Sprachverarbeitung jenseits der Spracherkennung

E. Nöth, J. Haas

1 Einleitung

In diesem Beitrag wollen wir Ansätze zur Interpretation gesprochener Sprache präsentieren. Wir zeigen zunächst, warum die direkte Übertragung der linguistischen Ansätze zur Interpretation von geschriebener Sprache auf gesprochene Sprache in der Regel nicht erfolgreich ist. Dies führt uns zur Entwicklung eines Konzepts für partielles Parsen, welches stochastische Verfahren und prosodische Information verwendet. Wir präsentieren vorläufige Ergebnisse für die einzelnen Wissensquellen, ein Test des Gesamtkonzepts steht noch aus.

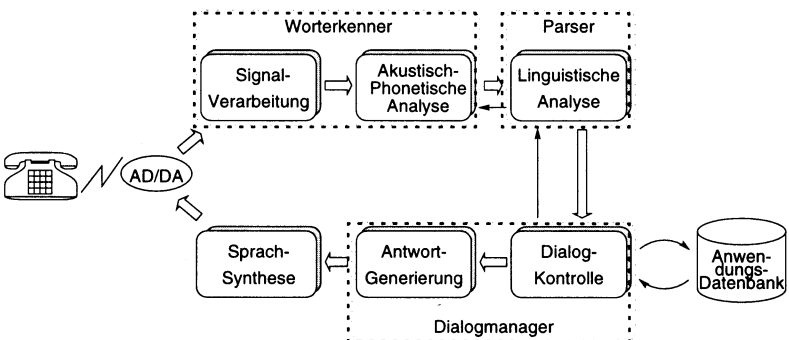

Abbildung 1: Systemarchitektur für ein sprachverstehendes Dialogsystem

Abbildung 1 zeigt die Systemarchitektur eines prototypischen Dialogsystems zum Führen von informationsabfragenden Dialogen. Das System wurde im Rahmen mehrerer Verbundprojekte am Lehrstuhl für Mustererkennung entwickelt und hat als Anwendungsgebiet die Auskunft über Intercity-Zugverbindungen. Mit dem System, das seit ca. vier Jahren über das öffentliche Telofonnetz zugänglich (09131/16287) ist, wurden mittlerweile ca. 3000 Dialoge gesammelt. Eine ausführliche Beschreibung des Systems findet sich z.B. in [Eck96,BAG97].

2 Linguistische Analyse

Im Rahmen dieses Beitrages beschäftigen wir uns mit dem Modul Linguistische Analyse. Eingabe des Moduls ist das Ergebnis der Worterkennung. Abbildung 2 zeigt mögliche Schnittstellen zwischen der Worterkennung und der linguistischen Analyse: Sie kann entweder die am besten passende Wortkette, ein Wortgitter oder ein Worthypothesen-

graph sein. Einer Worthypothese ist ein Anfangs- und Endezeitpunkt sowie ein Maß dafür zugeordnet, wie gut sie zu dem dazugehörigen akustischen Signal paßt. Die Worthypothesen in einem Gitter sind nicht verbunden. Mögliche Vorgänger und Nachfolger einer Worthypothese müssen während der linguistischen Analyse im Gitter erst gesucht werden. Ein Graph kann als spezielles Gitter angesehen werden, bei dem nur solche Worthypothesen betrachtet werden, welche auf einem Pfad liegen, der das Signal überdeckt. Eine Worthypothesenkette ist der Spezialfall eines linearen Worthypothesengraphen und ein Worthypothesengraph ist andererseits eine effiziente Kodierung der n am besten passenden Ketten, wobei n die Zahl der möglichen Pfade durch den Graphen ist. Für den Rest dieses Beitrages gehen wir von einem Worthypothesengraphen als Schnittstelle zur linguistischen Analyse aus.

