Der mixed-initiative Ansatz als Basis für benutzerfreundliche Sprachdialogsysteme

Axel Horndasch, Florian Gallwitz, Jürgen Haas, Elmar Nöth

Im Rahmen dieses Beitrags wird beschrieben, welche Eigenschaften automatische Sprachdialogsysteme haben sollten, um Anforderungen hinsichtlich einer angenehmen und effizienten Interaktion zu erfüllen. Dabei wird für das Systemdesign der mixed-initiative Ansatz als optimal betrachtet, da die damit erreichbare Benutzerfreundlichkeit für Sprachanwendungen die Akzeptanz von Sprachdialogsystemen bei Firmen und Endbenutzern erheblich verbessern kann. Es werden die Herangehensweise bei der Entwicklung neuer Applikationen sowie das dazu notwendige Know-how und hilfreiche Werkzeuge beleuchtet, zudem stellen die Autoren die erörterten theoretischen Konzepte anhand von Systemen, die kommerziell eingesetzt werden, beispielhaft dar.

1 Einleitung

Seit Jahren werden der automatischen Spracherkennung im Allgemeinen und Telefon-Dialogsystemen im Besonderen große Wachstumsraten im kommerziellen Bereich vorausgesagt. In einer 2004 veröffentlichten Studie des renommierten Marktforschungsinstituts Gartner [1] wird die Spracherkennung für Telefonie und Call-Center zwar noch zum „Emerging Technologies Hype Cycle" gezählt, allerdings bescheinigt man ihr gleichzeitig auch den Reifegrad der „Produktivität" (andere Reifegrade u.a.: „Technologieauslöser", „Desillusion").

Trotz der positiven Prognosen bringen viele Firmen und Endbenutzer dem Einsatz bzw. dem Gebrauch von Dialogsystemen immer noch eine gewisse Skepsis entgegen. Ein anderes Problem stellt der relativ geringe Bekanntheitsgrad von kommerziell verfügbaren Sprachanwendungen dar: in einer Studie des Fraunhofer-Instituts für Arbeitswirtschaft und Organisation (IAO) bejahten nur 54% die Frage „Kennen Sie Sprachapplikationen?" [2]. Die Autoren des IAO halten als Konsequenz neben „umfassender Öffentlichkeitsarbeit und Marketingkampagnen... insbesondere gute Beispielapplikationen" für notwendig.

Im Rahmen dieses Beitrags wird beschrieben, welche Eigenschaften diese „guten Beispielapplikationen" haben sollten, um Anforderungen hinsichtlich einer angenehmen und effizienten Interaktion zu erfüllen. Dabei wird für das Systemdesign der mixed-initiative Ansatz als optimal betrachtet, da die damit erreichbare Benutzerfreundlichkeit für Sprachanwendungen die Akzeptanz von Sprachdialogsystemen bei Firmen und Endbenutzern erheblich verbessern kann. Die Herangehensweise bei der Entwicklung neuer Applikationen sowie das dazu notwendige Know-how und hilfreiche Werkzeuge finden sich in Abschnitt 3. Im vierten Abschnitt werden die erörterten theoretischen Konzepte anhand von Systemen, die kommerziell eingesetzt werden, beispielhaft dargestellt.

2 Anforderungen an Dialogsysteme

Praktisch alle Dialogsysteme, die heutzutage kommerziell eingesetzt werden, sind entweder Informationssysteme (der Benutzer benötigt eine Information wie z.B. eine Flugverbindung) oder Transaktionssysteme (der Benutzer will eine Transaktion durchführen, beispielsweise ein Produkt bestellen, mit einer bestimmten Person verbunden werden oder Geld überweisen). Die Domäne für automatische Dialogsysteme ist in den meisten Fällen sehr eingeschränkt; das bedeutet, dass ein System dem Benutzer oft nur eine bestimmte Anwendung anbietet.

Trotz dieser Einschränkung stoßen klassische, in Menüs strukturierte IVR-Systeme, die nur auf Touchtone- oder Einzelworterkennung aufbauen, schnell an Grenzen: zu viele Alternativen in einer Systemausgabe führen dazu, dass der Benutzer den Überblick verliert. Ein hierarchischer Menü-Ansatz, der einer hohen Komplexität eher gerecht wird, hat wiederum den Nachteil, dass die Navigation aufwändiger wird und die Frustration, z.B. bei einer Rückkehr zum Hauptmenü wegen eines Fehlers, zunimmt.

