das hier erstellte Modul ebenfalls in anderen Domänen verwendet werden kann. Dementsprechend ist die Erstellung und Verwendung domänenspezifischer Datenbanken, wie das im Pegasus-Projekt verwendete Wörterbuch und die Bibliothek mit den hinterlegten Bedeutungen, für die in dieser Bachelorarbeit zu erstellenden initialen Handlungsrepräsentation keine mögliche Vorgehensweise.

### 3.2.5. Metafor

Metafor [HH05] versucht, wie Pegasus, natürliche Sprache auf Programmiersprachen abzubilden. Dabei ist das Ziel von Metafor nicht die Erstellung eines fertig ausführbaren Programmes, sondern lediglich die Erstellung eines groben Grundgerüsts, um die in natürlicher Sprache formulierten Gedanken in Python-Code zu überführen. Die Umsetzung von Metafor ist sehr mit der in dieser Arbeit angestrebten Vorgehensweise verwandt. Metafor versucht zunächst mithilfe des Sprachverarbeitungswerkzeuges **MontyLingua** [LS04] Wortarten, Subjekt-Verb-Objekt-Strukturen und semantische Rollen zu extrahieren, um darauf aufbauend weitere Operationen, wie eine Diskursanalyse, anzuwenden und somit die Abbildung auf Python-Code zu realisieren. Beispielsweise wird der geschriebene Text „*There is a bar with a bartender who makes drinks.*“ in folgenden Python-Code abgebildet:

```
class bar:
    the_bartender = bartender()
class bartender:
    make(drink): pass
```

Wie alle bisher betrachteten Systeme, wird Metafor ausschließlich mithilfe von geschriebener Sprache verwendet. Ob der hier vorgestellte Ansatz von Metafor auch mit gesprochener Sprache funktioniert, wird in dieser Ausarbeitung in Kapitel 4 diskutiert.

### 3.2.6. Shallow Semantic Parsing for Spoken Language Understanding

Die Publikation „*Shallow Semantic Parsing for Spoken Language Understanding*“ [CMR09] stellt einen Parser vor, der semantische Informationen aus gesprochener Sprache extrahiert. Dabei bezieht sich der Ansatz auf einen relativ neuen Themenbereich, welcher im englischen als *Spoken Language Understanding* bezeichnet wird. Im *Spoken Language Understanding* wird versucht, akustische Signale in eine textuelle Repräsentation zu überführen (engl. Speech Processing) und die anschließende Verarbeitung dieses Textes (engl. Natural Language Processing) zu vereinen. Der hier vorgestellte Ansatz führt zum Extrahieren von semantischen Informationen folgende vier Schritte aus: Zunächst wird der Eingabetext auf Verben untersucht (engl. Target Word Detection). Anschließend wird jedem entdeckten Verb ein *Frame* zugeordnet (engl. Frame Disambiguation). Ein Frame gibt im Vergleich zur PropBank-Annotation, die im Grundlagenkapitel unter 2.2.7 vorgestellt wurde, die semantischen Rollen eines Prädikates vor und wird in der FrameNet-Annotation verwendet.

Die Wörter, die im ersten Schritt als Nicht-Verben klassifiziert wurden, werden im dritten Schritt den jeweiligen Verben zugeordnet (engl. Boundary Detection). Zuletzt werden die semantischen Rollen mithilfe der Frames klassifiziert (engl. Role Classification).

Prinzipiell versucht dieser Ansatz eine Zuweisung von semantischen Rollen mithilfe der FrameNet-Annotation. Dabei werden auch hier vorhandene Satztrennungszeichen vorausgesetzt. Diese können mithilfe eines automatischen Spracherkenners, der Satztrennungszeichen anhand von Pausen und der Grundfrequenz der Schallwellen ableiten kann, vorhergesagt werden. Ob der hier betrachtete Ansatz auch für Eingabetexte ohne Satztrennungszeichen angewendet werden kann, wird im Analysekapitel unter 4 diskutiert.

### 3.2.7. CALO-Projekt

Als letztes Gesamtsystem soll das CALO-Projekt vorgestellt werden. Bei CALO (engl. Cognitive Assistant that Learns and Organizes) handelt es sich um ein von **DARPA** (engl. Defense Advanced Research Projects Agency) finanziertes Projekt, welches vor allem durch die Verwendung in Apples digitalen Sprachassistenten *Siri* bekannt wurde. Von den über 100 veröffentlichten wissenschaftlichen Publikationen zu diesem Projekt, wird der grundlegende Ansatz zur Sprachverarbeitung unter [BHH<sup>+</sup>03] vorgestellt. CALO verwendet eine agentenorientierte Softwarearchitektur, bei der einzelne Agenten eigenständige Aufgaben erfüllen und miteinander kommunizieren. Dabei wird der Eingabetext nicht mithilfe von syntaktischen oder semantischen Analysen strukturiert, sondern hauptsächlich nach Schlüsselwörtern durchsucht. Falls nicht genügend Schlüsselwörter beziehungsweise Informationen im Eingabetext vorhanden sind, werden weitere Informationen angefordert, um durch inkrementelles Zusammensetzen der Eingabe (engl. slot-filling) das Dialogsystem zu realisieren.

Auch das Gesamtprojekt „PARSE“ wird auf die, in dieser Ausarbeitung bereitgestellten, Handlungsrepräsentation Agenten anwenden, um das Ziel, ARMAR-III neue Anweisungen beizubringen, zu erreichen. Eine Besonderheit fällt bei aktuellen digitalen Sprachassistenten auf. Sowohl *Siri*, *Cortana* oder *Google Now* sind zwar in der Lage einzelne Fragen oder Aussagen zu verarbeiten. Besteht der Eingabetext jedoch aus mehreren Sätzen, scheitern aktuelle digitale Sprachassistenten an der Verarbeitung.

### 3.2.8. Tagging Spoken Language Using Written Language Statistics

In der wissenschaftlichen Publikation „*Tagging Spoken Language Using Written Language Statistics*“ [NGG<sup>+</sup>96] wird ein Experiment der Universität Göteborg vorgestellt. Dabei wird ein Wortartenerkennung zur Klassifikation von Texten verwendet, die gesprochene Sprache darstellen. Das Besondere hierbei ist, dass der Wortartenerkennung vorher auf geschriebene Sprache trainiert wurde. Das Ziel des Experiments war es also herauszufinden, ob ein Wortartenerkennung, der auf geschriebene Sprache trainiert wurde, verwertbare Klassifikationsergebnisse für gesprochene Sprache liefern kann. Um dies festzustellen wurde eine Evaluation des Wortartenerkenners mit einem 1360 Wörter umfassenden Testkorpus durchgeführt. Das Ergebnis hierbei war, dass eine Genauigkeit von um die 85 Prozent erzielt werden konnte. Die Berechnung des Genauigkeitsmaßes wird im Abschnitt 4.2 erläutert und anhand eines Beispiels verdeutlicht.

Die bei der Evaluation betrachteten Texte ähneln den Eingabetexten des in dieser Bachelorarbeit zu erstellenden Werkzeuges. Sie stellen gesprochene Sprache dar, die keine Punktierung, Wortfehler, Wiederholungen und Füllwörter enthält. Betrachtet man anstatt der schwedischen nun die englische Sprache und führt den hier betrachteten Ansatz weiter, kann man sich folgende Frage stellen: Welche NLP-Aufgaben können dadurch bewerkstelligt werden, Sprachverarbeitungswerkzeuge, die auf geschriebene Sprache optimiert wurden, auf gesprochene Sprache anzuwenden? Diese Frage soll im nächsten Kapitel detailliert analysiert werden.



## 4. Analyse

Ziel dieser Bachelorarbeit ist die Implementierung eines Werkzeuges, welches die Ausgabe eines automatischen Spracherkenners als Eingabe erhält und daraus eine initiale Handlungsrepräsentation generiert. Die initiale Handlungsrepräsentation, bei der es sich um eine Graphendarstellung handelt, wird im Anschluss verwendet, um ARMAR-III neue Anweisungen beizubringen. Dabei ist die Herausforderung, dass die Eingabetexte Füllwörter, Wortwiederholungen, Wortfehler und keine Satztrennungszeichen enthalten. Zusätzlich handelt es sich nicht um einzelne Anweisungen, sondern um Eingabetexte mit mehreren Sätzen. Ein beispielhafter Eingabetext könnte folgendermaßen lauten:

*„Go to the table and see if you can find a popcorn bag if so take it and bring it to me“*

Um dieses Ziel zu erreichen, wurden im Abschnitt 2.2 die **NLP**-Aufgaben vorgestellt. Dabei handelt es sich um zu bewerkstelligende Herausforderungen, um Informationen zu erlangen, die innerhalb einer initialen Handlungsrepräsentation dargestellt werden können. Dazu wurden im Abschnitt 3.1 Sprachverarbeitungswerkzeuge vorgestellt, die NLP-Aufgaben bewerkstelligen können. Anschließend wurden Gesamtsysteme betrachtet, die vergleichbare Problemstellungen bewältigen mussten. Bei dieser Betrachtung sind folgende Besonderheiten aufgefallen: Zunächst einmal erwarten die meisten Gesamtsysteme geschriebene Sprache als Eingabe und vermeiden somit die Schwierigkeiten von gesprochener Sprache (3.2.1 - 3.2.5). Zusätzlich wird in den meisten Ansätzen Domänenwissen in jeder Bearbeitungsphase des Eingabetextes vorausgesetzt. Beispielsweise besitzt das **Precise**-System (3.2.2) in jeder Verarbeitungsphase die Informationen, welche Datenbankelemente vorhanden sind. In Analogie zu dieser Ausarbeitung würde die Verwendung dieses Ansatzes bedeuten, dass die Funktionalitäten von ARMAR-III bekannt sind und in der Erstellung der initialen Handlungsrepräsentation verwendet werden kann. Bei der Konstriktion der hier erstellten Anwendung wird jedoch bewusst auf Domänenwissen verzichtet. Diese Anwendung soll möglichst unabhängig von der späteren Verwendung konstruiert werden, um das Modul auch in anderen Systemen wiederverwenden zu können. Bei den betrachteten Gesamtsystemen in den Unterabschnitten 3.2.3, 3.2.6 und 3.2.7, die gesprochene Sprache als Eingabe erwarten, werden Satztrennungszeichen in der Verarbeitung der Eingabetexte vorausgesetzt. Dabei werden entweder automatische Spracherkennung verwendet, die Satztrennungszeichen vorhersagen (3.2.7), oder in einem Vorverarbeitungsschritt Satztrennungszeichen mithilfe von maschinellen Lernverfahren hinzugefügt (3.2.3). Auch die Begrenzung des Eingabetextes auf einzelne Sätze ist ein häufig umgesetzter Ansatz (3.2.6). Diese Ausarbeitung

soll jedoch, im Rahmen des Gesamtprojektes „PARSE“, die Sprachverarbeitungsmöglichkeiten von Eingabetexten ohne Satztrennungszeichen untersuchen. Das Kernproblem bei der Vorhersage von Satztrennungszeichen ist die damit verbundene Unzuverlässigkeit. Ein Beispiel für einen Eingabetext mit falsch vorhergesagten Satztrennungszeichen könnte folgendermaßen aussehen:

*„Go to. the table and see if you can find a popcorn. bag if so take it and bring it to me“*[3]

Betrachtet man in diesem Beispiel ausschließlich die Punktation fällt auf, dass zwei Punkte falsch positiv (engl. *false positive*) und ein Punkt falsch negativ (engl. *false negative*) klassifiziert wurden. Die damit verbundene Problematik liegt in der sequentiellen Architektur, die innerhalb der Sprachverarbeitung angewendet wird und im Abschnitt 2.3 vorgestellt wurde. Im ersten Schritt wird der Eingabetext mithilfe der Tokenisierung in einzelne Sätze segmentiert. Somit würden im Beispiel unter [3] drei Sätze entstehen. Die anschließend vorgestellten NLP-Aufgaben, wie der Syntaxanalyse oder dem Zuweisen von semantischen Rollen, werden für jeden einzelnen Satz unabhängig voneinander ausgeführt. Folglich führen falsch gesetzte Satztrennungszeichen in allen darauffolgenden Sprachverarbeitungsschritten zu fehlerhaften Ergebnissen. Deshalb soll der Eingabetext in dieser Ausarbeitung ohne Satztrennungszeichen verarbeitet werden.

Durch die Analyse verwandter Arbeiten konnte unter anderem mit der wissenschaftlichen Publikation der Universität Göteborg (siehe Unterabschnitt 3.2.8) ein Ansatz abgeleitet werden, der die in dieser Arbeit verbundenen Herausforderungen umsetzen könnte. Der hier betrachtete und analysierte Ansatz ist, Informationen mithilfe von Standard-Sprachverarbeitungswerkzeugen aus dem Eingabetext zu extrahieren, um anschließend die Handlungsrepräsentation erstellen zu können. Dabei wurden die einzelnen Sprachverarbeitungswerkzeuge nicht auf Eingabetexte trainiert, die im Rahmen dieser Ausarbeitung zu erwarten sind. Deshalb muss die Ausgabe der Sprachverarbeitungswerkzeuge bezüglich der hier verwendeten Eingabetexte auf ihre Verwendbarkeit analysiert werden. Eine weitere Herangehensweise wäre die Standard-Sprachverarbeitungswerkzeuge mithilfe des Sprachkorpus, welcher unter [Gue15] erstellt wurde, neu zu trainieren. Jedoch beinhaltet der Sprachkorpus momentan noch nicht genügend Sprachaufnahmen, um die einzelnen Sprachverarbeitungswerkzeuge zu trainieren.

Um die Standard-Sprachverarbeitungswerkzeuge, die im Abschnitt 3.1 vorgestellt wurden, auf ihre Verwendbarkeit zu überprüfen, wurde ein Korpus von Eingabetexten erstellt. Dieser Korpus wird im Abschnitt 4.1 vorgestellt. Anschließend werden die Sprachverarbeitungswerkzeuge entsprechend ihrer Funktionalität analysiert, um die bestmögliche Handlungsrepräsentation zu gewährleisten. Dabei liegt der Schwerpunkt auf den Folgen der fehlenden Satztrennungszeichen, da genau diese Problematik in vergleichbaren Arbeiten bewusst umgangen wurde. Die bei der Analyse verwendeten Ausgaben der Sprachverarbeitungswerkzeuge, wurden im Anhang unter G in einer überarbeiteten Form bereitgestellt. Die ursprünglichen und unverarbeiteten Ausgaben der Sprachverarbeitungswerkzeuge befinden sich auf den Servern des IPD-Institutes<sup>1</sup>.

## 4.1. Analysekorpus

Zur Analyse der Sprachverarbeitungswerkzeuge im Abschnitt 3.1 wurde ein Korpus von 14 Texten zusammengestellt. Bei den Texten handelt es sich einerseits um gesprochene Anweisungen, die mithilfe eines automatischen Spracherkenners in Text überführt wurden, andererseits um eigens erstellte Texte. Die gesprochenen Anweisungen wurden aus

<sup>1</sup>[https://code.ipd.kit.edu/weigelt/kocybik\\_ba/tree/master/Analyse](https://code.ipd.kit.edu/weigelt/kocybik_ba/tree/master/Analyse)

dem Sprachkorpus unter [Gue15] entnommen. Dabei wurden bewusst nicht alle Aufnahmen für die Analyse verwendet, um für die Evaluation des hier erstellten Werkzeuges auf die restlichen Sprachaufnahmen zurückgreifen zu können. Jeder der 14 Texte liegt in zwei Ausführungen vor, einmal mit und einmal ohne Satztrennungszeichen. Durch den Vergleich der Ausgaben von Eingabetexten, die sich lediglich im Vorhandensein von Satztrennungszeichen unterscheiden, kann festgestellt werden, welchen Informationsverlust das Fehlen von Satztrennungszeichen für die Sprachverarbeitungswerkzeuge bedeutet. Zusätzlich beinhalten die Texte aus dem Analysekorpus Wortwiederholungen und Füllwörter, die eine Folge von gesprochenen Anweisungen darstellen. Auf Wortfehler, die aus der Nutzung eines automatischen Spracherkenners resultieren, wird im Analysekorpus bewusst verzichtet. Die Entscheidung, ob ein Wort fehlerhaft erkannt wurde, hängt hauptsächlich von der betrachteten Domäne ab. Das hier zu erstellende Werkzeug soll jedoch domänenunabhängig verwendbar sein, deshalb müssen Wortfehler separat aufgedeckt und behoben werden. Zusätzlich steht zum Zeitpunkt dieser Ausarbeitung noch nicht fest, welcher automatische Spracherkenner im Gesamtprojekt „PARSE“ verwendet werden soll. Ein Beispiel für einen Eingabetext des Analysekorpus sieht folgendermaßen aus:

Mit Satztrennungszeichen:

*„Go to the fridge and open the door. Locate the orange juice and take it. Close the door. Come over here and give the orange juice to me“*

Ohne Satztrennungszeichen:

*„Go to the fridge and open the door locate the orange juice and take it close the door come over here and give the orange juice to me“*

Hierbei handelt es sich um den zweiten Text des Analysekorpus, welcher sich im Anhang unter D befindet und einen selbsterstellten Text darstellt. Die im Analysekorpus beinhaltenen selbsterstellten Texte wurden in der Form konzipiert, dass Sie aus mehreren Sätzen bestehen, um somit gezielt die Schwächen von Sprachverarbeitungswerkzeugen bei fehlenden Satztrennungszeichen aufdecken zu können.

## 4.2. Wortartenerkennung

Die Wortartenerkennung stellt die Grundlage innerhalb der Sprachverarbeitung dar, um einen Eingabetext maschinell interpretieren zu können. Beispielsweise repräsentieren Verben häufig Aktionen oder Funktionen, die das jeweilige Zielsystem umsetzen soll. Deshalb soll das in dieser Bachelorarbeit zu erstellende Werkzeug in der Lage sein, die Wortartenerkennung mithilfe vorhandener Sprachverarbeitungswerkzeuge zu realisieren. Die in dieser Analyse gewonnenen Erkenntnisse dienen als Entscheidungskriterium für die umgesetzte Implementierung, die im Kapitel 5 vorgestellt wird.

Zur Analyse der Sprachverarbeitungswerkzeuge wurden zu jedem der 14 Texte des Analysekorpus Musterlösungen angefertigt. Diese sind im Anhang unter E zu finden. Hier eine beispielhafte Musterlösung zum sechsten Text des Analysekorpus:

[4]: *At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN. Take/VB it/PRP, come/VB over/RP here/RB and/CC give/VB it/PRP to/TO me/PRP.*

Mithilfe der Musterlösungen kann nun die Ausgabe jedes Wortartenerkenners ausgewertet werden. Dafür wird die Genauigkeit (engl. accuracy) [DJH09, S.189] der Ausgaben bezüglich der Musterlösung angegeben. Das Genauigkeitsmaß ist der Prozentsatz aller korrekt erkannten Wortarten hinsichtlich einer Musterlösung. Jedes Token in der Ausgabe eines Wortartenerkenners besitzt ein entsprechendes Pendant in der Musterlösung. Entspricht

die Wortart eines Token innerhalb der Ausgabe eines Wortartenerkenners nicht der Wortart der Musterlösung, wird dies als Fehler bewertet, ansonsten als korrekt zugewiesene Wortart. Um die Vergleichbarkeit zwischen den Genauigkeitswerten von Eingabetexten mit und ohne Satztrennungszeichen zu gewährleisten, wird jedes Token, das einem Satztrennungszeichen entspricht, ignoriert. Zur Verdeutlichung soll die Musterlösung unter [4] exemplarisch mit der Ausgabe des *Stanford POS Tagger* verglichen und die Genauigkeit angegeben werden. Der *Stanford POS Tagger* gibt für den sechsten Text des Analysekorpus folgende Wortarten aus:

Mit Satztrennungszeichen:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN ./.*  
*Take/VB it/PRP ./, come/VBP over/IN here/RB*  
*and/CC give/VB it/PRP to/TO me/PRP ./.*

Ohne Satztrennungszeichen:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN*  
*take/VBP it/PRP come/VB over/IN here/RB*  
*and/CC give/VB it/PRP to/TO me/PRP*

Die Abweichungen zwischen der Musterlösung und der Vorhersage des *Stanford POS Taggers* sind rot markiert. Somit ergeben sich sowohl für die Ausgabe mit, als auch für die Ausgabe ohne Satztrennungszeichen, zwei Fehler. Bei der Ausgabe mit Satztrennungszeichen wurden 20 Wortarten korrekt bestimmt, einschließlich der Satztrennungszeichen (blau markiert), wohingegen es bei der Ausgabe ohne Satztrennungszeichen lediglich 17 Wortarten sind. Zur Vergleichbarkeit werden die Satztrennungszeichen bei der Bestimmung von korrekten und inkorrekten Wortarten ignoriert. Somit ergeben sich für die Ausgabe des *Stanford POS Taggers* beim Eingabetext mit Satztrennungszeichen ebenfalls 17 korrekte Wortarten. Bei insgesamt 20 Token entspricht dies einer Genauigkeit von 85 Prozent ( $\frac{17}{20}$ ).

Insgesamt enthalten die 14 Texte des Analysekorpus 468 Wörter. Betrachtet wurden die Sprachverarbeitungswerkzeuge *OpenNLP* [Apa], *Illinois POS Tagger* [Ill], *Stanford POS Tagger* [KDCDY03], *ClearNLP* [JD03], *MBSP* [TVW05], *SENNA* [RJL<sup>+</sup>11] und *Stanford Parser* [DCD03]. Der *Stanford Parser* verwendet nicht die Wortartmarkierungen des *Stanford POS Tagger*. Aus diesem Grund sind die Werte des *Stanford Parsers* separat aufgeführt. Das Ergebnis der Analyse ist unter Tabelle 4.1 zu finden.

	Mit Satztrennungszeichen		Ohne Satztrennungszeichen	
	korrekt erkannt	Genauigkeit	korrekt erkannt	Genauigkeit
OpenNLP	430	0,9188	422	0,9017
Illinois POS Tagger	408	0,8718	397	0,8483
Stanford POS Tagger	433	0,9252	423	0,9038
ClearNLP	440	0,9402	425	0,9081
MBSP	420	0,8974	395	0,8440
SENNA	416	0,8889	419	0,8953
Stanford Parser	-	-	422	0,9017

Tabelle 4.1.: Analyse der Wortartenerkennung

Hierbei wurde zu jedem Wortartenerkennung, neben der absoluten Anzahl an korrekt erkannten Wortarten, die prozentuale Genauigkeit angegeben. Bis auf *SENNA*, führt bei allen Sprachverarbeitungswerkzeugen das Fehlen von Satztrennungszeichen zu einer schlechteren Wortartenerkennung. Die Ursache hierfür ist, dass *SENNA* bei den Eingabetexten mit Satztrennungszeichen einen häufigen Fehler innerhalb der Tokenisierung produziert. Tabelle 4.2 stellt einen exemplarischen Auszug dar.

Token	Wortart
Put	VB
the	DT
cup	NN
into	IN
<b>it.</b>	NN

Tabelle 4.2.: *SENNA*: Fehler bei der Tokenisierung

Die letzte Zeile der Tabelle enthält das fehlerhafte Token „*it.*“. Korrekt wäre das Erkennen von zwei einzelnen Token, nämlich von „*it*“ und „.“. Dieser Fehler führt dazu, dass das Wort „*it*“ als Nomen erkannt wurde. Dieses Phänomen trat insgesamt 23 Mal in den Ausgaben von Senna auf, wobei es 16 Mal zu einem Fehler in der Wortartenerkennung führte.

Ebenfalls auffallend in der Analyse unter Tabelle 4.1 ist das besonders schlechte Ergebnis der Werkzeuge *Illinois POS Tagger* und *MBSP*. Jedoch schneidet *MBSP* lediglich bei Texten ohne Satztrennungszeichen mangelhaft ab, wohingegen der *Illinois POS Tagger* bei jeglicher Wortartenerkennung schlechte Ergebnisse liefert.

Das bisherige Analyseergebnis deutet auf eine erhebliche Verschlechterung der Wortartenerkennung bei dem Fehlen von Satztrennungszeichen hin. Jedoch ist hierbei jede Klassifikation, die von der Musterlösung abweicht, als Fehler kategorisiert worden. Während der Durchführung dieser Analyse ist allerdings folgendes Phänomen aufgetreten: Die Sprachverarbeitungswerkzeuge haben bei den Texten ohne Satztrennungszeichen Verben häufig richtig erkannt, jedoch in der falschen Konjugation. So wurde beispielsweise „*put*“ durch *VBD* häufig als Verb in der Vergangenheitsform klassifiziert, wobei es sich meist um ein Verb in der Basisform (*VB*) handelt. Die Klassifikation eines Verbes dient in der späteren Verarbeitung der initialen Handlungsrepräsentation hauptsächlich dazu, einen Zusammenhang zwischen einem Wort und einer Basisfunktionalität des Gesamtsystems herzustellen. Bezogen auf ARMAR-III, stellen Verben meist einzelne Handlungen dar, die der humanoide Haushaltsroboter ausführen soll. Dabei ist die Konjugation des Verbes meist irrelevant, da, mithilfe des Sprachkorpus festgestellt, die Reihenfolge der einzelnen Aktionen bei der Nutzung von gesprochener Sprache, sequentiell interpretiert werden können. Aus diesem Grund wurde die Analyse ohne Berücksichtigung der Konjugation der Verben erneut durchgeführt. Das heißt ein Verb wird als korrekt markiert aufgefasst, falls das Sprachverarbeitungswerkzeug eines der Wortarten für Verben (*VB*, *VBG*, *VBN*, *VBP* oder *VBZ*) zugewiesen hat. Das Ergebnis der aufgearbeiteten Analyse wird in Tabelle 4.3 aufgeführt.

	Mit Satztrennungszeichen		Ohne Satztrennungszeichen	
	korrekt erkannt	Genauigkeit	korrekt erkannt	Genauigkeit
OpenNLP	437	0,9338	437	0,9338
Illinois POS Tagger	427	0,9124	426	0,9103
Stanford POS Tagger	438	0,9359	439	0,9380
ClearNLP	441	0,9423	432	0,9230
MBSP	433	0,9252	425	0,9081
SENNA	426	0,9103	444	0,9487
Stanford Parser	-	-	438	0,9359

Tabelle 4.3.: Analyse der Wortartenerkennung (Verbform irrelevant)

Das Ignorieren der Verbformen zeigt eine erhebliche Verbesserung der Wortartenerkennung für Texte ohne Satztrennungszeichen auf. Mit *SENNA*, dem *Stanford POS Tagger* und

*OpenNLP* finden sich sogar drei Werkzeuge, die vergleichbare, sogar teilweise bessere, Ergebnisse bei fehlenden Satztrennungszeichen erzielen können.

Dementsprechend kann geschlossen werden, dass das Fehlen von Satztrennungszeichen für einige Sprachverarbeitungswerkzeuge zu keinem Informationsverlust bezüglich der Wortartenerkennung führt. Jedoch ist während der Durchführung der Analyse in Tabelle 4.3 folgende Problematik aufgefallen. Durch das Fehlen der Satztrennungszeichen werden Verben häufig inkorrekt als Adjektive oder Nomen klassifiziert, falls ihnen ein Nomen vorausgeht. Diese Besonderheit soll an einer Ausgabe des *Stanford POS Taggers* aufgezeigt werden:

Mit Satztrennungszeichen:

*Go/VB to/TO the/DT dishwasher/NN ./.* *Open/VB it/PRP ./.*

Ohne Satztrennungszeichen:

*go/VB to/TO the/DT dishwasher/NN open/JJ it/PRP*

In diesem Beispiel wird „*open*“ bei dem Eingabetext ohne Satztrennungszeichen fälschlicherweise als Adjektiv gekennzeichnet, wohingegen bei vorhandener Satztrennung die korrekte Wortart bestimmt werden konnte. Dieses häufig auftretende Phänomen kann folgendermaßen begründet werden: Ein entscheidendes Merkmal für die Bestimmung der Wortarten, sowohl in regelbasierten als auch in stochastischen Verfahren, sind die benachbarten Wörter. Ob das Wort „*open*“ ein Adjektiv darstellt hängt im Allgemeinen vom Vorhandensein eines Nomens ab, auf das es sich bezieht. Bei dem hier genannten Beispiel kann der *Stanford POS Taggers* durch eine vorausgehende Satzsegmentierung erkennen, dass es keinen Bezug zwischen „*dishwasher*“ und „*open*“ besteht. Bei Texten mit fehlender Satztrennung, führt genau diese fehlende Information zur inkorrekten Klassifikation. Eine ähnliche Problematik soll folgendes Beispiel verdeutlichen:

Mit Satztrennungszeichen:

*Okay/UH Armar/NNP ./,* *go/VB to/TO the/DT table/NN ./.* *Grab/VB popcorn/NN ./.*

Ohne Satztrennungszeichen:

*Okay/UH Armar/NNP go/VB to/TO the/DT table/NN grab/NN popcorn/NN*

Hier besteht die Möglichkeit, dass es sich bei „*grab*“ um einen Griff handelt, welcher sich auf „*table*“ bezieht, also eine Art „*Tischgriff*“.

In den Analyseergebnissen der Tabelle 4.3 wurden zwar die Verbform als unwichtig angenommen, jedoch ist jede andere Art von Fehler mit gleichem Gewicht in die Analyse mit eingeflossen. Innerhalb der angestrebten Handlungsrepräsentation ist es dagegen von besonderem Interesse, Verben und Nomen korrekt zu erkennen. Verben sollen korrekt erkannt werden, da diese meist repräsentativ für einzelne Funktionen des humanoiden Haushaltsroboters stehen. Nomen stellen innerhalb dieser Domäne häufig Gegenstände dar, auf die eine bestimmte Aktion ausgeführt werden soll. Deshalb wurden für jedes Sprachverarbeitungswerkzeug die Fehlerarten und die jeweilige Anzahl bei der Erkennung von Nomen und Verben betrachtet. Die Ergebnisse dieser Analyse sind in Tabelle 4.4 dargestellt.

Dabei ist in der zweiten Spalte die Gesamtanzahl der fehlerhaften Klassifikationen angegeben, die von dem jeweiligen Sprachverarbeitungswerkzeug hinsichtlich der Klassifikation von Nomen und Verben verursacht wurden. Eine Auflistung der einzelnen Fehler wird in der dritten Spalte angegeben. Lediglich beim *Illinois POS Tagger* wurde auf diese Auflistung, aufgrund der hohen Fehleranzahl, verzichtet. Beispielsweise ist bei *OpenNLP* das Wort „*close*“ insgesamt drei Mal fehlerhaft als Adverb klassifiziert worden. Tabelle 4.4

	Insgesamt	Auflistung der Fehler
OpenNLP	10	push/NN, 3xclose/RB, hand/NN, 3xlook/NN, grab/NN, open/JJ
Illinois POS Tagger	22	-
Stanford POS Tagger	7	2xclose/RB, look/NN, grab/NN, 2xopen/JJ handle/VB
ClearNLP	14	Go/NN, 2xclose/RB, 3xOkay/NNP, dishwasher/JJ, close/JJ, grab/NN, 2xopen/JJ, handle/VB, Inside/VB, go/NNP
MBSP	14	push/NN, rack/JJ, 5xhand/NN, handle/VB 2xfridge/JJ, Grab/NN, 2xopen/JJ, look/NN
Senna	12	3xlook/NN, handle/VB, 3xhand/NN, 3xOkay/NNP, 2xopen/JJ
Stanford Parser	13	Push/NN, 3xokay/VBZ, 3xokay/NNP, handle/VB 2xlook/NN, open/JJ, Grab/NN, fridge/JJ

Tabelle 4.4.: Fehlerhaft klassifizierte Nomen und Verben

soll hauptsächlich aufzeigen, dass die unterschiedlichen Sprachverarbeitungswerkzeuge unabhängig voneinander, oftmals die gleichen Wörter fehlerhaft klassifizieren. Beispielsweise hat jedes Sprachverarbeitungswerkzeug das Wort „open“ mindestens einmal fehlerhaft klassifiziert. Somit führt ein Ansatz, wie die Wortartenerkennung mithilfe einer Mehrheitsentscheidung von mehreren Klassifikatoren durchzuführen, in diesem Fall zu keiner Verbesserung der Wortartenerkennung. Zusätzlich fällt in den Analyseergebnissen der Tabelle 4.4 auf, dass einige Sprachverarbeitungswerkzeuge lediglich Fehler bei der Identifizierung von Verben verursachen. So stellen beispielsweise alle aufgelistete Fehler des *Stanford POS Taggers* Wörter dar, die in der Musterlösung Verben darstellen.

Abschließend soll das Alleinstellungsmerkmal des Sprachverarbeitungswerkzeuges *Apache OpenNLP* diskutiert werden. Wie im Abschnitt unter 3.1.5 vorgestellt, kann *OpenNLP* mehrere Sequenzen von Wortarten, einschließlich ihrer Konfidenzen, zu einem Eingabetext vorhersagen. Jedoch wurden bei den Analyseergebnissen der Tabellen 4.1, 4.3 und 4.4, aufgrund der Vergleichbarkeit mit den anderen Sprachverarbeitungswerkzeugen, ausschließlich die Sequenz mit der höchsten Bewertung berücksichtigt. Deshalb wurde mit *Apache OpenNLP* eine zusätzliche Analyse zur Wortartenerkennung durchgeführt. Hierbei wurden zu jedem Eingabetext die drei besten Sequenzen, einschließlich ihrer Konfidenzen, betrachtet. In dieser Analyse wurde primär versucht, einen Zusammenhang zwischen geringen Konfidenzen und falsch klassifizierten Wortarten herzustellen. Dabei konnte eine Beobachtung gemacht werden: Wurde in mindestens einer der drei Sequenzen ein Wort als Verb klassifiziert, hat es sich auch in der Musterlösung um ein Verb gehandelt. Mithilfe dieser Heuristik kann die Fehleranzahl der Analyseergebnisse der Tabelle 4.4 für *Apache OpenNLP* halbiert werden. Bei den noch fünf fehlerhaft klassifizierten Verben handelt es sich um „hand/NN“, „open/JJ“, „look/NN“ und zwei Mal „close/RB“.

In diesem Abschnitt wurde gezeigt, dass die Wortartenerkennung auch ohne Satztrennungszeichen durchgeführt werden kann. Bei einigen Sprachverarbeitungswerkzeugen war das Ergebnis der Wortartenerkennung beim Fehlen von Satztrennungszeichen sogar deutlich besser. Jedoch treten in der Klassifikation von Verben und Nomen vermehrt Fehler auf, falls keine Satztrennungszeichen vorhanden waren. Die Allgemeingültigkeit der hier durchgeführten Analyse wird im Abschnitt unter 4.6 betrachtet. Im Implementierungskapitel unter 5 wird die Entscheidung, welche Sprachverarbeitungswerkzeuge für die Wortartenerkennung verwendet wurden, diskutiert.

### 4.3. Phrasenerkennung

Nachdem jedem einzelnen Token des Eingabetextes seine Wortart zugewiesen wurde, müssen die Zusammenhänge zwischen einzelnen Wörtern aufgedeckt werden. Auf welches Verb sich ein Adverb bezieht oder ob ein Adjektiv attributiv auf ein Nomen angewendet wurde, kann mithilfe der Phrasenerkennung aufgedeckt werden. Deshalb soll das in dieser Bachelorarbeit zu erstellende Werkzeug neben der Wortartenerkennung in der Lage sein, die Phrasenerkennung mithilfe vorhandener Sprachverarbeitungswerkzeuge zu realisieren. Bei dieser Analyse müssen die Erkenntnisse des vorherigen Abschnitts berücksichtigt werden. Wie im Grundlagenkapitel 2.2.5 erläutert, benötigen Sprachverarbeitungswerkzeuge zur Phrasenerkennung, neben dem segmentierten Eingabetext, die Wortarten jedes einzelnen Tokens. Folglich resultieren aus fehlerhaften Klassifikationen in der Wortartenerkennung, inkorrekte Ergebnisse hinsichtlich der Phrasenerkennung. Bei der Analyse der Wortartenerkennung wurde festgestellt, dass es deutliche Unterschiede zwischen den Klassifikationsergebnissen der einzelnen Sprachverarbeitungswerkzeuge gibt. Deshalb ist der hier angestrebte Ansatz, die Phrasenerkennung mithilfe eines offenen Sprachverarbeitungswerkzeuges (siehe Abschnitt 2.3) durchzuführen. Somit ist es möglich, die Wortartenerkennung mit den Sprachverarbeitungswerkzeugen durchzuführen, die nach der Analyse unter 4.2 das beste Klassifikationsergebnis erzielen konnten. Diese Wortarten können dann anschließend von einem, eventuell anderem, Sprachverarbeitungswerkzeug zur Phrasenerkennung verwendet werden. Natürlich wäre die Verwendung eines geschlossenen Sprachverarbeitungswerkzeuges, welches sowohl in der Wortarten als auch in der Phrasenerkennung bestmögliche Klassifikationsergebnisse erzielt, wünschenswert, jedoch konnte in den hier durchgeführten Analysen kein solches Werkzeug gefunden werden.

Das Vorgehen bei der Analyse der Phrasenerkennung unterscheidet sich nicht von der Analyse der Wortartenerkennung. Dabei wurde zu jedem der 14 Texte des Analysekorpus eine Musterlösung im IOB-Format (siehe Unterabschnitt 2.2.5) angefertigt. Diese Musterlösungen sind im Anhang unter F zu finden. Somit ergeben sich, wie bei der Wortartenerkennung, 468 zu klassifizierende Wörter, wobei die Satztrennungszeichen, zugunsten der Vergleichbarkeit, nicht berücksichtigt wurden. Die offenen Sprachverarbeitungswerkzeuge benötigen neben einem Eingabetext die Wortarten jedes einzelnen Tokens. In der hier durchgeführten Analyse wurden die Musterlösungen der Wortartklassifikation, siehe Anhang E, als Eingabe für die einzelnen Sprachverarbeitungswerkzeuge verwendet. Somit wird analysiert, welchen Fehler die Phrasenerkennung bei korrekten Wortarten verursacht. Die Analyseergebnisse sind in Tabelle 4.5 zu finden.

	Mit Satztrennungszeichen		Ohne Satztrennungszeichen	
	korrekt erkannt	Genauigkeit	korrekt erkannt	Genauigkeit
BIOS	464	0,9915	462	0,9871
OpenNLP	436	0,9316	436	0,9316

Tabelle 4.5.: Analyse der Phrasenerkennung

In dieser Tabelle sind zwei Sprachverarbeitungswerkzeuge, nämlich *BIOS* (3.1.3) und *Apache OpenNLP* (3.1.5), aufgeführt. Es wurden noch weitere Phrasenerkenner, wie die von *Illinois* (3.1.4) und *ClearNLP* (3.1.6) betrachtet. Jedoch war das Klassifikationsergebnis dieser Werkzeuge bereits nach der Auswertung weniger Texte schlechter als *BIOS*. Deshalb wurde zugunsten der in dieser Bachelorarbeit noch zu analysierenden NLP-Aufgaben, lediglich *Apache OpenNLP* als Vergleichswerkzeug in das aufbereitete Analyseergebnis mit einbezogen. *BIOS* hat bei Eingabetexten ohne Satztrennungszeichen lediglich sechs Fehler und damit nur zwei mehr als bei den Eingabetexten mit Satztrennungszeichen verursacht. Demgegenüber hat *OpenNLP* sowohl bei den Eingabetexten mit, als auch bei den Eingabetexten ohne Satztrennungszeichen, 32 Klassifikationsfehler verursacht. Da sich bei beiden

hier aufgeführten Sprachverarbeitungswerkzeugen das Klassifikationsergebnis beim Fehlen von Satztrennungszeichen kaum verschlechtert, kann angenommen werden, dass das Fehlen von Satztrennungszeichen bei korrekt vorliegenden Wortarten keinen signifikanten Informationsverlust in der Phrasenerkennung darstellt.

In diesem Abschnitt wurde gezeigt, dass die Phrasenerkennung auch ohne Satztrennungszeichen mit guten Ergebnissen durchgeführt werden kann. Hierbei muss jedoch in der Bewertung des Analyseergebnisses beachtet werden, dass die korrekten Wortarten bereits vorgegeben wurden.

## 4.4. Syntaxanalyse

Nach der Analyse der Wort- und Phrasenerkennung soll die Syntaxanalyse betrachtet werden. Hierbei wird untersucht, ob die Ausgaben von Syntaxanalysewerkzeugen auch für lange Eingabetexte ohne Satztrennungszeichen verwertbare Informationen enthalten. Im Rahmen des Grundlagenkapitels unter 2.2.4 wurden die zwei grundsätzlichen Ansätze zur Syntaxanalyse vorgestellt.