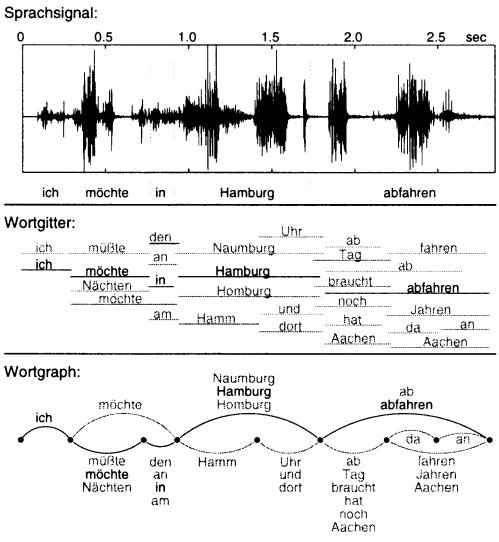

Abbildung 2: Mögliche Schnittstellen zwischen der Worterkennung und der linguistischen Analyse

Aufgabe der linguistischen Analyse ist es, den optimalen Pfad durch den Graph zu suchen in Bezug auf akustische Güte und linguistische Zulässigkeit. Hierfür werden meistens regelbasierte Grammatiksysteme verwendet, welche Worthypothesen zu höheren Konstrukten verknüpfen. Typische Ansätze sind beispielsweise HPSG (Head driven Phrase Structure Grammar) [PS87] und TUG (Trace Unification Grammar) [Sik93].
Die Ansätze basieren darauf, daß linguistisches (syntaktisches und semantisches) Wissen über die Wörter in einem Lexikon kodiert ist.

```
/* Regel zur Kombination von Artikel und Nomen */
          np ===> det, n if

/* Bedingungen */
1)        det:syn:head:major:artikel,
2)        n:syn:head:major:nomen,

 a)       [det,syn,head,kasus]    == [n,syn,head,kasus],
3b)       [det,syn,head,numerus]  == [n,syn,head,numerus],
 c)       [det,syn,head,genus]    == [n,syn,head,genus],

/* Aktionen beim Feuern der Regel */
4)        np:syn:head:major:nominal_phrase,

 a)       [np,syn,head,kasus]     == [det,syn,head,kasus],
5b)       [np,syn,head,numerus]   == [det,syn,head,numerus],
 c)       [np,syn,head,genus]     == [det,syn,head,genus],

6)        [np,sem]                == [n,sem].
```

Abbildung 3: Beispielregel einer Phrasenstrukturgrammatik

Abbildung 3 zeigt eine Regel, die es erlaubt, zwei Worthypothesen "det" und "n" zu einer Nominalphrase (Artikel gefolgt von einem Nomen) zusammenzubauen. Die Regel kann feuern, wenn der Bedingungsteil erfüllt ist (die Nummern beziehen sich auf die Nummern im Bild):

1) Der Syntaxeintrag im Lexikon für die Worthypothese "det" besagt, daß es sich um einen Artikel handelt.
2) Der Syntaxeintrag im Lexikon für die Worthypothese "n" besagt, daß es sich um ein Nomen handelt.
3a) - 3c) Die Syntaxeinträge im Lexikon für die beiden Hypothesen stimmen in Bezug auf Kasus, Numerus und Genus überein.

Falls die Regel feuert, wird ein neues Konstrukt "np" erzeugt, das folgende Eigenschaften hat:

4) Der Syntaxeintrag besagt, daß es sich um eine Nominalphrase handelt.
5a) - 5c) "np" übernimmt den Eintrag für Kasus, Numerus und Genus von "det".
6) "np" übernimmt die semantische Bedeutung von "n".