Ein „gutes" Dialogsystem sollte also in der Lage sein, natürlich gesprochene Sprache zu erkennen und zu verstehen. Für die Erkennung selbst bedeutet dies, dass ein Spracherkenner eingesetzt werden muss, der sich auch robust gegenüber spontan-sprachlichen Benutzeräußerungen zeigt. Hinsichtlich der Sprachmodellierung bietet es sich an, stochastische Grammatiken zu verwenden. Aktuelle Arbeiten zum Thema Spracherkennung, in denen auch der Umgang mit unbekannten Wörtern und der Variabilität bei der Erzeugung von Sprache behandelt wird, sind [3] und [4].

Um die Interaktion so natürlich wie möglich und damit benutzerfreundlich zu machen, müssen Besonderheiten, die in einem Dialog auftreten können, bedacht und durch ein intelligentes Dialogmanagement modelliert werden:

- Überbeantwortung von Fragen
  System: „Wohin wollen Sie fahren?"
  Benutzer: „Nach Hamburg, mit dem Zug um 17 Uhr."

Künstliche Intelligenz, Heft 3/2005, Seite 38-41,
ISSN 0933-1875, Böttcher IT Verlag, Bremen

- Verarbeitung von Out-Of-Focus-Antworten
  System: „Wann wollen Sie nach Hamburg fahren?"
  Benutzer: „Ich möchte übrigens drei Tickets buchen."

- Unterbrechen der Systemansage durch den Benutzer („Barge-In")

- Referenzen im Dialogkontext
  System: „Bayern München hat 2:2 gespielt."
  Benutzer: „Wo stehen die in der Tabelle?"

Ein Dialogsystem, das mit den aufgeführten Phänomenen umgehen kann, bietet dem Benutzer große Freiheiten. Allerdings ist das sture Festhalten am Konzept „say what you want at any time you want to" nicht immer sinnvoll: Kann ein Dialog in mehrere Teilaufgaben unterteilt werden (bei der Flugbuchung z.B. „Flugauswahl", „Eingabe der Daten für die Bezahlung") ist es oft besser, keinen direkten Wechsel von einem Subdialog in den anderen vorzusehen. Dadurch wird vermieden, dass Fehler bei der Spracherkennung zu schwer nachvollziehbaren Systemreaktionen führen. Mit zunehmender Leistungsfähigkeit der Spracherkennungstechnologie werden solche Beschränkungen des Dialogs in Zukunft aber immer weniger notwendig sein.

Durch die Anpassung des Prompting an die jeweilige Dialogsituation, z.B. abhängig vom Benutzerzustand, der aktuellen Qualität der Spracherkennung oder auch den Anforderungen an Sicherheit und Korrektheit der Eingabe, können weitere Feinheiten der Mensch-Mensch-Kommunikation nachempfunden werden. So kann ein erfolgreicher Dialogverlauf zu einer Relaxation des Interaktionsstils führen (das System generiert freiere Prompts, beispielsweise „Wie kann ich Ihnen noch helfen?"), bei Verständnisproblemen oder sicherheitsrelevanten Eingaben hingegen werden restriktivere Systemausgaben erzeugt („Wollen Sie nach Hamburg oder nach Homburg?", „Bitte buchstabieren Sie Ihr Kundenkennwort!"). In diesem Zusammenhang sind auch dynamische Bestätigungsstrategien zu sehen, die der jeweiligen Situation entsprechend entweder ganz wegfallen, bzw. implizit („Wann wollen Sie nach Hamburg?") oder explizit („Wollen Sie nach Hamburg?") sind.

Wünschenswert für jede Anwendung ist die Integration von allgemeinem und anwendungsspezifischem Wissen in das System: von der einfachen Prüfung hinsichtlich der Konsistenz der Eingabe (z.B. Datumsangaben, Kreditkartennummern) über die Interpretation mehrdeutiger Benutzeräußerungen (was bedeutet „acht Uhr" bei einer Kinoreservierung bzw. bei einem automatischen Wecksystem) bis hin zur Anpassung der Dialogstrategie auf Grund von Datenbankabfragen während des Dialogs (z.B. genügt in kleineren Städten oft schon der Ortsname zur sinnvollen Präsentation des Abendkinoprogramms).

Automatische Sprachdialogsysteme, die alle oder zumindest einen Großteil der bis hierhin erwähnten Eigenschaften aufweisen, können als mixed-initiative Systeme bezeichnet werden. Durch sie ist es einerseits möglich, unerfahrene Benutzer Schritt für Schritt durch einen Dialog zu führen. Andererseits können „Poweruser" durch das Nennen von mehreren Informationen in einer Äußerung mit demselben System sehr effizient zum Ziel gelangen.