Der erste Ansatz versucht die Syntax durch **Dependenzen** bzw. **Abhängigkeiten** aufzubauen. Jedoch kann von der Verwendung dieses Ansatzes kein vielversprechendes Ergebnis bezüglich der in dieser Ausarbeitung erwarteten Eingabetexte erwartet werden. Dies liegt an dem grundlegenden Aufbau der Dependenzgrammatik. Dabei wird angenommen, dass das Verb die Struktur eines Satzes festlegt und alle anderen Wörter unmittelbar oder mittelbar vom Verb abhängig sind. Bei Texten aus mehreren Sätzen, jedoch ohne Satztrennungszeichen, wird die Eingabe als einziger Satz erkannt und interpretiert. Aus dem Eingabetext extrahiert ein Dependenzparser zunächst genau ein Verb und versucht alle Wörter des Eingabetextes mit diesem Verb in Beziehung zu setzen. Somit entstehen syntaktische Beziehungen zwischen diesem extrahierten Verb und jedem einzelnen Wort des Eingabetextes, auch wenn mehrere Sätze dazwischen liegen und somit keine syntaktische Beziehung vorliegt. Deshalb wird keine eingehende Analyse bezüglich der Dependenzgrammatiken durchgeführt.

Der zweite Ansatz versucht die Syntax mithilfe von Konstituenten aufzubauen. Auch ein Konstituentenparser interpretiert einen Text ohne Satztrennungszeichen als einen einzigen Satz. Jedoch ist eine Untersuchung der Konstituentenbäume vielversprechend, da Teilbäume partiell korrekte Syntax enthalten können. Dieser Sachverhalt soll an dem Eingabetext „Go to the fridge. Open the door.“ erläutert werden.

Zunächst sind in Abbildung 4.1 die zwei korrekten Konstituentenbäume für beide Eingabesätze angegeben. Bei einem Eingabetext ohne Satztrennungszeichen, in diesem Beispiel also „Go to the fridge open the door“, könnte ein Konstituentenbaum wie in Abbildung 4.2 aussehen. Dieser Syntaxbaum ist inkorrekt, da beispielsweise das Verb „open“ fehlerhaft mit der Nominalphrase „the fridge“ in Beziehung gebracht wurde. Dennoch enthält dieser Konstituentenbaum die beiden Bäume aus Abbildung 4.1, wie die zwei eingekreisten Bereiche verdeutlichen sollen.

Nach genau solchen verwertbaren Informationen und Heuristiken wurde in der hier durchgeführten Analyse gesucht. Dabei wurden aus dem Analysekorpus die Eingabetexte ohne Satztrennungszeichen als Eingabe für die Konstituentenparser von *Stanford*, *Berkeley*, *Charniak* (siehe Unterabschnitt 3.1.8) und *SENNA* verwendet. Die dadurch entstandenen 56 Syntaxbäume sind im Anhang unter H in der Klammerdarstellung angegeben. Diese Klammerdarstellung ist aufgrund ihrer Verschachtelung sehr unübersichtlich und somit für eine Analyse ungeeignet. Deshalb wurden die Syntaxbäume mithilfe des Werkzeuges unter[MA03] visualisiert und auf den Servern des IPD-Institutes<sup>2</sup> zur Verfügung gestellt.

<sup>2</sup>[https://code.ipd.kit.edu/weigelt/kocybik\\_ba/tree/master/Analyse](https://code.ipd.kit.edu/weigelt/kocybik_ba/tree/master/Analyse)

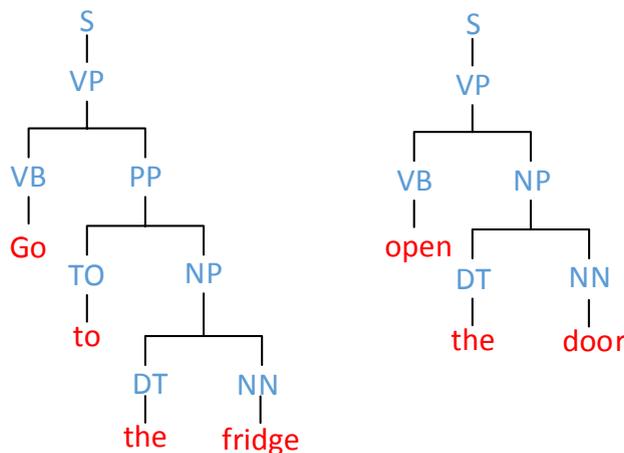


Abbildung 4.1.: Die zwei Konstituentenbäume für „Go to the fridge. Open the door.“

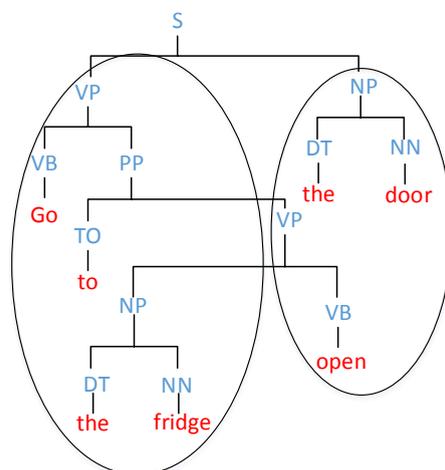


Abbildung 4.2.: Ausgabe des Stanford Parsers grafisch dargestellt

Bei der Analyse der Syntaxbäume ist lediglich eine Heuristik entstanden: Betrachtet man einen Knoten, welcher als Verb gekennzeichnet ist, stehen meist alle im Baum rechts befindlichen Knoten, die kein Verb darstellen, mit diesem Verb in Verbindung. Verwendet man beispielsweise diese Heuristik auf den Syntaxbaum aus Abbildung 4.2, stellt „Go“ den ersten Knoten dar, welcher durch „VB“ als Verb gekennzeichnet ist. Alle rechts befindlichen Knoten, hier „to“, „the“ und „fridge“, stehen somit mit dem Verb „Go“ in Verbindung. Das nächste und letzte Verb im Syntaxbaum ist „open“. Mithilfe der Heuristik stehen die Knoten „the“ und „door“ mit dem Verb in Zusammenhang. Diese Heuristik stützt sich jedoch auf der Annahme, dass die Eingabetexte ausschließlich aus Imperativsätzen bestehen. Für Eingabetexte ohne Satztrennungszeichen die nicht nur aus imperativen Sätzen bestehen, konnten keine Heuristiken, hinsichtlich der Syntaxanalyse mit Standard-Sprachverarbeitungswerkzeugen, gefunden werden, die verwertbare Informationen für die initiale Handlungsrepräsentation bereitstellen.

Aus der Zusammensetzung der Syntaxbäume bei Eingabetexten ohne Satztrennungszeichen konnte nur eine Heuristik abgeleitet werden. Dabei macht diese Heuristik Annahmen über die Satzstruktur des Eingabetextes. Wie später im Abschnitt 5.3 vorgestellt, können

die Informationen, welche die hier vorgestellte Heuristik mithilfe der Syntaxanalyse bereitstellt, auch durch eine alternative Herangehensweise extrahiert werden. Sollen jedoch keine Vorgaben über die Satzstruktur des Eingabetextes getroffen werden, ist das Ergebnis dieser Analyse, dass die Syntaxanalyse von Standard-Sprachverarbeitungswerkzeugen bei Eingabetexten ohne Satztrennungszeichen keine verwertbaren Ergebnisse liefert.

## 4.5. Zuweisung von semantischen Rollen

Wie im Grundlagenkapitel vorgestellt, baut die Zuweisung von semantischen Rollen auf Syntaxbäumen auf, um semantische Prädikat-Argument-Strukturen zu erkennen. Da im vorherigen Abschnitt gezeigt wurde, dass inkorrekte Syntaxbäume beim Fehlen von Satztrennungszeichen entstehen, sollten keine verwertbaren Ergebnisse erwartet werden. Um diese Hypothese zu bestätigen, wurden die Texte des Analysekorpus als Eingabe für die Sprachverarbeitungswerkzeuge *SENN*A, *ClearNLP* und *Illinois NLP Curator* verwendet, um die dabei entstandenen semantischen Rollen zu analysieren. Bei dieser Betrachtung konnte die Vermutung, dass Standard-Sprachverarbeitungswerkzeuge keine verwertbaren semantischen Rollen für Eingabetexte ohne Satztrennungszeichen produzieren, bestätigt werden. Die Ergebnisse der einzelnen Sprachverarbeitungswerkzeuge, bezüglich der Zuweisung von semantischen Rollen, befinden sich auf den Servern des IPD-Institutes<sup>3</sup>.

## 4.6. Fazit und Validität

Im Rahmen der hier durchgeführten Analysen muss beachtet werden, dass einige Einschränkungen getroffen werden mussten. Die verwendeten Eingabetexte des Analysekorpus beinhalten ausschließlich Anweisungen, die bezüglich der Interaktion mit ARMAR-III zu erwarten sind. Zusätzlich stellt der Analysekorpus mit 14 Texten, die meist eine imperative Satzstruktur aufweisen und sich untereinander sehr ähneln, einen verhältnismäßig kleinen Korpus dar. Nichtsdestotrotz genügt der Analysekorpus für die Bewertung des Ansatzes, Standard-Sprachverarbeitungswerkzeuge für die Informationsextraktion, im Rahmen des Gesamtprojektes „PARSE“, zu verwenden. Dabei lag der Schwerpunkt dieses Kapitels auf dem Fehlen von Satztrennungszeichen, da genau diese Problematik in verwandten Arbeiten bewusst umgangen wurde. Die Analyse konnte aufzeigen, dass die Wortarten- und Phrasenerkennung auch mit Eingabetexten ohne Satztrennungszeichen durchgeführt werden kann, ohne ein schlechteres Klassifikationsergebnis erwarten zu müssen. Dabei ist jedoch die Auswahl der Sprachverarbeitungswerkzeuge entscheidend. Im Rahmen der Sprachverarbeitung werden die NLP-Aufgaben der Tokenisierung, Wortartenerkennung und Phrasenerkennung auch als **flache Satzverarbeitung** [KUCC<sup>+</sup>10, S.264] bezeichnet. Bei den darauf aufbauenden NLP-Aufgaben, nämlich der Syntaxanalyse und der Zuweisung von semantischen Rollen, konnten die hier durchgeführten Analysen keine verwertbaren Informationen für Eingabetexte ohne Satztrennungszeichen aufdecken. Somit wird die in dieser Ausarbeitung erstellte initiale Handlungsrepräsentation die Ergebnisse der Wort- und Phrasenerkennung bereitstellen. Zusätzlich wird eine Heuristik zur Aufdeckung von einzelnen Anweisungen innerhalb einer Befehlsfolge vorgestellt, die bei Eingabetexten ohne Satztrennungszeichen und imperativen Satzstrukturen angewendet werden kann. Die Auswahl der dafür verwendeten Sprachverarbeitungswerkzeuge wird, einschließlich der Implementierungsumsetzung, im nächsten Kapitel vorgestellt.

<sup>3</sup>[https://code.ipd.kit.edu/weigelt/kocybik\\_ba/tree/master/Analyse](https://code.ipd.kit.edu/weigelt/kocybik_ba/tree/master/Analyse)



## 5. Entwurf und Implementierung

Dieses Kapitel beschreibt die Realisierung des in dieser Bachelorarbeit zu erstellenden Werkzeuges. Dabei wird zunächst der allgemeine Ansatz aus Kapitel 4 repetiert und der daraus resultierende Entwurf vorgestellt. Anschließend soll die Auswahl der verwendeten Standard-Sprachverarbeitungswerkzeuge diskutiert werden. Hierbei dienen die Analyseergebnisse aus Kapitel 4 als Entscheidungskriterium. Danach wird eine Heuristik vorgestellt, die Befehlsfolgen ohne Satztrennungszeichen in imperativer Satzstruktur erwartet und Befehls Grenzen erkennt. Zuletzt werden ausgewählte Implementierungsdetails, wie die verwendete interne Repräsentation und die Nutzung des implementierten Werkzeuges, vorgestellt.

### 5.1. Entwurf

Der in dieser Bachelorarbeit betrachtete Ansatz (siehe Kapitel 4) ist, Informationen mithilfe von Standard-Sprachverarbeitungswerkzeugen aus dem Eingabetext zu extrahieren und diese anschließend in einer Handlungsrepräsentation darzustellen. Dabei konnte im Analysekapitel festgestellt werden, dass die betrachteten Sprachverarbeitungswerkzeuge hinsichtlich der *flachen Satzverarbeitung* (siehe Abschnitt 4.6) verwertbare Informationen bereitstellen.

Bei der Analyse zur Wortartenerkennung (siehe Abschnitt 4.2) wurden mehrere Sprachverarbeitungswerkzeuge betrachtet. Dabei wurde ersichtlich, dass die Werkzeuge, abhängig von der betrachteten Herausforderung, unterschiedlich gute Klassifikationsergebnisse erzielen. Beispielsweise schneiden einige Wortartenerkennungswerkzeuge hinsichtlich der Analyse ohne Berücksichtigung der Verbformen (siehe Tabelle 4.3) besonders gut ab, wohingegen andere Sprachverarbeitungswerkzeuge Vorteile in der Klassifikation von Verben aufzeigen (siehe Tabelle 4.4). Deshalb soll das hier erstellte Werkzeug die Klassifikation von Wortarten durch eine Kombination mehrerer Wortartenerkennungswerkzeuge bewerkstelligen. Dadurch kann ein Wort beispielsweise genau dann als Verb klassifiziert werden, wenn ein Sprachverarbeitungswerkzeug dieses Wort als Verb kategorisiert, welches bei den Analyseergebnissen zur Klassifikation von Verben besonders gute Ergebnisse erzielen konnte. Durch solch eine Kombination von Sprachverarbeitungswerkzeugen soll die bestmögliche Wortartenklassifikation hinsichtlich der Analyseergebnisse gewährleistet werden. Um die Wortarten mithilfe mehrerer Sprachverarbeitungswerkzeuge zu bestimmen, muss jedes dieser Werkzeuge den Eingabetext erhalten und separat klassifizieren. Anschließend muss das hier erstellte Werkzeug die Ausgaben der unterschiedlichen Wortartenerkennungswerkzeuge entgegennehmen und mithilfe dieser Ausgaben die Wortarten bestimmen.

Neben der Wortartenerkennung soll das hier erstellte Werkzeug die Klassifikation von phrasalen Markierungen mithilfe von Standard-Sprachverarbeitungswerkzeugen ermöglichen. Bei der dabei durchgeführten Analyse (siehe Abschnitt 4.3) konnte genau ein Werkzeug, nämlich *BIOS*, gefunden werden, das die Phrasenerkennung hinreichend gut bewerkstelligt. Da der Phrasenerkennung neben dem Eingabetext die zugehörigen Wortarten benötigt, muss die Klassifikation der Wortarten vor der Phrasenerkennung stattfinden.

Konnten die Wortarten und phrasalen Markierungen aus dem Eingabetext extrahiert werden, kann die initiale Handlungsrepräsentation erstellt werden. Bei der verwendeten Handlungsrepräsentation handelt es sich um einen Graphen. Weshalb ein Graph als Handlungsrepräsentation verwendet wird, wurde im Rahmen der Bachelorarbeit von Michael Schneider [Sch15] diskutiert. Wie die initiale Handlungsrepräsentation erstellt wird und welche Informationen dargestellt werden, wird unter Abschnitt 5.5 diskutiert.

Abbildung 5.1 stellt den Entwurf des hier erstellten Werkzeuges graphisch dar.

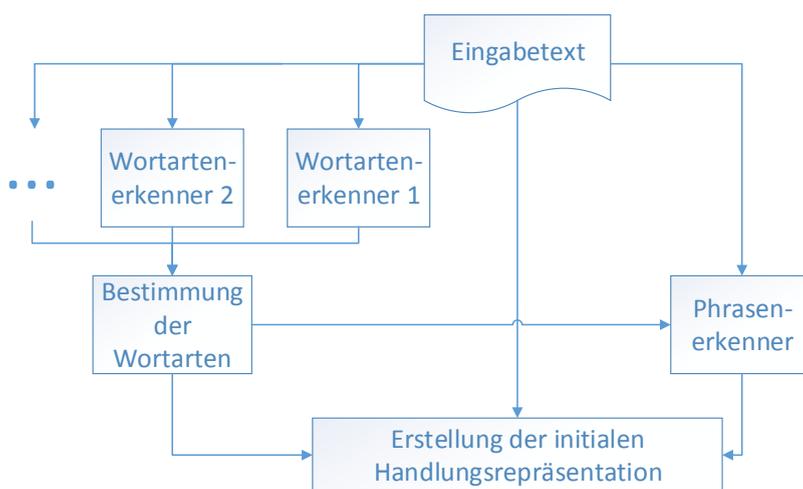


Abbildung 5.1.: Entwurf zur Kombination der verwendeten Sprachverarbeitungswerkzeuge

## 5.2. Die verwendeten Sprachverarbeitungswerkzeuge

Nachdem im vorherigen Abschnitt der grobe Entwurf des hier erstellten Werkzeuges vorgestellt wurde, soll dieser Abschnitt die verwendeten Standard-Sprachverarbeitungswerkzeuge vorstellen und begründen, weshalb diese Werkzeuge ausgewählt wurden.

Im Analyseergebnis unter Tabelle 4.3 hat *SENNA* das beste Klassifikationsergebnis hinsichtlich der Wortartenerkennung für Eingabetexte ohne Satztrennungszeichen erzielt. Deshalb wurde *SENNA* in dem hier erstellten Werkzeug verwendet. Zusätzlich wurde in der Analyse zur Wortartenerkennung (siehe Abschnitt 4.2) die Bedeutung der korrekten Klassifikation von Verben motiviert und eine entsprechende Analyse durchgeführt (siehe Tabelle 4.4). Bei dieser Analyse hat der *Stanford POS Tagger* besonders gute Klassifikationsergebnisse erzielen können. Um die bestmögliche Klassifikation auch unter Berücksichtigung der Verberkennung zu gewährleisten, wird die Wortartenerkennung, in dem in dieser Bachelorarbeit erstellten Werkzeug, durch die gemeinsame Verwendung von *SENNA* und dem *Stanford POS Tagger* realisiert. Dabei werden die Werkzeuge folgendermaßen kombiniert: *SENNA* und der *Stanford POS Tagger* führen zunächst separat die Wortartenmarkierung für den Eingabetext durch. Das in dieser Bachelorarbeit erstellte Werkzeug klassifiziert anschließend ein Token als Verb, falls *SENNA* oder der *Stanford POS Tagger* das Token

als Verb klassifiziert haben. In allen anderen Fällen wird das Klassifikationsergebnis von SENNA für die Wortartenerkennung verwendet.

Neben der Kombination von *SENNA* und dem *Stanford POS Tagger* hätte, statt dem Wortartenerkennung von Stanford, auch *Apache OpenNLP* verwendet werden können. *Apache OpenNLP* hat in der Analyse unter Tabelle 4.4 mit insgesamt fünf Klassifikationsfehlern, hinsichtlich der Erkennung von Verben, das beste Ergebnis erzielt. Demgegenüber konnte der *Stanford POS Tagger* mit sieben Klassifikationsfehlern lediglich das zweitbeste Ergebnis erzielen. Dennoch wurde der Wortartenerkennung von Stanford verwendet: *Apache OpenNLP* konnte das gute Klassifikationsergebnis von fünf nicht erkannten Verben lediglich durch die Heuristik erzielen, ein Token genau dann als Verb zu klassifizieren, wenn das Token in mindestens einer der drei besten Sequenzen als Verb erkannt wurde (siehe Abschnitt 4.2). Jedoch resultiert aus dieser Heuristik die Vermutung, das Token vermehrt als Verben klassifiziert werden, obwohl es sich um gar keine Verben handelt. Alternativ ausgedrückt besteht die Vermutung, dass durch diese Heuristik die Zuweisung von Verben „überschätzt“ wird. Diese Vermutung konnte jedoch mithilfe des Analysekorpus weder bestätigt noch widerlegt werden. Da aber die Möglichkeit besteht, dass Token vermehrt inkorrekt als Verben klassifiziert werden, wurde auf die Verwendung von *Apache OpenNLP* mit der vorgestellten Heuristik verzichtet.

Der Fokus des hier erstellten Werkzeuges liegt auf Eingabetexten ohne Satztrennungszeichen. Da im Rahmen der Analysen in Kapitel 4 aber auch Eingabetexte mit Satztrennungszeichen analysiert wurden, bietet das Werkzeug als Zusatzfunktionalität eine Schnittstelle an, die es ermöglicht, Eingabetexte mit Satztrennungszeichen zu verarbeiten. Dazu wird vor der Wortartenerkennung eine Tokenisierung auf Satzebene (siehe Unterabschnitt 2.2.1) durch das Tokenisierungswerkzeug von *Stanford* [MSB<sup>+</sup>14] ausgeführt. Anschließend wird die Wortartenerkennung satzweise ausschließlich mit SENNA durchgeführt. Durch diese Vorgehensweise können die 16 Fehler (siehe Tabelle 4.2), die *SENNA* durch die fehlerhafte Tokenisierung auf Satzebene produziert, beseitigt werden. Somit stellt *SENNA* das beste Klassifikationsergebnis, unter Berücksichtigung der beseitigten Fehler, im Analyseergebnis in Tabelle 4.3 für Eingabetexte mit Satztrennungszeichen dar. Wie bei der Verwendung des hier erstellten Werkzeuges zwischen Eingabetexten mit und ohne Satztrennungszeichen unterschieden werden kann, wird im Abschnitt 5.5 vorgestellt.

Wie bereits im Abschnitt 5.1 beschrieben, wird die Phrasenerkennung mit dem Sprachverarbeitungswerkzeug *BIOS* realisiert. *BIOS* konnte mit Abstand das beste Klassifikationsergebnis bezüglich der Phrasenerkennung erzielen (siehe Abschnitt 4.3).

### 5.3. Heuristik zum Erkennen von Befehls Grenzen

Die Standard-Sprachverarbeitungswerkzeuge konnten bei Eingabetexten ohne Satztrennungszeichen weder in der Syntaxanalyse, noch in der Zuweisung von semantischen Rollen verwertbare Ausgaben für die initiale Handlungsrepräsentation bereitstellen. Jedoch wären die Informationen, die durch diese NLP-Aufgaben extrahiert werden, hinsichtlich der Zusammenhänge einzelner Phrasen hilfreich. Die durch *BIOS* realisierte Phrasenerkennung stellt zwar erste Zusammenhänge zwischen einzelnen Token her, jedoch entstehen dadurch beispielsweise keine Zusammenhänge zwischen den einzelnen Verben des Eingabetextes und den jeweils zugehörigen Argumenten. Betrachtet man den Eingabetext „*Open the fridge and bring me the orange juice*“ würde die Ausgabe hinsichtlich der Wortarten- und Phrasenerkennung wie in Tabelle 5.1 aussehen. Diese Tabelle verdeutlicht, dass die bisherige Ausgabe weder einen Zusammenhang zwischen dem Verb „*open*“ und der Tokensequenz „*the fridge*“, noch zwischen dem Verb „*bring*“ und der Sequenz „*me the orange juice*“ herstellt. Diese Zusammenhänge sind jedoch hilfreich, um ARMAR-III neue Anweisungen beizubringen. Deshalb wurde in dieser Ausarbeitung eine Heuristik entwickelt,

die es ermöglicht solche Zusammenhänge festzustellen. Diese Heuristik wurde anhand der Texte des Analysekörpers abgeleitet, die konkret für das Gesamtprojekt „PARSE“ angefertigt wurden. Wie im Abschnitt 4.6 beschrieben, handelt es sich dabei um Eingabetexte in imperativer Satzstruktur, die keine Satztrennungszeichen enthalten. Das Werkzeug bietet für Eingabetexte, die von diesen Annahmen abweichen, eine Schnittstelle ohne die Durchführung der im Folgenden vorgestellten Heuristik an (siehe Abschnitt 5.5).

Token	Wortart	Chunk
open	VB	B-VP
the	DT	B-NP
fridge	NN	I-NP
and	CC	O
bring	VB	B-VP
me	PRP	B-NP
the	DT	B-NP
orange	NN	I-NP
juice	NN	I-NP

Tabelle 5.1.: Ausgabe für die Wortarten- und Phrasenerkennung

Die implementierte Heuristik geht im Allgemeinen folgendermaßen vor: Der Eingabetext wird als Folge von Befehlen in imperativer Satzstruktur interpretiert. Dabei wird angenommen, dass jeder Befehl, der eine imperative Satzstruktur besitzt, aus einer Verbphrase, die den Beginn des Befehls darstellt, und einer Folge von Nicht-Verbphrasen besteht. Ein neuer Befehl beginnt, nach einer Folge von Nicht-Verbphrasen, bei der nächsten identifizierten Verbphrase. Liegt nun der Eingabetext einschließlich der Wortarten oder der phrasalen Markierungen vor, kann jedem Token mithilfe der hier getroffenen Annahmen eine Befehlsnummer zugewiesen werden. Dabei erhalten alle Token, die einen Befehl darstellen, die gleiche Befehlsnummer. Durch die gleiche Befehlsnummer kann ein Zusammenhang zwischen den Verbphrasen und den folgenden Nicht-Verbphrasen eines Befehls hergestellt werden. Die hier implementierte Heuristik macht noch weitere Annahmen, wie beispielsweise, dass Befehle genau aus einem oder zwei aufeinanderfolgenden Verben bestehen müssen und dass neue Befehle nicht nur durch das Erkennen von Verbphrasen, nach einer Folge von Nicht-Verbphrasen, sondern auch durch Schlüsselwörter, wie „and“, eingeleitet werden können. Die implementierte Heuristik ist im Anhang unter J zu finden. Der genaue Ablauf des Algorithmus wird im Folgenden beschrieben:

**Eingabe:** Als Eingabe dienen die einzelnen Wörter einschließlich der zugehörigen Wortarten. Der Algorithmus betrachtet die einzelnen Wörter, entsprechend ihres Auftretens im Eingabetext, nacheinander. Initial ist die Variable der Befehlsnummer auf null gesetzt.

**1. Schritt:** Bei der Betrachtung des ersten Wortes wird überprüft, ob es sich um ein Verb handelt. Falls nicht, wird diesem Token die momentane Befehlsnummer zugewiesen und das nächste Token betrachtet. Dieses Vorgehen wird bis zum ersten erkannten Verb fortgesetzt.

**2. Schritt:** Bei der Erkennung des ersten Verbes bekommt dieses Verb ebenfalls die momentane Befehlsnummer zugewiesen. Anschließend wird für das nächste Token des Eingabetextes untersucht, ob es sich um ein Verb handelt: Handelt es sich um kein Verb, wird die Betrachtung **dieses** Tokens mit dem 3. Schritt des Algorithmus fortgesetzt. Stellt dieses Token hingegen ein Verb dar, wird diesem Token die momentane Befehlsnummer zugewiesen und das **darauffolgende** Token mithilfe des 3. Schrittes untersucht. Damit sollen zusammengehörige Verben, wie „is closed“, innerhalb eines Befehls zusammenfasst werden.

**3. Schritt:** In diesem Schritt wird nach dem Ende des Befehls, beziehungsweise nach dem Beginn eines neuen Befehls gesucht. Handelt es sich bei dem betrachteten Token um eines der Wörter „and“, „if“ oder „when“, wird die Variable der Befehlsnummer um eins inkrementiert, dem Token die inkrementierte Befehlsnummer zugewiesen und der Algorithmus im 1. Schritt, unter Betrachtung des **nächsten** Tokens, fortgesetzt. Wurde hingegen ein Verb erkannt, wird die Befehlsnummer um eins inkrementiert und der Algorithmus im 2. Schritt, unter Betrachtung **desselben** Tokens, welches das Verb darstellt, fortgesetzt. Trifft für das betrachtete Token keines dieser Bedingungen zu, wird ihm die momentane Befehlsnummer zugewiesen und die Suche nach einer Befehlsgrenze mit 3. Schritt des Algorithmus, unter der Betrachtung des **nächsten** Tokens, fortgesetzt.

Unter Tabelle 5.2 sind für das obige Beispiel die Befehlsnummern, mithilfe der hier vorgestellten Heuristik, angegeben.

Token	Wortart	Chunk	Befehlsnummer
open	VB	B-VP	0
the	DT	B-NP	0
fridge	NN	I-NP	0
and	CC	O	1
bring	VB	B-VP	1
me	PRP	B-NP	1
the	DT	B-NP	1
orange	NN	I-NP	1
juice	NN	I-NP	1

Tabelle 5.2.: Ausgabe für die Wortarten- und Phrasenerkennung

Abschließend muss die Abbildung 5.1, die den Entwurf des hier erstellten Werkzeuges skizziert, hinsichtlich der Zuweisung von Befehlsnummern angepasst werden. Die entsprechende Anpassung wurde in Abbildung 5.2 bereitgestellt. Hierbei unterscheidet sich die hier aufgeführte Abbildung dadurch, dass die Befehlsnummern mithilfe der hier vorgestellten Heuristik, neben der Wortarten und der phrasalen Markierungen, in der initialen Handlungsrepräsentation mit einbezogen werden.

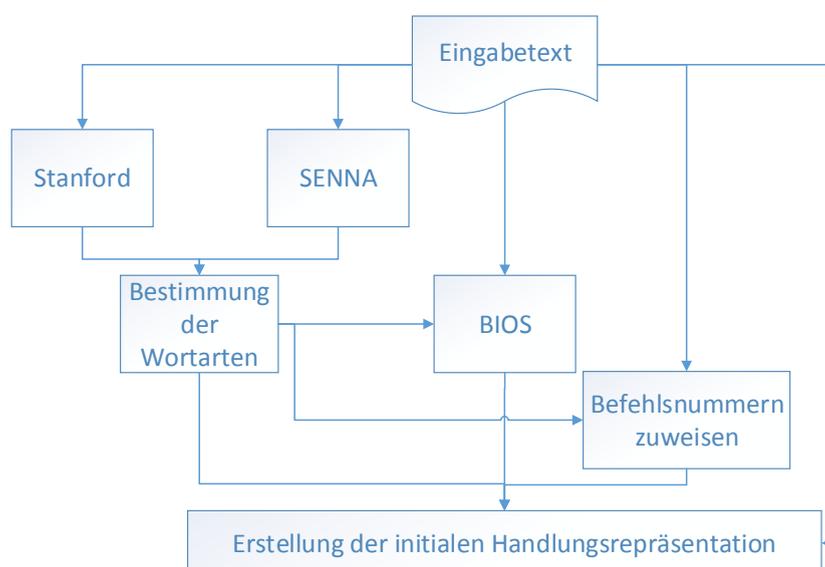


Abbildung 5.2.: Angepasster Entwurf

## 5.4. Interne Repräsentation

Bei dem in dieser Bachelorarbeit erstellten Werkzeug handelt es sich um ein in Java realisiertes Maven-Projekt. In den beiden vorherigen Abschnitten wurde erläutert, dass die Wortartenerkennung mithilfe der Kombination aus *SENN*A und dem *Stanford POS Tagger* durchgeführt wird. Anschließend wird mithilfe der erkannten Wortarten die Phrasenerkennung durch *BIOS* und die Zuweisung der Befehlsnummern mithilfe der vorgestellten Heuristik durchgeführt. Um die Kommunikation zwischen den einzelnen Sprachverarbeitungswerkzeugen *SENN*A, dem *Stanford POS Tagger* und *BIOS* zu vereinfachen, wurde eine interne Repräsentation entwickelt. Bei dieser Repräsentation handelt es sich um ein Feld (engl. array) der implementierten Klasse **Token**. Ein Objekt der Klasse **Token** repräsentiert, analog zur Wortsegmentierung im Unterabschnitt 2.2.1), genau ein Wort des Eingabetextes. Die **Token**-Klasse besitzt folgende Attribute:

- `position:int`  
Dieses Attribut speichert die Position des Tokens innerhalb des Eingabetextes.
- `partofSpeech:POSTag`  
Dieses Attribut speichert die Wortart des Tokens. Bei dem Typ `POSTag` handelt es sich um einen Aufzählungstyp (engl. enumeration type) der die Wortarten gemäß des **Penn Treebank-Standards** (siehe Anhang A) definiert.
- `chunk:ChunkTag`  
Dieses Attribut speichert die phrasale Markierung (engl. chunk) des Tokens. Bei dem Typ `ChunkTag` handelt es sich um einen Aufzählungstyp der die phrasalen Markierungen gemäß des **CoNLL-Standards** im IOB-Format (siehe Unterabschnitt 2.2.5 und Anhang B) definiert.
- `instructionNumber:int`  
Dieses Attribut speichert die Befehlsnummer des Tokens, welche durch den Algorithmus aus Abschnitt 5.3 generiert wird.

## 5.5. Verwendung des erstellten Werkzeuges

Um die Verwendung des in dieser Bachelorarbeit erstellten Werkzeuges zu verdeutlichen, sollen die Methoden vorgestellt werden, die es ermöglichen, einen Eingabetext, der die Ausgabe eines automatischen Spracherkenners darstellt, in die initiale Handlungsrepräsentation zu projizieren:

```
parse(text:String,periods:bool,opt:bool,imp:bool,list:WordPosType):Token[]
```

Diese Methode erhält den Eingabetext über den Parameter `text`. Mithilfe des Wahrheitswertes `periods` kann angegeben werden, ob der Eingabetext Satztrennungszeichen beinhaltet. Falls der Eingabetext Satztrennungszeichen beinhaltet, wird, wie in Abschnitt 5.2 beschrieben, der Eingabetext mithilfe des Tokenisierungswerkzeuges von Stanford auf Satzebene segmentiert. Wird `periods` auf Wahr gesetzt, wird der Wert des Parameters `opt` ignoriert. Mithilfe des Parameters `opt` kann gesteuert werden, ob die Wortartenerkennung ausschließlich mit *SENN*A (`opt = falsch`) oder durch die in Abschnitt 5.2 beschriebene Kombination von *SENN*A und dem *Stanford POS Tagger* (`opt = wahr`) durchgeführt werden soll. Der Parameter `opt` sollte angesichts der Analyseergebnisse immer auf Wahr gesetzt werden. Die Konfigurationsmöglichkeit dient lediglich zur Evaluation in Kapitel 6, um festzustellen, welcher Vorteil durch die Kombination von *SENN*A und dem *Stanford POS Tagger*, gegenüber der ausschließlichen Verwendung von *SENN*A, hinsichtlich des Klassifikationsergebnisses entsteht. Der Parameter `opt` wird bei Eingabetexten mit Satztrennungszeichen ignoriert, weil zur Klassifikation der Wortarten ausschließlich *SENN*A

verwendet werden soll (siehe Abschnitt 5.2), da hierbei das beste Klassifikationsergebnis erzielt werden konnte und eine Kombination mehrere Wortartenerkennung bei vorhandenen Satztrennungszeichen nicht analysiert wurde. Der Parameter `imp` gibt an, ob die in Abschnitt 5.3 vorgestellte Heuristik angewendet werden soll. Die Heuristik sollte ausschließlich auf Eingabetexte mit imperativer Satzstruktur angewendet werden. Der letzte Parameter `list` ermöglicht es einzelnen Wörtern des Eingabetextes eine feste Wortart zuzuordnen. Dazu stellt die Klasse `WordPosType` zwei String-Felder als Attribute bereit, nämlich `words:String[]` und `pos:PosTag[]`. Ein Element des Feldes `words` repräsentiert ein Wort, welches eine feste Wortart zugewiesen bekommen soll. Die Wortart muss dazu auf der identischen Position des Feldes `pos` hinterlegt werden. Soll beispielsweise „Armar“, unabhängig von den verwendeten Wortartenerkennung, als Eigenname klassifiziert werden, kann beispielsweise die Zeichenfolge „Armar“ im Feld `words` und die Wortart `NNP` im Feld `pos` jeweils auf Position null hinterlegt werden. Die Möglichkeit solch einer festen Zuweisung von Wortarten kann vor allem bei domänenabhängigen Wörtern hilfreich sein, die Folgefehler in der Phrasenerkennung und der Zuweisung von Befehlsnummern verursachen. Die hier vorgestellte Methode stellt als Rückgabewert ein Feld von `Token`-Objekten (siehe Abschnitt 5.4) bereit.

```
parse(text:String[], opt:bool, imp:bool, list:WordPosType):Token[]
```

Diese Methode ermöglicht es einen bereits auf Satzebene segmentierten Eingabetext mit Wortarten, phrasalen Markierungen und Befehlsnummern zu annotieren. Dabei repräsentiert ein Satz genau ein Element des Feldes `text`. Das Setzen eines abschließenden Punktes ist für die einzelnen Sätze optional. Die Parameter `opt`, `imp` und `list` sind analog zur vorherigen Methodenbeschreibung zu verstehen.

```
createAGGGraph(input:Token[], saveAGGtoPath:String):ParseGraph
```

Die zwei bereits vorgestellten Methoden ermöglichen die Informationsextraktion hinsichtlich der Wortartenmarkierung, der Phrasenerkennung und der Zuweisung von Befehlsnummern. Dabei dient in beiden Fällen ein Feld von `Token`-Objekten als Rückgabewert. Bei der in dieser Bachelorarbeit zu erstellenden initialen Handlungsrepräsentation soll es sich jedoch um einen AGG-Graphen (engl. Attributed Graph Grammar) [Tae04] handeln. Weshalb ein AGG-Graph als Handlungsrepräsentation verwendet wird und welche Vorteile damit verbunden sind, wurde im Rahmen einer weiteren Bachelorarbeit unter [Sch15] erarbeitet. In dieser Bachelorarbeit wurde zusätzlich eine Fassade entwickelt, die den Zugriff auf die AGG-Rahmenarchitektur vereinfacht und einen AGG-Graphen durch ein Objekt der Klasse `ParseGraph` darstellt. Die Methode `createAGGGraph(input:Token[], saveAGGtoPath:String):ParseGraph` bietet die Funktionalität an, ein Feld von `Token`-Objekten auf ein Objekt der Klasse `ParseGraph` abzubilden. Zusätzlich besteht die Möglichkeit, den AGG-Graphen durch die Angabe eines Dateipfades als XML-Datei zu hinterlegen. Soll keine XML-Datei hinterlegt werden, muss der Parameter `saveAGGtoPath` auf `null` gesetzt werden.

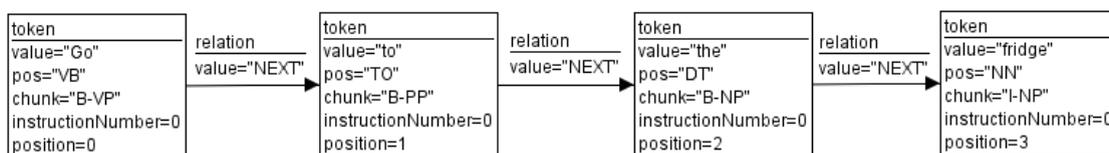


Abbildung 5.3.: Exemplarischer AGG-Graph

Um also einen Eingabetext, der die Ausgabe eines automatischen Spracherkenners darstellt, in die initiale Handlungsrepräsentation zu projizieren, muss das hier erstellte Werkzeug folgendermaßen verwendet werden: Zunächst muss eine der beiden `parse`-Methoden

mit dem Eingabetext aufgerufen werden, um ein Feld von **Token**-Objekten als Rückgabewert zu erhalten. Anschließend kann die `createAGGGraph`-Methode verwendet werden, um das **Token**-Feld auf den zugehörigen AGG-Graphen abzubilden, welcher die initiale Handlungsrepräsentation darstellt. Der dadurch erhaltene AGG-Graph stellt lediglich das **Token**-Feld als gerichteten Graphen dar. Dabei repräsentiert ein Token genau einem Knoten und aufeinanderfolgenden Wörter werden durch Kanten miteinander verbunden. Unter Abbildung 5.3 wurde zur Veranschaulichung die initiale Handlungsrepräsentation zum Eingabetext „*Go to the fridge*“ bereitgestellt, welche durch das in dieser Bachelorarbeit erstellte Werkzeug generiert wurde.

## 6. Evaluation

Dieses Kapitel befasst sich mit der Evaluation des in dieser Bachelorarbeit erstellten Werkzeuges. Der Evaluationsvorgang wurde mithilfe eines Evaluationskorpus durchgeführt, der sowohl händische Transkriptionen als auch Ausgaben eines automatischen Spracherkenners (ASR) enthält. Als automatischer Spracherkennung wurde die *AT&T WATSON API*<sup>1</sup> verwendet. Bei der hier durchgeführten Evaluation werden zunächst die Ausgaben des Werkzeuges für die einzelnen Texte des Korpus erzeugt. Anschließend sollen diese Ausgaben mit Musterlösungen (Gold-Standard) verglichen werden, um die Genauigkeit (siehe Abschnitt 4.2) des Werkzeuges, hinsichtlich der Wortartenmarkierung, Phrasenerkennung und Zuweisung von Befehlsnummern, zu berechnen. Die Evaluation wurde zusätzlich in verschiedenen Varianten durchgeführt, die sich neben den verwendeten Texten des Korpus, in den Konfigurationseinstellungen des Werkzeuges (siehe Abschnitt 5.5) unterscheiden. In den folgenden Abschnitten wird der Evaluationskorpus vorgestellt, gezeigt wie der Vergleich zwischen den Musterlösungen und den Ausgaben des Werkzeuges durchgeführt wird und die Evaluationsergebnisse diskutiert.