Mit Hilfe solcher Regeln kann man aus den Worthypothesen immer größere Konstrukte erzeugen, bis dadurch das Eingabesignal komplett überspannt wird. Das Kontrollmodul,

welches entscheidet, welche Regel für welche Teilkonstrukte als nächstes angewendet wird, nennen wir den Parser. Ziel von vielen Projekten im Bereich der geschriebenen Sprache ist es, Grammatiken und Parser zu entwickeln, welche einen möglichst hohen Anteil von möglichst unbeschränktem Text verarbeiten können. Diese Ansätze setzen allerdings meistens eine fehlerfreie und syntaktisch korrekte Eingabe voraus. Bereits die Annahme der Fehlerfreiheit ist für gesprochene Sprache nicht gegeben. Selbst die weltweit besten Erkenner sind in Bezug auf Fehlerfreiheit über sehr viele Anwendungen hinweg eine Größenordnung oder mehr schlechter als der Mensch [Lip97]. In unserem System ist die Erkennungsrate auf der am besten passenden Wortkette nur ca. 80%, d.h. zwei von zehn hypothetisierten Wörtern sind falsch, und ca. 4% der Wörter, die die Benutzer äußern, sind nicht im Erkennungslexikon des Systems [BAG97]. Auch die zweite Annahme ist nicht unproblematisch. Betrachten wir den folgenden Beispielsatz aus unserer Anwendung:

"<äh> ja grüß Gott ich würd' gern morgen nach Hamb <äh> nach Hannover fahren weil meine Großtante Elfriede da ihren <Pause> ich glaub' ihren fünfundachzigsten feiert"

Wie das Beispiel zeigt, müssen wir beim Parsen von spontaner Sprache folgende Phänomene modellieren:

1) ungrammatikalische Äußerungen
2) Nonverbalien (z.B. <äh>) und Geräusche
3) unbekannte Wörter
4) Abbrüche und Wiederaufnahmen

Allerdings ist auch ein vollständiges Parsen der Äußerung für die Anwendung überhaupt nicht notwendig. Es genügt für die Weiterführung des Dialogs, wenn wir die Äußerung in eine anwendungsabhängige semantische Repräsentation etwa in der folgenden Art überführen:

[Zielort:Hannover][relative_Tagesangabe:morgen]

Die semantisch wichtigen Einheiten nennen wir semantische Attribute. Statt die Verfahren der tiefen linguistischen Analyse robust gegen die oben genannten Phänomene zu machen, propagieren wir folgende Vorgehensweise:

1. Entwicklung von Grammatik-Fragmenten, welche die für die Anwendung notwendigen semantischen Attribute abdecken. Für unsere Anwendung sind dies Grammatikfragmente für "Zeitausdrücke", "Ortsangaben", etc.
2. Detektion semantischer Attribute mit einem "Orakel", also einem Verfahren, welches uns sehr schnell sagen kann, welche Grammatikfragmente wir für eine Äußerung verwenden sollen. In obigem Beispiel wären dies die Fragmente für "Ortsangaben" und "Zeitausdrücke". Wir werden für diese Aufgabe stochastische Klassifikatoren verwenden.

3. Lokalisierung von Startpunkten für einen lokalen Parser, also einen Parser, welcher an einer beliebigen Stelle im Worthypothesengraphen beginnt und nur so lange mit den Regeln der relevanten Grammatikfragmente größere Konstrukte erstellt, bis die semantischen Attribute gefunden werden. Wir werden zeigen, wie wir neben lexikalischer Information (syntaktische und semantische Wortklassen) auch prosodische Information über die betonten Stellen des Sprachsignals verwenden können.
4. Finden der korrespondierenden Werte zu den semantischen Attributen unter Verwendung der Grammatikfragmente und des lokalen Parsers.

Im folgenden wollen wir kurz die Funktionsweise des stochastischen "Orakels" andeuten.