3 Dialogsysteme der Praxis

Bei der Entwicklung neuer Applikationen ist die Kenntnis von anwendungsspezifischen Details von großer Bedeutung. Zwar bilden Erfahrungen aus vorangegangenen Projekten und umfassende Untersuchungen des Benutzerverhaltens im Vorfeld (z.B. durch die Befragung von Call-Center-Mitarbeitern zum jeweiligen Thema) eine gute Grundlage. Eine Verbesserung des Systems durch eine iterative Optimierung spielt jedoch eine wichtige Rolle. Ein typisches Beispiel für einen Entwicklungsablauf:

- Initiale Entwicklung und Test der Einzelkomponenten des Systems (Dialog, Erkennung, Anbindung von Telefonie/Datenbanken, etc.)
- Zusammenführung der Einzelkomponenten zu einem ersten anrufbaren System
- Testen des Systems durch die Entwickler (erfahrene, voreingenommene Benutzer)
- Testen durch „Freunde und Familie" (relativ erfahrene, unvoreingenommene Benutzer)
- Testen durch „freundliche" Benutzer (unerfahrene, unvoreingenommene, kooperative Benutzer)
- System allgemein verfügbar machen (alle Benutzer, evtl. aus einer bestimmten Zielgruppe)
- Optimierung und Anpassung (während aller Testphasen und des laufenden Betriebs)

Um schon in der Anfangsphase der Entwicklung gut testen zu können müssen einige technische Gegebenheiten der Einzelmodule und des Gesamtsystems gewährleistet sein. Bei der Verarbeitungsgeschwindigkeit einer Äußerung ist zu berücksichtigen, dass eine zu lange Wartezeit den Benutzer irritiert und die Synchronität des Gesprächs erheblich stört. In [5] bzw. [6] wird mit 2-3 Sekunden ein Schwellwert für diese Wartezeit angegeben. Idealerweise sollte sich die Antwortzeit aber zwischen 0,6 und 0,9 Sekunden bewegen. Hierbei spielen unter anderem die Latenzzeiten beim Zugriff auf Datenbanken, die Pufferung von Sprachdaten z.B. bei VoIP (jitter buffer) und die Generierung von Systemprompts eine Rolle. Auch eine robuste Detektion von Anfang und Ende einer Spracheingabe ist notwendig, um eine schnelle Reaktion des Systems zu ermöglichen.

Ein Großteil des zeitlichen Aufwandes besteht in der Erkennung, allerdings kann mit der Verarbeitung des Sprachsignals schon begonnen werden, während der Benutzer noch spricht. Um zumutbare Antwortzeiten zu garantieren, bieten moderne Spracherkenner ein Feature, das den Berechnungsaufwand von der verfügbaren Rechenkapazität abhängig macht. Der Suchraum für die beste Wortkette bzw. den besten Wortgraphen wird dann gegebenenfalls eingeschränkt. Die bei Lastspitzen mögliche, etwas geringere Erkennungsgenauigkeit wird deswegen in Kauf genommen, weil sie den Gesprächspartner weniger verwirrt als eine zu lange Wartezeit.

Die Robustheit der Spracherkennung mit Blick auf eine bestimmte Anwendung wird entscheidend von der Anpassung des Wortschatzes, der erkannt werden kann, mitbestimmt. In den einzelnen Testphasen ist deshalb darauf zu achten, wie Benutzer ihre Wünsche äußern, welche Synonyme für bereits integrierte Begriffe auftreten, etc. Neben diesen Ergebnissen, die auch für die inhaltliche Analyse verwendet werden können, spielen auch evtl. aufgenommene Benutzeräußerungen eine Rolle beim Fine-Tuning des Erkenners. Mit diesen Sprachdaten und einem Nachtraining der

Erkennerparameter lassen sich oftmals Verbesserungen bei der Erkennungsrate erzielen.

Um die Dialogkomponente für eine neue Applikation entsprechend zu konfigurieren, werden heutzutage oft Diagramme eingesetzt, die den Gesprächsfluss modellieren. Je nach Anspruch im Hinblick auf den mixed-initiative Ansatz können diese Flussdiagramme sehr komplex werden, es sei denn sie beschreiben nur sehr primitive Dialoge. Mit Hilfe von Call-Flow-Editoren ist eine automatische Umsetzung dieser Modelle zwar möglich, aber oft ist noch weitere Implementierungsarbeit in einer Programmiersprache (z.B. C++, Java) nötig. Ein weiterer Nachteil betrifft die Erweiterbarkeit eines so erstellten Systems. Wenn die Abfrage von weiteren Informationseinheiten im Laufe eines Projektes erforderlich wird, wirken sich die Veränderungen oft stark auf den programmierten Dialogfluss aus.