### 6.1. Evaluationskorpus und Vergleichsvorgang

Um das Werkzeug zu evaluieren, wird ein Korpus von Eingabetexten benötigt. Zur Erstellung dieses Evaluationskorpus wurden 20 Sprachaufnahmen des Sprachkorpus, welcher im Rahmen einer weiteren Bachelorarbeit unter [Gue15] erstellt wurde, zufällig ausgewählt. Zu jeder dieser Aufnahmen wurde eine händische Transkription und eine ASR-Ausgabe mit der *AT&T WATSON API* erstellt. Der in dieser Evaluation verwendete Korpus beinhaltet dementsprechend für die 20 zufällig ausgewählten Sprachaufnahmen die händischen Transkriptionen und ASR-Ausgaben. Im Folgenden sollen die Texte des Evaluationskorpus genauer betrachtet werden: Die Transkriptionen enthalten zwar keine Wortfehler, jedoch wurden bewusst Füllwörter und Wortwiederholungen in die Transkriptionen mit aufgenommen. Zusätzlich wurde auf das Setzen von Satztrennungszeichen verzichtet. Die 20 Transkriptionen sind im Anhang unter I.1 zu finden. Der zwölfte Text lautet beispielsweise:

*uhm go to to the fridge open the fridge uhm get the orange juice uhm close the fridge uhm bring the orange juice to me*

---

<sup>1</sup><http://developer.att.com/apis/speech>

Die 20 ASR-Ausgaben wurden mithilfe der *AT&T WATSON API* generiert. Dabei handelt es sich um einen als Webdienst bereitgestellten Spracherkenner, der Sprachaufnahmen als WAV-Datei entgegennimmt und in Textform bringt. Die, mithilfe der ASR-Ausgaben durchgeführte, Evaluation soll die Auswirkungen von Wortfehlern in der Ausgabe des hier erstellten Werkzeuges aufdecken. Es muss jedoch beachtet werden, dass dieser Spracherkenner vorher weder evaluiert, noch auf der Domäne ARMAR-III neue Anweisungen beizubringen, trainiert wurde. Die Auswahl des Spracherkenners ist zufällig, da zum Zeitpunkt dieser Bachelorarbeit noch nicht feststeht, welcher Spracherkenner im Gesamtprojekt „*PARSE*“ verwendet werden soll. Die ASR-Ausgaben sind im Anhang unter I.2 zu finden. Als beispielhafte Ausgabe wird der zwölfte Text aufgeführt:

*i'm go to learn to french open french and get the orange juice close the fridge um bring the arches to me*

Die 20 Sprachaufnahmen wurden von insgesamt 13 Probanden erstellt. In den Transkriptionen entstanden somit insgesamt 414 Wörter, die 74 Befehle darstellen. Im Gegensatz dazu ist zu beachten, dass der automatische Spracherkenner Wortfehler produziert. Deshalb entspricht die Wortanzahl der 20 ASR-Texte mit 395 Wörtern nicht den Transkriptionen.

Die hier durchgeführte Evaluation wurde in mehreren Varianten durchgeführt. Dabei unterscheiden sich die Varianten neben der Konfiguration des Werkzeuges (siehe Abschnitt 5.5) darin, ob die Transkriptionen oder die ASR-Ausgaben betrachtet wurden. Im Folgenden soll die durchgeführte Evaluation, abhängig von den Transkriptionen beziehungsweise den ASR-Texten, vorgestellt werden.

### 6.1.1. Evaluationsvorgang: Transkriptionen

Für jede der 20 Transkriptionen des Evaluationskorpus wurde eine Musterlösung angefertigt, welche für jedes Token die korrekte Wortart, phrasale Markierung und Befehlsnummer vorgibt. Die Musterlösungen befinden sich im Anhang unter I.3. Diese Musterlösungen sollen mit den Ausgaben des Werkzeuges für die Transkriptionen des Evaluationskorpus verglichen werden. In diesem Vergleich kann, wie in den Analysen der Standard-Sprachverarbeitungswerkzeuge zur Wortartenerkennung in Abschnitt 4.2, die Genauigkeit (engl. *accuracy*) des hier erstellten Werkzeuges für die Wortartenmarkierung, Phrasenerkennung und Zuweisung von Befehlsnummern angegeben werden. Im Falle der Wortartenerkennung gibt die Genauigkeit den Prozentsatz aller korrekt erkannten Wortarten hinsichtlich der Musterlösung an. Dazu wird für jedes Token der 20 Transkriptionen die Wortart, die das hier erstellte Werkzeug in seiner Ausgabe produziert, mit der Wortart der Musterlösung verglichen. Stimmen die Wortarten überein, handelt es sich um eine korrekt erkannte Wortart. Unterscheiden sich die Wortarten, wurde eine falsch klassifizierte Wortart identifiziert. Somit lässt sich die Genauigkeit mit folgender Formel berechnen:

$$\text{Genauigkeit} = \frac{\text{Anzahl der korrekt erkannten Wortarten}}{\text{Gesamtanzahl der zu klassifizierenden Wortarten}}$$

Für die Phrasenerkennung ist die Genauigkeit analog zu berechnen, dabei werden lediglich statt den Wortarten, die phrasalen Markierungen betrachtet. Nur bei der Betrachtung der Befehlsnummern reicht ein einfacher Vergleich zwischen der Musterlösung und Ausgabe des Werkzeuges nicht aus. Bei der Zuweisung von Befehlsnummern, die jeweils um eins inkrementiert werden, führen Fehler an einer Stelle, bei einem einfachen Vergleich der Befehlsnummern, zwischen der Musterlösung und der Ausgabe des Werkzeuges, bei den darauffolgenden Token zu Folgefehlern. Zur Verdeutlichung soll das Beispiel in Tabelle 6.1 betrachtet werden.

In dieser Tabelle wurden zwei Folgen von Befehlsnummern für den in der ersten Spalte dargestellten Eingabesatz „*open the fridge after that grab the orange juice*“ angegeben.

Token	Befehlsnummer	
	Musterlösung	Beispiel
open	0	0
the	0	0
fridge	0	0
after	0	1
that	0	1
grab	1	2
the	1	2
orange	1	2
juice	1	2

Tabelle 6.1.: Genauigkeitsmaß hinsichtlich der Befehlsnummern

Die zweite Spalte stellt die korrekten Befehlsnummern dar. In der dritten Spalte wurde eine inkorrekte Folge von Befehlsnummern angegeben. Hierbei wurde die Tokensequenz „*after that*“ fehlerhaft als Befehl erkannt. Im Hinblick auf die in Abschnitt 5.3 vorgestellte Heuristik, könnte diese fehlerhafte Folge von Befehlsnummern genau dann entstehen, wenn die Token „*open*“, „*after*“ und „*grab*“ als Verben klassifiziert wurden. Anhand dieses Beispiels wird deutlich, dass ein einfacher Vergleich der Befehlsnummern nicht ausreicht. Durch einen einfachen Vergleich zwischen der Musterlösung und der beispielhaften Befehlsfolge würden sich sechs inkorrekte („*after*“, „*that*“, „*grab*“, „*the*“, „*orange*“ und „*juice*“) und drei korrekte („*open*“, „*the*“ und „*fridge*“) Befehlsnummern ergeben. Jedoch wurde die Wortfolge „*grab the orange juice*“ korrekt als zusammengehöriger Befehl erkannt. Da aber die Befehlsnummer, aufgrund des fehlerhaft identifizierten Befehls „*after that*“, nicht übereinstimmt, kommt es bei solch einem einfachen Vergleich zu Folgefehlern. Das Ziel in der Angabe der Genauigkeit muss demnach sein, solche Folgefehler zu ignorieren. Dazu wurde in der hier durchgeführten Evaluation ein Token hinsichtlich der Zuweisung von Befehlsnummern genau dann als fehlerhaft eingestuft, wenn das Token nicht in dem Befehl enthalten ist, welcher durch die Musterlösung vorgegeben ist. In der Ausgabe der dritten Spalte sind demnach die Token „*after*“ und „*that*“ nicht dem Befehl der Musterlösung („*open the fridge after that*“) zugeordnet. Somit ergeben sich zwei inkorrekte und sieben korrekte Befehlsnummern zur Berechnung der Genauigkeit.

### 6.1.2. Evaluationsvorgang: ASR-Texte

In diesem Abschnitt wird beschrieben, wie die Evaluation mit den ASR-Texten durchgeführt wird. Dabei sollte der Evaluationsvorgang analog zum Vorgang aus Unterabschnitt 6.1.1 aufgebaut werden, um die Vergleichbarkeit zwischen den Evaluationsvarianten zu gewährleisten:

Zur Evaluation des Werkzeuges soll auch in diesem Fall das Genauigkeitsmaß verwendet werden. Dabei sind zunächst die Ausgaben des hier erstellten Werkzeuges für jeden der 20 ASR-Texte notwendig. Diese Ausgaben sollen mit den Musterlösungen, die sich im Anhang unter I.3 befinden, verglichen werden. Dabei handelt es sich um die Musterlösungen der Transkriptionen. Falls die Ausgabe des automatischen Spracherkenners der Transkription entspricht, wird wie im Unterabschnitt 6.1.1 vorgegangen, um zu entscheiden, ob es sich um korrekte oder inkorrekte Wortarten, phrasale Markierungen oder Befehlsnummern handelt. Jedoch tritt in der hier durchgeführten Evaluation ausschließlich der Fall auf, dass sich die Transkriptionen und die ASR-Texte voneinander unterscheiden, obwohl bei der Erstellung dieselbe Sprachaufnahme verwendet wurde. Dies liegt an den Wortfehlern, die bei der Verwendung eines automatischen Spracherkenners auftreten. Deshalb muss vor dem Vergleich der Token, die ASR-Ausgabe mit der Wortfolge der Transkription ausgerichtet werden.

Die Ausrichtung der Texte wird wie folgt umgesetzt: Ist ein Wort in der ASR-Ausgabe zu finden, welches einem Wort der Musterlösung entspricht, wird ein Vergleich zwischen den Token durchgeführt. Dabei müssen die zwei zu vergleichenden Wörter nicht identisch sein. Es genügt, wenn klar zu Erkennen ist, dass das Wort in der ASR-Ausgabe einem Wort aus der Musterlösung entspricht. Bei einem Vergleich zwischen den Token wird wie im Unterabschnitt 6.1.1 vorgegangen, um zu entscheiden, ob es sich um korrekte oder inkorrekte Wortarten, phrasale Markierungen oder Befehlsnummern handelt. Ist kein Wort in der ASR-Ausgabe zu finden, das einem Wort in der Musterlösung entspricht, wird ein Fehler hinsichtlich der Wortart, phrasalen Markierung und Befehlsnummer eingerechnet. Zur Verdeutlichung des Evaluationsvorganges wird das Beispiel aus Tabelle 6.2 betrachtet:

Transkription	ASR Ausgabe
	I/PRP/NP/0
go/VB/B-VP/0	→ go/VB/B-VP/0
to/TO/B-VP/0	→ to/TO/B-VP/0
the/DT/B-NP/0	
fridge/NN/I-NP/0	→ bridge/NN/I-NP/0

Tabelle 6.2.: Beispiel zum Vergleich der Wortfolgen

In der ersten Spalte ist die Musterlösung des transkribierten Textes „*go to the fridge*“ in der Darstellungsform *Wort/Wortart/phrasale Markierung/Befehlsnummer* angegeben. Diese Musterlösung soll exemplarisch mit der Ausgabe des Werkzeuges für den ASR-Text „*I go to fridge*“ evaluiert werden: Da sich die Texte der Musterlösung und des automatischen Spracherkenners unterscheiden, muss der Text vor dem Vergleich der Token ausgerichtet werden. Hierfür wird zunächst das erste Wort „*go*“ der Musterlösung betrachtet. Da der ASR-Text das Wort „*go*“ beinhaltet, können diese beiden Token hinsichtlich der Wortart, phrasalen Markierung und Befehlsnummer verglichen werden. Dabei ist festzustellen, dass alle Klassifikationen korrekt sind. Anschließend kann das Wort „*to*“ der Musterlösung betrachtet werden. Auch hier ist das Wort im ASR-Text mit den korrekten Klassifikationen angegeben. Für das Wort „*the*“ kann hingegen kein Wort im ASR-Text gefunden werden. In diesem Fall wird jeweils ein Fehler für die Wortartenmarkierung, Phrasenerkennung und Zuweisung der Befehlsnummer eingerechnet. Das letzte Wort der Musterlösung „*fridge*“ wurde in diesem Beispiel vom automatischen Spracherkenner inkorrekt als „*bridge*“ erkannt. Da aber klar zu Erkennen ist, dass das Wort „*bridge*“ dem Wort „*fridge*“ entspricht, wird ein Vergleich zwischen den beiden Token durchgeführt. Bei dem Vergleich ist festzustellen, dass alle Klassifikationen korrekt sind. Mit dem letzten Wort der Musterlösung ist der Vergleich abgeschlossen. Da die Ausrichtung der Wörter von der Musterlösung aus durchgeführt wird, bleibt die Präposition „*I*“ des ASR-Textes in der Evaluation unberücksichtigt. Somit ergeben sich in diesem Beispiel jeweils drei korrekte Wortarten, phrasale Markierungen und Befehlsnummern. Wohingegen jeweils eine inkorrekte Wortart, phrasale Markierung und Befehlsnummer vorzufinden ist.

Dieses vorgestellte Vorgehen hat den Vorteil, dass die Tokenanzahl der Musterlösung in die Berechnung der Genauigkeit fließt. Dadurch können die Evaluationsergebnisse der Transkriptionen mit denen der ASR-Texte verglichen werden.

## 6.2. Evaluationsvarianten

In diesem Abschnitt sollen die unterschiedlichen Evaluationsvarianten vorgestellt werden. Dabei unterscheiden sich die Varianten hinsichtlich der verwendeten Texte des Evaluationskorpus (Transkriptionen und ASR-Texte) und der Konfiguration des Werkzeuges (siehe Abschnitt 5.5). Im Folgenden werden die Evaluationsvarianten beschrieben:

**1. Variante:** In der ersten Evaluationsvariante wurden die Transkriptionen des Evaluationskorpus berücksichtigt (siehe Anhang I.1). Für jeden der 20 Transkriptionen wurde eine Ausgabe mit dem hier erstellten Werkzeug erstellt. Dabei wurde in den Konfigurationsmöglichkeiten des Werkzeuges ausschließlich *SENNA* zur Wortartenerkennung verwendet. Die Ausgaben dieser Evaluationsvariante sind im Anhang unter I.4 zu finden.

**2. Variante:** Die zweite Evaluationsvariante unterscheidet sich von der ersten Variante lediglich dadurch, dass zur Wortartenerkennung die Kombination aus *SENNA* und dem *Stanford POS Tager* verwendet wurde (siehe Abschnitt 5.2). Die Ausgaben des Werkzeuges für diese Evaluationsvariante sind im Anhang unter I.5 zu finden.

**3. Variante:** Die dritte Evaluationsvariante unterscheidet sich von der zweiten Variante dadurch, dass zusätzlich eine Liste von vorgegebenen Wort-Wortart-Paaren vorgegeben wurde (siehe Abschnitt 5.5). Dabei wurde „*Armar*“ als Eigename, „*Okay*“ und „*Hey*“ als Interjektion und „*please*“ als Adverb vorgegeben. Die Auswahl dieser Paare wurde anhand der Evaluationsergebnisse der 2. Variante abgeleitet, um die Wortartenerkennung zu optimieren. Die Ausgabe des Werkzeuges ist im Anhang unter I.6 zu finden.

**4. Variante:** In der letzten Variante wurde der Evaluationskorpus aus den ASR-Texten berücksichtigt (siehe Anhang I.2). Für jeden der 20 ASR-Texte wurde eine Ausgabe mit dem hier erstellten Werkzeug erstellt. Dabei wurde in den Konfigurationsmöglichkeiten des Werkzeuges die Kombination aus *SENNA* und dem *Stanford POS Tager* zur Wortartenerkennung verwendet. Die Ausgaben sind im Anhang unter I.7 zu finden.

### 6.3. Evaluationsergebnisse

Bei der Berechnung der Genauigkeit ergaben sich je Evaluationsvariante genau 414 zu klassifizierende Token. Das Evaluationsergebnis ist in Tabelle 6.3 zu finden. Dabei gibt die Tabelle für jede Evaluationsvariante die absolute Anzahl der inkorrekten Token hinsichtlich der Wortartenmarkierung, Phrasenerkennung und Zuweisung von Befehlsnummern zuzüglich der Genauigkeiten an.

Variante	Wortarten		phasale Markierung		Befehlsnummer	
	inkorrekt	Genauigkeit	inkorrekt	Genauigkeit	inkorrekt	Genauigkeit
1	26	0,9372	37	0,9106	20	0,9517
2	18	0,9565	29	0,9300	4	0,9903
3	3	0,9928	42	0,8986	1	0,9976
4	119	0,7126	120	0,7101	130	0,6860

Tabelle 6.3.: Evaluationsergebnisse

Bevor die Ergebnisse der Evaluation diskutiert werden, soll die Allgemeingültigkeit der hier durchgeführten Evaluation betrachtet werden: Sowohl die Transkriptionen als auch die ASR-Texte wurden aus den Sprachaufnahmen erstellt, die Anweisungen an den humanoiden Haushaltsroboter ARMAR-III darstellen. Somit beinhalten die Texte ausschließlich Anweisungen, die bezüglich der Interaktion mit ARMAR-III zu erwarten sind. Zusätzlich stellen die 414 zu klassifizierenden Wörter pro Evaluationsvariante, die meist eine imperative Satzstruktur aufweisen und sich untereinander sehr ähneln, einen verhältnismäßig kleinen Evaluationskorpus dar. Nichtsdestotrotz genügt die hier durchgeführte Evaluation aus, um den Ansatz, Standard-Sprachverarbeitungswerkzeuge für die Informationsextraktion im Rahmen des Gesamtprojektes „PARSE“, zu bewerten. Im Folgenden soll das Evaluationsergebnis aus Tabelle 6.3 hinsichtlich der Wortartenerkennung, Phrasenerkennung und Zuweisung von Befehlsnummern separat betrachtet werden.

### 6.3.1. Wortartenerkennung

In diesem Abschnitt sollen die Evaluationsergebnisse hinsichtlich der Wortartenerkennung diskutiert werden. Die Ergebnisse der Wortartenerkennung sind für diese Diskussion in Tabelle 6.4 dargestellt. Dabei stellt die Tabelle einen Auszug aus Tabelle 6.3 dar und gibt die absolute Anzahl der inkorrekten Token zuzüglich der Genauigkeiten für die Wortartenmarkierung an.

Variante	Wortarten	
	inkorrekt	Genauigkeit
1	26	0,9372
2	18	0,9565
3	3	0,9928
4	119	0,7126

Tabelle 6.4.: Evaluationsergebnisse hinsichtlich der Wortartenerkennung

Die erste Variante konnte mit insgesamt 26 inkorrekt klassifizierten Wortarten eine Genauigkeit von 93,72 Prozent erzielen. Hierbei wurde lediglich *SENNA* zur Wortartenerkennung verwendet. Bei sechs der insgesamt 26 fehlerhaften Klassifikationen handelt es sich um nicht erkannte Verben. Zusätzlich wurde das Wort „*please*“ acht Mal fehlerhaft als Verb klassifiziert, obwohl es sich in der Musterlösung in allen Fällen um ein Adverb gehandelt hat. Die zweite Variante konnte bei 18 inkorrekten Wortarten eine Genauigkeit von 95,65 Prozent erreichen. Durch den Vergleich der ersten beiden Varianten hinsichtlich der Verberkennung konnte festgestellt werden, dass die zweite Variante die sechs Verben, welche in der ersten Variante nicht identifiziert wurden, erkennen konnte. Dementsprechend konnte die Evaluation, die durch die Analyse geschlussfolgerte Vermutung bestätigen, dass die Verwendung des *Stanford POS Taggers* zu einer besseren Identifizierung von Verben führt. Jedoch wurde auch in der zweiten Variante das Wort „*please*“ acht Mal inkorrekt als Verb klassifiziert. Um diesen Fehler zu beseitigen, wurde in der dritten Variante ein entsprechender Eintrag in die Liste hinzugefügt, damit das Wort „*please*“ als Adverb klassifiziert wird. Die dritte Variante konnte dementsprechend alle Fehler hinsichtlich der Verberkennung beseitigen und mit drei inkorrekt klassifizierten Wörtern und einer Genauigkeit von 99,28 Prozent das beste Evaluationsergebnis hinsichtlich der Wortartenerkennung erzielen. Zuletzt soll die vierte Evaluationsvariante betrachtet werden. Mit 119 inkorrekten Klassifikationen und einer Genauigkeit von 71,26 Prozent ist das Klassifikationsergebnis in dieser Variante am schlechtesten. Diese Evaluationsvariante unterscheidet sich von den ersten drei Varianten dadurch, dass die ASR-Texte des Evaluationskorpus betrachtet wurden. Im Unterabschnitt 6.1.2 wurde beschrieben, dass bei dieser Variante für jedes Token der Musterlösung nach einem entsprechenden Token im ASR-Text gesucht wurde. Falls kein Token gefunden werden konnte, wurde ein Fehler unter anderem für die Wortartenerkennung eingerechnet. Bei der hier durchgeführten Evaluation sind durch das Fehlen von Token in den ASR-Texten insgesamt 80 Fehler entstanden. Dies entspricht um die zwei Drittel aller aufgetretenen Fehler hinsichtlich der Wortartenerkennung. Ignoriert man diese 80 inkorrekten Wortarten, weil die Klassifikation des Werkzeuges für diese Wörter nicht betrachtet werden konnte, reduziert sich die absolute Anzahl an inkorrekten Klassifikationen auf 39 und die Genauigkeit steigt bei nur noch 334 zu klassifizierenden Wörtern auf 88, 32 Prozent. Wie die in dieser Evaluation erzielten Genauigkeitswerte einzuordnen sind, wird im Unterabschnitt 6.3.4 mithilfe von Referenzwerten diskutiert.

### 6.3.2. Befehlsnummern

In diesem Abschnitt sollen die Evaluationsergebnisse hinsichtlich der Zuweisung von Befehlsnummern diskutiert werden. Dazu sind die Ergebnisse in Tabelle 6.5 dargestellt. Diese

Tabelle stellt einen Auszug aus Tabelle 6.3 dar und gibt die absolute Anzahl der inkorrekten Token zuzüglich der Genauigkeit an.

Variante	Befehlsnummern	
	inkorrekt	Genauigkeit
1	20	0,9517
2	4	0,9903
3	1	0,9976
4	130	0,6860

Tabelle 6.5.: Evaluationsergebnisse hinsichtlich der Zuweisung von Befehlsnummern

Die im vorherigen Unterabschnitt 6.3.1 diskutierten Erkenntnisse zur Verbklassifikation haben direkte Auswirkungen auf die Evaluationsergebnisse hinsichtlich der Zuweisung von Befehlsnummern, da die Heuristik (siehe Abschnitt 5.3) unter anderem die identifizierten Verben als Entscheidungskriterium verwendet. Bei den ersten drei Varianten konnten alle inkorrekten Befehlsnummern auf die fehlerhaft klassifizierten Verben zurückgeführt werden: Die erste Variante konnte mit 20 inkorrekt zugewiesenen Befehlsnummern eine Genauigkeit von 95,17 Prozent erzielen. Wie im Unterabschnitt 6.3.1 beschrieben, konnte die erste Variante hinsichtlich der Verbklassifikation sechs Verben nicht identifizieren und das Wort „*please*“ wurde acht Mal als Verb klassifiziert, obwohl es sich in allen Fällen um ein Adverb gehandelt hat. Aus allen sechs nicht identifizierten Verben resultierte jeweils eine inkorrekte Befehlsnummer. Bei der Betrachtung des Wortes „*please*“ konnte festgestellt werden, dass in lediglich drei der acht fehlerhaften Verbklassifikationen eine inkorrekte Befehlsnummer resultierte. Bei den verbleibenden fünf Verbklassifikationen kam es zu keinem Klassifikationsfehler, weil ein korrekt identifiziertes Verb auf das Wort „*please*“ folgte (z.B. „*please open*“). Da die Heuristik zwei aufeinanderfolgende Verben in einem Befehl zusammenfasst (siehe Abschnitt 5.3), konnten fehlerhafte Befehlsnummern vermieden werden. Die restlichen elf Fehler der ersten Variante stellen Folgefehler der inkorrekten Verbklassifikation dar. Die zweite Variante konnte mit vier inkorrekt zugewiesenen Befehlsnummern eine Genauigkeit von 99,03 Prozent erzielen. Wie im Unterabschnitt 6.3.1 beschrieben, tritt in dieser Evaluationsvariante hinsichtlich der Verbklassifikation nur die fehlerhafte Klassifikation des Wortes „*please*“ auf. Analog zur ersten Evaluationsvariante resultierten hierdurch drei inkorrekte Befehlsnummern. Der vierte Fehler entstand durch eine Anweisung im siebten Text des Evaluationskorpus („*it's next to the cereal box*“), welche nicht der imperativen Satzstruktur entspricht. Dieser Fehler entstand folglich auch in der dritten Evaluationsvariante und sorgt damit für eine Genauigkeit von 99,76 Prozent.

Auch in der Zuweisung der Befehlsnummern fällt das Evaluationsergebnis in der vierten Variante deutlich ab. Aufgrund der 80 Token der Musterlösung, die kein Token in den ASR-Texten besitzen, wurden 130 Befehlsnummern inkorrekt zugewiesen und ergeben eine Genauigkeit von 68,60 Prozent. Ignoriert man erneut die 80 inkorrekten Token, reduziert sich die absolute Anzahl der inkorrekten Befehlsnummern auf 50 und die Genauigkeit steigt auf 85,32 Prozent. Wie die in dieser Evaluation erzielten Genauigkeitswerte einzuordnen sind, wird ebenfalls im Unterabschnitt 6.3.4 mithilfe von Referenzwerten diskutiert.

### 6.3.3. Phrasenerkennung

In diesem Abschnitt sollen die Evaluationsergebnisse hinsichtlich der Phrasenerkennung diskutiert werden. Dazu sind die Ergebnisse in Tabelle 6.6 dargestellt.

Grundsätzlich resultieren die inkorrekten phrasalen Markierungen aus den Fehlern der Wortartenerkennung. Die erste Evaluationsvariante konnte mit 37 inkorrekten phrasalen Markierungen eine Genauigkeit von 91,06 Prozent erzielen. Im Vergleich dazu konnte die

Variante	phrasale Markierung	
	inkorrekt	Genauigkeit
1	37	0,9106
2	29	0,9300
3	42	0,8986
4	120	0,7101

Tabelle 6.6.: Evaluationsergebnisse hinsichtlich der Phrasenerkennung

zweite Evaluationsvariante aufgrund der besseren Wortartenerkennung mit 29 inkorrekten phrasalen Markierungen und einer Genauigkeit von 93 Prozent ein besseres Ergebnis erzielen. Die dritte Variante besitzt mit insgesamt 42 inkorrekten phrasalen Markierungen, trotz der besten Wortartenmarkierung, das schlechteste Ergebnis der ersten drei Evaluationsvarianten. Dieses schlechte Ergebnis konnte auf eine Besonderheit zurückgeführt werden: Im Evaluationskorpus folgt häufig auf das Adverb „*please*“ ein anschließendes Verb. Diese Wortsequenz „*please*+*Verb*“ wurde innerhalb der Musterlösung als eine Verbphrase klassifiziert. Tabelle 6.7 verdeutlicht dies anhand eines Auszugs der Musterlösung für den Text „*please open the fridge*“. Hierbei wird die Wortsequenz „*please open*“ als Verbphrase vorgegeben.

Token	Wortart	phrasale Markierung
Please	RB	B-VP
open	VB	I-VP
the	DT	B-NP
fridge	NN	I-NP

Tabelle 6.7.: Musterlösung für *please open the fridge* (Phrasenerkennung)

Bei der dritten Evaluationsvariante klassifiziert das Werkzeug jedoch die Wortsequenz „*please*+*Verb*“ als unabhängige Phrasen. Dabei wird „*please*“ als Adverbphrase und das anschließende Verb als Verbphrase klassifiziert. Für das obige Beispiel wurde die Ausgabe des Werkzeuges hinsichtlich der dritten Evaluationsvariante in Tabelle 6.8 bereitgestellt.

Token	Wortart	phrasale Markierung
Please	RB	B-ADVP
open	VB	B-VP
the	DT	B-NP
fridge	NN	I-NP

Tabelle 6.8.: Ausgabe für *please open the fridge* (3. Variante)

Der hier beschriebene Fehler tritt jedoch nicht bei den ersten beiden Evaluationsvarianten auf, weil hier das Wort „*please*“ inkorrekt als Verb klassifiziert wurde. Durch die inkorrekte Klassifikation von „*please*“ innerhalb der Wortartenerkennung, klassifiziert der Phrasenerkennung *BIOS* die Wortsequenz „*please+Verb*“ korrekt als zusammengehörige Verbphrase. Die Ausgabe des Werkzeuges für die beiden ersten Evaluationsvarianten, hinsichtlich des Beispieltextes „*please open the fridge*“, ist in Tabelle 6.9 dargestellt.

Die Entscheidung, ob die Wortsequenz „*please*+*Verb*“ eine einzige Verbphrase oder separate Adverb- und Verbphrasen darstellen, hängt vom Ersteller der Musterlösung ab. Objektiv sind beide Annotationsmöglichkeiten korrekt. Deshalb wurden die Evaluationsergebnisse aus Tabelle 6.6 neu aufgearbeitet. In den aufgearbeiteten Evaluationsergebnissen unter Tabelle 6.10 wurden im Fall der Wortsequenz „*please*+*Verb*“ sowohl eine einzige Verbphrase,

Token	Wortart	phrasale Markierung
Please	VB	B-VP
open	VB	I-VP
the	DT	B-NP
fridge	NN	I-NP

Tabelle 6.9.: Ausgabe für *please open the fridge* (1. und 2.Variante)

als auch separate Adverb- und Verbphrasen hinsichtlich der Phrasenerkennung als korrekte Klassifikationen angesehen. In diesem aufbereiteten Evaluationsergebnis haben sich ausschließlich die Werte für die dritte Evaluationsvariante verändert. Durch diese Veränderung konnte die dritte Variante mit 22 inkorrekten phrasalen Markierungen und einer Genauigkeit von 94,69 Prozent das beste Klassifikationsergebnis für die Phrasenerkennung erzielen.

Variante	phrasale Markierung	
	inkorrekt	Genauigkeit
1	37	0,9106
2	29	0,9300
3	22	0,9469
4	120	0,7101

Tabelle 6.10.: Evaluationsergebnisse hinsichtlich der Phrasenerkennung

Auch in der Phrasenerkennung fällt das Evaluationsergebnis in der vierten Variante deutlich ab. Aufgrund der 80 Token der Musterlösung die kein Token in den ASR-Texten besitzen, wurden 120 phrasale Markierungen inkorrekt zugewiesen und ergeben eine Genauigkeit von 71,01 Prozent. Ignoriert man die 80 inkorrekten Token reduziert sich die absolute Anzahl der inkorrekten phrasalen Markierungen auf 40 und die Genauigkeit steigt auf 88,02 Prozent.

### 6.3.4. Bewertung

Um die in der Evaluation errechneten Genauigkeitswerte einordnen zu können, sollen die Evaluationsergebnisse mit Referenzwerten verglichen werden: Bei aktuellen Klassifikatoren liegt die Genauigkeit der Wortartenerkennung bei um die 96-97 Prozent [DJH09, S.189]. Ein Wert um die 94 Prozent [EFB00] wird bei aktuellen Phrasenerkennern angegeben. Hierbei muss jedoch berücksichtigt werden, dass sich diese Werte auf Klassifikatoren beziehen, die geschriebene Sprache klassifizieren und vor der Berechnung der Genauigkeit eine Trainingsphase auf geschriebener Sprache durchlaufen haben. Unter [NGG<sup>+</sup>96] konnte ein Wortartenerkennner für die schwedische Sprache gefunden werden, der auf geschriebener Sprache trainiert und anschließend zur Klassifikation von gesprochener Sprache verwendet wurde. Die Genauigkeit dieses Wortartenerkenners liegt bei um die 85 Prozent. Das hier erstellte Werkzeug sollte mindestens in der Konfiguration verwendet werden, die zur Wortartenerkennung die Kombination von *SENNA* und dem *Stanford POS Tagger* verwendet (siehe Abschnitt 5.5). Diese Konfiguration wurde in der zweiten und vierten Evaluationsvariante verwendet. Die zweite Evaluationsvariante konnte mit 95,65 Prozent in der Wortartenerkennung (siehe Tabelle 6.5) und 94,69 Prozent in der Phrasenerkennung (siehe Tabelle 6.10) ein annähernd gutes Klassifikationsergebnis erzielen, wie aktuelle Spracherkennner die auf geschriebener Sprache evaluiert wurden. Außerdem konnte mit der Heuristik zur Zuweisung von Befehlsnummern ein hervorragendes Ergebnis von 99,03 Prozent (siehe Tabelle 6.5) erreicht werden. Die vierte Evaluationsvariante konnte hingegen mit

Genauigkeiten von 71,26 Prozent in der Wortartenerkennung, 71,01 Prozent in der Phrasenerkennung und 68,6 Prozent in der Zuweisung von Befehlsnummern ein vergleichsweise schlechtes Evaluationsergebnis erzielen. Jedoch sind um die zwei drittel aller Klassifikationsfehler dadurch entstanden, dass der automatische Spracherkenner die einzelnen Wörter der Sprachaufnahmen nicht korrekt erkennen konnte. Aus dieser Feststellung resultiert der Ausblick, dass hinsichtlich des Gesamtprojektes „*PARSE*“ die Auswahl des automatischen Spracherkenners wesentlich ist, um mit dem hier erstellten Werkzeug Genauigkeiten erhalten zu können, die mit der zweiten Evaluationsvariante vergleichbar sind.

## 7. Zusammenfassung und Ausblick

In dieser Bachelorarbeit wurde ein Werkzeug implementiert, das einen Text auf eine initiale Handlungsrepräsentation für natürliche Sprache abbildet. Bei den Eingabetexten handelt es sich um gesprochene Anweisungen an den humanoiden Haushaltsroboter ARMAR-III, die mithilfe eines automatischen Spracherkenners in Text überführt werden. Die damit verbundenen Herausforderungen waren, dass die Ausgabe des automatischen Spracherkenners keine Punktierung, Wortfehler, Wiederholungen und Füllwörter enthalten. Auf eine Behandlung der Wortfehler wurde in dem hier erstellten Werkzeug verzichtet, da die Behebung von Wortfehlern eine domänenabhängige Aufgabe darstellt und das hier erstellte Werkzeug als zusätzliche Anforderung möglichst unabhängig von der Domäne, ARMAR-III neue Anweisungen beizubringen, konstruiert werden sollte. Bei der Betrachtung von Gesamtsystemen (siehe Abschnitt 3.2), die ähnliche Problemstellungen adressieren, konnte festgestellt werden, dass die größte Herausforderung dieser Bachelorarbeit das Extrahieren von Informationen aus einem Eingabetext ohne Satztrennungszeichen darstellt. Die meisten betrachteten Gesamtsysteme umgehen diese Herausforderung, indem sie Satztrennungszeichen für die Informationsextraktion voraussetzen. Das hier erstellte Werkzeug adressiert hingegen gezielt das Problem der fehlenden Satztrennungszeichen. Dazu wurde der Ansatz, aktuelle Standard-Sprachverarbeitungswerkzeuge, die auf geschriebener Sprache optimiert wurden, zu verwenden, um Informationen aus dem Eingabetext zu extrahieren, analysiert. Diese extrahierten Informationen können anschließend in der initialen Handlungsrepräsentation dargestellt werden. Um diesen Ansatz zu analysieren, wurden insgesamt elf Sprachverarbeitungswerkzeuge (siehe Abschnitt 3.1) betrachtet. Dazu wurden die einzelnen Sprachverarbeitungswerkzeuge hinsichtlich der Wortartenerkennung, Phrasenerkennung, Syntaxanalyse und der Zuweisung von semantischen Rollen mit einem 468 Wörter umfassenden Korpus (siehe Abschnitt 4.1) analysiert. Durch die Analyse konnte festgestellt werden, dass Standard-Sprachverarbeitungswerkzeuge für die Wortartenerkennung und Phrasenerkennung verwertbare Informationen extrahieren können. Zusätzlich führte diese Analyse zur Erkenntnis, dass Standard-Sprachverarbeitungswerkzeuge ungeeignet sind, um syntaktische und semantische Informationen aus einem Eingabetext ohne Satztrennungszeichen zu extrahieren. Neben dieser durchgeführten Analyse wurde eine Heuristik entworfen, die aus einem Eingabetext ohne Satztrennungszeichen, der aus einer Folge von Befehlen in imperativer Satzstruktur besteht, einzelne Befehle erkennt (siehe Abschnitt 5.3). Somit stellt das hier erstellte Werkzeug für jedes Wort des Eingabetextes die Wortart, phrasale Markierung und Befehlsnummer dar. Zur Wortartenerkennung werden die Klassifikatoren *SENNA* und der *Stanford POS Tagger* verwendet. Dabei werden die Wortartenmarker folgendermaßen kombiniert: Das in dieser Bachelorarbeit

erstellte Werkzeug klassifiziert ein Wort als Verb, falls *SENNA* oder der *Stanford POS Tagger* das Wort als Verb klassifiziert haben. In allen anderen Fällen wird das Klassifikationsergebnis von *SENNA* für die Wortartenerkennung verwendet. Zur Phrasenerkennung wird anschließend das Sprachverarbeitungswerkzeug *BIOS* verwendet, da *BIOS* das beste Klassifikationsergebnis erzielen konnte (siehe Abschnitt 4.3). In der Evaluation konnte die Konfiguration des Werkzeuges, die zur Wortartenerkennung *SENNA* und den *Stanford POS Tagger* verwendet, für die händischen Transkriptionen des Evaluationskorpus eine Genauigkeit von 95,65 Prozent in der Wortartenerkennung, 94,69 Prozent in der Phrasenerkennung und 99,03 Prozent in der Zuweisung von Befehlsnummern erzielen (siehe Unterabschnitt 6.3.4). Jedoch konnte die Evaluation auch zeigen, dass die Auswahl des automatischen Spracherkenners entscheidend ist und das Auftreten von Wortfehler minimiert werden muss, um diese Genauigkeitswerte annähernd erzielen zu können. Deshalb sollte sich eine weiterführende Arbeit mit der Auswahl des automatischen Spracherkenners für das Gesamtprojekt „*PARSE*“ beschäftigen. Um die Anzahl der Wortfehler zu minimieren, sollte eine entsprechende Trainingsphase des automatischen Spracherkenners in Betracht gezogen werden. Hierbei könnten die Sprachaufnahmen und zugehörigen Transkriptionen der Bachelorarbeit unter [Gue15] verwendet werden, um beispielsweise durch überwachte Lernenverfahren (engl. Supervised Learning) die Wortfehlerrate des automatischen Spracherkenners zu minimieren.

In der Analyse in Kapitel 4 wurde festgestellt, dass Sprachverarbeitungswerkzeuge, die auf geschriebener Sprache optimiert wurden, ungeeignet sind, um syntaktische und semantische Informationen aus einem Eingabetext ohne Satztrennungszeichen zu extrahieren. Dabei handelt es sich um Eingabetexte, die aus mehreren Befehlen bestehen. Da das hier erstellte Werkzeug jedoch einzelne Befehle mit einer Genauigkeit von 99,03 Prozent (siehe Unterabschnitt 6.3.2) erkennt, können eventuell doch syntaktische und semantische Informationen mithilfe von Standard-Sprachverarbeitungswerkzeugen, die auf geschriebener Sprache optimiert wurden, extrahiert werden. Dafür muss lediglich ein Befehl nach dem anderen von den Sprachverarbeitungswerkzeugen analysiert werden. Zur Verdeutlichung soll die Ausgabe des hier erstellten Werkzeuges für den Eingabetext „*go to the fridge open it grab the orange juice*“ hinsichtlich der Befehlsnummern betrachtet werden:

Wörter	go	to	the	fridge	open	it	grab	the	orange	juice
Befehlsnummer	0	0	0	0	1	1	2	2	2	2

In der Analyse in Kapitel 4 wurde immer die gesamte Folge an Befehlen als Eingabe für die Sprachverarbeitungswerkzeuge verwendet. In diesem Beispiel also der Eingabetext „*go to the fridge open it grab the orange juice*“. Nun kann jedoch versucht werden ausschließlich den Befehl „*go to the fridge*“ als Eingabe für ein Sprachverarbeitungswerkzeug zu verwenden, welches syntaktische oder semantische Informationen extrahiert. Da der Befehl in diesem Beispiel genau einem Satz entspricht, sollten verwertbare Informationen erwartet werden können, obwohl die Sprachverarbeitungswerkzeuge auf geschriebener Sprach optimiert wurden. Analog können die Befehle „*open it*“ und „*grab the orange juice*“ untersucht werden. Durch dieses Vorgehen könnte das Problem der fehlenden Satztrennungszeichen hinsichtlich der Syntaxanalyse und der Zuweisung von semantischen Rollen für das Gesamtprojekt „*PARSE*“ gelöst werden. Eine weiterführende Arbeit könnte diese Vermutung untersuchen.