3 Stochastische Attributdetektion

Als "Orakel" benutzen wir sogenannte n-gramm Sprachmodelle [Jel90]. Wir gehen davon aus, daß uns ein Trainingskorpus zur Verfügung steht, für das wir zu jeder Äußerung eine Markierung der darin enthaltenen semantischen Attribute haben. Für alle Wortfolgen der Länge n in der Transliteration erhöhen wir dann einen Zähler, der auf die in der Äußerung enthaltenen semantischen Attribute hinweist. Sei n=3, so würden wir z.B. für "ja grüß Gott", "grüß Gott ich", "Gott ich würde", ... den Zähler für "Zielort" und "relative_Tagesangabe" erhöhen. Auf diese Weise können wir einen statistischen Klassifikator trainieren, der uns für jede Wortfolge angibt, welche Attribute darin enthalten sind. Diese Vorhersagen des semantischen Gehalts können wir anschließend verwenden, um die entsprechenden Grammatikfragmente auf die Äußerung anzuwenden.
Für die von uns durchgeführten Experimente verwenden wir ca. 10000 Äußerungen, die mit unserem Zugauskunftssystem [Eck96,BAG97] gesammelt wurden. Für die stochastische Attributdetektion konzentrieren wir uns auf die drei semantischen Attribute CITY, DATE und TIME. Mit CITY sind alle Wortfolgen gekennzeichnet, die einen syntaktischen Konstrukt mit einem Ortsnamen enthalten. DATE umfaßt alle möglichen Datumsangaben wie beispielsweise die relative Angabe des Datums *morgen* aber auch die explizite Nennung *am ersten neunten*. Alle Zeitangaben sind mit dem semantischen Attribut TIME markiert, wobei auch hier die Bandbreite von relativen Zeitangaben *später* bis zu konkreten Zeitpunkten *um neun Uhr neun* reicht.
Die Klassifikationsaufgabe, die wir mittels der n-gramm Sprachmodelle zu lösen versuchen, besteht nun darin, daß wir nicht nur entscheiden wollen, welche Attribute in einer Wortfolge auftreten, sondern wir wollen auch Information über den weiteren semantischen Gehalt gewinnen. Konkret wollen wir für jedes Attribut folgendes unterscheiden:

1) Das Attribut tritt in der Wortfolge auf, und ansonsten keine weiteren semantischen Attribute CITY1, DATE1, TIME1.
2) Das Attribut tritt in der Wortfolge auf, und dazu noch weitere semantische Attribute CITY+, DATE+, TIME+.
3) Das Attribut tritt in der Wortfolge nicht auf NOCITY, NODATE, NOTIME.

Dazu trainieren wir wie oben beschrieben drei n-gramm Sprachmodelle, das erste mit allen Wortfolgen, deren semantische Repräsentation lediglich einen einzigen CITY Eintrag haben, das zweite mit den Wortfolgen, deren semantische Repräsentation einen CITY Eintrag und mindestens einen weiteren Eintrag haben und das dritte mit solchen Wortfolgen, die in ihrer Repräsentation keinen CITY Eintrag haben. Dieselbe Vorgehensweise übernehmen wir natürlich auch für die Orakel, die DATE und TIME vorhersagen.
Wenn wir nun eine Wortfolge zur Analyse vom Spracherkenner geliefert bekommen, berechnen wir für die drei Orakel jeweils die Wahrscheinlichkeit, daß der aktuelle Satz zu dem n-gramm Sprachmodell paßt. Das Modell, das die höchste Wahrscheinlichkeit hat, gewinnt, und wir entscheiden uns für die entsprechende semantische Klasse. Die Erkennungsraten, die wir mit dieser Methode für die drei semantischen Attribute CITY, DATE und TIME erreichen, sind in folgender Tabelle dargestellt.

	gesprochen				erkannt			
	1	+	NO	RR	1	+	NO	RR
CITY1	**82**	17	0		**52**	29	19	
CITY+	10	**90**	0	**86**	11	**79**	11	**74**
NOCITY	6	9	**85**		8	14	**78**	
DATE1	**92**	8	0		**72**	11	17	
DATE+	12	**84**	5	**85**	6	**60**	34	**79**
NODATE	6	9	**85**		5	13	**83**	
TIME1	**91**	9	0		**71**	11	18	
TIME+	9	**89**	2	**85**	6	**66**	28	**81**
NOTIME	7	8	**85**		7	9	**84**	