Als weiterführende Arbeiten sind alle Aufgaben zu nennen, die im Gesamtprojekt „*PARSE*“ bewerkstelligt werden müssen, um ARMAR-III neue Anweisungen beizubringen. Dabei könnte sich ein Agent um die Entfernung von Füllwörtern und semantisch irrelevanten Begriffen kümmern. Zusätzlich könnte ein Agent die Koreferenzanalyse durchführen und Informationen, wie Synonyme und Lemmata mithilfe einer Datenbank (z.B. **WordNet**

[Mil95]) in die Handlungsrepräsentation mit aufnehmen. Ebenfalls stellt die Abbildung der Handlungsrepräsentation auf den von ARMAR-III ausführbaren Programmiercode eine zu lösende Herausforderung dar. Auch wenn es sich hier nur um eine Teilmenge der noch zu bewerkstellenden Aufgaben handelt, ist zu erkennen, dass viele einzelne Komponenten zusammengeführt werden müssen, um ARMAR-III neue Anweisungen beizubringen. Somit stellt das hier erstellte Werkzeug nicht mehr als eine einzelne Komponente dar, die zur Erfüllung dieser Aufgabe beiträgt.



# Literaturverzeichnis

- [AA10] ALESSANDRA, Giordani ; ALESSANDRO, Moschitti: Semantic Mapping Between Natural Language Questions and SQL Queries via Syntactic Pairing. In: *Natural Language Processing and Information Systems* Bd. 5723, 2010, S. 207–221
- [AMOH03] ANA-MARIA, Popescu ; OREN, Etzioni ; HENRY, Kautz: Towards a theory of natural language interfaces to databases. In: *IUI '03 Proceedings of the 8th international conference on Intelligent user interfaces*, 2003, S. 149–157
- [Apa] *openNLP*. <http://opennlp.apache.org/>
- [BHH<sup>+</sup>03] BABAK, Hodjat ; HORACIO, Franco ; HARRY, Bratt ; KRISTIN, Precoda ; ANDREAS, Stolcke ; ANAND, Venkataraman ; DIMITRA, Vergyri ; JING, Zheng: *Iterative Statistical Language Model Generation for Use with an Agent-Oriented Natural Language Interface*. 2003
- [Bra00] BRANTS, Thorsten: TnT: a statistical part-of-speech tagger. In: *ANLC '00 Proceedings of the sixth conference on Applied natural language processing*, 2000, S. 224–231
- [CFCJL98] COLLIN F., Baker ; CHARLES J., Fillmore ; LOWE, John B.: The Berkeley FrameNet Project. In: *COLING '98 Proceedings of the 17th international conference on Computational linguistics* Bd. 1, 1998, S. 86–90
- [Cha00] CHARNIAK: A Maximum-Entropy-Inspired Parser. In: *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, 2000, S. 132–139
- [CMR09] COPPOLA, Bonaventura ; MOSCHITTI, Alessandro ; RICCARDI, Giuseppe: Shallow Semantic Parsing for Spoken Language Understanding. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, Association for Computational Linguistics, 2009, 85–88
- [Col99] COLLINS, Michael: *Head-Driven Statistical Models for Natural Language Parsing*, University of Pennsylvania, Diss., 1999
- [DCD03] DAN, Klein ; CHRISTOPHER D., Manning: Accurate Unlexicalized Parsing. In: *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, 2003, S. 423–430
- [DJH09] DANIEL, Jurafsky ; JAMES H., Martin: *Speech and Language Processing*. Pearson, 2009
- [EFB00] ERIK F., Tjong Kim S. ; BUCHHOLZ, Sabine: Introduction to the CoNLL-2000 Shared Task: Chunking. In: *Proceedings of the 2Nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural*

- Language Learning - Volume 7*. Stroudsburg, PA, USA : Association for Computational Linguistics, 2000 (ConLL '00), 127-132
- [EFT02] ERIK F. TJONG, Kim S.: Introduction to the CoNLL-2002 shared task: language-independent named entity recognition. In: *COLING-02 proceedings of the 6th conference on Natural language learning* Bd. 22, 2002, S. 1–4
- [Gue15] GUENES, Zeynep: *Aufbau eines Sprachkorpus zur Programmierung autonomer Roboter mittels natürlicher Sprache*, Karlsruher Institut für Technologie (KIT) – IPD Tichy, Bachelor's Thesis, Mai 2015. [https://code.ipd.kit.edu/weigelt/parse/wikis/Theses/guenes\\_ba](https://code.ipd.kit.edu/weigelt/parse/wikis/Theses/guenes_ba)
- [HH05] HUGO, Liu ; HENRY, Lieberman: Metafor: Visualizing stories as code. In: *10th International Conference on Intelligent User Interfaces*, ACM Press, 2005, S. 305–307
- [III] *Cognitive Computation Group-Software Packages*. <http://cogcomp.cs.illinois.edu/page/software/>
- [JD03] JINHO D., Choi: *Optimization of Natural Language Processing Components for Robustness and Scalability*, University of Pennsylvania, Diss., 2003
- [KDCDY03] KRISTINA, Toutanova ; DAN, Klein ; CHRISTOPHER D., Manning ; YORAM, Singer: Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In: *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*. Stroudsburg, PA, USA : Association for Computational Linguistics, 2003 (NAACL '03), 173-180
- [Kor] *Koreferenz*. <https://de.wikipedia.org/wiki/Koreferenz>
- [KUCC<sup>+</sup>10] KAI-UWE, Carstensen ; CHRISTIAN, Ebert ; CORNELIA, Ebert ; SUSANNE, Jekat ; RALF, Klabunde ; HAGEN, Langer: *Computerlinguistik und Sprachtechnologie*. Spektrum, 2010
- [LS04] LIU, H. ; SINGH, P.: ConceptNet - A Practical Commonsense Reasoning Tool-Kit. In: *BT Technology Journal* 22 (2004), Oktober, Nr. 4, S. 211–226. – ISSN 1358–3948
- [MA03] MEI, Eisenbach ; ANDRE, Eisenbach: *phpSyntaxTree - drawing syntax trees made easy*. <http://ironcreek.net/phpsyntaxtree/?PHPSESSID=ui5d0d3pabcbijjcvn9714nc5>. Version: 2003
- [MAB03] MITCHELL, Marcus ; ANN, Taylo ; BEATRICE, Santorini: *THE PENN TREE-BANK: AN OVERVIEW*. 2003
- [MBS] *Penn Treebank II tag set*. <http://www.clips.ua.ac.be/pages/mbsp-tags>
- [MCBCD06] MARIE-CATHERINE, de M. ; BILL, MacCartney ; CHRISTOPHER D., Manning: *Generating Typed Dependency Parses from Phrase Structure Parses*. 2006
- [MDK05] MARTHA, Palmer ; DANIEL, Gildea ; KINGSBURY, Paul: The Proposition Bank: An Annotated Corpus of Semantic Roles. In: *Computational Linguistics* Bd. 31, 2005, S. 71–106
- [Mil95] MILLER, George A.: WordNet: a lexical database for English. In: *Communications of the ACM* 38 (1995), nov., S. 39–41

- [MSB<sup>+</sup>14] MANNING, Christopher D. ; SURDEANU, Mihai ; BAUER, John ; FINKEL, Jenny ; BETHARD, Steven J. ; MCCLOSKEY, David: The Stanford CoreNLP Natural Language Processing Toolkit. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2014, 55-60
- [MSK] MARK SLEE, Aditya A. ; KWIATKOWSKI, Marc: Thrift: Scalable Cross-Language Services Implementation
- [NGG<sup>+</sup>96] NIVRE, Joakim ; GRÖNQVIST, Leif ; GUSTAFSSON, Malin ; LAGER, Torbjörn ; SOFKOVA, Sylvana: Tagging Spoken Language Using Written Language Statistics. In: *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*. Stroudsburg, PA, USA : Association for Computational Linguistics, 1996 (COLING '96), 1078-1081
- [Ope] *OpenNLP - Models for 1.5 series*. <http://opennlp.sourceforge.net/models-1.5/>
- [Pro] *PropBank FrameSets*. <http://verbs.colorado.edu/framesets/>
- [RJL<sup>+</sup>11] RONAN, Collobert ; JASON, Weston ; LEON, Bottou ; MICHAEL, Karlen ; KORAY, Kavukcuoglu ; PAVEL, Kuksa: Natural Language Processing (Almost) from Scratch. In: *J. Mach. Learn. Res.* 12 (2011), November, 2493–2537. <http://dl.acm.org/citation.cfm?id=1953048.2078186>. – ISSN 1532–4435
- [RM06] ROMAN, Knoell ; MIRA, Mezini: Pegasus: first steps toward a naturalistic programming language. In: *OOPSLA '06 Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications*, 2006, S. 542–559
- [Sch15] SCHNEIDER, Michael: *Entwurf einer Handlungsrepräsentation für gesprochene Sprache*, Karlsruher Institut für Technologie (KIT) – IPD Tichy, Bachelor's Thesis, Mai 2015. [https://code.ipd.kit.edu/weigelt/parse/wikis/Theses/schneider\\_ba](https://code.ipd.kit.edu/weigelt/parse/wikis/Theses/schneider_ba)
- [SLRD06] SLAV, Petrov ; LEON, Barrett ; ROMAIN, Thibaux ; DAN, Klein: Learning Accurate, Compact, and Interpretable Tree Annotation. In: *ACL-44 Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, 2006, S. 433–440
- [Sta] *The Stanford Natural Language Processing Group*. <http://nlp.stanford.edu/software/index.shtml>
- [Sur] SURDEANU, Mihai: *BIOS - A suite of syntactico-semantic analyzers for English*. <http://www.talp.upc.edu/index.php/technology/tools/textual-processing-tools/151-bios>
- [Tae04] TAENTZER, Gabriele: AGG: A Graph Transformation Environment for Modeling and Validation of Software. In: PFALTZ, JohnL. (Hrsg.) ; NAGL, Manfred (Hrsg.) ; BÖHLEN, Boris (Hrsg.): *Applications of Graph Transformations with Industrial Relevance* Bd. 3062, Springer Berlin Heidelberg, 2004 (Lecture Notes in Computer Science). – ISBN 978–3–540–22120–3, 446-453
- [Tam06] TAMIM, Asfour: Humanoid Robots, 2006 6th IEEE-RAS International Conference on. In: *Humanoid Robots, 2006 6th IEEE-RAS International Conference*, 2006

- [TLHW13] THANH-LE HA, Jan Niehues Mohammed Mediani Eunah Cho Yuqi Zhang Isabel S. Teresa Herrmann H. Teresa Herrmann ; WAIBEL, Alex: The KIT Translation Systems for IWSLT 2013. In: *IWSLT 2013* (2013)
- [TVW05] TOM, de S. ; VINCENT, Van A. ; WALTER, Daelemans: Clips - Memory-based Shallow Parser for Python. In: *CLiPS Technical Report Series (CTRS)* Bd. 2, Cambridge University Press, 2005
- [Yan08] YANG, Jun: *Review of Multi-Instance Learning and Its applications*. <http://www.cs.cmu.edu/~juny/MILL/>. Version: 2008

# Anhang

## A. Penn Treebank part-of-speech tags

Tag	Description
CC	Coordinating conjunction
CD	Cardinal number
DT	Determiner
EX	Existential <i>there</i>
FW	Foreign word
IN	Preposition or subordinating conjunction
JJ	Adjective
JJR	Adjective, comparative
JJS	Adjective, superlative
LS	List item marker
MD	Modal
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
NNPS	Proper noun, plural
PDT	Predeterminer
POS	Possessive ending
PRP	Personal pronoun
PRP\$	Possessive pronoun
RB	Adverb
RBR	Adverb, comparative
RBS	Adverb, superlative
RP	Particle
SYM	Symbol
TO	<i>to</i>
UH	Interjection
VB	Verb, base form
VBD	Verb, past tense
VBG	Verb, gerund or present participle
VBN	Verb, past participle
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present
WDT	Wh-determiner
WP	Wh-pronoun
WP\$	Possessive wh-pronoun
WRB	Wh-adverb

## B. Chunk tags

Tag	Description
NP	Noun phrase
PP	Prepositional phrase
VP	Verb phrase
ADVP	Adverb phrase
ADJP	Adjective phrase
SBAR	Subordinating conjunction
PRT	Particle
INTJ	Interjection

## C. Einige grammatikalische Relationen aus [MCBCD06]

Argument Dependencies	Description
nsubj	nominal subject
csubj	clausal subject
doj	direct object
iobj	indirect object
pobj	object of preposition
Modifier Dependencies	Description
tmod	temporal modifier
det	determiner
prep	prepositional modifier

## D. Analysekorpus

Nr. 1:

Mit Satztrennung:

*Go to the dishwasher. Open the dishwasher's door. Locate its upper rack. Pull it out. Go to the kitchen table and locate the green cup. Take it. Go back to the dishwasher. Put the green cup in the rack. Push the rack back and close the dishwasher's door.*

Ohne Satztrennung:

*Go to the dishwasher open the dishwasher's door locate its upper rack pull it out go to the kitchen table and locate the green cup take it go back to the dishwasher put the green cup in the rack push the rack back and close the dishwasher's door*

Nr. 2:

Mit Satztrennung:

*Go to the fridge and open the door. Locate the orange juice and take it. Close the door. Come over here and give the orange juice to me.*

Ohne Satztrennung:

*Go to the fridge and open the door locate the orange juice and take it close the door come over here and give the orange juice to me*

Nr. 3:

Mit Satztrennung:

*Go to the kitchen table. Locate the the popcorn bag. Take it. Come over here. Hand the popcorn bag over.*

Ohne Satztrennung:

*Go to the kitchen table locate the the popcorn bag take it come over here hand the popcorn*

*bag over*

Nr. 4:

Mit Satztrennung:

*Okay ARMAR, go to the kitchen table and take the green cup from there. Look out for the dishwasher. If you can see it, go over there. Open its door. Pull out the top basket. Put the cup into it. Finally, close the dishwasher's door.*

Ohne Satztrennung:

*Okay ARMAR go to the kitchen table and take the green cup from there look out for the dishwasher if you can see it go over there open its door pull out the top basket put the cup into it finally close the dishwasher's door*

Nr. 5:

Mit Satztrennung:

*Okay ARMAR, look for the refrigerator. If you can see it, go over there. Grab the door handle and pull. Look what's inside the fridge. There should be an orange juice somewhere. If you can see it, take it and close the door. Come over here and hand it over.*

Ohne Satztrennung:

*Okay ARMAR look for the refrigerator if you can see it go over there grab the door handle and pull look what's inside the fridge there should be an orange juice somewhere if you can see it take it and close the door come over here and hand it over*

Nr. 6:

Mit Satztrennung:

*At the kitchen table there is a popcorn bag. Take it, come over here and give it to me.*

Ohne Satztrennung:

*At the kitchen table there is a popcorn bag take it come over here and give it to me*

Nr. 7:

Mit Satztrennung:

*Locate the dishwasher. If you can see it, go there. When you're there, open its front door and pull out the upper rack. Go to the kitchen table and see what's there. If you can see a green cup, take it. Move back towards the dishwasher and put the cup into it. Finally, push back the basket and close the door.*

Ohne Satztrennung:

*Locate the dishwasher if you can see it go there when you're there open its front door and pull out the upper rack go to the kitchen table and see what's there if you can see a green cup take it move back towards the dishwasher and put the cup into it finally push back the basket and close the door*

Nr. 8:

Mit Satztrennung:

*Inside the ice box, there is an orange juice. So, go over there, open the door and take it. When you've got it, close the door, carry the orange juice over here and hand it over.*

Ohne Satztrennung:

*Inside the ice box there is an orange juice so go over there open the door and take it when you've got it close the door carry the orange juice over here and hand it over.*

Nr. 9:

Mit Satztrennung:

*Okay ARMAR, look for the table. Go there and take the popcorn. After taking the popcorn, come here and hand it over.*

Ohne Satztrennung:

*Okay ARMAR look for the table go there and take the popcorn after taking the popcorn come here and hand it over*

Nr. 10:

Mit Satztrennung:

*Go to the kitchen table and see if you can find a popcorn bag. If so, take it and bring it to me.*

Ohne Satztrennung:

*Go to the kitchen table and see if you can find a popcorn bag if so take it and bring it to me*

Nr. 11:

Mit Satztrennung:

*Okay Armar, go to the table. Grab popcorn. Come to me. Give me the popcorn which is in your hand.*

Ohne Satztrennung:

*Okay Armar go to the table grab popcorn come to me give me the popcorn which is in your hand*

Nr. 12:

Mit Satztrennung:

*Okay Armar, go to the table. Take the green cup. Go to the dishwasher. Open it. Put the green cup into the dishwasher and close it.*

Ohne Satztrennung:

*Okay Armar go to the table take the green cup go to the dishwasher open it put the green cup into the dishwasher and close it*

Nr. 13:

Mit Satztrennung:

*Okay ARMAR, go to the fridge. Look for the orange juice and take it. Go to the table. Pour the juice into the green cup. Put down the juice and bring me the green cup from the table.*

Ohne Satztrennung:

*Okay ARMAR go to the fridge look for the orange juice and take it go to the table pour the juice into the green cup put down the juice and bring me the green cup from the table*

Nr. 14:

Mit Satztrennung:

*Okay Armar, go to the fridge. Open it. Look for the orange juice and take it. Close the fridge and bring the juice to me.*

Ohne Satztrennung:

*Okay Armar go to the fridge open it look for the orange juice and take it close the fridge and bring the juice to me*

## E. Musterlösungen Wortartenerkennung

Nr. 1:

*Go/VB to/TO the/DT dishwasher/NN. Open/VB the/DT dishwasher/NN 's/POS door/NN . Locate/VB its/PRP\$ upper/JJ rack/NN. Pull/VB it/PRP out/RP. Go/VB to/TO the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN. Take/VB it/PRP. Go/VB back/RB to/TO the/DT dishwasher/NN. Put/VB the/DT green/JJ cup/NN in-/IN the/DT rack/NN. Push/VB the/DT rack/NN back/RB and/CC close/VB the/DT dishwasher/NN 's/POS door/NN.*

Nr. 2:

*Go/VB to/TO the/DT fridge/NN and/CC open/VB the/DT door/NN. Locate/VB the/DT orange/NN juice/NN and/CC take/VB it/PRP. Close/VB the/DT door/NN. Come/VB over/RP here/RB and/CC give/VB the/DT orange/NN juice/NN to/TO me/PRP.*

Nr. 3:

*Go/VB to/TO the/DT kitchen/NN table/NN. Locate/VB the/DT popcorn/NN bag/NN. Take/VB it/PRP. Come/VB over/RP here/RB. Hand/VB the/DT popcorn/NN bag/NN over/RP.*

Nr. 4:

*Okay/UH ARMAR/NNP go/VB to/TO the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/RB. Look/VB out/IN for/IN the/DT dishwasher/NN. If/IN you/PRP can/MD see/VB it/PRP go/VB over/RP there/RB. Open/VB its/PRP\$ door/NN. Pull/VB out/RB the/DT top/JJ basket/NN. Put/VB the/DT cup/NN into/IN it/PRP. Finally/RB close/VB the/DT dishwasher/NN 's/POS door/NN.*

Nr. 5:

*Okay/UH ARMAR/NNP look/VB for/IN the/DT refrigerator/NN. If/IN you/PRP can/MD see/VB it/PRP go/VB over/RP there/RB. Grab/VB the/DT door/NN handle/NN and/C pull/VB. Look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN. There/EX should/MD be/VB an/DT orange/NN juice/NN somewhere/RB. If/IN you/PRP can/MD see/VB it/PRP, take/VB it/PRP and/CC close/VB the/DT door/NN. Come/VB over/RP here/RB and/CC hand/VB it/PRP over/RP.*

Nr. 6:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN. Take/VB it/PRP, come/VB over/RP here/RB and/CC give/VB it/PRP to/TO me/PRP.*

Nr. 7:

*Locate/VB the/DT dishwasher/NN. If/IN you/PRP can/MD see/VB it/PRP, go/VB there/RB. When/WRB you/PRP 're/VBP there/RB, open/VB its/PRP\$ front/NN door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN. Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB. If/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN, take/VB it/PRP. Move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VB the/DT cup/NN into/IN it/PRP. Finally/RB, push/VB back/IN the/DT basket/NN and/CC close/VB the/DT door/NN.*

Nr. 8:

*Inside/PRP the/DT ice/NN box/NN, there/EX is/VBZ an/DT orange/NN juice/NN.*

*So/IN, go/VB over/RP there/RB, open/VB the/DT door/NN and/CC take/VB it/PRP. When/WRB you/PRP 've/VBP got/VBN it/PRP, close/VB the/DT door/NN, carry/VB the/DT orange/NN juice/NN over/RP here/RB and/CC hand/VB It/PRP over/RP.*

Nr. 9:

*Okay/UH ARMAR/NNP, look/VBP for/IN the/DT table/NN. Go/VB there/RB and/CC take/VB the/DT popcorn/NN. After/IN taking/VBG the/DT popcorn/NN, come/VB here/RB and/CC hand/VB it/PRP over/RP.*

Nr. 10:

*Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD find/VB a/DT popcorn/NN bag/NN. If/IN so/IN, take/VB it/PRP and/CC bring/VB it/PRP to/TO me/PRP.*

Nr. 11:

*Okay/UH Armar/NNP, go/VBP to/TO the/DT table/NN. Grab/VB popcorn/NN. Come/VB to/TO me/PRP. Give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ in/IN your/PRP\$ hand/NN.*

Nr. 12:

*Okay/UH Armar/NNP, go/VBP to/TO the/DT table/NN. Take/VB the/DT green/JJ cup/NN. Go/VB to/TO the/DT dishwasher/NN. Open/VB it/PRP. Put/VB the/DT green/JJ cup/NN into/IN the/DT dishwasher/NN and/CC close/VB it/PRP.*

Nr. 13:

*Okay/UH ARMAR/NNP, go/VBP to/TO the/DT fridge/NN. Look/VB for/IN the/DT orange/NN juice/NN and/CC take/VB it/PRP. Go/VB to/TO the/DT table/NN. Pour/VB the/DT juice/NN into/IN the/DT green/JJ cup/NN. Put/VB down/IN the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN.*

Nr. 14:

*Okay/UH Armar/NNP, go/VBP to/TO the/DT fridge/NN. Open/VB it/PRP. Look/VB for/IN the/DT orange/NN juice/NN and/CC take/VB it/PRP. Close/VB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/TO me/PRP.*

## F. Musterlösungen Phrasenerkennung

Nr. 1:

*Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O Locate/B-VP its/B-NP upper/I-NP rack/I-NP ./O Pull/B-VP it/B-NP out/B-PRT ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O locate/B-VP the/B-NP green/I-NP cup/I-NP ./O Take/B-VP it/B-NP ./O Go/B-VP back/B-ADVP to/B-PP the/B-NP dishwasher/I-NP ./O Put/B-VP the/B-NP green/I-NP cup/I-NP in/B-PP the/B-NP rack/I-NP ./O Push/B-VP the/B-NP rack/I-NP back/B-ADVP and/O close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Nr. 2:

*Go/B-VP to/B-PP the/B-NP fridge/I-NP and/O open/B-VP the/B-NP door/I-NP ./O Locate/B-VP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Close/B-VP the/B-NP door/I-NP ./O Come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP the/B-NP orange/I-NP juice/I-NP to/B-PP me/B-NP ./O*

Nr. 3:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP ./O Locate/B-VP the/B-NP popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ./O Come/B-VP over/B-PRT here/B-ADVP ./O Hand/B-VP the/B-NP popcorn/I-NP bag/I-NP over/B-PRT ./O*

Nr. 4:

*Okay/B-INTJ ARMAR/B-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O take/B-VP the/B-NP green/I-NP cup/I-NP from/B-PP there/B-ADVP ./O Look/B-VP out/B-PP for/B-PP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP ./O Open/B-VP its/B-NP door/I-NP ./O Pull/B-VP out/B-ADVP the/B-NP top/I-NP basket/I-NP ./O Put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./O Finally/B-ADVP close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Nr. 5:

*Okay/B-INTJ ARMAR/B-NP look/B-VP for/B-PP the/B-NP refrigerator/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP ./O Grab/B-VP the/B-NP door/I-NP handle/I-NP and/O pull/B-VP ./O Look/B-VP what/B-NP 's/B-VP inside/B-PP the/B-NP fridge/I-NP ./O There/B-NP should/B-VP be/I-NP an/B-NP orange/I-NP juice/I-NP somewhere/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP take/I-NP it/B-NP and/O close/B-VP the/B-NP door/I-NP ./O Come/B-VP over/B-PRT here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT ./O*

Nr. 6:

*At/B-PP the/B-NP kitchen/I-NP table/I-NP there/B-NP is/B-VP a/B-NP popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ./O come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 7:

*Locate/B-VP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP ./O go/B-VP there/B-ADVP ./O When/B-ADVP you/B-NP 're/B-VP there/B-ADVP ./O open/B-VP its/B-NP front/I-NP door/I-NP and/O pull/B-VP out/B-PRT*

*the/B-NP upper/I-NP rack/I-NP ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP what/B-NP 's/B-VP there/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP a/B-NP green/I-NP cup/I-NP ./O take/B-VP it/B-NP ./O Move/B-VP back/B-ADVP towards/B-PP the/B-NP dishwasher/I-NP and/O put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./O Finally/B-ADVP ./O push/B-VP back/B-PP the/B-NP basket/I-NP and/O close/B-VP the/B-NP door/I-NP ./O*

Nr. 8:

*Inside/B-PP the/B-NP ice/I-NP box/I-NP ./O there/B-NP is/B-VP an/B-NP orange/I-NP juice/I-NP ./O So/B-PP ./O go/B-VP over/B-PRT there/B-ADVP ./O open/B-VP the/B-NP door/I-NP and/O take/B-VP it/B-NP ./O When/B-ADVP you/B-NP 've/B-VP got/I-NP it/B-NP ./O close/B-VP the/B-NP door/I-NP ./O carry/B-VP the/B-NP orange/I-NP juice/I-NP over/B-PRT here/B-ADVP and/O hand/B-VP It/B-NP over/B-PRT ./O*

Nr. 9:

*Okay/B-INTJ ARMAR/B-NP ./O look/B-VP for/B-PP the/B-NP table/I-NP ./O Go/B-VP there/B-ADVP and/O take/B-VP the/B-NP popcorn/I-NP ./O After/B-PP taking/B-VP the/B-NP popcorn/I-NP ./O come/B-VP here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT ./O*

Nr. 10:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP if/B-SBAR you/B-NP can/B-VP find/I-NP a/B-NP popcorn/I-NP bag/I-NP ./O If/B-SBAR so/I-SBAR ./O take/B-VP it/B-NP and/O bring/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 11:

*Okay/B-INTJ Armar/B-NP ./O go/B-VP to/B-PP the/B-NP table/I-NP ./O Grab/B-VP popcorn/B-NP ./O Come/B-VP to/B-PP me/B-NP ./O Give/B-VP me/B-NP the/B-NP popcorn/I-NP which/B-NP is/B-VP in/B-PP your/B-NP hand/I-NP ./O*

Nr. 12:

*Okay/B-INTJ Armar/B-NP ./O go/B-VP to/B-PP the/B-NP table/I-NP ./O Take/B-VP the/B-NP green/I-NP cup/I-NP ./O Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP it/B-NP ./O Put/B-VP the/B-NP green/I-NP cup/I-NP into/B-PP the/B-NP dishwasher/I-NP and/O close/B-VP it/B-NP ./O*

Nr. 13:

*Okay/B-INTJ ARMAR/B-NP ./O go/B-VP to/B-PP the/B-NP fridge/I-NP ./O Look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Go/B-VP to/B-PP the/B-NP table/I-NP ./O Pour/B-VP the/B-NP juice/I-NP into/B-PP the/B-NP green/I-NP cup/I-NP ./O Put/B-VP down/B-PP the/B-NP juice/I-NP and/O bring/B-VP me/B-NP the/B-NP green/I-NP cup/I-NP from/B-PP the/B-NP table/I-NP ./O*

Nr. 14:

*Okay/B-INTJ Armar/B-NP ./O go/B-VP to/B-PP the/B-NP fridge/I-NP ./O Open/B-VP it/B-NP ./O Look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Close/B-VP the/B-NP fridge/I-NP and/O bring/B-VP the/B-NP juice/I-*

*NP to/B-PP me/B-NP ./O*

## G. Ausgaben der Sprachverarbeitungswerkzeuge

### G.1. Stanford POS Tagger

Nr. 1:

Mit Satztrennung:

*Go/VB to/TO the/DT dishwasher/NN ./ Open/VB the/DT dishwasher/NN 's/POS door/NN ./ Locate/VB its/PRP\$ upper/JJ rack/NN ./ Pull/VB it/PRP out/RP ./ Go/VB to/TO the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN ./ Take/VB it/PRP ./ Go/VB back/RB to/TO the/DT dishwasher/NN ./ Put/VB the/DT green/JJ cup/NN in/IN the/DT rack/NN ./ Push/VB the/DT rack/NN back/RB and/CC close/RB the/DT dishwasher/NN 's/POS door/NN ./*

Ohne Satztrennung:

*Go/VB to/TO the/DT dishwasher/NN open/VB the/DT dishwasher/NN 's/POS door/NN locate/VB its/PRP\$ upper/JJ rack/NN pull/VB it/PRP out/RP go/VB to/TO the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN take/VBP it/PRP go/VB back/RB to/TO the/DT dishwasher/NN put/VBD the/DT green/JJ cup/NN in/IN the/DT rack/NN push/VB the/DT rack/NN back/RB and/CC close/RB the/DT dishwasher/NN 's/POS door/NN*

Nr. 2:

Mit Satztrennung:

*Go/VB to/TO the/DT fridge/NN and/CC open/VB the/DT door/NN ./ Locate/VB the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./ Close/RB the/DT door/NN ./ Come/VB over/IN here/RB and/CC give/VB the/DT orange/JJ juice/NN to/TO me/PRP ./*

Ohne Satztrennung:

*Go/VB to/TO the/DT fridge/NN and/CC open/VB the/DT door/NN locate/VB the/DT orange/JJ juice/NN and/CC take/VB it/PRP close/VB the/DT door/NN come/VBN over/IN here/RB and/CC give/VB the/DT orange/JJ juice/NN to/TO me/PRP*

Nr. 3:

Mit Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN ./ Locate/VB the/DT the/DT popcorn/NN bag/NN ./ Take/VB it/PRP ./ Come/VB over/IN here/RB ./ Hand/VB the/DT popcorn/NN bag/NN over/IN ./*

Ohne Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN locate/VBP the/DT the/DT popcorn/NN bag/NN take/VBP it/PRP come/VB over/RB here/RB hand/VB the/DT popcorn/NN bag/NN over/IN*

Nr. 4:

Mit Satztrennung:

*Okay/UH ARMAR/NNP ,/, go/VB to/TO the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/RB ./ Look/VB out/RP for/IN the/DT dishwasher/NN ./ If/IN you/PRP can/MD see/VB it/PRP ,/, go/VB over/RB there/RB ./ Open/VB its/PRP\$ door/NN ./ Pull/VB out/RP the/DT top/JJ basket/NN*

*./.* Put/VB the/DT cup/NN into/IN it/PRP *./.* Finally/RB *./,* close/RB the/DT dishwasher/NN 's/POS door/NN *./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NN go/VB to/TO the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/EX look/VB out/RP for/IN the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/IN there/EX open/VB its/PRP\$ door/NN pull/VB out/RP the/DT top/JJ basket/NN put/VBD the/DT cup/NN into/IN it/PRP finally/RB close/RB the/DT dishwasher/NN 's/POS door/NN *./.**

Nr. 5:

Mit Satztrennung:

*Okay/UH ARMAR/NN *./,* look/NN for/IN the/DT refrigerator/NN *./.* If/IN you/PRP can/MD see/VB it/PRP *./,* go/VB over/RB there/RB *./.* Grab/VB the/DT door/NN handle/VB and/CC pull/VB *./.* Look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN *./.* There/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB *./.* If/IN you/PRP can/MD see/VB it/PRP *./,* take/VB it/PRP and/CC close/VB the/DT door/NN *./.* Come/VB over/IN here/RB and/CC hand/VB it/PRP over/RP *./.**

Ohne Satztrennung:

*Okay/UH ARMAR/NNP look/VB for/IN the/DT refrigerator/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/RB there/RB grab/VB the/DT door/NN handle/VB and/CC pull/VB look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN there/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB if/IN you/PRP can/MD see/VB it/PRP take/VB it/PRP and/CC close/VB the/DT door/NN come/VBN over/IN here/RB and/CC hand/VB it/PRP over/RP *./.**

Nr. 6:

Mit Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN *./.* Take/VB it/PRP *./,* come/VBP over/IN here/RB and/CC give/VB it/PRP to/TO me/PRP *./.**

Ohne Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN take/VBP it/PRP come/VB over/IN here/RB and/CC give/VB it/PRP to/TO me/PRP *./.**

Nr. 7:

Mit Satztrennung:

*Locate/VB the/DT dishwasher/NN *./.* If/IN you/PRP can/MD see/VB it/PRP *./,* go/VB there/RB *./.* When/WRB you/PRP 're/VBP there/RB *./,* open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN *./.* Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB *./.* If/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN *./,* take/VB it/PRP *./.* Move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VBD the/DT cup/NN into/IN it/PRP *./.* Finally/RB *./,* push/VB back/RP the/DT basket/NN and/CC close/VB the/DT door/NN *./.**

Ohne Satztrennung:

*Locate/VB the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB there/RB when/WRB you/PRP 're/VBP there/EX open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB if/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN take/VBP it/PRP move/VB back/RB towards/IN the/DT dis-*

*hwasher/NN and/CC put/VBD the/DT cup/NN into/IN it/PRP finally/RB push/VB back/RP the/DT basket/NN and/CC close/VB the/DT door/NN ./.*

Nr. 8:

Mit Satztrennung:

*Inside/IN the/DT ice/NN box/NN ./, there/EX is/VBZ an/DT orange/JJ juice/NN ./.  
So/RB ./, go/VBP over/IN there/RB ./, open/VB the/DT door/NN and/CC take/VB  
it/PRP ./.* *When/WRB you/PRP 've/VBP got/VBD it/PRP ./, close/RB the/DT  
door/NN ./, carry/VBP the/DT orange/JJ juice/NN over/IN here/RB and/CC hand/VB  
it/PRP over/RP ./.*

Ohne Satztrennung:

*Inside/IN the/DT ice/NN box/NN there/EX is/VBZ an/DT orange/JJ juice/NN so/IN  
go/NN over/IN there/RB open/JJ the/DT door/NN and/CC take/VB it/PRP when/WRB  
you/PRP 've/VBP got/VBD it/PRP close/VB the/DT door/NN carry/VBP the/DT oran-  
ge/JJ juice/NN over/IN here/RB and/CC hand/VB it/PRP over/RP ./.*

Nr. 9:

Mit Satztrennung:

*Okay/UH ARMAR/NN ./, look/NN for/IN the/DT table/NN ./.* *Go/VB there/RB an-  
d/CC take/VB the/DT popcorn/NN ./.* *After/IN taking/VBG the/DT popcorn/NN ./,  
come/VBN here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NNP look/VB for/IN the/DT table/NN go/VBP there/RB and/CC  
take/VB the/DT popcorn/NN after/IN taking/VBG the/DT popcorn/NN come/VBN her-  
e/RB and/CC hand/VB it/PRP over/RP ./.*

Nr. 10:

Mit Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD  
find/VB a/DT popcorn/NN bag/NN ./.* *If/IN so/RB ./, take/VB it/PRP and/CC  
bring/VB it/PRP to/TO me/PRP ./.*

Ohne Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD  
find/VB a/DT popcorn/NN bag/NN if/IN so/RB take/VB it/PRP and/CC bring/VB  
it/PRP to/TO me/PRP ./.*

Nr. 11:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN ./.* *Grab/VB popcorn/NN  
./.* *Come/VB to/TO me/PRP ./.* *Give/VB me/PRP the/DT popcorn/NN which/WDT  
is/VBZ in/IN your/PRP\$ hand/NN ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN grab/NN popcorn/NN co-  
me/VBN to/TO me/PRP give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ  
in/IN your/PRP\$ hand/NN ./.*

Nr. 12:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN ./.* *Take/VB the/DT green/JJ  
cup/NN ./.* *Go/VB to/TO the/DT dishwasher/NN ./.* *Open/VB it/PRP ./.* *Put/VB  
the/DT green/JJ cup/NN into/IN the/DT dishwasher/NN and/CC close/VB it/PRP ./.*

Ohne Satztrennung:

*take/VB the/DT green/JJ cup/NN go/VB to/TO the/DT dishwasher/NN open/JJ it/PRP put/VBD the/DT green/JJ cup/NN into/IN the/DT dishwasher/NN and/CC close/VB it/PRP ./.*

Nr. 13:

Mit Satztrennung:

*Okay/UH ARMAR/NNP ./, go/VB to/TO the/DT fridge/NN ./ Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./ Go/VB to/TO the/DT table/NN ./ Pour/VB the/DT juice/NN into/IN the/DT green/JJ cup/NN ./ Put/VB down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NNP ./, go/VB to/TO the/DT fridge/NN look/NN for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP go/VB to/TO the/DT table/NN pour/VBP the/DT juice/NN into/IN the/DT green/JJ cup/NN put/VBD down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN ./.*

Nr. 14:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT fridge/NN ./ Open/VB it/PRP ./ Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./ Close/RB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/TO me/PRP ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT fridge/NN open/VBP it/PRP look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP close/VB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/TO me/PRP ./.*

## G.2. OpenNLP Wortartenerkennung

Nr. 1:

Mit Satztrennung:

*Go/VB to/TO the/DT dishwasher/NN ./ Open/VB the/DT dishwasher/NN 's/POS door/NN ./ Locate/VB its/PRP\$ upper/JJ rack/NN ./ Pull/VB it/PRP out/RP ./ Go/VB to/TO the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN ./ Take/VB it/PRP ./ Go/VB back/RB to/TO the/DT dishwasher/NN ./ Put/VB the/DT green/JJ cup/NN in/IN the/DT rack/NN ./ Push/NNP the/DT rack/NN back/RB and/CC close/VB the/DT dishwasher/NN 's/POS door/NN ./.*

Ohne Satztrennung:

*Go/VB to/TO the/DT dishwasher/NN open/VB the/DT dishwasher/NN 's/POS door/NN locate/VBP its/PRP\$ upper/JJ rack/NN pull/VBP it/PRP out/RP go/VB to/TO the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN take/VB it/PRP go/VB back/RB to/TO the/DT dishwasher/NN put/VBD the/DT green/JJ cup/NN in/IN the/DT rack/NN push/NN the/DT rack/NN back/RB and/CC close/VB the/DT dishwasher/NN 's/POS door/NN*

Nr. 2:

Mit Satztrennung:

*Go/VB to/TO the/DT fridge/NN and/CC open/VB the/DT door/NN ./ Locate/NN*

*the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./.* *Close/RB the/DT door/NN ./.* *Come/VB over/IN here/RB and/CC give/VB the/DT orange/NN juice/NN to/TO me/PRP ./.*

Ohne Satztrennung:

*Go/VB to/TO the/DT fridge/NN and/CC open/VB the/DT door/NN locate/VBP the/DT orange/JJ juice/NN and/CC take/VB it/PRP close/RB the/DT door/NN come/VB over/IN here/RB and/CC give/VB the/DT orange/NN juice/NN to/TO me/PRP*

Nr. 3:

Mit Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN ./.* *Locate/VB the/DT the/DT popcorn/NN bag/NN ./.* *Take/VB it/PRP ./.* *Come/VB over/IN here/RB ./.* *Hand/VB the/DT popcorn/NN bag/NN over/IN ./.*

Ohne Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN locate/VBP the/DT the/DT popcorn/NN bag/NN take/VB it/PRP come/VB over/IN here/RB hand/NN the/DT popcorn/NN bag/NN over/IN*

Nr. 4:

Mit Satztrennung:

*Okay/UH ARMAR/NN ,/, go/VB to/TO the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/RB ./.* *Look/VB out/RP for/IN the/DT dishwasher/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ,/, go/VB over/IN there/RB ./.* *Open/VB its/PRP\$ door/NN ./.* *Pull/VB out/IN the/DT top/NN basket/NN ./.* *Put/VB the/DT cup/NN into/IN it/PRP ./.* *Finally/RB ,/, close/VB the/DT dishwasher/NN 's/POS door/NN ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NN go/VB to/TO the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/RB look/VB out/RP for/IN the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/IN there/RB open/VB its/PRP\$ door/NN pull/VB out/IN the/DT top/NN basket/NN put/VBD the/DT cup/NN into/IN it/PRP finally/RB close/VB the/DT dishwasher/NN 's/POS door/NN*

Nr. 5:

Mit Satztrennung:

*Okay/UH ARMAR/NN ,/, look/NN for/IN the/DT refrigerator/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ,/, go/VB over/IN there/RB ./.* *Grab/VB the/DT door/NN handle/NN and/CC pull/VB ./.* *Look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN ./.* *There/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB ./.* *If/IN you/PRP can/MD see/VB it/PRP ,/, take/VB it/PRP and/CC close/VB the/DT door/NN ./.* *Come/VB over/IN here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NN look/NN for/IN the/DT refrigerator/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/IN there/RB grab/VB the/DT door/NN handle/NN and/CC pull/VB look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN there/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB if/IN you/PRP can/MD see/VB it/PRP take/VB it/PRP and/CC close/VB the/DT door/NN come/VB over/IN here/RB and/CC hand/VB it/PRP over/IN*

Nr. 6:

Mit Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN ./.* *Take/VB it/PRP ,/, come/VB over/IN here/RB and/CC give/VB it/PRP to/TO me/PRP ./.*

Ohne Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/NN bag/NN take/VB it/PRP come/VB over/IN here/RB and/CC give/VB it/PRP to/TO me/PRP*

Nr. 7:

Mit Satztrennung:

*Locate/VB the/DT dishwasher/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ,/, go/VB there/RB ./.* *When/WRB you/PRP 're/VBP there/RB ,/, open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN ./.* *Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/POS there/EX ./.* *If/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN ,/, take/VB it./PRP\$ Move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VB the/DT cup/NN into/IN it/PRP ./.* *Finally/RB ,/, push/VB back/RB the/DT basket/NN and/CC close/VB the/DT door/NN ./.*

Ohne Satztrennung:

*Locate/VB the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB there/RB when/WRB you/PRP 're/VBP there/RB open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN go/VBP to/TO the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB if/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN take/VBP it/PRP move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VB the/DT cup/NN into/IN it/PRP finally/RB push/VB back/RB the/DT basket/NN and/CC close/VB the/DT door/NN*

Nr. 8:

Mit Satztrennung:

*Inside/IN the/DT ice/NN box/NN ,/, there/EX is/VBZ an/DT orange/JJ juice/NN ./.* *So/RB ,/, go/VB over/IN there,/DT open/JJ the/DT door/NN and/CC take/VB it/PRP ./.* *When/WRB you/PRP 've/VBP got/VBD it/PRP ,/, close/RB the/DT door/NN ,/, carry/VBP the/DT orange/JJ juice/NN over/IN here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Inside/IN the/DT ice/NN box/NN there/RB is/VBZ an/DT orange/JJ juice/NN so/RB go/VB over/IN there/RB open/VB the/DT door/NN and/CC take/VB it/PRP when/WRB you/PRP 've/VBP got/VBD it/PRP close/RB the/DT door/NN carry/VBP the/DT orange/JJ juice/NN over/IN here/RB and/CC hand/VB it/PRP over/IN*

Nr. 9:

Mit Satztrennung:

*Okay/UH ARMAR/NN ,/, look/NN for/IN the/DT table/NN ./.* *Go/VB there/RB and/CC take/VB the/DT popcorn/NN ./.* *After/IN taking/VBG the/DT popcorn/NN ,/, come/VBN here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NN look/NN for/IN the/DT table/NN go/VB there/RB and/CC take/VB the/DT popcorn/NN after/IN taking/VBG the/DT popcorn/NN come/VBN here/RB and/CC hand/VB it/PRP over/IN*

Nr. 10:

Mit Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD find/VB a/DT popcorn/NN bag/NN ./.* *If/IN so/RB ./, take/VB it/PRP and/CC bring/VB it/PRP to/TO me/PRP ./.*

Ohne Satztrennung:

*Go/VB to/TO the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD find/VB a/DT popcorn/NN bag/NN if/IN so/RB take/VB it/PRP and/CC bring/VB it/PRP to/TO me/PRP*

Nr. 11:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN ./.* *Grab/VB popcorn/NN ./.* *Come/VB to/TO me/PRP ./.* *Give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ in/IN your/PRP\$ hand/NN ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN grab/NN popcorn/NN come/VBN to/TO me/PRP give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ in/IN your/PRP\$ hand/NN*

Nr. 12:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT table/NN ./.* *Take/VB the/DT green/JJ cup/NN ./.* *Go/VB to/TO the/DT dishwasher/NN ./.* *Open/VB it/PRP ./.* *Put/VBD the/DT green/JJ cup/NN into/IN the/DT dishwasher/NN and/CC close/VB it/PRP ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP go/VB to/TO the/DT table/NN take/VBP the/DT green/JJ cup/NN go/VB to/TO the/DT dishwasher/NN open/JJ it/PRP put/VBD the/DT green/JJ cup/NN into/IN the/DT dishwasher/NN and/CC close/VB it/PRP*

Nr. 13:

Mit Satztrennung:

*Okay/UH ARMAR/NN ./, go/VB to/TO the/DT fridge/NN ./.* *Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./.* *Go/VB to/TO the/DT table/NN ./.* *Pour/PDT the/DT juice/NN into/IN the/DT green/JJ cup/NN ./.* *Put/VBD down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NN go/VB to/TO the/DT fridge/NN look/NN for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP go/VB to/TO the/DT table/NN pour/VBP the/DT juice/NN into/IN the/DT green/JJ cup/NN put/VBD down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN*

Nr. 14:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/TO the/DT fridge/NN ./.* *Open/VB it/PRP ./.* *Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./.* *Close/NNP the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/TO me/PRP ./.*

Ohne Satztrennung:

*Okay/UH Armair/NNP go/VB to/TO the/DT fridge/NN open/VB it/PRP look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP close/RB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/TO me/PRP*

### G.3. Illinois POS Tagger

Nr. 1:

Mit Satztrennung:

*VB/Go TO/to DT/the NN/dishwasher ./ . NNP/Open DT/the JJR/dishwasher POS/'s NN/door ./ . NN/Locate PRP\$/its JJ/upper NN/rack ./ . NNP/Pull PRP/it RP/out ./ . VB/Go TO/to DT/the NN/kitchen NN/table CC/and VB/locate DT/the JJ/green NN/cup ./ . VB/Take PRP/it ./ . VB/Go RB/back TO/to DT/the NN/dishwasher ./ . VB/Put DT/the JJ/green NN/cup IN/in DT/the NN/rack ./ . VBP/Push DT/the NN/rack RB/back CC/and RB/close DT/the JJR/dishwasher POS/'s NN/door ./ .*

Ohne Satztrennung:

*VB/Go TO/to DT/the JJR/dishwasher JJ/open DT/the NN/dishwasher POS/'s NN/door VBP/locate PRP\$/its JJ/upper NN/rack VBP/pull PRP/it RP/out VBP/go TO/to DT/the NN/kitchen NN/table CC/and VB/locate DT/the JJ/green NN/cup VBP/take PRP/it VB/go RB/back TO/to DT/the JJR/dishwasher NN/put DT/the JJ/green NN/cup IN/in DT/the NN/rack VB/push DT/the NN/rack RB/back CC/and RB/close DT/the JJR/dishwasher POS/'s NN/door*

Nr. 2:

Mit Satztrennung:

*VB/Go TO/to DT/the JJ/fridge CC/and JJ/open DT/the NN/door ./ . NN/Locate DT/the NN/orange NN/juice CC/and VB/take PRP/it ./ . NNP/Close DT/the NN/door ./ . VBN/Come IN/over RB/here CC/and VB/give DT/the NN/orange NN/juice TO/to PRP/me ./ .*

Ohne Satztrennung:

*VB/Go TO/to DT/the JJ/fridge CC/and JJ/open DT/the NN/door VB/locate DT/the NN/orange NN/juice CC/and VB/take PRP/it VB/close DT/the NN/door VBN/come IN/over RB/here CC/and VB/give DT/the NN/orange NN/juice TO/to PRP/me*

Nr. 3:

Mit Satztrennung:

*VB/Go TO/to DT/the NN/kitchen NN/table ./ . NN/Locate DT/the DT/the NN/popcorn NN/bag ./ . VB/Take PRP/it ./ . VBN/Come RP/over RB/here ./ . NNP/Hand DT/the JJ/popcorn NN/bag IN/over ./ .*

Ohne Satztrennung:

*VB/Go TO/to DT/the NN/kitchen NN/table VB/locate DT/the DT/the NN/popcorn NN/bag VBP/take PRP/it VBP/come IN/over RB/here NN/hand DT/the JJ/popcorn NN/bag IN/over ./ .*

Nr. 4:

Mit Satztrennung:

*UH/Okay NNP/ARMAR ./ , VBP/go TO/to DT/the NN/kitchen NN/table CC/and VB/take DT/the JJ/green NN/cup IN/from RB/there ./ . VB/Look IN/out IN/for DT/the NN/dishwasher ./ . IN/If PRP/you MD/can VB/see PRP/it ./ , VBP/go IN/over RB/there ./ . NNP/Open PRP\$/its NN/door ./ . NNP/Pull IN/out DT/the JJ/top NN/basket ./ . VB/Put DT/the NN/cup IN/into PRP/it ./ . RB/Finally ./ , RB/close DT/the JJR/dishwasher POS/'s NN/door ./ .*

Ohne Satztrennung:

*UH/Okay NNP/ARMAR VBP/go TO/to DT/the NN/kitchen NN/table CC/and VB/take DT/the JJ/green NN/cup IN/from EX/there VB/look RP/out IN/for DT/the NN/dishwasher IN/if PRP/you MD/can VB/see PRP/it VB/go IN/over RB/there JJ/open PRP\$/its NN/door VB/pull RP/out DT/the JJ/top NN/basket VBD/put DT/the NN/cup IN/into PRP/it RB/finally RB/close DT/the JJR/dishwasher POS/'s NN/door ./.*

Nr. 5:

Mit Satztrennung:

*UH/Okay NNP/ARMAR ./, VBP/look IN/for DT/the NN/refrigerator ./ IN/If PRP/you MD/can VB/see PRP/it ./, VBP/go IN/over RB/there ./ NNP/Grab DT/the NN/door VB/handle CC/and VB/pull ./ VB/Look WP/what VBZ/'s IN/inside DT/the NN/fridge ./ EX/There MD/should VB/be DT/an NN/orange NN/juice RB/somewhere ./ IN/If PRP/you MD/can VB/see PRP/it ./, VB/take PRP/it CC/and VB/close DT/the NN/door ./ VBN/Come IN/over RB/here CC/and VB/hand PRP/it IN/over ./.*

Ohne Satztrennung:

*UH/Okay NNP/ARMAR VB/look IN/for DT/the NN/refrigerator IN/if PRP/you MD/can VB/see PRP/it VB/go IN/over EX/there VB/grab DT/the NN/door VB/handle CC/and VB/pull VB/look WP/what VBZ/'s IN/inside DT/the NN/fridge RB/there MD/should VB/be DT/an NN/orange NN/juice RB/somewhere IN/if PRP/you MD/can VB/see PRP/it VB/take PRP/it CC/and VB/close DT/the NN/door VBN/come IN/over RB/here CC/and VB/hand PRP/it IN/over ./.*

Nr. 6:

Mit Satztrennung:

*IN/At DT/the NN/kitchen NN/table EX/there VBZ/is DT/a JJ/popcorn NN/bag ./ VB/TAke PRP/it ./, VBP/come IN/over RB/here CC/and VB/give PRP/it TO/to PRP/me ./.*

Ohne Satztrennung:

*IN/At DT/the NN/kitchen NN/table EX/there VBZ/is DT/a JJ/popcorn NN/bag VBP/take PRP/it VBP/come IN/over RB/here CC/and VB/give PRP/it TO/to PRP/me ./.*

Nr. 7:

Mit Satztrennung:

*NN/Locate DT/the NN/dishwasher ./ IN/If PRP/you MD/can VB/see PRP/it ./, VBP/go RB/there ./ WRB/When PRP/you VBP/'re RB/there ./, JJ/open PRP\$/its JJ/front NN/door CC/and VB/pull RP/out DT/the JJ/upper NN/rack ./ VB/Go TO/to DT/the NN/kitchen NN/table CC/and VB/see WP/what VBZ/'s RB/there ./ IN/If PRP/you MD/can VB/see DT/a JJ/green NN/cup ./, VB/take PRP/it ./ VB/Move RB/back IN/towards DT/the JJR/dishwasher CC/and VBD/put DT/the NN/cup IN/into PRP/it ./ RB/Finally ./, VB/push RP/back DT/the NN/basket CC/and VB/close DT/the NN/door ./.*

Ohne Satztrennung:

*NN/Locate DT/the NN/dishwasher IN/if PRP/you MD/can VB/see PRP/it VB/go RB/there WRB/when PRP/you VBP/'re RB/there VB/open PRP\$/its JJ/front NN/door CC/and VB/pull RP/out DT/the JJ/upper NN/rack VBP/go TO/to DT/the NN/kitchen NN/table CC/and VB/see WP/what VBZ/'s RB/there IN/if PRP/you MD/can VB/see DT/a JJ/green NN/cup VB/take PRP/it VB/move RB/back IN/towards DT/the JJR/dishwasher CC/and VBD/put DT/the NN/cup IN/into PRP/it RB/finally VBP/push RB/back DT/the NN/basket CC/and VB/close DT/the NN/door ./.*

Nr. 8:

Mit Satztrennung:

*IN/Inside DT/the NN/ice NN/box ,/, EX/there VBZ/is DT/an NN/orange NN/juice ./.  
RB/So ,/, VBP/go IN/over RB/there ,/, JJ/open DT/the NN/door CC/and VB/take  
PRP/it ./.* *WRB/When PRP/you VBP/'ve VBN/got PRP/it ,/, RB/close DT/the NN/  
door ,/, VBP/carry DT/the NN/orange NN/juice IN/over RB/here CC/and VB/hand  
PRP/it IN/over ./.*

Ohne Satztrennung:

*IN/Inside DT/the NN/ice NN/box EX/there VBZ/is DT/an NN/orange NN/juice RB/so  
VBP/go IN/over RB/there JJ/open DT/the NN/door CC/and VB/take PRP/it WR-  
B/when PRP/you VBP/'ve VBN/got PRP/it RB/close DT/the NN/door VBP/carry  
DT/the NN/orange NN/juice IN/over RB/here CC/and VB/hand PRP/it IN/over ./.*

Nr. 9:

Mit Satztrennung:

*UH/Okay NNP/ARMAR ,/, VBP/look IN/for DT/the NN/table ./.* *VB/Go RB/there  
CC/and VB/take DT/the NN/popcorn ./.* *IN/After VBG/taking DT/the NN/popcorn ,/,  
VBP/come RB/here CC/and VB/hand PRP/it IN/over ./.*

Ohne Satztrennung:

*UH/Okay NNP/ARMAR VB/look IN/for DT/the NN/table VB/go RB/there CC/and  
VB/take DT/the VBN/popcorn IN/after VBG/taking DT/the JJ/popcorn VBP/come RB/he-  
re CC/and VB/hand PRP/it IN/over ./.*

Nr. 10:

Mit Satztrennung:

*VB/Go TO/to DT/the NN/kitchen NN/table CC/and VB/see IN/if PRP/you MD/can  
VB/find DT/a JJ/popcorn NN/bag ./.* *IN/If RB/so ,/, VB/take PRP/it CC/and VB/-  
bring PRP/it TO/to PRP/me ./.*

Ohne Satztrennung:

*VB/Go TO/to DT/the NN/kitchen NN/table CC/and VB/see IN/if PRP/you MD/can  
VB/find DT/a JJ/popcorn NN/bag IN/if RB/so VB/take PRP/it CC/and VB/bring  
PRP/it TO/to PRP/me ./.*

Nr. 11:

Mit Satztrennung:

*UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/table ./.* *NNP/Grab VBP/popcorn  
./.* *VBN/Come TO/to PRP/me ./.* *VB/Give PRP/me DT/the NN/popcorn WDT/which  
VBZ/is IN/in PRP\$/your NN/hand ./.*

Ohne Satztrennung:

*UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/table NN/grab NN/popcorn VB-  
N/come TO/to PRP/me VBP/give PRP/me DT/the NN/popcorn WDT/which VBZ/is  
IN/in PRP\$/your NN/hand ./.*

Nr. 12:

Mit Satztrennung:

*UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/table ./.* *VB/Take DT/the JJ/  
green NN/cup ./.* *VB/Go TO/to DT/the NN/dishwasher ./.* *VB/Open PRP/it ./.* *VB/Put  
DT/the JJ/green NN/cup IN/into DT/the JJR/dishwasher CC/and VB/close PRP/it ./.*

Ohne Satztrennung:

UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/table VB/take DT/the JJ/green NN/cup VBP/go TO/to DT/the NN/dishwasher JJ/open PRP/it VBD/put DT/the JJ/green NN/cup IN/into DT/the JJR/dishwasher CC/and VB/close PRP/it ./.

Nr. 13:

Mit Satztrennung:

UH/Okay NNP/ARMAR ,/, VBP/go TO/to DT/the NN/fridge ./ VB/Look IN/for DT/the NN/orange NN/juice CC/and VB/take PRP/it ./ VB/Go TO/to DT/the NN/table ./ NNP/Pour DT/the NN/juice IN/into DT/the JJ/green NN/cup ./ VB/Put IN/down DT/the NN/juice CC/and VB/bring PRP/me DT/the JJ/green NN/cup IN/from DT/the NN/table ./.

Ohne Satztrennung:

UH/Okay NNP/ARMAR ,/, VBP/go TO/to DT/the JJ/fridge NN/look IN/for DT/the NN/orange NN/juice CC/and VB/take PRP/it VB/go TO/to DT/the NN/table VBP/pour DT/the NN/juice IN/into DT/the JJ/green NN/cup VBD/put RP/down DT/the NN/juice CC/and VB/bring PRP/me DT/the JJ/green NN/cup IN/from DT/the NN/table ./.

Nr. 14:

Mit Satztrennung:

UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/fridge ./ VB/Open PRP/it ./ VB/Look IN/for DT/the NN/orange NN/juice CC/and VB/take PRP/it ./ NNP/Close DT/the NN/fridge CC/and VB/bring DT/the NN/juice TO/to PRP/me ./.

Ohne Satztrennung:

UH/Okay NNP/Armar ,/, VBP/go TO/to DT/the NN/fridge JJ/open PRP/it VBP/look IN/for DT/the NN/orange NN/juice CC/and VB/take PRP/it VB/close DT/the NN/fridge CC/and VB/bring DT/the NN/juice TO/to PRP/me ./.

#### G.4. ClearNLP

Nr. 1:

Mit Satztrennung:

Go/NNP to/IN the/DT dishwasher/NN ./ Open/VB the/DT dishwasher/NN 's/POS door/NN ./ Locate/VB its/PRP\$ upper/JJ rack/NN ./ Pull/VB it/PRP out/RP ./ Go/VB to/IN the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN ./ Take/VB it/PRP ./ Go/VB back/RB to/IN the/DT dishwasher/NN ./ Put/VB the/DT green/JJ cup/NN in/IN the/DT rack/NN ./ Push/VB the/DT rack/NN back/RB and/CC close/VB the/DT dishwasher/NN 's/POS door/NN ./.

Ohne Satztrennung:

Go/NNP to/IN the/DT dishwasher/NN open/VB the/DT dishwasher/NN 's/POS door/NN locate/VB its/PRP\$ upper/JJ rack/NN pull/VB it/PRP out/RP go/VB to/IN the/DT kitchen/NN table/NN and/CC locate/VB the/DT green/JJ cup/NN take/VB it/PRP go/VB back/RB to/IN the/DT dishwasher/NN put/VBD the/DT green/JJ cup/NN in/IN the/DT rack/NN push/VBP the/DT rack/NN back/RB and/CC close/VB the/DT dishwasher/NN 's/POS door/NN

Nr. 2:

Mit Satztrennung:

Go/VB to/IN the/DT fridge/NN and/CC open/VB the/DT door/NN ./ Locate/VB the/DT

*orange/NN juice/NN and/CC take/VB it/PRP ./.* *Close/VB the/DT door/NN ./.* *Come/VB over/RP here/RB and/CC give/VB the/DT orange/NN juice/NN to/IN me/PRP ./.*

Ohne Satztrennung:

*Go/VB to/IN the/DT fridge/NN and/CC open/VB the/DT door/NN locate/VB the/DT orange/NN juice/NN and/CC take/VB it/PRP close/RB the/DT door/NN come/VB over/RB here/RB and/CC give/VB the/DT orange/NN juice/NN to/IN me/PRP*

Nr. 3:

Mit Satztrennung:

*Go/VB to/IN the/DT kitchen/NN table/NN ./.* *Locate/VB the/DT the/DT popcorn/JJ bag/NN ./.* *Take/VB it/PRP ./.* *Come/VB over/RB here/RB ./.* *Hand/VB the/DT popcorn/JJ bag/NN over/RP ./.*

Ohne Satztrennung:

*Go/VB to/IN the/DT kitchen/NN table/NN locate/VBP the/DT the/DT popcorn/JJ bag/NN take/VB it/PRP come/VB over/RP here/RB hand/VB the/DT popcorn/JJ bag/NN over/RP ./.*

Nr. 4:

Mit Satztrennung:

*Okay/NNP ARMAR/NNP ./,* *go/VB to/IN the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/RB ./.* *Look/VB out/RP for/IN the/DT dishwasher/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ./,* *go/VB over/RB there/RB ./.* *Open/VB its/PRP\$ door/NN ./.* *Pull/VB out/RP the/DT top/JJ basket/NN ./.* *Put/VB the/DT cup/NN into/IN it/PRP ./.* *Finally/RB ./,* *close/VB the/DT dishwasher/NN 's/POS door/NN ./.*

Ohne Satztrennung:

*Okay/NNP ARMAR/NNP go/VBP to/IN the/DT kitchen/NN table/NN and/CC take/VB the/DT green/JJ cup/NN from/IN there/EX look/VB out/RP for/IN the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/RB there/RB open/VB its/PRP\$ door/NN pull/VB out/RP the/DT top/JJ basket/NN put/VBD the/DT cup/NN into/IN it/PRP finally/RB close/VBP the/DT dishwasher/NN 's/POS door/NN ./.*

Nr. 5:

Mit Satztrennung:

*Okay/UH ARMAR/NNP ./,* *look/VB for/IN the/DT refrigerator/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ./,* *go/VB over/RB there/RB ./.* *Grab/VB the/DT door/NN handle/VB and/CC pull/VB ./.* *Look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN ./.* *There/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB ./.* *If/IN you/PRP can/MD see/VB it/PRP ./,* *take/VB it/PRP and/CC close/VB the/DT door/NN ./.* *Come/VB over/RP here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Okay/NNP ARMAR/NNP look/VB for/IN the/DT refrigerator/NN if/IN you/PRP can/MD see/VB it/PRP go/VB over/RB there/RB grab/VB the/DT door/NN handle/VB and/CC pull/VB look/VB what/WP 's/VBZ inside/IN the/DT fridge/NN there/EX should/MD be/VB an/DT orange/JJ juice/NN somewhere/RB if/IN you/PRP can/MD see/VB it/PRP take/VB it/PRP and/CC close/VB the/DT door/NN come/VB over/RB here/RB and/CC hand/VB it/PRP over/RP ./.*

Nr. 6:

Mit Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/JJ bag/NN ./.* *Take/VB it/PRP ./,* *come/VB over/RP here/RB and/CC give/VB it/PRP to/IN me/PRP ./.*

Ohne Satztrennung:

*At/IN the/DT kitchen/NN table/NN there/EX is/VBZ a/DT popcorn/JJ bag/NN take/VB it/PRP come/VB over/RB here/RB and/CC give/VB it/PRP to/IN me/PRP ./.*

Nr. 7:

Mit Satztrennung:

*Locate/VB the/DT dishwasher/NN ./.* *If/IN you/PRP can/MD see/VB it/PRP ./,* *go/VB there/RB ./.* *When/WRB you/PRP 're/VBP there/RB ./,* *open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN ./.* *Go/VB to/IN the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB ./.* *If/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN ./,* *take/VB it/PRP ./.* *Move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VBD the/DT cup/NN into/IN it/PRP ./.* *Finally/RB ./,* *push/VB back/RB the/DT basket/NN and/CC close/VB the/DT door/NN ./.*

Ohne Satztrennung:

*Locate/VB the/DT dishwasher/NN if/IN you/PRP can/MD see/VB it/PRP go/VB there/RB when/WRB you/PRP 're/VBP there/RB open/VB its/PRP\$ front/JJ door/NN and/CC pull/VB out/RP the/DT upper/JJ rack/NN go/VB to/IN the/DT kitchen/NN table/NN and/CC see/VB what/WP 's/VBZ there/RB if/IN you/PRP can/MD see/VB a/DT green/JJ cup/NN take/VB it/PRP move/VB back/RB towards/IN the/DT dishwasher/NN and/CC put/VBD the/DT cup/NN into/IN it/PRP finally/RB push/VB back/RB the/DT basket/NN and/CC close/VB the/DT door/NN ./.*

Nr. 8:

Mit Satztrennung:

*Inside/VB the/DT ice/NN box/NN ./,* *there/EX is/VBZ an/DT orange/JJ juice/NN ./.* *So/RB ./,* *go/VB over/RB there/RB ./,* *open/VB the/DT door/NN and/CC take/VB it/PRP ./.* *When/WRB you/PRP 've/VBP got/VBN it/PRP ./,* *close/VB the/DT door/NN ./,* *carry/VB the/DT orange/JJ juice/NN over/IN here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Inside/VB the/DT ice/NN box/NN there/EX is/VBZ an/DT orange/JJ juice/NN so/RB go/VB over/RB there/RB open/VB the/DT door/NN and/CC take/VB it/PRP when/WRB you/PRP 've/VBP got/VBN it/PRP close/VB the/DT door/NN carry/VBP the/DT orange/NN juice/NN over/RB here/RB and/CC hand/VB it/PRP over/RP ./.*

Nr. 9:

Mit Satztrennung:

*Okay/UH ARMAR/NNP ./,* *look/VB for/IN the/DT table/NN ./.* *Go/VB there/RB and/CC take/VB the/DT popcorn/NN ./.* *After/IN taking/VBG the/DT popcorn/NN ./,* *come/VB here/RB and/CC hand/VB it/PRP over/RP ./.*

Ohne Satztrennung:

*Okay/NNP ARMAR/NNP look/VB for/IN the/DT table/NN go/VB there/RB and/CC take/VB the/DT popcorn/NN after/IN taking/VBG the/DT popcorn/NN come/VB here/RB and/CC hand/VB it/PRP over/RP ./.*

Nr. 10:

Mit Satztrennung:

*Go/VB to/IN the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD find/VB a/DT popcorn/JJ bag/NN ./.* *If/IN so/RB ./, take/VB it/PRP and/CC bring/VB it/PRP to/IN me/PRP ./.*

Ohne Satztrennung:

*Go/NNP to/IN the/DT kitchen/NN table/NN and/CC see/VB if/IN you/PRP can/MD find/VB a/DT popcorn/JJ bag/NN if/IN so/RB take/VB it/PRP and/CC bring/VB it/PRP to/IN me/PRP ./.*

Nr. 11:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/IN the/DT table/NN ./.* *Grab/NNP popcorn/NN ./.* *Come/VB to/IN me/PRP ./.* *Give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ in/IN your/PRP\$ hand/NN ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP go/VB to/IN the/DT table/NN grab/NN popcorn/NN come/VB to/IN me/PRP give/VB me/PRP the/DT popcorn/NN which/WDT is/VBZ in/IN your/PRP\$ hand/NN*

Nr. 12:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/IN the/DT table/NN ./.* *Take/VB the/DT green/JJ cup/NN ./.* *Go/VB to/IN the/DT dishwasher/NN ./.* *Open/VB it/PRP ./.* *Put/VB the/DT green/JJ cup/NN into/IN the/DT dishwasher/JJ and/CC close/JJ it/PRP*

Ohne Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/IN the/DT table/NN take/VB the/DT green/JJ cup/NN go/VB to/IN the/DT dishwasher/NN open/JJ it/PRP put/VBD the/DT green/JJ cup/NN into/IN the/DT dishwasher/JJ and/CC close/JJ it/PRP*

Nr. 13:

Mit Satztrennung:

*Okay/UH ARMAR/NNP ./, go/VB to/IN the/DT fridge/NN ./.* *Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./.* *Go/VB to/IN the/DT table/NN ./.* *Pour/VB the/DT juice/NN into/IN the/DT green/JJ cup/NN ./.* *Put/VB down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN ./.*

Ohne Satztrennung:

*Okay/UH ARMAR/NNP ./, go/VB to/IN the/DT fridge/NN look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP go/VB to/IN the/DT table/NN pour/VB the/DT juice/NN into/IN the/DT green/JJ cup/NN put/VBD down/RP the/DT juice/NN and/CC bring/VB me/PRP the/DT green/JJ cup/NN from/IN the/DT table/NN*

Nr. 14:

Mit Satztrennung:

*Okay/UH Armar/NNP ./, go/VB to/IN the/DT fridge/NN ./.* *Open/VB it/PRP ./.* *Look/VB for/IN the/DT orange/JJ juice/NN and/CC take/VB it/PRP ./.* *Close/VB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/IN me/PRP ./.*

Ohne Satztrennung:

*Okay/UH Armar/NNP go/VB to/IN the/DT fridge/NN open/JJ it/PRP look/VBP for/IN*

*the/DT orange/JJ juice/NN and/CC take/VB it/PRP close/RB the/DT fridge/NN and/CC bring/VB the/DT juice/NN to/IN me/PRP*

## G.5. MBSP

### Format: Wort/Wortart/Chunking

Nr. 1:

Mit Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP dishwasher/NN/NP^ ././- Open/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^ ././- Locate/VB/VP its/PRP\$/NP upper/JJ/NP^ rack/NN/NP^ ././- Pull/VB/VP it/PRP/NP out/RP/PRT ././- Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- locate/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ ././- Take/VB/VP it/PRP/NP ././- Go/VB/VP back/RB/ADVP to/TO/PP the/DT/NP dishwasher/NN/NP^ ././- Put/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ in/IN/PP the/DT/NP rack/NN/NP^ ././- Push/NN/NP the/NNP/NP^ rack/NN/NP^ back/RB/ADVP and/CC/- close/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^ ././-*

Ohne Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP dishwasher/NN/NP^ open/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^ locate/VB/VP its/PRP\$/NP upper/JJ/NP^ rack/NN/NP^ pull/VBP/VP it/PRP/NP out/RP/PRT go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- locate/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ take/VBP/VP it/PRP/NP go/VB/VP back/RB/ADVP to/TO/PP the/DT/NP dishwasher/NN/NP^ put/VBD/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ in/IN/PP the/DT/NP rack/JJ/NP^ push/NN/NP^ the/DT/NP rack/NN/NP^ back/RB/ADVP and/CC/- close/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^*

Nr. 2:

Mit Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP fridge/NN/NP^ and/CC/- open/VB/VP the/DT/NP door/NN/NP^ ././- Locate/VB/VP the/DT/NP orange/JJ/NP^ juice/NN/NP^ and/CC/- take/VB/VP it/PRP/NP ././- Close/VB/VP the/DT/NP door/NN/NP^ ././- Come/VB/N/PP over/IN/PP^ here/RB/NP and/CC/- give/VB/VP the/DT/NP orange/JJ/NP^ juice/NN/NP^ to/TO/PP me/PRP/NP ././-*

Ohne Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP fridge/NN/NP^ and/CC/- open/VB/VP the/DT/NP door/NN/NP^ locate/VBP/VP the/DT/NP orange/JJ/NP^ juice/NN/NP^ and/CC/- take/VB/VP it/PRP/NP close/VB/VP the/DT/NP door/NN/NP^ come/VBN/VP over/IN/PP here/RB/NP and/CC/- give/VB/VP the/DT/NP orange/JJ/NP^ juice/NN/NP^ to/TO/PP me/PRP/NP*

Nr. 3:

Mit Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ ././- Locate/VB/VP the/DT/NP the/DT/NP^ popcorn/JJ/NP^ bag/NN/NP^ ././- Take/VB/VP it/PRP/NP ././- Come/VBN/PP over/IN/PP^ here/RB/NP ././- Hand/VBN/VP the/DT/NP popcorn/JJ/NP^ bag/NN/NP^ over/IN/ADVP ././-*

Ohne Satztrennung:

*Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ locate/VBP/VP*

*the/DT/NP the/DT/NP^ popcorn/JJ/NP^ bag/NN/NP^ take/VBP/VP it/PRP/NP come/VBP/VP over/IN/PP here/RB/ADVP hand/NN/NP the/DT/NP popcorn/JJ/NP^ bag/NN/NP^ over/IN/ADVP ././-*

Nr. 4:

Mit Satztrennung:

*Okay/UH/INTJ ARMAR/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- take/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ from/IN/PP there/RB/NP ././- Look/VB/VP out/IN/PP for/IN/PP the/DT/NP dishwasher/NN/NP^ ././- If/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP ,/,/- go/VB/VP over/IN/PP there/RB/ADVP ././- Open/VB/VP its/PRP\$/NP door/NN/NP^ ././- Pull/VB/VP out/RP/PRT the/DT/NP top/JJ/NP^ basket/NN/NP^ ././- Put/VB/VP the/DT/NP cup/NN/NP^ into/IN/PP it/PRP/NP ././- Finally/RB/ADVP ,/,/- close/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^ ././-*

Ohne Satztrennung:

*Okay/JJ/NP ARMAR/NNP/NP^ go/VBP/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- take/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ from/IN/PP there/RB/NP look/VB/VP out/RP/PRT for/IN/PP the/DT/NP dishwasher/NN/NP^ if/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP go/VB/VP over/IN/PP there/RB/NP open/JJ/VP its/PRP\$/NP door/NN/NP^ pull/VBP/VP out/RP/PRT the/DT/NP top/JJ/NP^ basket/NN/NP^ put/VBD/VP the/DT/NP cup/NN/NP^ into/IN/PP it/PRP/NP finally/RB/ADVP close/VB/VP the/DT/NP dishwasher/NN/NP^ 's/POS/NP door/NN/NP^ ././-*

Nr. 5:

Mit Satztrennung:

*Okay/UH/INTJ ARMAR/NNP/NP ,/,/- look/VBP/VP for/IN/PP the/DT/NP refrigerator/NN/NP^ ././- If/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP ,/,/- go/VB/VP over/IN/PP there/RB/ADVP ././- Grab/IN/PP the/DT/NP door/NN/NP^ handle/VB/VP and/CC/VP^ pull/VB/VP^ ././- Look/VB/VP what/WP/NP 's/VBZ/VP inside/IN/PP the/DT/NP fridge/NN/NP^ ././- There/EX/NP should/MD/VP be/VB/VP^ an/DT/NP orange/JJ/NP^ juice/NN/NP^ somewhere/RB/NP^ ././- If/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP ,/,/- take/VBP/VP it/PRP/NP and/CC/- close/VB/VP the/DT/NP door/NN/NP^ ././- Come/VBN/PP over/IN/PP^ here/RB/ADVP and/CC/- hand/NN/NP it/PRP/NP over/RP/PRT ././-*

Ohne Satztrennung:

*Okay/JJ/NP ARMAR/NNP/NP^ look/VB/VP for/IN/PP the/DT/NP refrigerator/NN/NP^ if/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP go/VB/VP over/IN/PP there/RB/NP grab/VB/VP the/DT/NP door/NN/NP^ handle/VB/VP and/CC/VP^ pull/VB/VP^ look/VB/VP^ what/WP/NP 's/VBZ/VP inside/IN/PP the/DT/NP fridge/NN/NP^ there/RB/ADVP should/MD/VP be/VB/VP^ an/DT/NP orange/JJ/NP^ juice/NN/NP^ somewhere/RB/ADVP if/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ it/PRP/NP take/VB/VP it/PRP/NP and/CC/- close/VB/VP the/DT/NP door/NN/NP^ come/VBN/VP over/IN/PP here/RB/ADVP and/CC/- hand/NN/NP it/PRP/NP over/RP/PRT ././-*

Nr. 6:

Mit Satztrennung:

*At/IN/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ there/EX/NP is/VBZ/VP a/DT/NP popcorn/JJ/NP^ bag/NN/NP^ ././- Take/VB/VP it/PRP/NP ,/,/- come/VBP/VP over-*

/IN/PP here/RB/NP and/CC/- give/VB/VP it/PRP/NP to/TO/PP me/PRP/NP ././-

Ohne Satztrennung:

At/IN/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ there/EX/NP is/VBZ/VP  
a/DT/NP popcorn/JJ/NP^ bag/NN/NP^ take/VBP/VP it/PRP/NP come/VBP/VP over-  
/IN/PP here/RB/NP and/CC/- give/VB/VP it/PRP/NP to/TO/PP me/PRP/NP

Nr. 7:

Mit Satztrennung:

Locate/VB/VP the/DT/NP dishwasher/NN/NP^ ././- If/IN/SBAR you/PRP/NP can/MD/VP  
see/VB/VP^ it/PRP/NP ,./,- go/VB/VP there/RB/ADVP ././- When/WRB/  
ADVP you/PRP/NP 're/VBP/VP there/RB/ADVP ,./,- open/VB/VP its/PRP\$/NP  
front/NN/NP^ door/NN/NP^ and/CC/- pull/VB/VP out/RP/PRT the/DT/NP up-  
per/JJ/NP^ rack/NN/NP^ ././- Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ ta-  
ble/NN/NP^ and/CC/- see/VB/VP what/WP/NP 's/VBZ/VP there/RB/ADVP ././-  
If/IN/SBAR you/PRP/NP can/MD/VP see/VB/VP^ a/DT/NP green/JJ/NP^  
cup/NN/NP^  
,./,- take/VBP/VP it/PRP/NP ././- Move/VB/VP back/RB/ADVP towards/IN/PP  
the/DT/NP dishwasher/NN/NP^ and/CC/- put/VBD/VP the/DT/NP cup/NN/NP^ in-  
to/IN/PP it/PRP/NP ././- Finally/RB/ADVP ,./,- push/VB/VP back/RP/PRT  
the/DT/NP basket/NN/NP^ and/CC/- close/VB/VP the/DT/NP door/NN/NP^ ././-

Ohne Satztrennung:

Locate/VB/VP the/DT/NP dishwasher/NN/NP^ if/IN/SBAR you/PRP/NP can/MD/VP  
see/VB/VP^ it/PRP/NP go/VB/VP there/RB/ADVP when/WRB/ADVP you/PRP/NP  
're/VBP/VP there/RB/NP open/VB/VP its/PRP\$/NP front/NN/NP^ door/NN/NP^  
and/CC/- pull/VB/VP out/RP/PRT the/DT/NP upper/JJ/NP^ rack/NN/NP^ go/VB-  
P/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- see/VB/VP  
what/WP/NP 's/VBZ/VP there/RB/NP if/IN/SBAR you/PRP/NP can/MD/VP see/V-  
B/VP^ a/DT/NP green/JJ/NP^ cup/NN/NP^ take/VBP/VP it/PRP/NP move/VB/VP  
back/RB/ADVP towards/IN/PP the/DT/NP dishwasher/NN/NP^ and/CC/- put/VB-  
D/VP the/DT/NP cup/NN/NP^ into/IN/PP it/PRP/NP finally/RB/ADVP push/V-  
B/VP back/RP/PRT the/DT/NP basket/NN/NP^ and/CC/- close/VB/VP the/DT/NP  
door/NN/NP^ ././-

Nr. 8:

Mit Satztrennung:

Inside/IN/PP the/DT/NP ice/NN/NP^ box/NN/NP^ ,./,- there/EX/NP is/VBZ/VP  
an/DT/NP orange/JJ/NP^ juice/NN/NP^ ././- So/RB/ADVP ,./,- go/VB/VP over-  
/IN/PP there/RB/ADVP ,./,- open/VB/VP the/DT/NP door/NN/NP^ and/CC/- ta-  
ke/VB/VP it/PRP/NP ././- When/WRB/ADVP you/PRP/NP 've/VBP/VP got/VB-  
N/VP^ it/PRP/NP ,./,- close/VB/VP the/DT/NP door/NN/NP^ ,./,- carry/VBP/VP  
the/DT/NP orange/JJ/NP^ juice/NN/NP^ over/IN/ADVP here/RB/ADVP^ and/CC/-  
hand/NN/NP it/PRP/NP over/RP/PRT ././-

Ohne Satztrennung:

Inside/IN/PP the/DT/NP ice/NN/NP^ box/NN/NP^ there/EX/NP is/VBZ/VP an/DT/NP  
orange/JJ/NP^ juice/NN/NP^ so/RB/VP go/VBP/VP^ over/IN/PP there/RB/NP open/V-  
B/VP the/DT/NP door/NN/NP^ and/CC/- take/VB/VP it/PRP/NP when/WRB/ADVP  
you/PRP/NP 've/VBP/VP got/VBN/VP^ it/PRP/NP close/VB/VP the/DT/NP  
door/NN/NP^ carry/VBP/VP the/DT/NP orange/JJ/NP^ juice/NN/NP^ over/IN/ADVP  
here/RB/ADVP^ and/CC/- hand/NN/NP it/PRP/NP over/RP/PRT ././-

Nr. 9:

Mit Satztrennung:

Okay/UH/INTJ ARMAR/NNP/NP ,/,/- look/VBP/VP for/IN/PP the/DT/NP table/NN/NP^ ././- Go/VB/VP there/RB/ADVP and/CC/- take/VB/VP the/DT/NP popcorn/NN/NP^ ././- After/IN/PP taking/VBG/VP the/DT/NP popcorn/NN/NP^ ,/,/- come/VBP/VP here/RB/ADVP and/CC/- hand/NN/NP it/PRP/NP over/RP/PRT ././-

Ohne Satztrennung:

Okay/JJ/NP ARMAR/NNP/NP^ look/VB/VP for/IN/PP the/DT/NP table/NN/NP^ go/VB- P/VP there/RB/ADVP and/CC/- take/VB/VP the/DT/NP popcorn/NN/NP^ after/IN/PP taking/VBG/VP the/DT/NP popcorn/NN/NP^ come/VBN/VP here/RB/ADVP and/CC/- hand/NN/NP it/PRP/NP over/RP/PRT ././-

Nr. 10:

Mit Satztrennung:

Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- see/V- B/VP if/IN/SBAR you/PRP/NP can/MD/VP find/VB/VP^ a/DT/NP popcorn/JJ/NP^ bag/NN/NP^ ././- If/IN/SBAR so/RB/ADVP ,/,/- take/VBP/VP it/PRP/NP and/CC/- bring/VB/VP it/PRP/NP to/TO/PP me/PRP/NP ././-

Ohne Satztrennung:

Go/VB/VP to/TO/PP the/DT/NP kitchen/NN/NP^ table/NN/NP^ and/CC/- see/V- B/VP if/IN/SBAR you/PRP/NP can/MD/VP find/VB/VP^ a/DT/NP popcorn/JJ/NP^ bag/NN/NP^ if/IN/SBAR so/RB/ADVP take/VB/VP it/PRP/NP and/CC/- bring/V- B/VP it/PRP/NP to/TO/PP me/PRP/NP ././-

Nr. 11:

Mit Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ ././- Grab/NNP/NP popcorn/NN/NP^ ././- Come/VBN/PP to/TO/PP me/PRP/NP ././- Give/VB/VP me/PRP/NP the/DT/NP popcorn/NN/NP^ which/WDT/NP is/VBZ/VP in/IN/PP your/PRP\$/NP hand/NN/NP^ ././-

Ohne Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ grab/NN/NP^ popcorn/NN/NP^ come/VBP/VP to/TO/PP me/PRP/NP give/VB/VP me/PRP/NP the/DT/NP popcorn/NN/NP^ which/WDT/NP is/VBZ/VP in/IN/PP your/PRP\$/NP hand/NN/NP^ ././-

Nr. 12:

Mit Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ ././- Take/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ ././- Go/VB/VP to/TO/PP the/DT/NP dishwasher/NN/NP^ ././- Open/V- B/VP it/PRP/NP ././- Put/VB/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ into/IN/PP the/DT/NP dishwasher/NN/NP^ and/CC/- close/VB/VP it/PRP/NP ././-

Ohne Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ ta- ke/VBP/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ go/VBP/VP to/TO/PP the/DT/NP dishwasher/NN/NP^ open/VBP/VP it/PRP/NP put/VBD/VP the/DT/NP green/JJ/NP^ cup/NN/NP^ into/IN/PP the/DT/NP dishwasher/NN/NP^ and/CC/- close/VB/VP it/PRP/NP ././-

Nr. 13:

Mit Satztrennung:

Okay/UH/INTJ ARMAR/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP fridge/NN/NP^  
 ././- Look/VB/VP for/IN/PP the/DT/NP orange/JJ/NP^ juice/NN/NP^ and/CC/- ta-  
 ke/VB/VP it/PRP/NP ././- Go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ ././-  
 Pour/IN/PP the/DT/NP juice/NN/NP^ into/IN/PP the/DT/NP green/JJ/NP^ cup/NN/NP^  
 ././- Put/VB/VP down/RP/PRT the/DT/NP juice/NN/NP^ and/CC/- bring/VB/VP  
 me/PRP/NP the/DT/NP green/JJ/NP^ cup/NN/NP^ from/IN/PP the/DT/NP  
 table/NN/NP^ ././-

Ohne Satztrennung:

Okay/UH/INTJ ARMAR/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP fridge/JJ/NP^  
 look/NN/NP^ for/IN/PP the/DT/NP orange/JJ/NP^ juice/NN/NP^ and/CC/- take/V-  
 B/VP it/PRP/NP go/VB/VP to/TO/PP the/DT/NP table/NN/NP^ pour/VBP/VP  
 the/DT/NP juice/NN/NP^ into/IN/PP the/DT/NP green/JJ/NP^ cup/NN/NP^ put/VB-  
 D/VP down/RP/PRT the/DT/NP juice/NN/NP^ and/CC/- bring/VB/VP me/PRP/NP  
 the/DT/NP green/JJ/NP^ cup/NN/NP^ from/IN/PP the/DT/NP table/NN/NP^ ././-

Nr. 14:

Mit Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP fridge/NN/NP^  
 ././- Open/VB/VP it/PRP/NP ././- Look/VB/VP for/IN/PP the/DT/NP orange/JJ/NP^  
 juice/NN/NP^ and/CC/- take/VB/VP it/PRP/NP ././- Close/VB/VP the/DT/NP  
 fridge/NN/NP^  
 and/CC/- bring/VB/VP the/DT/NP juice/NN/NP^ to/TO/PP me/PRP/NP ././-

Ohne Satztrennung:

Okay/UH/INTJ Armar/NN/NP ,/,/- go/VB/VP to/TO/PP the/DT/NP fridge/JJ/NP^  
 open/JJ/NP^ it/PRP/NP look/VBP/VP for/IN/PP the/DT/NP orange/JJ/NP^ jui-  
 ce/NN/NP^ and/CC/- take/VB/VP it/PRP/NP close/VB/VP the/DT/NP fridge/NN/NP^  
 and/CC/- bring/VB/VP the/DT/NP juice/NN/NP^ to/TO/PP me/PRP/NP ././-

## G.6. SENNA

### Format: Wort/Wortart/Chunking

Nr. 1:

Mit Satztrennung:

Go/VB/S-VP to/TO/S-PP the/DT/B-NP dishwasher/NN/E-NP ././O Open/NNP/S-  
 NP the/DT/B-NP dishwasher/NN/E-NP 's/POS/B-NP door/NN/E-NP ././O Locate/VB/S-  
 VP its/PRP\$/B-NP upper/JJ/I-NP rack/NN/E-NP ././O Pull/VB/S-VP it/PRP/S-NP  
 out./IN/S-PP Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-  
 NP and/CC/O locate/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP ././O  
 Take/VB/S-VP it./PRP/S-NP Go/VB/S-VP back/RB/S-ADVP to/TO/S-PP the/DT/B-  
 NP  
 dishwasher/NN/E-NP ././O Put/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP  
 in/IN/S-PP the/DT/B-NP rack/NN/E-NP ././O Push/VB/S-VP the/DT/B-NP rack/NN/E-  
 NP back/RB/S-ADVP and/CC/O close/VB/S-VP the/DT/B-NP dishwasher/NN/E-NP  
 's/POS/B-NP door/NN/E-NP ././O

Ohne Satztrennung:

Go/VB/S-VP to/TO/S-PP the/DT/B-NP dishwasher/NN/E-NP open/VB/S-VP the/DT/B-  
 NP dishwasher/NN/E-NP 's/POS/B-NP door/NN/E-NP locate/VB/S-VP its/PRP\$/B-  
 NP upper/JJ/I-NP rack/NN/E-NP pull/VB/S-VP it/PRP/S-NP out/RP/S-ADVP go/VB/S-  
 VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O locate/VB/S-  
 VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP take/VBP/S-VP it/PRP/S-NP go/VB/S-

*VP back/RB/S-ADVP to/TO/S-PP the/DT/B-NP dishwasher/NN/E-NP put/VBD/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP in/IN/S-PP the/DT/B-NP rack/NN/E-NP push/VB/S-VP the/DT/B-NP rack/NN/E-NP back/RB/S-ADVP and/CC/O close/VB/S-VP the/DT/B-NP dishwasher/NN/E-NP 's/POS/B-NP door/NN/E-NP*

Nr. 2:

Mit Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP fridge/NN/E-NP and/CC/O open/VB/S-VP the/DT/B-NP door/NN/E-NP ././O Locate/VB/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP and/CC/O take/VB/S-VP it./PRP/S-NP Close/VB/S-VP the/DT/B-NP door/NN/E-NP ././O Come/VB/S-VP over/RB/S-ADVP here/RB/S-ADVP and/C-C/O give/VB/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP to/TO/S-PP me././S-ADJP*

Ohne Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP fridge/NN/E-NP and/CC/O open/VB/S-VP the/DT/B-NP door/NN/E-NP locate/VB/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP and/CC/O take/VB/S-VP it/PRP/S-NP close/VB/S-VP the/DT/B-NP door/NN/E-NP come/VB/S-VP over/RB/S-ADVP here/RB/S-ADVP and/CC/O give/VB/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP to/TO/S-PP me/PRP/S-NP*

Nr. 3:

Mit Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP ././O Locate/VB/S-VP the/DT/B-NP the/DT/I-NP popcorn/NN/I-NP bag/NN/E-NP ././O Take/VB/S-VP it./PRP/S-NP Come/VB/S-VP over/RB/B-ADVP here/RB/E-ADVP ././O Hand/NN/S-NP the/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP over/IN/S-ADVP ././O*

Ohne Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP locate/VB/S-VP the/DT/B-NP the/DT/I-NP popcorn/NN/I-NP bag/NN/E-NP take/VBP/S-VP it/PRP/S-NP come/VB/S-VP over/RP/S-PRT here/RB/S-ADVP hand/VBP/S-VP the/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP over/IN/S-ADVP ././O*

Nr. 4:

Mit Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP ,./O go/VBP/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O take/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP from/IN/S-PP there././O Look/VB/S-VP out/RP/S-PRT for/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP ././O If/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP ,./O go/VB/S-VP over/RB/B-ADVP there././E-ADVP Open/VB/S-VP its/PRP\$/B-NP door/NN/E-NP ././O Pull/VB/S-VP out/RP/S-PRT the/DT/B-NP top/JJ/I-NP basket/NN/E-NP ././O Put/VB/S-VP the/DT/B-NP cup/NN/E-NP into/IN/S-PP it./NN/S-NP Finally/RB/S-ADVP ,./O close/RB/S-ADVP the/DT/B-NP dishwasher/NN/E-NP 's/POS/B-NP door/NN/E-NP ././O*

Ohne Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O take/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP from/IN/S-PP there/EX/S-NP look/VBP/S-VP out/RP/S-PRT for/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP go/VB/S-VP over/RP/S-PRT there/RB/S-NP open/VB/S-*

*VP its/PRP\$/B-NP door/NN/E-NP pull/VB/S-VP out/RP/S-PRT the/DT/B-NP top/JJ/I-NP basket/NN/E-NP put/VBD/S-VP the/DT/B-NP cup/NN/E-NP into/IN/S-PP it/PRP/S-NP finally/RB/B-VP close/VB/E-VP the/DT/B-NP dishwasher/NN/E-NP 's/POS/B-NP door/NN/E-NP ././O*

Nr. 5:

Mit Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP ././O look/VBP/S-VP for/IN/S-PP the/DT/B-NP refrigerator/NN/E-NP ././O If/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP ././O go/VB/S-VP over/RB/B-ADVP there././E-ADVP Grab/VB/S-VP the/DT/B-NP door/NN/E-NP handle/VB/B-VP and/CC/I-VP pull/VB/E-VP ././O Look/VB/S-VP what/WP/S-NP 's/VBZ/S-VP inside/IN/S-PP the/DT/B-NP fridge/NN/E-NP ././O There/EX/S-NP should/MD/B-VP be/VB/E-VP an/DT/B-NP orange/NN/I-NP juice/NN/E-NP somewhere/RB/S-ADVP ././O If/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP ././O take/VB/S-VP it/PRP/S-NP and/C-C/O close/VB/S-VP the/DT/B-NP door/NN/E-NP ././O Come/VB/S-VP over/RB/S-ADVP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Ohne Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/I-NP look/NN/E-NP for/IN/S-PP the/DT/B-NP refrigerator/NN/E-NP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP go/VB/S-VP over/RP/S-PRT there/RB/S-ADVP grab/VB/S-VP the/DT/B-NP door/NN/E-NP handle/VB/B-VP and/CC/I-VP pull/VB/E-VP look/VB/S-VP what/WP/S-NP 's/VBZ/S-VP inside/IN/S-PP the/DT/B-NP fridge/NN/E-NP there/EX/S-NP should/MD/B-VP be/VB/E-VP an/DT/B-NP orange/NN/I-NP juice/NN/E-NP somewhere/RB/B-SBAR if/IN/E-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP take/VB/S-VP it/PRP/S-NP and/CC/O close/VB/S-VP the/DT/B-NP door/NN/E-NP come/VB/S-VP over/RB/S-ADVP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Nr. 6:

Mit Satztrennung:

*At/IN/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP there/EX/S-NP is/VBZ/S-VP a/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP ././O Take/VB/S-VP it/PRP/S-NP ././O come/VBP/S-VP over/RP/S-PRT here/RB/S-ADVP and/CC/O give/VB/S-VP it/PRP/S-NP to/TO/B-VP me./VB/E-VP*

Ohne Satztrennung:

*At/IN/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP there/EX/S-NP is/VBZ/S-VP a/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP take/VBP/S-VP it/PRP/S-NP come/VB/S-VP over/RP/S-PRT here/RB/S-ADVP and/CC/O give/VB/S-VP it/PRP/S-NP to/TO/S-PP me/PRP/S-NP*

Nr. 7:

Mit Satztrennung:

*Locate/VB/S-VP the/DT/B-NP dishwasher/NN/E-NP ././O If/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP ././O go/VBP/S-VP there././S-ADJP When/WRB/S-ADVP you/PRP/S-NP 're/VBP/S-VP there/RB/S-ADVP ././O open/VB/S-VP its/PRP\$/B-NP front/JJ/I-NP door/NN/E-NP and/CC/O pull/VB/S-VP out/RP/S-PRT the/DT/B-NP upper/JJ/I-NP rack/NN/E-NP ././O Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O see/VB/S-VP what/WP/S-NP 's/VBZ/S-VP there././S-ADJP If/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-*

*VP a/DT/B-NP green/JJ/I-NP cup/NN/E-NP ,/,/O take/VB/S-VP it./PRP/S-NP Move/VB/S-VP back/RB/S-ADVP towards/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP and/CC/O put/VBD/S-VP the/DT/B-NP cup/NN/E-NP into/IN/S-PP it./NN/S-NP Finally/RB/S-ADVP ,/,/O push/VB/S-VP back/RP/S-ADVP the/DT/B-NP basket/NN/E-NP and/CC/O close/VB/S-VP the/DT/B-NP door/NN/E-NP ././O*

Ohne Satztrennung:

*Locate/VB/S-VP the/DT/B-NP dishwasher/NN/E-NP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP it/PRP/S-NP go/VB/S-VP there/RB/S-ADVP when/WRB/S-ADVP you/PRP/S-NP 're/VBP/S-VP there/RB/S-ADVP open/VB/S-VP its/PRP\$/B-NP front/JJ/I-NP door/NN/E-NP and/CC/O pull/VB/S-VP out/RP/S-PRT the/DT/B-NP upper/JJ/I-NP rack/NN/E-NP go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O see/VB/S-VP what/WP/S-NP 's/VBZ/S-VP there/RB/S-ADVP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP see/VB/E-VP a/DT/B-NP green/JJ/I-NP cup/NN/E-NP take/VBP/S-VP it/PRP/S-NP move/VB/S-VP back/RB/S-ADVP towards/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP and/C-C/O put/VBD/S-VP the/DT/B-NP cup/NN/E-NP into/IN/S-PP it/PRP/S-NP finally/RB/S-ADVP push/VB/S-VP back/RP/S-ADVP the/DT/B-NP basket/NN/E-NP and/CC/O close/VB/S-VP the/DT/B-NP door/NN/E-NP ././O*

Nr. 8:

Mit Satztrennung:

*Inside/IN/S-PP the/DT/B-NP ice/NN/I-NP box/NN/E-NP ,/,/O there/EX/S-NP is/VBZ/S-VP an/DT/B-NP orange/NN/I-NP juice/NN/E-NP ././O So/RB/S-ADVP ,/,/O go/VBP/S-VP over/RB/S-ADVP there/RB/S-ADVP ,/,/O open/VB/S-VP the/DT/B-NP door/NN/E-NP and/CC/O take/VB/S-VP it./PRP/S-NP When/WRB/S-ADVP you/PRP/S-NP 've/VBP/B-VP got/VBN/E-VP it/PRP/S-NP ,/,/O close/VB/S-VP the/DT/B-NP door/NN/E-NP ,/,/O carry/VBP/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP over/IN/S-PP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Ohne Satztrennung:

*Inside/IN/S-PP the/DT/B-NP ice/NN/I-NP box/NN/E-NP there/EX/S-NP is/VBZ/S-VP an/DT/B-NP orange/NN/I-NP juice/NN/E-NP so/RB/S-ADVP go/VBP/S-VP over/IN/S-PP there/RB/S-ADVP open/VB/S-VP the/DT/B-NP door/NN/E-NP and/C-C/O take/VB/S-VP it/PRP/S-NP when/WRB/S-ADVP you/PRP/S-NP 've/VBP/B-VP got/VBN/E-VP it/PRP/S-NP close/VB/S-VP the/DT/B-NP door/NN/E-NP carry/VBP/S-VP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP over/IN/S-PP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Nr. 9:

Mit Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP ,/,/O look/VBP/S-VP for/IN/S-PP the/DT/B-NP table/NN/E-NP ././O Go/VB/S-VP there/RB/S-ADVP and/CC/O take/VB/S-VP the/DT/B-NP popcorn/NN/E-NP ././O After/IN/S-PP taking/VBG/S-VP the/DT/B-NP popcorn/NN/E-NP ,/,/O come/VBN/S-VP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Ohne Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/I-NP look/NN/E-NP for/IN/S-PP the/DT/B-NP table/NN/E-NP go/VBP/S-VP there/RB/S-ADVP and/CC/O take/VB/S-VP the/DT/B-NP popcorn/NN/E-NP after/IN/S-PP taking/VBG/S-VP the/DT/B-NP popcorn/NN/E-NP come/VBN/S-VP here/RB/S-ADVP and/CC/O hand/NN/S-VP it/PRP/S-NP over/RP/S-PRT ././O*

Nr. 10:

Mit Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O see/VB/S-VP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP find/VB/E-VP a/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP ././O If/IN/S-SBAR so/RB/O ././O take/VB/S-VP it/PRP/S-NP and/CC/O bring/VB/S-VP it/PRP/S-NP to/TO/B-VP me./VB/E-VP*

Ohne Satztrennung:

*Go/VB/S-VP to/TO/S-PP the/DT/B-NP kitchen/NN/I-NP table/NN/E-NP and/CC/O see/VB/S-VP if/IN/S-SBAR you/PRP/S-NP can/MD/B-VP find/VB/E-VP a/DT/B-NP popcorn/NN/I-NP bag/NN/E-NP if/IN/S-SBAR so/RB/O take/VBP/S-VP it/PRP/S-NP and/CC/O bring/VB/S-VP it/PRP/S-NP to/TO/S-PP me/PRP/S-NP*

Nr. 11:

Mit Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP ././O go/VB/S-VP to/TO/S-PP the/DT/B-NP table/NN/E-NP ././O Grab/VB/S-VP popcorn/NN/S-NP ././O Come/VB/B-VP to/TO/I-VP me./VB/E-VP Give/VB/S-VP me/PRP/S-NP the/DT/B-NP popcorn/NN/E-NP which/WDT/S-NP is/VBZ/S-VP in/IN/S-PP your/PRP\$/B-NP hand/NN/E-NP ././O*

Ohne Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP table/NN/E-NP grab/VB/S-VP popcorn/NN/S-NP come/VBN/S-VP to/TO/S-PP me/PRP/S-NP give/VB/S-VP me/PRP/S-NP the/DT/B-NP popcorn/NN/E-NP which/WDT/S-NP is/VBZ/S-VP in/IN/S-PP your/PRP\$/B-NP hand/NN/E-NP ././O*

Nr. 12:

Mit Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP ././O go/VB/S-VP to/TO/S-PP the/DT/B-NP table/NN/E-NP ././O Take/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP ././O Go/VB/S-VP to/TO/S-PP the/DT/B-NP dishwasher/NN/E-NP ././O Open/JJ/B-NP it././E-NP Put/VB/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP into/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP and/CC/O close/RB/B-ADJP it././E-ADJP*

Ohne Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP table/NN/E-NP take/VBP/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP dishwasher/NN/E-NP open/JJ/S-ADJP it/PRP/S-NP put/VBD/S-VP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP into/IN/S-PP the/DT/B-NP dishwasher/NN/E-NP and/CC/O close/VB/S-VP it/PRP/S-NP*

Nr. 13:

Mit Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP ././O go/VBP/S-VP to/TO/S-PP the/DT/B-NP fridge/NN/E-NP ././O Look/VB/S-VP for/IN/S-PP the/DT/B-NP orange/NN/I-NP juice/NN/E-NP and/CC/O take/VB/S-VP it./PRP/S-NP Go/VB/S-VP to/TO/S-PP the/DT/B-NP table/NN/E-NP ././O Pour/VB/S-VP the/DT/B-NP juice/NN/E-NP into/IN/S-PP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP ././O Put/VB/S-VP down/RP/S-PRT the/DT/B-NP juice/NN/E-NP and/CC/O bring/VB/S-VP me/PRP/S-NP the/DT/B-NP green/JJ/I-NP cup/NN/E-NP from/IN/S-PP the/DT/B-NP table/NN/E-NP ././O*

Ohne Satztrennung:

*Okay/JJ/B-NP ARMAR/NNP/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP  
fridge/NN/I-NP look/NN/E-NP for/IN/S-PP the/DT/B-NP orange/NN/I-NP juice/NN/E-  
NP and/CC/O take/VB/S-VP it/PRP/S-NP go/VB/S-VP to/TO/S-PP the/DT/B-NP  
table/NN/E-NP pour/VBP/S-VP the/DT/B-NP juice/NN/E-NP into/IN/S-PP the/DT/B-  
NP green/JJ/I-NP cup/NN/E-NP put/VBD/S-VP down/RP/S-PRT the/DT/B-NP  
juice/NN/E-NP and/CC/O bring/VB/S-VP me/PRP/S-NP the/DT/B-NP green/JJ/I-  
NP cup/NN/E-NP from/IN/S-PP the/DT/B-NP table/NN/E-NP*

Nr. 14:

Mit Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP ,/,/O go/VB/S-VP to/TO/S-PP the/DT/B-NP  
fridge/NN/E-NP ././O Open/JJ/B-NP it././E-NP Look/VB/S-VP for/IN/S-PP the/DT/B-  
NP orange/NN/I-NP juice/NN/E-NP and/CC/O take/VB/S-VP it./PRP/S-NP Close/VB/S-  
VP the/DT/B-NP fridge/NN/E-NP and/CC/O bring/VB/S-VP the/DT/B-NP juice/NN/E-  
NP to/TO/S-PP me././S-ADJP*

Ohne Satztrennung:

*Okay/NNP/B-NP Armar/NNP/E-NP go/VBP/S-VP to/TO/S-PP the/DT/B-NP  
fridge/NN/E-NP open/JJ/S-ADJP it/PRP/S-NP look/VBP/S-VP for/IN/S-PP the/DT/B-  
NP orange/NN/I-NP juice/NN/E-NP and/CC/O take/VB/S-VP it/PRP/S-NP close/VB/S-  
VP the/DT/B-NP fridge/NN/E-NP and/CC/O bring/VB/S-VP the/DT/B-NP juice/NN/E-  
NP to/TO/S-PP me/PRP/S-NP*

## G.7. BIOS

Nr. 1:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP the/B-NP dishwasher/I-  
NP 's/B-NP door/I-NP ./O Locate/B-VP its/B-NP upper/I-NP rack/I-NP ./O Pull/B-  
VP it/B-NP out/B-PRT ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/  
O locate/B-VP the/B-NP green/I-NP cup/I-NP ./O Take/B-VP it/B-NP ./O Go/B-  
VP back/B-ADVP to/B-PP the/B-NP dishwasher/I-NP ./O Put/B-VP the/B-NP green/I-  
NP cup/I-NP in/B-PP the/B-NP rack/I-NP ./O Push/B-VP the/B-NP rack/I-NP back/B-  
ADVP and/O close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP dishwasher/I-NP open/B-VP the/B-NP dishwasher/I-NP  
's/B-NP door/I-NP locate/B-VP its/B-NP upper/I-NP rack/I-NP pull/B-VP it/B-NP  
out/B-PRT go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O locate/B-VP  
the/B-NP green/I-NP cup/I-NP take/B-VP it/B-NP go/B-VP back/B-ADVP to/B-PP  
the/B-NP dishwasher/I-NP put/B-VP the/B-NP green/I-NP cup/I-NP in/B-PP the/B-  
NP rack/I-NP push/B-VP the/B-NP rack/I-NP back/B-ADVP and/O close/B-VP the/B-  
NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Nr. 2:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP fridge/I-NP and/O open/B-VP the/B-NP door/I-NP ./O  
Locate/B-VP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Close/B-  
VP the/B-NP door/I-NP ./O Come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP  
the/B-NP orange/I-NP juice/I-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP fridge/I-NP and/O open/B-VP the/B-NP door/I-NP locate/B-  
VP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP close/B-VP the/B-NP*

door/I-NP come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP the/B-NP orange/I-NP juice/I-NP to/B-PP me/B-NP ./O

Nr. 3:

Mit Satztrennung:

Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP ./O Locate/B-VP the/B-NP popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ./O Come/B-VP over/B-PRT here/B-ADVP ./O Hand/B-VP the/B-NP popcorn/I-NP bag/I-NP over/B-PRT ./O

Ohne Satztrennung:

Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP locate/B-VP the/B-NP popcorn/I-NP bag/I-NP take/B-VP it/B-NP come/B-VP over/B-PRT here/B-ADVP hand/B-VP the/B-NP popcorn/I-NP bag/I-NP over/B-PRT ./O

Nr. 4:

Mit Satztrennung:

Okay/B-INTJ ARMAR/B-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O take/B-VP the/B-NP green/I-NP cup/I-NP from/B-PP there/B-ADVP ./O Look/B-VP out/B-PP for/B-PP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP ./O Open/B-VP its/B-NP door/I-NP ./O Pull/B-VP out/B-ADVP the/B-NP top/I-NP basket/I-NP ./O Put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./O Finally/B-ADVP close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O

Ohne Satztrennung:

Okay/B-INTJ ARMAR/B-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O take/B-VP the/B-NP green/I-NP cup/I-NP from/B-PP there/B-ADVP look/B-VP out/B-PP for/B-PP the/B-NP dishwasher/I-NP if/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP open/B-VP its/B-NP door/I-NP pull/B-VP out/B-ADVP the/B-NP top/I-NP basket/I-NP put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP finally/B-ADVP close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O

Nr. 5:

Mit Satztrennung:

Okay/B-INTJ ARMAR/B-NP look/B-VP for/B-PP the/B-NP refrigerator/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP ./O Grab/B-VP the/B-NP door/I-NP handle/I-NP and/B-VP pull/I-NP ./O Look/B-VP what/B-NP 's/B-VP inside/B-PP the/B-NP fridge/I-NP ./O There/B-NP should/B-VP be/I-NP an/B-NP orange/I-NP juice/I-NP somewhere/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/I-NP take/I-NP it/B-NP and/O close/B-VP the/B-NP door/I-NP ./O Come/B-VP over/B-PRT here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT ./O

Ohne Satztrennung:

Okay/B-INTJ ARMAR/B-NP look/B-VP for/B-PP the/B-NP refrigerator/I-NP if/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP grab/B-VP the/B-NP door/I-NP handle/I-NP and/B-VP pull/I-NP look/I-NP what/B-NP 's/B-VP inside/B-PP the/B-NP fridge/I-NP there/B-NP should/B-VP be/I-NP an/B-NP orange/I-NP juice/I-NP somewhere/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/I-NP take/I-NP it/B-NP and/O close/B-VP the/B-NP door/I-NP Come/B-VP over/B-PRT here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT

Nr. 6:

Mit Satztrennung:

*At/B-PP the/B-NP kitchen/I-NP table/I-NP there/B-NP is/B-VP a/B-NP popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ,/O come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP it/B-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*At/B-PP the/B-NP kitchen/I-NP table/I-NP there/B-NP is/B-VP a/B-NP popcorn/I-NP bag/I-NP take/B-VP it/B-NP come/B-VP over/B-PRT here/B-ADVP and/O give/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 7:

Mit Satztrennung:

*Locate/B-VP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-VP it/B-NP ,/O go/B-VP there/B-ADVP ./O When/B-ADVP you/B-NP 're/B-VP there/B-ADVP ,/O open/B-VP its/B-NP front/I-NP door/I-NP and/O pull/B-VP out/B-PRT the/B-NP upper/I-NP rack/I-NP ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP what/B-NP 's/B-VP there/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-VP a/B-NP green/I-NP cup/I-NP ,/O take/B-VP it/B-NP ./O Move/B-VP back/B-ADVP towards/B-PP the/B-NP dishwasher/I-NP and/O put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./O Finally/B-ADVP ,/O push/B-VP back/B-PP the/B-NP basket/I-NP and/O close/B-VP the/B-NP door/I-NP ./O*

Ohne Satztrennung:

*Locate/B-VP the/B-NP dishwasher/I-NP if/B-SBAR you/B-NP can/B-VP see/I-VP it/B-NP go/B-VP there/B-ADVP when/B-ADVP you/B-NP 're/B-VP there/I-VP open/I-VP its/B-NP front/I-NP door/I-NP and/O pull/B-VP out/B-PRT the/B-NP upper/I-NP rack/I-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP what/B-NP 's/B-VP there/B-ADVP if/B-SBAR you/B-NP can/B-VP see/I-VP a/B-NP green/I-NP cup/I-NP take/B-VP it/B-NP move/B-VP back/B-ADVP towards/B-PP the/B-NP dishwasher/I-NP and/O put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP finally/B-VP push/I-VP back/B-PP the/B-NP basket/I-NP and/O close/B-VP the/B-NP door/I-NP*

Nr. 8:

Mit Satztrennung:

*Inside/B-NP the/B-NP ice/I-NP box/I-NP ,/O there/B-NP is/B-VP an/B-NP orange/I-NP juice/I-NP ./O So/B-PP ,/O go/B-VP over/B-PRT there/B-ADVP ,/O open/B-VP the/B-NP door/I-NP and/O take/B-VP it/B-NP ./O When/B-ADVP you/B-NP 've/B-VP got/I-VP it/B-NP ,/O close/B-VP the/B-NP door/I-NP ,/O carry/B-VP the/B-NP orange/I-NP juice/I-NP over/B-PRT here/B-ADVP and/O hand/B-VP It/B-NP over/B-PRT ./O*

Ohne Satztrennung:

*Inside/B-NP the/B-NP ice/I-NP box/I-NP there/B-NP is/B-VP an/B-NP orange/I-NP juice/I-NP so/B-SBAR go/B-VP over/B-PRT there/B-ADVP open/B-VP the/B-NP door/I-NP and/O take/B-VP it/B-NP when/B-ADVP you/B-NP 've/B-VP got/I-VP it/B-NP close/B-VP the/B-NP door/I-NP carry/B-VP the/B-NP orange/I-NP juice/I-NP over/B-PRT here/B-ADVP and/O hand/B-VP It/B-NP over/B-PRT ./O*

Nr. 9:

Mit Satztrennung:

*Okay/B-INTJ ARMAR/B-NP ,/O look/B-VP for/B-PP the/B-NP table/I-NP ./O Go/B-VP there/B-ADVP and/O take/B-VP the/B-NP popcorn/I-NP ./O After/B-PP taking/B-*

*VP the/B-NP popcorn/I-NP ,/O come/B-VP here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT ./O*

Ohne Satztrennung:

*Okay/B-INTJ ARMAR/B-NP look/B-VP for/B-PP the/B-NP table/I-NP go/B-VP there/B-ADVP and/O take/B-VP the/B-NP popcorn/I-NP after/B-PP taking/B-VP the/B-NP popcorn/I-NP come/B-VP here/B-ADVP and/O hand/B-VP it/B-NP over/B-PRT ./O*

Nr. 10:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP if/B-SBAR you/B-NP can/B-VP find/I-VP a/B-NP popcorn/I-NP bag/I-NP ./O If/B-SBAR so/I-SBAR ,/O take/B-VP it/B-NP and/O bring/B-VP it/B-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP if/B-SBAR you/B-NP can/B-VP find/I-VP a/B-NP popcorn/I-NP bag/I-NP if/B-SBAR so/I-SBAR take/B-VP it/B-NP and/O bring/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 11:

Mit Satztrennung:

*Okay/B-INTJ Armar/B-NP ,/O go/B-VP to/B-PP the/B-NP table/I-NP ./O Grab/B-VP popcorn/B-NP ./O Come/B-VP to/B-PP me/B-NP ./O Give/B-VP me/B-NP the/B-NP popcorn/I-NP which/B-NP is/B-VP in/B-PP your/B-NP hand/I-NP ./O*

Ohne Satztrennung:

*Okay/B-INTJ Armar/B-NP go/B-VP to/B-PP the/B-NP table/I-NP grab/B-VP popcorn/B-NP come/B-VP to/B-PP me/B-NP give/B-VP me/B-NP the/B-NP popcorn/I-NP which/B-NP is/B-VP in/B-PP your/B-NP hand/I-NP ./O*

Nr. 12:

Mit Satztrennung:

*Okay/B-INTJ Armar/B-NP ,/O go/B-VP to/B-PP the/B-NP table/I-NP ./O Take/B-VP the/B-NP green/I-NP cup/I-NP ./O Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP it/B-NP ./O Put/B-VP the/B-NP green/I-NP cup/I-NP into/B-PP the/B-NP dishwasher/I-NP and/O close/B-VP it/B-NP ./O*

Ohne Satztrennung:

*Okay/B-INTJ Armar/B-NP go/B-VP to/B-PP the/B-NP table/I-NP take/B-VP the/B-NP green/I-NP cup/I-NP go/B-VP to/B-PP the/B-NP dishwasher/I-NP open/B-VP it/B-NP put/B-VP the/B-NP green/I-NP cup/I-NP into/B-PP the/B-NP dishwasher/I-NP and/O close/B-VP it/B-NP ./O*

Nr. 13:

Mit Satztrennung:

*Okay/B-INTJ ARMAR/B-NP ,/O go/B-VP to/B-PP the/B-NP fridge/I-NP ./O Look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Go/B-VP to/B-PP the/B-NP table/I-NP ./O Pour/B-VP the/B-NP juice/I-NP into/B-PP the/B-NP green/I-NP cup/I-NP ./O Put/B-VP down/B-PP the/B-NP juice/I-NP and/O bring/B-VP me/B-NP the/B-NP green/I-NP cup/I-NP from/B-PP the/B-NP table/I-NP ./O*

Ohne Satztrennung:

*Okay/B-INTJ ARMAR/B-NP go/B-VP to/B-PP the/B-NP fridge/I-NP look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP go/B-VP to/B-PP the/B-NP table/I-NP pour/B-VP the/B-NP juice/I-NP into/B-PP the/B-NP green/I-*

*NP cup/I-NP put/B-VP down/B-PP the/B-NP juice/I-NP and/O bring/B-VP me/B-NP the/B-NP green/I-NP cup/I-NP from/B-PP the/B-NP table/I-NP ./O*

Nr. 14:

Mit Satztrennung:

*Okay/B-INTJ Armar/B-NP ./O go/B-VP to/B-PP the/B-NP fridge/I-NP ./O Open/B-VP it/B-NP ./O Look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Close/B-VP the/B-NP fridge/I-NP and/O bring/B-VP the/B-NP juice/I-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Okay/B-INTJ Armar/B-NP go/B-VP to/B-PP the/B-NP fridge/I-NP open/B-VP it/B-NP look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP close/B-VP the/B-NP fridge/I-NP and/O bring/B-VP the/B-NP juice/I-NP to/B-PP me/B-NP ./O*

## G.8. OpenNLP Phrasenerkennung

Nr. 1:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O Locate/B-VP its/B-NP upper/I-NP rack/I-NP ./O Pull/B-VP it/I-NP out/B-PPT ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O locate/B-VP the/B-NP green/I-NP cup/I-NP ./O Take/B-VP it/B-NP ./O Go/B-VP back/B-ADVP to/B-PP the/B-NP dishwasher/I-NP ./O Put/B-VP the/B-NP green/I-NP cup/I-NP in/B-PP the/B-NP rack/I-NP ./O Push/B-VP the/B-NP rack/I-NP back/B-ADVP and/O close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP dishwasher/I-NP open/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP locate/B-VP its/B-NP upper/I-NP rack/I-NP pull/B-VP it/B-NP out/B-VP go/I-NP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O locate/B-VP the/B-NP green/I-NP cup/I-NP take/B-VP it/B-NP go/B-VP back/B-ADVP to/B-PP the/B-NP dishwasher/I-NP put/B-VP the/B-NP green/I-NP cup/I-NP in/B-PP the/B-NP rack/I-NP push/B-VP the/B-NP rack/I-NP back/B-ADVP and/O close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Nr. 2:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP fridge/I-NP and/O open/B-VP the/B-NP door/I-NP ./O Locate/B-VP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Close/B-VP the/B-NP door/I-NP ./O Come/B-VP over/I-NP here/B-ADVP and/O give/B-VP the/B-NP orange/I-NP juice/I-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP fridge/I-NP and/O open/B-VP the/B-NP door/I-NP locate/B-VP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP close/B-VP the/B-NP door/I-NP come/B-VP over/O here/B-ADVP and/O give/B-VP the/B-NP orange/I-NP juice/I-NP to/B-PP me/B-NP ./O*

Nr. 3:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP ./B-VP Locate/I-NP the/B-NP*

*popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ./O Come/B-VP over/I-NP here/B-ADVP ./O Hand/B-VP the/B-NP popcorn/I-NP bag/I-NP over/I-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP locate/B-VP the/B-NP popcorn/I-NP bag/I-NP take/B-VP it/B-NP come/B-VP over/B-ADJP here/B-ADVP hand/B-VP the/B-NP popcorn/I-NP bag/I-NP over/I-NP ./O*

Nr. 4:

Mit Satztrennung:

*Okay/B-NP ARMAR/I-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O take/B-VP the/B-NP green/I-NP cup/I-NP from/B-PP there/B-NP ./O Look/B-VP out/B-PRT for/B-PP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-ADVP there/B-ADVP ./O Open/B-VP its/B-NP door/I-NP ./O Pull/B-VP out/B-ADVP the/B-NP top/I-NP basket/I-NP ./O Put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./B-VP Finally/I-NP close/I-NP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Ohne Satztrennung:

*Okay/B-NP ARMAR/I-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O take/B-VP the/B-NP green/I-NP cup/I-NP from/B-PP there/B-NP look/B-VP out/B-ADVP for/B-PP the/B-NP dishwasher/I-NP if/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-ADVP open/B-VP its/B-NP door/I-NP pull/B-VP out/B-ADVP the/B-NP top/I-NP basket/I-NP put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP finally/B-ADVP close/B-VP the/B-NP dishwasher/I-NP 's/B-NP door/I-NP ./O*

Nr. 5:

Mit Satztrennung:

*Okay/B-NP ARMAR/I-NP look/B-VP for/B-PP the/B-NP refrigerator/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-ADVP there/I-ADVP ./O Grab/B-VP the/B-NP door/I-NP handle/I-NP and/O pull/B-VP ./O Look/B-VP what/B-NP 's/B-VP inside/B-PP the/B-NP fridge/I-NP ./O There/B-NP should/B-VP be/I-NP an/B-NP orange/I-NP juice/I-NP somewhere/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/I-NP take/I-NP it/B-NP and/O close/B-VP the/B-NP door/I-NP ./O Come/B-VP over/I-NP here/B-ADVP and/O hand/B-VP it/B-NP over/B-VP ./O*

Ohne Satztrennung:

*Okay/B-NP ARMAR/I-NP look/B-VP for/B-PP the/B-NP refrigerator/I-NP if/B-SBAR you/B-NP can/B-VP see/I-NP it/B-NP go/B-VP over/B-PRT there/B-NP grab/B-VP the/B-NP door/I-NP handle/I-NP and/O pull/B-VP look/I-NP what/B-NP 's/B-VP inside/B-PP the/B-NP fridge/I-NP there/B-NP should/B-VP be/I-NP an/B-NP orange/I-NP juice/I-NP somewhere/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-NP it/I-NP take/I-NP it/B-NP and/O close/B-VP the/B-NP door/I-NP Come/B-VP over/I-NP here/B-ADVP and/O hand/B-VP it/B-NP over/O*

Nr. 6:

Mit Satztrennung:

*At/B-PP the/B-NP kitchen/I-NP table/I-NP there/B-ADVP is/B-VP a/B-NP popcorn/I-NP bag/I-NP ./O Take/B-VP it/B-NP ,/O come/B-VP over/B-ADVP here/B-ADVP and/O give/B-VP it/B-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*At/B-PP the/B-NP kitchen/I-NP table/I-NP there/B-ADVP is/B-VP a/B-NP popcorn/I-*

*NP bag/I-NP take/B-VP it/B-NP come/B-VP over/B-ADVP here/B-ADVP and/O give/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 7:

Mit Satztrennung:

*Locate/B-VP the/B-NP dishwasher/I-NP ./O If/B-SBAR you/B-NP can/B-VP see/I-VP it/B-NP ,/O go/B-VP there/B-ADVP ./O When/B-ADVP you/B-NP 're/B-VP there/B-ADVP ,/O open/B-VP its/B-NP front/I-NP door/I-NP and/O pull/B-VP out/B-PRT the/B-NP upper/I-NP rack/I-NP ./O Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP what/B-NP 's/B-VP there/B-ADVP ./O If/B-SBAR you/B-NP can/B-VP see/I-VP a/B-NP green/I-NP cup/I-NP ,/O take/B-VP it/B-NP ./O Move/B-VP back/B-ADVP towards/B-PP the/B-NP dishwasher/I-NP and/O put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP ./O Finally/B-ADVP ,/O push/B-VP back/B-SBAR the/B-NP basket/I-NP and/O close/B-VP the/B-NP door/I-NP ./O*