Tabelle 1: Klassifikation semantischer Attribute mit n-gramm Klassifikatoren

Wie wir sehen, ist eine Vorhersage des semantischen Inhalts einer Wortfolge mit n-gramm Sprachmodellen durchaus in einer akzeptablen Qualität möglich, auch auf der erkannten Kette, bei der, wie oben beschrieben, durchschnittlich zwei von zehn hypothetisierten Wörtern falsch sind. Für unser angedachtes Gesamtsystem ist diese Vorgehensweise zur Detektion semantischer Attribute und damit einer Vorauswahl der einzusetzenden Grammatikfragmente sehr gut geeignet, insbesondere, da eine Fehlentscheidung im günstigen Fall lediglich dazu führt, daß erst mit einem falschen Grammatikfragment erfolglos und danach mit dem richtigen Fragment nach einem enthaltenen semantischen Attribut gesucht wird.

4 Finden von Startpunkten für die Grammatikfragmente

Der nächste Schritt in der Analyse des Sprachsignals ist die Bestimmung von Startpunkten für die ausgewählten Grammatikfragmente. Eine Möglichkeit ist die Verwendung

lexikalischer Information über syntaktische und semantische Klassen der Worthypothesen. Eine Regel könnte z.B. lauten:

Falls ein Zeitausdruck gesucht wird, starte an jedem Zahlwort eine Analyse

Hier betrachten wir die Verwendung von prosodischer Information über betonte Stellen im Sprachsignal zum Finden von Startpunkten. Wir gehen davon aus, daß die Benutzer eines Dialogsystems kooperative Sprecher sind und - ebenso wie in Mensch-Mensch-Dialogen - die für die Interpretation der Äußerung wichtigen Wörter hervorheben.
In [Kie97,Kom97] beschreiben wir ein Verfahren, mit dem für jede Worthypothese die Wahrscheinlichkeit berechnet wird, daß die Worthypothese betont ist. Hierfür werden Merkmale berechnet, welche den Tonhöhenverlauf, die Energie-Kontur, die Dauerverhältnisse, die Pausen und die Sprechgeschwindigkeit beschreiben. Die Merkmale werden als Eingabe in ein Neuronales Netz benutzt. Die beiden Klassen betont/unbetont werden mit 83% erkannt. Das Verfahren wurde auf der oben erwähnten Stichprobe von Mensch-Maschine-Dialogen getestet.

"immer" betonte Wörter (Betonungswahrscheinlichkeit >0.8 bei >80% der Beobachtungen)		
Rang	Wort alle Äußerungen	Wort TIME+ Äußerungen
1	Nürnberg	nein
2	Verbindung	zehn
3	Freitag	zwölf
4	Stuttgart	achtzehn
5	Frankfurt	sechzehn
6	Zugverbindung	Nürnberg
7	zehn	Erlangen
8	Sonntag	fünfzehn
9	Samstag	dreizehn
10	zwölf	neunzehn
11	Montag	siebzehn
12	Mittwoch	vierzehn
13	Würzburg	nachmittag
14	Bamberg	Freitag
15	achtzehn	dreiundzwanzig

Tabelle 2: "Immer" betonte Wörter, geordnet nach der Zahl der Beobachtungen

Tabelle 2 zeigt - geordnet nach der Zahl der Beobachtungen - die Wörter, die für alle Äußerungen und für die TIME+ Äußerungen fast immer (hier: in mehr als 80% der Beobachtungen) als betont klassifiziert wurden. Man erkennt deutlich, daß alle diese Wörter für die Anwendung "Zugauskunft" sehr wichtig sind. Bei den TIME+ Äußerungen fällt auf, daß das häufigste Wort *nein* ist. Dies läßt sich damit erklären, daß bei Korrekturen (z.B. *nein, um siebzehn Uhr* das Wort *nein* ein "Dialogmarker" und Korrektursignal ist und

vom Benutzer stark betont wird. Außer den Wörtern Erlangen, Nürnberg und Freitag sind alle weiteren Wörter Zahlwörter und Teil eines Zeitausdrucks. Betrachten wir für die gesprochenen Wörter die automatisch berechneten Betonungsmaße, so zeigt sich, daß in 81% der Fälle das absolute Maximum der Betonungsbewertungen aller Wörter innerhalb der Zeitangabe liegt, und in 18% der Fälle ein relatives Maximum innerhalb der Zeitangabe liegt.

5 Zusammenfassung und Ausblick

Automatische sprachverstehende Systeme gewinnen zunehmend an Bedeutung. Sobald man von Benutzerseite eine freie Eingabe zuläßt, muß ein solches System spontansprachliche Phänomene verarbeiten können. Traditionelle Ansätze setzen allerdings häufig eine - im Sinne einer Grammatik für geschriebene Sprache - korrekte Eingabe voraus. Alternativ zur Erweiterung von bestehenden Grammatiken in Richtung auf Robustheit gegen spontansprachliche Phänomene schlagen wir die Entwicklung von anwendungsorientierten Grammatikfragmenten vor. Wir zeigen, daß es möglich ist, mit stochastischen Klassifikatoren die in einer Äußerung enthaltenen semantischen Attribute mit einer hohen Sicherheit vorherzusagen. Die Position der semantischen Attribute innerhalb der Äußerung kann mit Betonungsinformation ebenfalls sicher vorhergesagt werden. Zur Zeit arbeiten wir daran, die statistische Vorhersage und die Betonungbewertung in das Parsing zu integrieren und die anwendungsabhängigen Grammatikfragmente zu entwickeln.

Literaturverzeichnis

[BAG97] Boros, M. / Aretoulaki, M. / Gallwitz, F. / Nöth, E. / Niemann, H.: *Semantic Processing of Out-Of-Vocabulary Words in a Spoken Dialogue System.* In: Proc. European Conf. on Speech Communication and Technology. Band 4, Rhodes 1997, S. 1887-1890.

[Blo97] Block, H.U.: *The Language Components in Verbmobil.* In: Proc. Int. Conf. on Acoustics, Speech and Signal Processing. Band 1, München 1997, S. 79-82.

[Eck96] Eckert, W.: *Gesprochener Mensch-Maschine-Dialog. Berichte aus der Informatik.* Aachen 1996.

[Han96] Hanrieder, G.: *Inkrementelles Parsing gesprochener Sprache mit einer linksassoziativen Unifikationsgrammatik.* Dissertation, Universität Erlangen-Nürnberg 1996.

[Jel90] Jelinek, F.: *Self-organized Language Modeling for Speech Recognition.* In: Readings in Speech Recognition. Hsg. v. A. Waibel / K.-F. Lee, San Mateo 1990, S. 450-506.

[Kie97] Kießling, A.: *Extraktion und Klassifikation prosodischer Merkmale in der automatischen Sprachverarbeitung. Berichte aus der Informatik.* Aachen 1997.

[Kom97] Kompe, R.: *Prosody in Speech Understanding Systems. Lecture Notes for Artificial Intelligence.* Berlin 1997.

[Lip97] Lippmann, R. P.: *Speech Recognition by Machines and Humans.* In: Speech Communication. 22(1), 1997, S. 1-15.

[PS87] Pollard, C. / Sag, I.: *Information-Based Syntax and Semantics.* Stanford 1987.

[Sik93] Sikkel, N.: *Parsing Schemata.* CIP-Gegevens Koninklijke Bibliotheek 1993.