Ohne Satztrennung:

*Locate/B-VP the/B-NP dishwasher/I-NP if/B-SBAR you/B-NP can/B-VP see/I-VP it/B-NP go/B-VP there/B-ADVP when/B-ADVP you/B-NP 're/B-VP there/I-VP open/I-VP its/B-NP front/I-NP door/I-NP and/O pull/B-VP out/B-PRT the/B-NP upper/I-NP rack/I-NP go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP what/B-NP 's/B-VP there/B-ADVP if/B-SBAR you/B-NP can/B-VP see/I-VP a/B-NP green/I-NP cup/I-NP take/B-VP it/B-NP move/B-VP back/B-ADVP towards/B-PP the/B-NP dishwasher/I-NP and/O put/B-VP the/B-NP cup/I-NP into/B-PP it/B-NP finally/B-ADVP push/B-VP back/B-SBAR the/B-NP basket/I-NP and/O close/B-VP the/B-NP door/I-NP*

Nr. 8:

Mit Satztrennung:

*Inside/B-NP the/B-NP ice/I-NP box/I-NP ,/O there/B-NP is/B-VP an/B-NP orange/I-NP juice/I-NP ./O So/O ,/O go/B-VP over/B-ADVP there/I-ADVP ,/O open/B-VP the/B-NP door/I-NP and/O take/B-VP it/B-NP ./O When/B-ADVP you/B-NP 've/B-VP got/I-VP it/B-NP ,/O close/B-VP the/B-NP door/I-NP ,/O carry/B-VP the/B-NP orange/I-NP juice/I-NP over/B-VP here/B-ADVP and/O hand/B-VP It/B-NP over/B-VP ./O*

Ohne Satztrennung:

*Inside/B-NP the/B-NP ice/I-NP box/I-NP there/B-NP is/B-VP an/B-NP orange/I-NP juice/I-NP so/B-PP go/B-NP over/B-VP there/I-VP open/I-VP the/B-NP door/I-NP and/O take/B-VP it/B-NP when/B-ADVP you/B-NP 've/B-VP got/I-VP it/B-NP close/B-VP the/B-NP door/I-NP carry/B-VP the/B-NP orange/I-NP juice/I-NP over/B-VP here/B-ADVP and/O hand/B-VP It/B-NP over/B-VP ./O*

Nr. 9:

Mit Satztrennung:

*Okay/B-NP ARMAR/I-NP ,/O look/B-VP for/B-PP the/B-NP table/I-NP ./O Go/B-VP there/B-ADVP and/O take/B-VP the/B-NP popcorn/I-NP ./O After/B-PP taking/B-VP the/B-NP popcorn/I-NP ,/O come/B-VP here/B-ADVP and/O hand/B-VP it/B-NP over/B-VP ./O*

Ohne Satztrennung:

*Okay/B-NP ARMAR/I-NP look/B-VP for/B-PP the/B-NP table/I-NP go/B-VP there/B-ADVP and/O take/B-VP the/B-NP popcorn/I-NP after/B-PP taking/B-VP the/B-NP popcorn/I-NP come/B-VP here/B-ADVP and/O hand/B-VP it/B-NP over/B-VP ./O*

Nr. 10:

Mit Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP if/B-SBAR you/B-NP can/B-VP find/I-NP a/B-NP popcorn/I-NP bag/I-NP ./O If/B-ADVP so/I-ADVP ,/O take/B-VP it/B-NP and/O bring/B-VP it/B-NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Go/B-VP to/B-PP the/B-NP kitchen/I-NP table/I-NP and/O see/B-VP if/B-SBAR you/B-NP can/B-VP find/I-NP a/B-NP popcorn/I-NP bag/I-NP if/B-PP so/B-PP take/B-NP it/B-NP and/O bring/B-VP it/B-NP to/B-PP me/B-NP ./O*

Nr. 11:

Mit Satztrennung:

*Okay/B-NP Armar/I-NP ,/O go/B-VP to/B-PP the/B-NP table/I-NP ./B-VP Grab/I-NP popcorn/I-NP ./O Come/B-VP to/B-PP me/B-NP ./O Give/B-VP me/B-NP the/B-NP popcorn/I-NP which/B-NP is/B-VP in/B-PP your/B-NP hand/I-NP ./O*

Ohne Satztrennung:

*Okay/B-NP Armar/I-NP go/B-VP to/B-PP the/B-NP table/I-NP grab/I-NP popcorn/I-NP come/B-VP to/B-PP me/B-NP give/B-VP me/B-NP the/B-NP popcorn/I-NP which/B-NP is/B-VP in/B-PP your/B-NP hand/I-NP ./O*

Nr. 12:

Mit Satztrennung:

*Okay/B-NP Armar/I-NP ,/O go/B-VP to/B-PP the/B-NP table/I-NP ./B-VP Take/I-NP the/B-NP green/I-NP cup/I-NP ./O Go/B-VP to/B-PP the/B-NP dishwasher/I-NP ./O Open/B-VP it/B-NP ./O Put/B-VP the/B-NP green/I-NP cup/I-NP into/B-PP the/B-NP dishwasher/I-NP and/O close/B-VP it/B-NP ./O*

Ohne Satztrennung:

*Okay/B-NP Armar/I-NP go/B-VP to/B-PP the/B-NP table/I-NP take/B-VP the/B-NP green/I-NP cup/I-NP go/B-VP to/B-PP the/B-NP dishwasher/I-NP open/B-VP it/B-NP put/B-VP the/B-NP green/I-NP cup/I-NP into/B-PP the/B-NP dishwasher/I-NP and/O close/B-VP it/B-NP ./O*

Nr. 13:

Mit Satztrennung:

*Okay/B-NP ARMAR/I-NP ,/O go/B-VP to/B-PP the/B-NP fridge/I-NP ./B-VP Look/I-NP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP ./O Go/B-VP to/B-PP the/B-NP table/I-NP ./B-VP Pour/I-NP the/B-NP juice/I-NP into/B-PP the/B-NP green/I-NP cup/I-NP ./O Put/B-VP down/B-PP the/B-NP juice/I-NP and/O bring/B-VP me/B-NP the/B-NP green/I-NP cup/I-NP from/B-PP the/B-NP table/I-NP ./O*

Ohne Satztrennung:

*Okay/B-NP ARMAR/I-NP go/B-VP to/B-PP the/B-NP fridge/I-NP look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP go/B-VP to/B-PP the/B-NP table/I-NP pour/B-VP the/B-NP juice/I-NP into/B-PP the/B-NP green/I-NP cup/I-NP put/B-VP down/B-PP the/B-NP juice/I-NP and/O bring/B-VP me/B-NP the/B-NP green/I-NP cup/I-NP from/B-PP the/B-NP table/I-NP ./O*

Nr. 14:

Mit Satztrennung:

*Okay/B-NP Armar/I-NP ,/O go/B-VP to/B-PP the/B-NP fridge/I-NP ./O Open/B-VP*

*it/B-NP ./O Look/B-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP  
it/B-NP ./O Close/B-VP the/B-NP fridge/I-NP and/O bring/B-VP the/B-NP juice/I-  
NP to/B-PP me/B-NP ./O*

Ohne Satztrennung:

*Okay/B-NP Armar/I-NP go/B-VP to/B-PP the/B-NP fridge/I-NP open/B-VP it/I-VP  
look/I-VP for/B-PP the/B-NP orange/I-NP juice/I-NP and/O take/B-VP it/B-NP close/B-  
VP the/B-NP fridge/I-NP and/O bring/B-VP the/B-NP juice/I-NP to/B-PP me/B-NP  
./O*

## H. Syntaxbäume

### H.1. SENNA

Nr. 1:

*(S1(VP(VP VB (PP TO (NP DT NN)) (VP VB (NP(NP DT NN POS) NN) (VP VB  
(NP PRP\$ JJ NN) (VP VB (NP PRP) (ADVP RP) (VP VB (PP TO (NP DT NN  
NN)))))) CC (VP VB (NP DT JJ NN)) (VP VBP (S(NP PRP) (VP VB (ADVP RB)  
(PP TO (NP DT NN)))))) (VP VBD (S(NP(NP DT JJ NN) (PP IN (NP DT NN)))  
(VP(VP VB (NP DT NN) (ADVP RB)) CC (VP VB (NP(NP DT NN POS) NN))))))*

Nr. 2:

*(S1(VP(VP VB (PP TO (NP DT NN))) CC (VP VB (NP DT NN)) (VP VB (NP DT  
NN NN)) CC (VP VB (S(NP PRP) (VP VB (S(NP DT NN) (VP(VP VB (ADVP RB)  
(ADVP RB)) CC (VP VB (NP DT NN NN) (PP TO (NP PRP))))))))))*

Nr. 3:

*(S1(S(VP VB (PP TO (NP DT NN NN)) (VP VB (SBAR(S(NP DT DT NN NN) (VP  
VBP (SBAR(S(NP PRP) (VP(VP VB (ADVP RP) (ADVP RB)) (VP VBP) (SBAR(S(NP  
DT NN NN) (ADVP IN)))))))))) .))*

Nr. 4:

*(S1(S(NP JJ NNP) (VP VBP (PP TO (NP DT NN NN))) CC (VP VB (NP(NP DT JJ  
NN) (PP IN (NP EX)))) VBP (ADVP RP) (PP IN (NP DT NN)) IN (NP PRP) MD  
VB (NP PRP) VB RP (NP RB) VB (NP PRP\$ NN) VB (ADVP RP) (NP DT JJ NN)  
VBD (NP DT NN) (PP IN (NP PRP)) (ADVP RB) (VP VB (NP(NP DT NN POS)  
NN)) .))*

Nr. 5:

*(S1(NP(NP(ADJP JJ NNP) NN) (PP IN (NP DT NN)) (SBAR IN (S(NP PRP) (VP  
MD (SBAR VB (S(NP PRP) (VP VB (ADVP RP) (ADVP RB) (SBAR VB (S(NP DT  
NN) (VP VB CC VB VB (SBAR(S(SBAR(WHNP WP) (S(VP VBZ (PP IN (NP DT  
NN)))))) (NP EX) (VP MD (VP VB (ADVP(NP DT NN NN) RB) (SBAR IN (S(NP  
PRP) (VP MD (VP VB (S(NP PRP) (VP(VP VB (NP PRP)) CC (VP VB (S(NP DT  
NN) (VP VB (UCP(ADVP RB) (ADVP RB) CC (NP(NP(NP NN) (NP PRP)) (ADVP  
RP)))))))))))))))))) .))*

Nr. 6:

*(S1(S(PP IN (NP DT NN NN)) (NP EX) (VP VBZ (SBAR(S(NP DT NN NN) (VP VBP  
(SBAR(S(NP PRP) (VP(VP VB (ADVP RP) (ADVP RB)) CC (VP VB (NP PRP) (PP*

*TO (NP PRP))))))))))*

Nr. 7:

*(S1(S(S(VP VB (NP DT NN))) IN (NP PRP) MD VB (NP PRP) VB (ADVP RB) (WH-ADVP WRB) (NP PRP) VBP (ADVP RB) (VP VB (NP PRP\$ JJ NN)) CC VB (ADVP RP) (NP DT JJ NN) (VP VB (PP TO (NP DT NN NN))) CC VB (WHNP WP) VBZ (ADVP RB) IN (NP PRP) MD VB (NP DT JJ NN) VBP (NP PRP) (VP VB (ADVP RB) (PP IN (NP DT NN))) CC VBD (NP DT NN) (PP IN (NP PRP)) (ADVP RB) (VP VB (ADVP RP) (NP DT NN)) CC (VP VB (NP DT NN)) .))*

Nr. 8:

*(S1(S(PP IN (NP DT NN NN)) (NP EX) VBZ (NP DT NN NN) (ADVP RB) VBP (ADVP IN RB) (VP VB (NP DT NN)) CC (VP VB (NP PRP) (SBAR(WHADVP WRB) (S(NP PRP) (VP VBP (VP VBN (S(NP PRP) (VP VB (NP DT NN)))))))))) (VP VBP (NP DT NN NN) (ADVP IN RB)) CC (NP(NP(NP NN) (NP PRP)) (ADVP RP)) .))*

Nr. 9:

*(S1(S(NP(NP(ADJP JJ NNP) NN) (PP IN (NP DT NN))) (VP(VP VBP (ADVP RB)) CC (VP VB (NP DT NN) (PP IN (S(VP VBG (S(NP DT NN) (VP VBN (ADVP RB))))))) CC (VP(NP(NP NN) (NP PRP)) (ADVP RP))) .))*

Nr. 10:

*(S1(S(S(VP VB (PP TO (NP DT NN NN)) (VP VB (SBAR(S(NP DT DT NN NN) (VP VBP (SBAR(S(NP PRP) (VP(VP VB (ADVP RP) (ADVP RB)) (VP VBP) (SBAR(S(NP DT NN NN) (ADVP IN)))))))))) .))*

Nr. 11:

*(S1(S(NP NNP NNP) (VP VBP (S(PP TO (NP DT NN)) (VP VB (S(NP NN) (VP VBN (PP TO (NP PRP)) (VP VB (NP PRP) (NP(NP DT NN) (SBAR(WHNP WDT) (S(VP VBZ (PP IN (NP PRP\$ NN)))))))))) .))*

Nr. 12:

*(S1(S(S(NP NNP NNP) (VP VBP (PP TO (NP DT NN))) (VP VBP (SBAR(S(NP DT JJ NN) (VP VBP (PP TO (NP DT NN)) (ADJP JJ) (SBAR(S(NP PRP) (VP(VP VBD (NP DT JJ NN) (PP IN (NP DT NN))) CC (VP VB (NP PRP))))))))))*

Nr. 13:

*(S1(S(NP JJ NNP) (VP(VP VBP (PP TO (NP(NP DT NN NN) (PP IN (NP DT NN NN)))))) CC (VP VB (NP PRP) (VP VB (PP TO (NP DT NN)))) (VP(VP VBP (NP(NP DT NN) (PP IN (NP DT JJ NN))) (VP(VP VBD (ADVP RP) (NP DT NN)) CC (VP VB (NP PRP) (NP(NP DT JJ NN) (PP IN (NP DT NN))))))*

Nr. 14:

*(S1(S(NP NNP NNP) (VP VBP (PP TO (NP DT NN)) (PP(ADJP JJ) (SBAR(S(NP PRP) (VP(VP VBP (PP IN (NP DT NN NN))) CC (VP VB (SBAR(S(NP PRP) (VP(VP VB (NP DT NN) CC (VP VB (NP DT NN) (PP TO (NP PRP))))))))))*

## H.2. Charniak

Nr. 1:

*(S1 (S (VP (VB Go) (PP (TO to) (NP (DT the) (JJ dishwasher) (JJ open) (NP (DT the) (NN dishwasher) (POS 's) (NN door)))) (S (VP (VB locate) (S (NP (PRP\$ its) (JJ upper) (NN rack)) (VP (VB pull) (NP (PRP it)) (ADVP (RB out)) (S (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VB locate) (S (NP (DT the) (JJ green) (NN cup)) (VP (VB take) (S (NP (PRP it)) (VP (VB go) (ADVP (RB back) (PP (TO to) (NP (NP (DT the) (NN dishwasher)) (VP (VBD put) (S (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN in) (NP (DT the) (NN rack)))) (VP (VP (VB push) (NP (DT the) (NN rack)) (ADVP (RB back))) (CC and) (VP (VB close) (NP (NP (DT the) (NN dishwasher) (POS 's) (NN door))))))))))))))))))))))))))))))))))*

Nr. 2:

*(S1 (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN fridge)))) (CC and) (VP (VB open) (NP (DT the) (NN door))) (VP (VB locate) (NP (DT the) (NN orange) (NN juice))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VB close) (NP (DT the) (NN door)) (VP (VP (VB come) (ADVP (RB over) (RB here))) (CC and) (VP (VB give) (NP (DT the) (NN orange) (NN juice)) (PP (TO to) (NP (PRP me))))))))))))))*

Nr. 3:

*(S1 (S (VP (VB Go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table))) (S (VP (VB locate) (NP (DT the) (DT the) (JJ popcorn) (NN bag)) (VP (VB take) (SBAR (S (NP (PRP it)) (VP (VB come) (PRT (RP over)) (ADVP (RB here)) (VP (VB hand) (NP (DT the) (JJ popcorn) (NN bag)) (PP (IN over))))))))))))))*

Nr. 4:

*(S1 (S (INTJ (UH Okay)) (VP (VB ARMAR) (S (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VB take) (S (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN from) (NP (EX there)))) (VP (VB look) (PRT (RP out)) (PP (IN for) (NP (DT the) (NN dishwasher))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VB go) (PP (IN over) (NP (RB there)))) (S (VP (VB open) (NP (PRP\$ its) (NN door)) (S (VP (VB pull) (PRT (RP out)) (S (NP (DT the) (JJ top) (NN basket)) (VP (VB put) (NP (DT the) (NN cup)) (PP (IN into) (S (NP (PRP it)) (ADVP (RB finally)) (VP (VB close) (NP (NP (DT the) (NN dishwasher) (POS 's) (NN door))))))))))))))))))))))))))))))))))*

Nr. 5:

*(S1 (S (INTJ (UH Okay)) (NP (NNP ARMAR)) (VP (VB look) (PP (IN for) (NP (DT the) (NN refrigerator))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VB go) (PRT (RP over)) (S (NP (EX there)) (VP (VB grab) (NP (DT the) (NN door)) (VP (VP (VB handle)) (CC and) (VP (VB pull) (S (VP (VB look) (SBAR (WHNP (WP what)) (S (VP (AUX 's) (PP (IN inside) (NP (NP (DT the) (NN fridge)) (SBAR (S (NP (EX there)) (VP (MD should) (VP (AUX be) (NP (DT an) (NN orange) (NN juice)) (SBAR (RB somewhere) (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VP (VB take) (NP (PRP it))) (CC and) (VP (VB close) (NP (DT the) (NN door))) (VP (VB come) (ADVP (IN over) (RB here))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over))))))))))))))))))))))))))))))))))*

Nr. 6:

*(S1 (S (PP (IN At) (NP (DT the) (NN kitchen) (NN table))) (S (NP (EX there)) (VP*

(AUX is) (NP (DT a) (JJ popcorn) (NN bag)))) (VP (VB take) (NP (PRP it)) (VP (VP (VB come) (PRT (RP over)) (ADVP (RB here)))) (CC and) (VP (VB give) (NP (PRP it)) (PP (TO to) (NP (PRP me))))))))))

Nr. 7:

(S1 (S (VP (VB Locate) (NP (DT the) (NN dishwasher)) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VP (VB go) (ADVP (RB there)) (SBAR (WHADVP (WRB when)) (S (NP (PRP you)) (VP (AUX 're) (ADVP (RB there)) (VP (VB open) (NP (PRP\$ its) (JJ front) (NN door)))))))) (CC and) (VP (VB pull) (PRT (RP out)) (S (NP (DT the) (JJ upper) (NN rack)) (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VBP see) (SBAR (WHNP (WP what)) (S (VP (AUX 's) (ADVP (RB there)) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (DT a) (JJ green) (NN cup)) (VP (VB take) (S (NP (PRP it)) (VP (VP (VB move) (PRT (RP back)) (PP (IN towards) (NP (DT the) (NN dishwasher)))) (CC and) (VP (VB put) (NP (DT the) (NN cup)) (SBAR (IN into) (S (NP (PRP it)) (ADVP (RB finally)) (VP (VP (VB push) (PRT (RP back)) (NP (DT the) (NN basket)))) (CC and) (VP (VB close) (NP (DT the) (NN door))))))))))))))))))))))))))

Nr. 8:

(S1 (S (PP (IN Inside) (NP (NP (DT the) (NN ice) (NN box)) (SBAR (S (NP (EX there)) (VP (AUX is) (NP (DT an) (NN orange) (NN juice)))))) (RB so) (VP (VP (VB go) (PRT (RP over)) (ADVP (RB there)) (VP (VB open) (NP (DT the) (NN door)))) (CC and) (VP (VB take) (NP (PRP it)) (SBAR (WHADVP (WRB when)) (S (NP (PRP you)) (VP (AUX 've) (VP (VBD got) (S (NP (PRP it)) (VP (VB close) (NP (DT the) (NN door)) (S (VP (VP (VB carry) (NP (DT the) (NN orange) (NN juice)) (ADVP (IN over) (RB here))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over))))))))))))))))))

Nr. 9:

(S1 (S (NP (JJ Okay) (NN ARMAR)) (VP (VB look) (PP (IN for) (NP (DT the) (NN table)))) (S (VP (VP (VB go) (ADVP (RB there))) (CC and) (VP (VB take) (NP (DT the) (NN popcorn)) (PP (IN after) (S (VP (VBG taking) (S (NP (DT the) (NN popcorn)) (VP (VP (VB come) (ADVP (RB here))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over))))))))))))))

Nr. 10:

(S1 (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VB see) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB find) (NP (DT a) (JJ popcorn) (NN bag)) (SBAR (IN if) (S (ADVP (RB so)) (VP (VP (VB take) (NP (PRP it)) (CC and) (VP (VB bring) (NP (PRP it)) (PP (TO to) (NP (PRP me))))))))))))))

Nr. 11:

(S1 (S (INTJ (UH Okay)) (NP (NNP Armar)) (, ,) (VP (VB go) (PP (TO to) (NP (DT the) (NN table))) (S (NP (NN grab) (NN popcorn)) (VP (VBN come) (PP (TO to) (S (NP (PRP me)) (VP (VB give) (NP (PRP me)) (NP (NP (DT the) (NN popcorn)) (SBAR (WHNP (WDT which)) (S (VP (AUX is) (PP (IN in) (NP (PRP\$ your) (NN hand))))))))))))))

Nr. 12:

(S1 (S (NP (JJ Okay) (NNP Armar)) (, ,) (VP (VB go) (PP (TO to) (NP (DT the) (NN

table))) (S (VP (VB take) (S (NP (DT the) (JJ green) (NN cup)) (VP (VB go) (PP (TO to) (NP (DT the) (NN dishwasher) (NN open)))) (S (NP (PRP it)) (VP (VP (VB put) (NP (DT the) (JJ green) (NN cup)) (PP (IN into) (NP (DT the) (NN dishwasher)))) (CC and) (VP (VB close) (NP (PRP it))))))))))

Nr. 13:

(S1 (S (INTJ (UH Okay)) (NP (NNP ARMAR)) (VP (VP (VB go) (PP (TO to) (NP (NP (DT the) (JJ fridge) (NN look)) (PP (IN for) (NP (DT the) (NN orange) (NN juice)))))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VB go) (PP (TO to) (NP (DT the) (NN table)))) (S (VP (VB pour) (NP (DT the) (NN juice)) (PP (IN into) (NP (DT the) (JJ green) (NN cup)))) (VP (VP (VB put) (PRT (RP down)) (NP (DT the) (NN juice))) (CC and) (VP (VB bring) (NP (PRP me)) (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN from) (NP (DT the) (NN table))))))))))

Nr. 14:

(S1 (S (INTJ (UH Okay)) (NP (NNP Armar)) (VP (VB go) (PP (TO to) (NP (DT the) (NN fridge)))) (S (VP (VP (VB open) (S (NP (PRP it)) (VP (VB look) (PP (IN for) (NP (DT the) (NN orange) (NN juice)))))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VP (VB close) (NP (DT the) (NN fridge))) (CC and) (VP (VB bring) (NP (DT the) (NN juice)) (PP (TO to) (NP (PRP me))))))))))

### H.3. Stanford

Nr. 1:

(ROOT (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN dishwasher)))) (S (VP (VB open) (SBAR (S (NP (NP (DT the) (NN dishwasher) (POS 's)) (NN door)) (VP (VBP locate) (SBAR (S (NP (PRP\$ its) (JJ upper) (NN rack)) (VP (VBP pull) (SBAR (S (NP (PRP it)) (ADVP (RB out)) (VP (VBP go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))))))))) (CC and) (VP (VB locate) (S (NP (DT the) (JJ green) (NN cup)) (VP (VB take) (S (NP (PRP it)) (VP (VB go) (ADVP (RB back)) (PP (TO to) (NP (NP (DT the) (NN dishwasher)) (VP (VBN put) (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN in) (NP (DT the) (NN rack) (NN push)))) (ADVP (NP (DT the) (NN rack)) (RB back)))))) (CC and) (VP (VB close) (NP (NP (DT the) (NN dishwasher) (POS 's)) (NN door))))))

Nr. 2:

(ROOT (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN fridge)))) (CC and) (VP (VB open) (S (NP (DT the) (NN door)) (VP (VB locate) (NP (DT the) (JJ orange) (NN juice)))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VB close) (S (NP (DT the) (NN door)) (VP (VP (VB come) (PRT (RP over)) (ADVP (RB here))) (CC and) (VP (VB give) (NP (DT the) (JJ orange) (NN juice)) (PP (TO to) (NP (PRP me))))))))))

Nr. 3:

(ROOT (S (NP (NP (NNP Go)) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (VP (VBP locate) (SBAR (S (NP (DT the) (DT the) (NN popcorn) (NN bag)) (VP (VBP take) (SBAR (S (NP (PRP it)) (VP (VBP come) (PRT (RP over)) (S (ADVP (RB here)) (VP (VB hand) (NP (DT the) (NN popcorn) (NN bag)) (PRT (RP over)))))))))) (. .)))

Nr. 4:

(ROOT (S (NP (NNP Okay) (NNP ARMAR)) (VP (VP (VB go) (PP (TO to) (NP (DT

*the* (NN kitchen) (NN table)))) (CC and) (VP (VB take) (NP (DT the) (JJ green) (NN cup)) (PP (IN from) (S (NP (EX there)) (VP (VB look) (PRT (RP out)) (PP (IN for) (NP (DT the) (NN dishwasher)))))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (SBAR (S (NP (PRP it)) (VP (VBP go) (PRT (RP over)) (S (NP (EX there)) (VP (VB open) (S (NP (PRP\$ its) (NN door)) (VP (VB pull) (PRT (RP out)) (NP (DT the) (JJ top) (NN basket)) (S (VP (VB put) (NP (DT the) (NN cup)) (PP (IN into) (NP (PRP it)) (ADVP (RB finally)) (VP (VB close) (NP (NP (DT the) (NN dishwasher) (POS 's) (NN door)))))))))))))) ( . .)))

Nr. 5:

(ROOT (S (S (INTJ (UH Okay)) (NP (NNP ARMAR)) (VP (VBP look) (PP (IN for) (NP (DT the) (NN refrigerator))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VB go) (PRT (RP over)))))))))) (NP (EX there)) (VP (VBP grab) (S (NP (DT the) (NN door)) (VP (VP (VB handle) (CC and) (VB pull) (NP (NP (NN look)) (SBAR (WHNP (WP what)) (S (VP (VBZ 's) (ADVP (RB inside)) (NP (NP (DT the) (NN fridge)) (SBAR (S (NP (EX there)) (VP (MD should) (VP (VB be) (NP (DT an) (JJ orange) (NN juice)) (SBAR (ADVP (RB somewhere)) (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VB take) (NP (PRP it)))))))))))))) (CC and) (VP (VB close) (NP (NP (DT the) (NN door)) (VP (VBN come) (PRT (RP over)) (ADVP (RB here)))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over)))) ( . .)))

Nr. 6:

(ROOT (S (PP (IN At) (NP (DT the) (NN kitchen) (NN table))) (NP (EX there)) (VP (VBZ is) (S (NP (DT a) (NN popcorn) (NN bag)) (VP (VB take) (S (NP (PRP it)) (VP (VP (VB come) (PRT (RP over)) (ADVP (RB here))) (CC and) (VP (VB give) (NP (PRP it)) (PP (TO to) (NP (PRP me)))))))) ( . .)))

Nr. 7:

(ROOT (S (VP (VB Locate) (NP (DT the) (NN dishwasher)) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VP (VB go) (ADVP (RB there)) (SBAR (WHADVP (WRB when)) (S (NP (PRP you)) (VP (VBP 're) (ADVP (RB there)) (VP (JJ open) (NP (PRP\$ its) (JJ front) (NN door)))))) (CC and) (VP (VB pull) (PRT (RP out)) (NP (DT the) (JJ upper) (NN rack)) (S (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table))) (CC and) (VP (VB see) (SBAR (WHNP (WP what)) (S (VP (VBZ 's) (ADVP (RB there)) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (DT a) (JJ green) (NN cup)) (VP (VB take) (S (NP (PRP it)) (VP (VB move) (PRT (RP back)) (PP (IN towards) (NP (DT the) (NN dishwasher)))))))))) (CC and) (VP (VB put) (NP (DT the) (NN cup)) (PP (IN into) (S (NP (PRP it)) (VP (ADVP (RB finally)) (VP (VB push) (PRT (RP back)) (NP (DT the) (NN basket)))))) (CC and) (VP (VB close) (NP (DT the) (NN door)))))) ( . .)))

Nr. 8:

(ROOT (S (PP (IN Inside) (NP (DT the) (NN ice) (NN box))) (NP (EX there)) (VP (VBZ is) (NP (NP (DT an) (JJ orange) (NN juice)) (SBAR (IN so) (S (VP (VBP go) (PRT (RP over)) (S (NP (EX there)) (VP (VP (VB open) (NP (DT the) (NN door)) (CC and) (VP (VB take) (NP (PRP it)) (SBAR (WHADVP (WRB when)) (S (NP (PRP you)) (VP (VBP 've) (VP (VBN got) (S (NP (PRP it)) (VP (VB close) (S (NP (DT the) (NN door)) (VP (VB carry) (NP (DT the) (JJ orange) (NN juice)) (PP (IN over) (NP

(RB here)))))))))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over))))))))))  
 (. .))

Nr. 9:

(ROOT (S (NP (NNP Okay) (NNP ARMAR)) (VP (VBP look) (SBAR (IN for) (S (NP (DT the) (NN table)) (VP (VP (VB go) (ADVP (RB there))) (CC and) (VP (VB take) (NP (DT the) (NN popcorn)) (PP (IN after) (S (VP (VBG taking) (S (NP (DT the) (NN popcorn)) (VP (VB come) (ADVP (RB here)))))))))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over)))))))))) (. .))

Nr. 10:

(ROOT (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VB see) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB find) (NP (DT a) (NN popcorn) (NN bag)) (SBAR (IN if) (S (ADVP (RB so)) (VP (VBP take) (NP (PRP it)))))))))) (CC and) (VP (VB bring) (NP (PRP it)) (PP (TO to) (NP (PRP me)))))) (. .))

Nr. 11:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VBP go) (PP (TO to) (NP (NP (DT the) (NN table) (NN grab) (NN popcorn)) (VP (VBN come) (SBAR (TO to) (S (NP (PRP me)) (VP (VBP give) (NP (PRP me)) (NP (NP (DT the) (NN popcorn)) (SBAR (WHNP (WDT which)) (S (VP (VBZ is) (PP (IN in) (NP (PRP\$ your) (NN hand)))))))))))))) (. .))

Nr. 12:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VBP go) (PP (TO to) (NP (DT the) (NN table))) (S (VP (VB take) (S (NP (DT the) (JJ green) (NN cup)) (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN dishwasher))) (S (VP (VB open) (S (NP (PRP it)) (VP (VB put) (NP (DT the) (JJ green) (NN cup)) (PP (IN into) (NP (DT the) (NN dishwasher)))))))))) (CC and) (VP (VB close) (NP (PRP it)))))) (. .))

Nr. 13:

(ROOT (S (NP (NNP Okay) (NNP ARMAR)) (VP (VP (VB go) (PP (TO to) (NP (NP (DT the) (JJ fridge) (NN look)) (PP (IN for) (NP (DT the) (JJ orange) (NN juice)))))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VB go) (PP (TO to) (NP (DT the) (NN table))) (S (VP (VB pour) (NP (DT the) (NN juice)) (PP (IN into) (NP (NP (DT the) (JJ green) (NN cup)) (VP (VBN put) (PRT (RP down)) (NP (DT the) (NN juice)))))))))) (CC and) (VP (VB bring) (NP (PRP me)) (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN from) (NP (DT the) (NN table)))))) (. .))

Nr. 14:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VBP go) (PP (TO to) (NP (DT the) (NN fridge))) (S (VP (VB open) (S (NP (PRP it)) (VP (VP (VB look) (PP (IN for) (NP (DT the) (JJ orange) (NN juice)))) (CC and) (VP (VB take) (S (NP (PRP it)) (VP (VB close) (NP (DT the) (NN fridge)))))) (CC and) (VP (VB bring) (NP (DT the) (NN juice)) (PP (TO to) (NP (PRP me)))))) (. .))



(VBZ is) (NP (NP (DT a) (JJ popcorn) (NN bag))) (SBAR (S (VP (VB take) (NP (PRP it)) (VP (VP (VB come) (PRT (RP over)) (ADVP (RB here)))) (CC and) (VP (VB give) (NP (PRP it)) (PP (TO to) (NP (PRP me)))))))))) (. .))

Nr. 7:

( (S (VP (VB Locate) (NP (DT the) (NN dishwasher))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (S (NP (PRP it)) (VP (VP (VB go) (ADVP (RB there)) (SBAR (WHADVP (WRB when)) (S (ADVP (RB you're)) (NP (EX there)) (VP (VBP open) (NP (PRP\$ its) (JJ front) (NN door)))))) (CC and) (VP (VB pull) (PRT (RP out)) (NP (DT the) (JJ upper) (NN rack)) (VP (VP (VB go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table)))) (CC and) (VP (VB see) (NP (NNS what's)) (ADVP (RB there)))))) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VB see) (NP (DT a) (JJ green) (NN cup)) (VP (VB take) (NP (PRP it)) (VP (VP (VB move) (ADVP (RB back)) (PP (IN towards) (NP (DT the) (NN dishwasher)))) (CC and) (VP (VB put) (NP (DT the) (NN cup)) (PP (IN into) (S (NP (PRP it)) (ADVP (RB finally)) (VP (VP (VB push) (PRT (RP back)) (NP (DT the) (NN basket))) (CC and) (VP (VB close) (NP (DT the) (NN door.)))))))))))))) )

Nr. 8:

(ROOT (S (PP (IN Inside) (NP (DT the) (NN ice) (NN box))) (NP (EX there)) (VP (VP (VBZ is) (NP (DT an) (NN orange) (NN juice))) (CC so) (VP (VBP go) (PRT (RP over)) (SBAR (S (NP (EX there)) (VP (VP (VP (VB open) (NP (DT the) (NN door))) (CC and) (VP (VB take) (NP (PRP it)) (SBAR (WHADVP (WRB when)) (S (NP (PRP you)) (VP (VBP 've) (VP (VBN got) (S (NP (PRP it)) (VP (VB close) (SBAR (S (NP (DT the) (NN door)) (VP (VBP carry) (NP (DT the) (NN orange) (NN juice)) (ADVP (IN over) (RB here)))))))))) (CC and) (VP (VB hand) (NP (PRP it)) (PRT (RP over)))))) ( . .)))

Nr. 9:

(ROOT (S (NP (NNP Okay) (NNP ARMAR)) (VP (VBP look) (PP (IN for) (NP (DT the) (NN table))) (S (VP (VP (VB go) (ADVP (RB there))) (CC and) (VP (VP (VB take) (NP (DT the) (NN popcorn)) (PP (IN after) (S (VP (VBG taking) (S (NP (DT the) (NN popcorn)) (VP (VB come) (ADVP (RB here)))))) (CC and) (VP (VB hand) (NP (PRP it)) (ADVP (RB over)))))) ( . .)))

Nr. 10:

(ROOT (S (VP (VP (VB Go) (PP (TO to) (NP (DT the) (NN kitchen) (NN table))) (CC and) (VP (VB see) (SBAR (IN if) (S (NP (PRP you)) (VP (MD can) (VP (VP (VB find) (NP (DT a) (JJ popcorn) (NN bag))) (CONJP (IN if) (RB so)) (VP (VP (VB take) (NP (PRP it)) (CC and) (VP (VB bring) (NP (PRP it)) (PP (TO to) (NP (PRP me)))))) ( . .)))

Nr. 11:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VB go) (PP (TO to) (NP (DT the) (NN table))) (NP (NN grab) (NN popcorn)) (VP (VB come) (PP (TO to) (NP (PRP me))) (VP (VB give) (NP (PRP me)) (NP (NP (DT the) (NN popcorn)) (SBAR (WHNP (WDT which)) (S (VP (VBZ is) (PP (IN in) (NP (PRP\$ your) (NN hand)))))) ( . .)))

Nr. 12:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VBP go) (PP (TO to) (NP (DT the)

(NN table))) (S (VP (VB take) (NP (NP (DT the) (JJ green) (NN cup)) (SBAR (S (VP (VP (VB go) (PP (TO to) (NP (NP (DT the) (NN dishwasher) (NN open)) (SBAR (S (NP (PRP it)) (VP (VBP put) (NP (DT the) (JJ green) (NN cup)) (PP (IN into) (NP (DT the) (NN dishwasher)))))))))) (CC and) (VP (VB close) (NP (PRP it))))))))) (. .))

Nr. 13:

(ROOT (S (NP (NNP Okay) (NNP ARMAR)) (VP (VP (VB go) (PP (TO to) (NP (DT the) (JJ fridge) (NN look))) (PP (IN for) (NP (DT the) (NN orange) (NN juice)))) (CC and) (VP (VBP take) (S (NP (PRP it)) (VP (VB go) (PP (TO to) (NP (DT the) (NN table)))))) (SQ (VBP pour) (NP (NP (DT the) (NN juice)) (PP (IN into) (NP (DT the) (JJ green) (NN cup)))) (VP (VP (VB put) (PRT (RP down)) (NP (DT the) (NN juice))) (CC and) (VP (VB bring) (NP (PRP me)) (NP (NP (DT the) (JJ green) (NN cup)) (PP (IN from) (NP (DT the) (NN table))))))))) (. .))

Nr. 14:

(ROOT (S (NP (NNP Okay) (NNP Armar)) (VP (VBP go) (PP (TO to) (NP (DT the) (NN fridge))) (S (VP (VB open) (S (NP (PRP it)) (VP (VP (VB look) (PP (IN for) (NP (DT the) (NN orange) (NN juice)))) (CC and) (VP (VB take) (NP (PRP it)) (VP (VP (VB close) (NP (DT the) (NN fridge))) (CC and) (VP (VB bring) (NP (DT the) (NN juice)) (PP (TO to) (NP (PRP me))))))))) (. .))

## I. Evaluation

### I.1. Transkriptionen

Nr. 1:

*Okay Armar go to the fridge open it look for the orange juice and take it close the fridge and bring the juice to me*

Nr. 2:

*Armar could you please take the green cup from the kitchen table and then move to the dishwasher and open it and then put the cup into the dishwasher and close it again*

Nr. 3:

*Hey Armar go to kitchen table take the popcorn bag and bring it to me*

Nr. 4:

*Armar please bring me the popcorn bag from the table*

Nr. 5:

*Armar open the fridge and bring me the orange juice*

Nr. 6:

*Can you grab me the box of popcorn on the table it's next to the cereal box*

Nr. 7:

*Can you please bring me the orange juice from the fridge please*

Nr. 8:

*Can you put the green cup from the kitchen table in the dishwasher*

Nr. 9:

*Hey Armar please go to the kitchen table on the right side and hand me the popcorn*

Nr. 10:

*Hey Armar please take the green cup open the dishwasher and put the green cup in the dishwasher and then close the dishwasher again*

Nr. 11:

*uhm go to the table uhm take the green cup uhm then go to the dishwasher open the dishwasher uhm put the green green cup ehm in the dishwasher and close the dishwasher*

Nr. 12:

*uhm go to to the fridge open the fridge uhm get the orange juice uhm close the fridge uhm bring the orange juice to me*

Nr. 13:

*go to the table in front of the window and take the popcorn and bring it to me*

Nr. 14:

*go to the table and take the green cup then bring the cup to the dishwasher first open the dishwasher put the cup inside it and then close the dishwasher*

Nr. 15:

*go to the fridge open the fridge and take the orange juice bring the orange juice to me please*

Nr. 16:

*go to the table grab the green cup turn and go to the dishwasher if it is closed please open it find a free place put the green cup into it and close the dishwasher*

Nr. 17:

*go to the fridge open it if it's closed searched the orange juice and grab it pull it out of the fridge and close the door*

Nr. 18:

*Armar go to the table and give me the popcorn bag*

Nr. 19:

*can you bring me the popcorn from the table*

Nr. 20:

*hello Armar uhm I need some popcorn could you please go to the table grab the popcorn box turn to turn to me and then come back and hand it to me*

## I.2. Ausgaben der Google Speech To Text API

Nr. 1:

*okay armor go to the french open it look for the orange juice and take it close the fridge and bring the dreads to me*

Nr. 2:

*comma could you please take the green car on the kitchen table and then moved to the dishwasher and open it and then pick up into the dishwasher so sick*

Nr. 3:

*hey comma go to cancel tennis take the popcorn bag and bring it to me*

Nr. 4:

*the are mom please bring the pop on back from the table*

Nr. 5:

*armagh on the fridge and bring your insurance*

Nr. 6:

*can you grab me the box of pop corn on the table next to the cereal box*

Nr. 7:

*can you please bring me orange juice from the fridge please*

Nr. 8:

*can you put a green path from the kitchen to end of the session*

Nr. 9:

*hey comma please go into the kitchen taylor on the right side in huntington popcorn*

Nr. 10:

*hey anna please take the green pop open the dishwasher and put the green cap in the dishwasher and stove the dishwasher*

Nr. 11:

*ulingo to taylor take the green car then go to the dishwasher open the dishwasher and put the green green cop at amc in the dish washer um and killswitch*

Nr. 12:

*i'm go to learn to french open french and get the orange juice close the fridge um bring the arches to me*

Nr. 13:

*go to the table in front of the window uh and take the popcorn and bring it to me*

Nr. 14:

*go to do table and take to cream cup note bring to talk to the dishwasher first open the dishwasher for pick up inside and then close it before*

Nr. 15:

*the go to the french open french and take the orange juice bread orange juice mean*

Nr. 16:

*go to the table grab the green top turn and go to the dishwasher if it is closed please of let's find a free place put the green tap into it and does the dishwasher*

Nr. 17:

*go to the french open and if it's coast search the the orange juice and grab it close it out of to french toast or*

Nr. 18:

*i'm gonna go to the temple of come back*

Nr. 19:

*can you bring the popcorn from the taylor*

Nr. 20:

*hello babe i need some popcorn can you please go to the table grab the book on bugs turn to turn to me and then come back and the man*

### I.3. Musterlösung zur Evaluation

**Format: Wort/Wortart/Chunking/Befehlsnummer**

Nr. 1:

*Okay/Uh/B-NP/0 Armar/NNP/I-NP/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 it/PRP/B-NP/1 look/VB/B-VP/2 for/IN/B-PP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 and/CC/O/3 take/VB/B-VP/3 it/PRP/B-NP/3 close/VB/B-VP/4 the/DT/B-NP/4 fridge/NN/I-NP/4 and/CC/O/5 bring/VB/B-VP/5 the/DT/B-NP/5 juice/NN/I-NP/5 to/TO/B-PP/5 me/PRP/B-NP/5*

Nr. 2:

*Armar/NNP/B-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/RB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 and/CC/O/1 then/RB/B-VP/1 move/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 open/VB/B-VP/2 it/PRP/B-NP/2 and/CC/O/3 then/RB/B-ADVP/3 put/VB/B-VP/3 the/DT/B-NP/3 cup/NN/I-NP/3 into/IN/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 and/C-C/O/4 close/VB/B-VP/4 it/PRP/B-NP/4 again/RB/B-ADVP/4*

Nr. 3:

*Hey/UH/B-NP/0 Armar/NNP/I-NP/0 go/VB/B-VP/0 to/TO/B-PP/0 kitchen/NN/B-NP/0 table/NN/I-NP/0 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 4:

*Armar/NNP/B-NP/0 please/RB/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 bag/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 5:

*Armar/NNP/B-NP/0 open/VB/B-VP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 and/CC/O/1 bring/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 orange/NN/I-NP/1 juice/NN/I-NP/1*

Nr. 6:

*Can/MD/B-VB/0 you/PRP/B-NP/0 grab/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 box/NN/I-NP/0 of/IN/B-PP/0 popcorn/NN/B-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 it/PRP/B-NP/1 's/VBZ/B-VP/1 next/IN/B-PP/1 to/TO/I-PP/1 the/DT/B-NP/1 cereal/NN/I-NP/1 box/NN/I-NP/1*

Nr. 7:

*Can/MD/B-VB/0 you/PRP/B-NP/0 please/RB/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 please/RB/B-ADVP/0*

Nr. 8:

*Can/MD/B-VB/0 you/PRP/B-NP/0 put/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 the/DT/B-NP/0 dishwasher/NN/I-NP/0*

Nr. 9:

*Hey/UH/B-NP/0 Armar/NNP/I-NP/0 please/RB/B-VP/0 go/VB/I-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 right/JJ/I-NP/0 side/NN/I-NP/0 and/CC/O/1 hand/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1*

Nr. 10:

*Hey/UH/B-NP/0 Armar/NNP/I-NP/0 please/RB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 put/VB/B-VP/2 the/DT/B-NP/2 green/JJ/I-NP/2 cup/NN/I-NP/2 in/IN/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 and/CC/O/3 then/RB/B-VP/3 close/VB/I-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 again/RB/B-ADVP/3*

Nr. 11:

*uhm/UH/B-INTJ/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 uhm/UH/B-INTJ/0 take/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1*

*uhm/UH/B-INTJ/1 then/RB/B-ADVP/1 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 uhm/UH/B-INTJ/3 put/VB/B-VP/4 the/DT/B-NP/4 green/JJ/I-NP/4 green/JJ/I-NP/4 cup/NN/I-NP/4 ehm/UH/B-INTJ/4 in/IN/B-PP/4 the/DT/B-NP/4 dishwasher/NN/I-NP/4 and/CC/O/5 close/VB/B-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 12:

*uhm/UH/B-INTJ/0 go/VB/B-VP/0 to/TO/B-PP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 uhm/UH/B-INTJ/1 get/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 uhm/UH/B-NP/2 close/VB/B-VP/3 the/DT/B-NP/3 fridge/NN/I-NP/3 uhm/UH/B-INTJ/3 bring/VB/B-VP/4 the/DT/B-NP/4 orange/NN/I-NP/4 juice/NN/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4*

Nr. 13:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 front/NN/B-NP/0 of/IN/B-PP/0 the/DT/B-NP/0 window/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 14:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 then/RB/B-ADVP/1 bring/VB/B-VP/2 the/DT/B-NP/2 cup/NN/I-NP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 first/RB/B-ADVP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 put/VB/B-VP/4 the/DT/B-NP/4 cup/NN/I-NP/4 inside/IN/B-PP/4 it/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 close/VB/I-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 15:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 bring/VB/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 to/TO/B-PP/3 me/PRP/B-NP/3 please/RB/B-ADVP/3*

Nr. 16:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 grab/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 turn/VB/B-VP/2 and/CC/O/3 go/VB/B-VP/3 to/TO/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 if/IN/B-SBAR/4 it/PRP/B-NP/4 is/VBZ/B-VP/4 closed/VBN/I-VP/4 please/RB/B-VP/4 open/VB/I-VP/5 it/PRP/B-NP/5 find/VB/B-VP/6 a/DT/B-NP/6 free/JJ/I-NP/6 place/NN/I-NP/6 put/VB/B-VP/7 the/DT/B-NP/7 green/JJ/I-NP/7 cup/NN/I-NP/7 into/IN/B-PP/7 it/PRP/B-NP/7 and/CC/O/8 close/VB/B-VP/8 the/DT/B-NP/8 dishwasher/NN/I-NP/8*

Nr. 17:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 it/PRP/B-NP/1 if/IN/B-SBAR/2 it/PRP/B-NP/2 's/VBZ/B-VP/2 closed/VBN/I-VP/2 searched/VBD/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 and/C-*

*C/O/4 grab/VB/B-VP/4 it/PRP/B-NP/4 pull/VB/B-VP/5 it/PRP/B-NP/5 out/IN/B-PP/5 of/IN/I-PP/5 the/DT/B-NP/5 fridge/NN/I-NP/5 and/CC/O/6 close/VB/B-VP/6 the/DT/B-NP/6 door/NN/I-NP/6*

Nr. 18:

*Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 give/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1*

Nr. 19:

*can/MD/B-VP/0 you/PRP/B-NP/0 bring/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 20:

*hello/UH/B-INTJ/0 Armar/NNP/B-NP/0 uhm/UH/B-INTJ/0 I/PRP/B-NP/0 need/VB/B-VP/0 some/JJ/B-NP/0 popcorn/NN/I-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/RB/B-ADVP/0 go/VB/B-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1 grab/VB/B-VP/2 the/DT/B-NP/2 popcorn/NN/I-NP/2 box/NN/I-NP/2 turn/VB/B-VP/3 to/TO/B-PP/3 turn/VB/B-VP/4 to/TO/B-PP/4 me/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 come/VB/I-VP/5 back/RB/B-ADVP/5 and/CC/O/6 hand/VB/B-VP/6 it/PRP/B-NP/6 to/TO/B-PP/6 me/PRP/B-NP/6*

#### I.4. Ausgabe mit SENNA

Nr. 1:

*Okay/NNP/B-NP/0 Armar/NNP/I-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/JJ/B-ADJP/0 it/PRP/B-NP/0 look/VBP/B-VP/1 for/IN/B-PP/1 the/DT/B-NP/1 orange/NN/I-NP/1 juice/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 it/PRP/B-NP/2 close/VB/B-VP/3 the/DT/B-NP/3 fridge/NN/I-NP/3 and/C-C/O/4 bring/VB/B-VP/4 the/DT/B-NP/4 juice/NN/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4*

Nr. 2:

*Armar/NNP/B-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/VB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 and/CC/O/1 then/RB/B-VP/1 move/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 open/VB/B-VP/2 it/PRP/B-NP/2 and/CC/O/3 then/RB/B-ADVP/3 put/VBD/B-VP/3 the/DT/B-NP/3 cup/NN/I-NP/3 into/IN/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 and/C-C/O/4 close/VB/B-VP/4 it/PRP/B-NP/4 again/RB/B-ADVP/4*

Nr. 3:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 kitchen/NN/B-NP/0 table/NN/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 4:

*Armar/NNP/B-NP/0 please/VBP/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-*

*NP/0 popcorn/NN/I-NP/0 bag/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 5:

*Armar/JJ/B-NP/0 open/JJ/I-NP/0 the/DT/I-NP/0 fridge/NN/I-NP/0 and/CC/O/0 bring/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0*

Nr. 6:

*Can/MD/O/0 you/PRP/B-NP/0 grab/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 box/NN/I-NP/0 of/IN/B-PP/0 popcorn/NN/B-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 it/PRP/B-NP/0 's/VBZ/B-VP/1 next/JJ/B-NP/1 to/TO/B-PP/1 the/DT/B-NP/1 cereal/NN/I-NP/1 box/NN/I-NP/1*

Nr. 7:

*Can/MD/O/0 you/PRP/B-NP/0 please/VB/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 please././O/0*

Nr. 8:

*Can/MD/O/0 you/PRP/B-NP/0 put/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 the/DT/B-NP/0 dishwasher/NN/I-NP/0*

Nr. 9:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 please/MD/B-VP/0 go/VB/I-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 right/JJ/I-NP/0 side/NN/I-NP/0 and/CC/O/1 hand/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1*

Nr. 10:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 please/VBP/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 open/JJ/B-ADJP/0 the/DT/B-NP/0 dishwasher/NN/I-NP/0 and/CC/O/1 put/VBD/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 in/IN/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 then/RB/B-VP/2 close/VB/I-VP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 again/RB/B-ADVP/2*

Nr. 11:

*uhm/UH/B-INTJ/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 uhm/UH/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 uhm/UH/I-NP/1 then/RB/B-ADVP/1 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 uhm/UH/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 green/JJ/I-NP/4 green/JJ/I-NP/4 cup/NN/I-NP/4 ehm/UH/I-NP/4 in/IN/B-PP/4 the/DT/B-NP/4 dishwasher/NN/I-NP/4 and/CC/O/5 close/VB/B-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 12:

*uhm/UH/B-INTJ/0 go/VBP/B-VP/0 to/TO/B-PP/0 to/TO/B-PP/0 the/DT/B-NP/0*

*fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 uhm/UH/B-VP/1 get/VB/I-VP/2 the/DT/B-NP/2 orange/JJ/I-NP/2 juice/NN/I-NP/2 uhm/UH/I-NP/2 close/NN/I-NP/2 the/DT/B-NP/2 fridge/NN/I-NP/2 uhm/UH/B-VP/2 bring/VB/I-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 to/TO/B-PP/3 me/PRP/B-NP/3*

Nr. 13:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 front/NN/B-NP/0 of/IN/B-PP/0 the/DT/B-NP/0 window/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 14:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 then/RB/B-ADVP/1 bring/VB/B-VP/2 the/DT/B-NP/2 cup/NN/I-NP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 first/RB/B-ADVP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 cup/NN/I-NP/4 inside/IN/B-PP/4 it/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 close/VB/I-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 15:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 bring/VB/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 to/TO/B-VP/3 me/PRP/B-NP/3 please/VB/B-VP/4*

Nr. 16:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 grab/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 turn/NN/I-NP/1 and/CC/O/2 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 if/IN/B-SBAR/3 it/PRP/B-NP/3 is/VBZ/B-VP/3 closed/VBN/I-VP/3 please/RB/B-ADVP/3 open/VB/B-VP/4 it/PRP/B-NP/4 find/VB/B-VP/5 a/DT/B-NP/5 free/JJ/I-NP/5 place/NN/I-NP/5 put/VBD/B-VP/6 the/DT/B-NP/6 green/JJ/I-NP/6 cup/NN/I-NP/6 into/IN/B-PP/6 it/PRP/B-NP/6 and/CC/O/7 close/VB/B-VP/7 the/DT/B-NP/7 dishwasher/NN/I-NP/7*

Nr. 17:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 it/PRP/B-NP/1 if/IN/B-SBAR/2 it/PRP/B-NP/2 's/VBZ/B-VP/2 closed/VBN/I-VP/2 searched/VBD/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 and/CC/O/4 grab/VB/B-VP/4 it/PRP/B-NP/4 pull/VB/B-VP/5 it/PRP/B-NP/5 out/IN/B-PP/5 of/IN/B-PP/5 the/DT/B-NP/5 fridge/NN/I-NP/5 and/CC/O/6 close/VB/B-VP/6 the/DT/B-NP/6 door/NN/I-NP/6*

Nr. 18:

*Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 give/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1*

Nr. 19:

*can/MD/O/0 you/PRP/B-NP/0 bring/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 20:

*hello/SYM/O/0 Armar/JJ/B-ADJP/0 uhm/UH/I-ADJP/0 I/PRP/B-NP/0 need/VBP/B-VP/0 some/DT/B-NP/0 popcorn/NN/I-NP/0 could/MD/B-VP/0 you/PRP/I-VP/0 please/VB/I-VP/1 go/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1 grab/VB/B-VP/2 the/DT/B-NP/2 popcorn/NN/I-NP/2 box/NN/I-NP/2 turn/NN/I-NP/2 to/TO/B-VP/2 turn/VB/I-VP/3 to/TO/B-PP/3 me/PRP/B-NP/3 and/CC/O/4 then/RB/B-VP/4 come/VB/I-VP/4 back/RB/B-ADVP/4 and/CC/O/5 hand/NN/B-NP/5 it/PRP/B-NP/5 to/TO/B-PP/5 me/PRP/B-NP/5*

### I.5. Ausgabe mit SENNA und Stanford

Nr. 1:

*okay/JJ/B-NP/0 Armar/NNP/I-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VBP/B-VP/1 it/PRP/B-NP/1 look/VBP/B-VP/2 for/IN/B-PP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 and/CC/O/3 take/VB/B-VP/3 it/PRP/B-NP/3 close/VB/B-VP/4 the/DT/B-NP/4 fridge/NN/I-NP/4 and/CC/O/5 bring/VB/B-VP/5 the/DT/B-NP/5 juice/NN/I-NP/5 to/TO/B-PP/5 me/PRP/B-NP/5*

Nr. 2:

*Armar/NNP/B-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/VB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 and/CC/O/1 then/RB/B-VP/1 move/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 open/VB/B-VP/2 it/PRP/B-NP/2 and/CC/O/3 then/RB/B-ADVP/3 put/VBD/B-VP/3 the/DT/B-NP/3 cup/NN/I-NP/3 into/IN/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 and/C-O/4 close/VB/B-VP/4 it/PRP/B-NP/4 again/RB/B-ADVP/4*

Nr. 3:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 kitchen/NN/B-NP/0 table/NN/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 4:

*Armar/NNP/B-NP/0 please/VBP/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 bag/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 5:

*Armar/JJ/B-NP/0 open/VB/B-VP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 and/CC/O/1 bring/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 orange/NN/I-NP/1 juice/NN/I-NP/1*

Nr. 6:

*Can/MD/O/0 you/PRP/B-NP/0 grab/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 box/NN/I-NP/0 of/IN/B-PP/0 popcorn/NN/B-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 it/PRP/B-NP/0 's/VBZ/B-VP/1 next/JJ/B-NP/1 to/TO/B-PP/1 the/DT/B-NP/1 cereal/NN/I-NP/1 box/NN/I-NP/1*

Nr. 7:

*Can/MD/O/0 you/PRP/B-NP/0 please/VB/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 please/VB/B-VP/1*

Nr. 8:

*Can/MD/O/0 you/PRP/B-NP/0 put/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 the/DT/B-NP/0 dishwasher/NN/I-NP/0*

Nr. 9:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 please/VBP/B-VP/0 go/VB/I-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 right/JJ/I-NP/0 side/NN/I-NP/0 and/CC/O/1 hand/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1*

Nr. 10:

*Hey/NNP/B-NP/0 Armar/NNP/I-NP/0 please/VBP/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 put/VBD/B-VP/2 the/DT/B-NP/2 green/JJ/I-NP/2 cup/NN/I-NP/2 in/IN/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 and/CC/O/3 then/RB/B-VP/3 close/VB/I-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 again/RB/B-ADVP/3*

Nr. 11:

*uhm/UH/B-INTJ/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 uhm/UH/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 uhm/UH/I-NP/1 then/RB/B-ADVP/1 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 uhm/UH/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 green/JJ/I-NP/4 green/JJ/I-NP/4 cup/NN/I-NP/4 ehm/UH/I-NP/4 in/IN/B-PP/4 the/DT/B-NP/4 dishwasher/NN/I-NP/4 and/CC/O/5 close/VB/B-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 12:

*uhm/UH/B-INTJ/0 go/VBP/B-VP/0 to/TO/B-PP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 uhm/UH/B-VP/1 get/VB/I-VP/2 the/DT/B-NP/2 orange/JJ/I-NP/2 juice/NN/I-NP/2 uhm/UH/B-VP/2 close/VB/I-VP/3 the/DT/B-NP/3 fridge/NN/I-NP/3 uhm/UH/B-VP/3 bring/VB/I-VP/4 the/DT/B-NP/4 orange/NN/I-NP/4 juice/NN/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4*

Nr. 13:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 front/*

*NN/B-NP/0 of/IN/B-PP/0 the/DT/B-NP/0 window/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 14:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 then/RB/B-ADVP/1 bring/VB/B-VP/2 the/DT/B-NP/2 cup/NN/I-NP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 first/RB/B-ADVP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 cup/NN/I-NP/4 inside/IN/B-PP/4 it/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 close/VB/I-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 15:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 bring/VBP/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 to/TO/B-VP/3 me/PRP/B-NP/3 please/VB/B-VP/4*

Nr. 16:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 grab/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 turn/VB/B-VP/2 and/CC/O/3 go/VB/I-VP/3 to/TO/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 if/IN/B-SBAR/4 it/PRP/B-NP/4 is/VBZ/B-VP/4 closed/VBN/I-VP/4 please/VBP/B-VP/5 open/VB/I-VP/5 it/PRP/B-NP/5 find/VB/B-VP/6 a/DT/B-NP/6 free/JJ/I-NP/6 place/NN/I-NP/6 put/VBD/B-VP/7 the/DT/B-NP/7 green/JJ/I-NP/7 cup/NN/I-NP/7 into/IN/B-PP/7 it/PRP/B-NP/7 and/CC/O/8 close/VB/B-VP/8 the/DT/B-NP/8 dishwasher/NN/I-NP/8*

Nr. 17:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 it/PRP/B-NP/1 if/IN/B-SBAR/2 it/PRP/B-NP/2 's/VBZ/B-VP/2 closed/VBN/I-VP/2 searched/VBD/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 and/CC/O/4 grab/VB/B-VP/4 it/PRP/B-NP/4 pull/VB/B-VP/5 it/PRP/B-NP/5 out/IN/B-PP/5 of/IN/B-PP/5 the/DT/B-NP/5 fridge/NN/I-NP/5 and/CC/O/6 close/VB/B-VP/6 the/DT/B-NP/6 door/NN/I-NP/6*

Nr. 18:

*Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 give/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1*

Nr. 19:

*can/MD/O/0 you/PRP/B-NP/0 bring/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 20:

*hello/SYM/O/0 Armar/JJ/B-ADJP/0 uhm/UH/I-ADJP/0 I/PRP/B-NP/0 need/VBP/B-VP/0 some/DT/B-NP/0 popcorn/NN/I-NP/0 could/MD/B-VP/0 you/PRP/I-VP/0*

please/VB/I-VP/1 go/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1  
 grab/VB/B-VP/2 the/DT/B-NP/2 popcorn/NN/I-NP/2 box/NN/I-NP/2 turn/VB/B-VP/3  
 to/TO/I-VP/3 turn/VB/I-VP/4 to/TO/B-PP/4 me/PRP/B-NP/4 and/CC/O/5  
 then/RB/B-VP/5 come/VB/I-VP/5 back/RB/B-ADVP/5 and/CC/O/6 hand/VB/B-VP/6  
 it/PRP/B-NP/6 to/TO/B-PP/6 me/PRP/B-NP/6

## I.6. Ausgabe mit SENNA, Stanford und Optimierungsliste

Nr. 1:

Okay/UH/B-INTJ/0 Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-  
 NP/0 fridge/NN/I-NP/0 open/VBP/B-VP/1 it/PRP/B-NP/1 look/VBP/B-VP/2 for/IN/B-  
 PP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 and/CC/O/3 take/VB/B-  
 VP/3 it/PRP/B-NP/3 close/VB/B-VP/4 the/DT/B-NP/4 fridge/NN/I-NP/4 and/C-  
 C/O/5 bring/VB/B-VP/5 the/DT/B-NP/5 juice/NN/I-NP/5 to/TO/B-PP/5 me/PRP/B-  
 NP/5

Nr. 2:

Armar/NNP/B-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/RB/B-ADVP/0  
 take/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0  
 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 and/CC/O/1 then/RB/B-VP/1  
 move/VB/I-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2  
 open/VB/B-VP/2 it/PRP/B-NP/2 and/CC/O/3 then/RB/B-ADVP/3 put/VBD/B-VP/3  
 the/DT/B-NP/3 cup/NN/I-NP/3 into/IN/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-  
 NP/3 and/CC/O/4 close/VB/B-VP/4 it/PRP/B-NP/4 again/RB/B-ADVP/4

Nr. 3:

Hey/UH/B-INTJ/0 Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 kitchen/NN/B-  
 NP/0 table/NN/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-  
 NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2

Nr. 4:

Armar/NNP/B-NP/0 please/RB/B-ADVP/0 bring/VB/B-VP/0 me/PRP/B-NP/0  
 the/DT/B-NP/0 popcorn/NN/I-NP/0 bag/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0  
 table/NN/I-NP/0

Nr. 5:

Armar/NNP/B-NP/0 open/VB/B-VP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 and/CC/O/1  
 bring/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 orange/NN/I-NP/1 juice/NN/I-  
 NP/1

Nr. 6:

Can/MD/O/0 you/PRP/B-NP/0 grab/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0  
 box/NN/I-NP/0 of/IN/B-PP/0 popcorn/NN/B-NP/0 on/IN/B-PP/0 the/DT/B-NP/0  
 table/NN/I-NP/0 it/PRP/B-NP/0 's/VBZ/B-VP/1 next/JJ/B-NP/1 to/TO/B-PP/1  
 the/DT/B-NP/1 cereal/NN/I-NP/1 box/NN/I-NP/1

Nr. 7:

Can/MD/O/0 you/PRP/B-NP/0 please/RB/B-ADVP/0 bring/VB/B-VP/0 me/PRP/B-

*NP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 please/RB/B-ADVP/0*

Nr. 8:

*Can/MD/O/0 you/PRP/B-NP/0 put/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 the/DT/B-NP/0 dishwasher/NN/I-NP/0*

Nr. 9:

*Hey/UH/B-INTJ/0 Armar/NNP/B-NP/0 please/RB/B-ADVP/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 right/JJ/I-NP/0 side/NN/I-NP/0 and/CC/O/1 hand/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1*

Nr. 10:

*Hey/UH/B-INTJ/0 Armar/NNP/B-NP/0 please/RB/B-ADVP/0 take/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 cup/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 put/VBD/B-VP/2 the/DT/B-NP/2 green/JJ/I-NP/2 cup/NN/I-NP/2 in/IN/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 and/CC/O/3 then/RB/B-VP/3 close/VB/I-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 again/RB/B-ADVP/3*

Nr. 11:

*uhm/UH/B-INTJ/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 uhm/UH/I-NP/0 take/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 uhm/UH/I-NP/1 then/RB/B-ADVP/1 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 uhm/UH/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 green/JJ/I-NP/4 green/JJ/I-NP/4 cup/NN/I-NP/4 ehm/UH/I-NP/4 in/IN/B-PP/4 the/DT/B-NP/4 dishwasher/NN/I-NP/4 and/CC/O/5 close/VB/B-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 12:

*uhm/UH/B-INTJ/0 go/VBP/B-VP/0 to/TO/B-PP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 uhm/UH/B-VP/1 get/VB/I-VP/2 the/DT/B-NP/2 orange/JJ/I-NP/2 juice/NN/I-NP/2 uhm/UH/B-VP/2 close/VB/I-VP/3 the/DT/B-NP/3 fridge/NN/I-NP/3 uhm/UH/B-VP/3 bring/VB/I-VP/4 the/DT/B-NP/4 orange/NN/I-NP/4 juice/NN/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4*

Nr. 13:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 front/NN/B-NP/0 of/IN/B-PP/0 the/DT/B-NP/0 window/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2*

Nr. 14:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 then/RB/B-ADVP/1 bring/VB/B-VP/2 the/DT/B-NP/2 cup/NN/I-NP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-*

*NP/2 first/RB/B-ADVP/2 open/VB/B-VP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 put/VBD/B-VP/4 the/DT/B-NP/4 cup/NN/I-NP/4 inside/IN/B-PP/4 it/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 close/VB/I-VP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5*

Nr. 15:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 fridge/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 bring/VBP/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 to/TO/B-PP/3 me/PRP/B-NP/3 please/RB/B-ADVP/3*

Nr. 16:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 grab/VBP/B-VP/1 the/DT/B-NP/1 green/JJ/I-NP/1 cup/NN/I-NP/1 turn/VB/B-VP/2 and/CC/I-VP/3 go/VB/I-VP/3 to/TO/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 if/IN/B-SBAR/4 it/PRP/B-NP/4 is/VBZ/B-VP/4 closed/VBN/I-VP/4 please/RB/B-ADVP/4 open/VB/B-VP/5 it/PRP/B-NP/5 find/VB/B-VP/6 a/DT/B-NP/6 free/JJ/I-NP/6 place/NN/I-NP/6 put/VBD/B-VP/7 the/DT/B-NP/7 green/JJ/I-NP/7 cup/NN/I-NP/7 into/IN/B-PP/7 it/PRP/B-NP/7 and/CC/O/8 close/VB/B-VP/8 the/DT/B-NP/8 dishwasher/NN/I-NP/8*

Nr. 17:

*go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 open/VB/B-VP/1 it/PRP/B-NP/1 if/IN/B-SBAR/2 it/PRP/B-NP/2 's/VBZ/B-VP/2 closed/VBN/I-VP/2 searched/VBD/B-VP/3 the/DT/B-NP/3 orange/NN/I-NP/3 juice/NN/I-NP/3 and/C-C/O/4 grab/VB/B-VP/4 it/PRP/B-NP/4 pull/VB/B-VP/5 it/PRP/B-NP/5 out/IN/B-PP/5 of/IN/B-PP/5 the/DT/B-NP/5 fridge/NN/I-NP/5 and/CC/O/6 close/VB/B-VP/6 the/DT/B-NP/6 door/NN/I-NP/6*

Nr. 18:

*Armar/NNP/B-NP/0 go/VBP/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 and/CC/O/1 give/VB/B-VP/1 me/PRP/B-NP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 bag/NN/I-NP/1*

Nr. 19:

*can/MD/O/0 you/PRP/B-NP/0 bring/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0*

Nr. 20:

*hello/SYM/O/0 Armar/NNP/B-NP/0 uhm/UH/I-NP/0 I/PRP/B-NP/0 need/VBP/B-VP/0 some/DT/B-NP/0 popcorn/NN/I-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/RB/B-ADVP/0 go/VB/B-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1 grab/VB/B-VP/2 the/DT/B-NP/2 popcorn/NN/I-NP/2 box/NN/I-NP/2 turn/VB/B-VP/3 to/TO/I-VP/3 turn/VB/I-VP/4 to/TO/B-PP/4 me/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5 come/VB/I-VP/5 back/RB/B-ADVP/5 and/CC/O/6 hand/VB/B-VP/6 it/PRP/B-NP/6 to/TO/B-PP/6 me/PRP/B-NP/6*

### I.7. Ausgaben für die ASR Texte (SENNA und Stanford)

Nr. 1:

*okay/JJ/B-NP/0 armor/NN/I-NP/0 go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 french/JJ/I-NP/0 open/JJ/I-NP/0 it/PRP/B-NP/0 look/VBP/B-VP/1 for/IN/B-PP/1 the/DT/B-NP/1 orange/NN/I-NP/1 juice/NN/I-NP/1 and/CC/O/2 take/VB/B-VP/2 it/PRP/B-NP/2 close/VB/B-VP/3 the/DT/B-NP/3 fridge/NN/I-NP/3 and/CC/O/4 bring/VB/B-VP/4 the/DT/B-NP/4 dreads/NNS/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4*

Nr. 2:

*comma/NN/B-NP/0 could/MD/B-VP/0 you/PRP/B-NP/0 please/VB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 car/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 table/NN/I-NP/0 and/CC/O/1 then/RB/B-ADVP/1 moved/VBD/B-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 open/VB/B-VP/2 it/PRP/B-NP/2 and/CC/O/3 then/RB/B-VP/3 pick/VB/I-VP/3 up/RP/B-PRT/3 into/IN/B-PP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3 so/RB/B-ADJP/3 sick/JJ/I-ADJP/3*

Nr. 3:

*hey/SYM/O/0 comma/NN/B-NP/0 go/VB/B-VP/0 to/TO/I-VP/0 cancel/VB/I-VP/1 tennis/NN/B-NP/1 take/VBP/B-VP/2 the/DT/B-NP/2 popcorn/NN/I-NP/2 bag/NN/I-NP/2 and/CC/O/3 bring/VB/B-VP/3 it/PRP/B-NP/3 to/TO/B-PP/3 me/PRP/B-NP/3*

Nr. 4:

*the/DT/B-NP/0 are/VBP/B-VP/0 mom/NN/B-NP/0 please/VBP/B-VP/1 bring/VB/I-VP/1 the/DT/B-NP/1 pop/NN/I-NP/1 on/IN/B-PP/1 back/RB/B-ADVP/1 from/IN/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1*

Nr. 5:

*armagh/NN/B-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 and/CC/O/0 bring/VB/B-VP/0 your/PRP\$/B-NP/0 insurance/NN/I-NP/0*

Nr. 6:

*can/MD/O/0 you/PRP/B-NP/0 grab/VB/B-VP/0 me/PRP/B-NP/0 the/DT/B-NP/0 box/NN/I-NP/0 of/IN/B-PP/0 pop/JJ/B-NP/0 corn/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 next/JJ/B-NP/0 to/TO/B-PP/0 the/DT/B-NP/0 cereal/NN/I-NP/0 box/NN/I-NP/0*

Nr. 7:

*can/MD/O/0 you/PRP/B-NP/0 please/VB/B-VP/0 bring/VB/I-VP/0 me/PRP/B-NP/0 orange/JJ/B-NP/0 juice/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 fridge/NN/I-NP/0 please./O/0*

Nr. 8:

*can/MD/O/0 you/PRP/B-NP/0 put/VB/B-VP/0 a/DT/B-NP/0 green/JJ/I-NP/0 path/NN/I-NP/0 from/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 to/TO/B-VP/0 end/VB/I-VP/1 of/IN/B-PP/1 the/DT/B-NP/1 session/NN/I-NP/1*

Nr. 9:

hey/SYM/B-NP/0 comma/NN/I-NP/0 please/VBP/B-VP/0 go/VB/I-VP/0 into/IN/B-PP/0 the/DT/B-NP/0 kitchen/NN/I-NP/0 taylor/NN/I-NP/0 on/IN/B-PP/0 the/DT/B-NP/0 right/JJ/I-NP/0 side/NN/I-NP/0 in/IN/B-PP/0 huntington/NN/B-NP/0 popcorn/NN/I-NP/0

Nr. 10:

hey/NN/B-NP/0 anna/NN/I-NP/0 please/VB/B-VP/0 take/VB/I-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 pop/NN/I-NP/0 open/VB/B-VP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 and/CC/O/2 put/VBD/B-VP/2 the/DT/B-NP/2 green/JJ/I-NP/2 cap/NN/I-NP/2 in/IN/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 and/CC/I-NP/3 stove/NN/I-NP/3 the/DT/B-NP/3 dishwasher/NN/I-NP/3

Nr. 11:

vingo/NN/B-NP/0 to/TO/B-PP/0 taylor/NN/B-NP/0 take/VB/B-VP/0 the/DT/B-NP/0 green/JJ/I-NP/0 car/NN/I-NP/0 then/RB/B-ADVP/0 go/VB/B-VP/1 to/TO/B-PP/1 the/DT/B-NP/1 dishwasher/NN/I-NP/1 open/VB/B-VP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2 and/CC/O/3 put/VBD/B-VP/3 the/DT/B-NP/3 green/JJ/I-NP/3 green/JJ/I-NP/3 cop/NN/I-NP/3 at/IN/B-PP/3 amc/NN/B-NP/3 in/IN/B-PP/3 the/DT/B-NP/3 dish/NN/I-NP/3 washer/NN/I-NP/3 um/UH/I-NP/3 and/CC/I-NP/4 killswitch/NNS/I-NP/4

Nr. 12:

i/PRP/B-NP/0 'm/VBP/B-VP/0 go/VB/I-VP/0 to/TO/I-VP/0 learn/VB/I-VP/1 to/TO/B-PP/1 french/JJ/B-NP/1 open/JJ/I-NP/1 french/JJ/I-NP/1 and/CC/I-NP/2 get/VB/B-VP/2 the/DT/B-NP/2 orange/NN/I-NP/2 juice/NN/I-NP/2 close/RB/B-ADVP/2 the/DT/B-NP/2 fridge/NN/I-NP/2 um/UH/B-VP/2 bring/VB/I-VP/3 the/DT/B-NP/3 arches/NNS/I-NP/3 to/TO/B-PP/3 me/PRP/B-NP/3

Nr. 13:

go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 in/IN/B-PP/0 front/NN/B-NP/0 of/IN/B-PP/0 the/DT/B-NP/0 window/NN/I-NP/0 uh/UH/I-NP/0 and/CC/O/1 take/VB/B-VP/1 the/DT/B-NP/1 popcorn/NN/I-NP/1 and/CC/O/2 bring/VB/B-VP/2 it/PRP/B-NP/2 to/TO/B-PP/2 me/PRP/B-NP/2

Nr. 14:

go/VB/B-VP/0 to/TO/I-VP/0 do/VB/I-VP/1 table/NN/B-NP/1 and/CC/O/2 take/VB/B-VP/2 to/TO/I-VP/2 cream/VB/I-VP/3 cup/NN/B-NP/3 note/NN/I-NP/3 bring/VB/B-VP/4 to/TO/I-VP/4 talk/VB/I-VP/5 to/TO/B-PP/5 the/DT/B-NP/5 dishwasher/NN/I-NP/5 first/RB/B-ADVP/5 open/VB/B-VP/6 the/DT/B-NP/6 dishwasher/NN/I-NP/6 for/IN/B-PP/6 pick/VB/B-VP/7 up/RP/B-PRT/7 inside/RB/B-ADVP/7 and/CC/O/8 then/RB/B-VP/8 close/VB/I-VP/8 it/PRP/B-NP/8 before/IN/B-PP/8

Nr. 15:

the/DT/B-NP/0 go/NN/I-NP/0 to/TO/B-PP/0 the/DT/B-NP/0 french/JJ/I-NP/0 open/JJ/I-NP/0 french/JJ/I-NP/0 and/CC/O/0 take/VB/B-VP/0 the/DT/B-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 bread/NN/I-NP/0 orange/NN/I-NP/0 juice/NN/I-NP/0 mean./O/0

Nr. 16:

go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 table/NN/I-NP/0 grab/VBP/B-VP/1  
 the/DT/B-NP/1 green/JJ/I-NP/1 top/NN/I-NP/1 turn/NN/I-NP/1 and/CC/O/2  
 go/VB/B-VP/2 to/TO/B-PP/2 the/DT/B-NP/2 dishwasher/NN/I-NP/2  
 if/IN/B-SBAR/3 it/PRP/B-NP/3 is/VBZ/B-VP/3 closed/JJ/B-NP/3 please/NN/I-NP/3  
 of/IN/B-PP/3 let/VB/B-VP/4 's/POS/B-NP/4 find/VB/B-VP/5 a/DT/B-NP/5  
 free/JJ/I-NP/5 place/NN/I-NP/5 put/VBD/B-VP/6 the/DT/B-NP/6 green/JJ/I-NP/6  
 tap/NN/I-NP/6 into/IN/B-PP/6 it/PRP/B-NP/6 and/CC/O/7 does/VBZ/B-VP/7  
 the/DT/B-NP/7 dishwasher/NN/I-NP/7

Nr. 17:

go/VB/B-VP/0 to/TO/B-PP/0 the/DT/B-NP/0 french/JJ/I-NP/0 open/JJ/I-NP/0  
 and/CC/I-NP/1 if/IN/B-SBAR/1 it/PRP/B-NP/1 's/VBZ/B-VP/1 coast/NN/B-NP/1  
 search/NN/I-NP/1 the/DT/B-NP/1 the/DT/I-NP/1 orange/NN/I-NP/1 juice/NN/I-NP/1  
 and/CC/O/2 grab/VB/B-VP/2 it/PRP/B-NP/2 close/VB/B-VP/3 it/PRP/B-NP/3  
 out/IN/B-PP/3 of/IN/B-PP/3 to/TO/B-PP/3 french/JJ/B-NP/3 toast/NN/I-NP/3 or/C-  
 C/O/3

Nr. 18:

i/PRP/B-NP/0 'm/VBP/B-VP/0 gonna/RB/B-ADVP/0 go/VB/B-VP/1 to/TO/B-PP/1  
 the/DT/B-NP/1 temple/NN/I-NP/1 of/IN/B-PP/1 come/VBN/B-VP/2  
 back/RB/B-ADVP/2

Nr. 19:

can/MD/O/0 you/PRP/B-NP/0 bring/VB/B-VP/0 the/DT/B-NP/0 popcorn/NN/I-NP/0  
 from/IN/B-PP/0 the/DT/B-NP/0 taylor/NN/I-NP/0

Nr. 20:

hello/SYM/O/0 babe/NN/B-NP/0 i/PRP/B-NP/0 need/VBP/B-VP/0 some/DT/B-NP/0  
 popcorn/NN/I-NP/0 can/MD/B-VP/0 you/PRP/I-NP/0 please/VB/I-NP/1 go/VB/I-  
 VP/1 to/TO/B-PP/1 the/DT/B-NP/1 table/NN/I-NP/1 grab/VB/B-VP/2 the/DT/B-  
 NP/2 book/NN/I-NP/2 on/IN/B-PP/2 bugs/NNS/B-NP/2 turn/VBP/B-VP/3 to/TO/I-  
 VP/3 turn/VB/I-NP/4 to/TO/B-PP/4 me/PRP/B-NP/4 and/CC/O/5 then/RB/B-VP/5  
 come/VB/I-NP/5 back/RB/B-ADVP/5 and/CC/O/6 the/DT/B-NP/6 man/NN/I-NP/6

## J. Heuristik zur Zuweisung von Befehlsnummern

```
public int [] InstructionNumber(String [] words, String [] pos) {
    if (words.length == pos.length) {
        int [] list = new int[words.length];
        int instrNr = 0;
        int verbCounter = 0;
        for (int i = 0; i < words.length; i++) {
            if (verbCounter == 0) {
                if (pos[i].startsWith("VB")
                    && i < words.length - 1
                    && pos[i+1].startsWith("VB")) {
                    list[i] = instrNr;
                    list[i+1] = instrNr;
                    i++;
                    verbCounter++;
                }
                else if (pos[i].startsWith("VB")){
                    list[i] = instrNr;
                    verbCounter++;
                }
                else {
                    list[i] = instrNr;
                }
            }
            else {
                //words which represent instruction boundaries
                if (words[i].toLowerCase().equals("and")
                    || words[i].toLowerCase().equals("if")
                    || words[i].toLowerCase().equals("when")) {
                    verbCounter = 0;
                    instrNr++;
                    list[i] = instrNr;
                }
                else if (pos[i].startsWith("VB")) {
                    instrNr++;
                    verbCounter = 0;
                    i--; //repeat loop
                }
                else {
                    list[i] = instrNr;
                }
            }
        }
        return list;
    }
    else {
        return null;
    }
}
```