Abhilfe kann eine Dialog-Engine leisten, bei der der Dialogablauf nicht explizit vorgegeben wird, sondern die zur Laufzeit eine Systemantwort berechnet. Als Eingangsgrößen dienen neben der aktuellen Benutzeräußerung die Belegung der Informationseinheiten (auch bezeichnet als „Slots") und der Datenbankinhalt. Anwendungsspezifische Slots werden bei einem im Kern applikationsunabhängigen Dialogmanager beispielsweise in einer Konfigurationsdatei angegeben. Die in Abschnitt 4 beschriebene Kinoanwendung verwendet z.B. die Slots Zeit, Datum, Filmtitel, Stadt, Kinoname. Andere (zum Teil slotspezifische) Informationen, die auch in der Konfiguration einer Anwendung enthalten sind, umfassen Systemprompts, Grammatikregeln, Synonymtabellen, initiale Bestätigungsstrategie, etc.

Je weiter die Entwicklung einer neuen Sprachapplikation voranschreitet, desto wichtiger ist das sogenannte „hear and feel", also die Außenwirkung einer Anwendung. Einen wichtigen Beitrag hierzu leistet eine gute Systemausgabe. Neben der Wortwahl bei den Prompts, ist vor allem die Stimme des Systems von entscheidender Bedeutung. Um eine hohe Qualität zu erreichen ist für kommerziell eingesetzte Systeme der Einsatz von vorab aufgenommenen Sprachschnipseln mit professionellen Sprechern unabdingbar. Nur für die Präsentation stark dynamischer Inhalte sollte ein Sprachsynthese-Modul eingesetzt werden. Mit den Prompts selbst ist es möglich, das Verhalten und die Befindlichkeit des Benutzers zu beeinflussen. So können Systemansagen wie „Bitte sprechen Sie laut und deutlich!" dazu führen, dass die Kooperativität leidet, weil der Benutzer verärgert wird („Ich spreche laut und deutlich, du verstehst mich nur nicht!").

Im Bundesligainformationssystem, das in Abschnitt 4 beschrieben wird, wird der akustische Eindruck der Applikation auf den Anrufer noch durch das Einspielen einer Stadionatmosphäre unterstrichen. Weitere Möglichkeiten in diesem Zusammenhang bieten sogenannte „Earcons", die von Anrufern mit bestimmten Aktionen in Verbindung gebracht werden.

Als Fazit dieses Abschnitts lässt sich sagen, dass nur Experten, die über viel Erfahrung in ihrem jeweiligen Fachgebiet verfügen (Spracherkennung, Dialogdesign, Integration von Telefonie/Datenbanken/externen Diensten), in der Lage sind, qualitativ hochwertige Sprachanwendungen nach dem mixed-initiative Ansatz zu erstellen. Bei der Entwicklung neuer Applikationen sind einige Optimierungsschritte notwendig, um Robustheit und Benutzerfreundlichkeit des Systems iterativ zu verbessern. Allerdings können der Einsatz von intelligenten Werkzeugen (z.B. eine konfigurierbare Dialog-Engine) und der direkte Zugriff auf die einzelnen Technologiekomponenten (z.B. zum Nachtraining des Spracherkenners) viel Entwicklungszeit einsparen helfen.

System:	*Hallo, hier ist das Kinoinformationssystem von Filmtips...*
Benutzer:	*[Barge-In] Wann kommt heute Abend in Nürnberg der Film „Alexander"?*
System:	*Sie möchten heute zwischen 18 und 22 Uhr in Nürnberg den Film „Alexander" sehen?*
Benutzer:	*Ja.*
System:	*Der Film „Alexander" wird heute... gezeigt.*
Benutzer:	*Läuft der auch in Erlangen?*
System:	*Der Film „Alexander" wird heute zwischen 18 und 22 Uhr in Erlangen nicht gezeigt.*

Abbildung 1: Dialogbeispiel, Kinoinformationssystem

4 Kommerziell eingesetzte Systeme

Die Applikationen, die in diesem Abschnitt vorgestellt werden, sind mixed-initiative Systeme im kommerziellen Einsatz. Das Bundesligaergebnis- und Tabellen-Informationssystem BERTI, mit dem sich Fußballinteressierte schnell einen Überblick verschaffen können, wurde mit dem VOICE Award 2004 in der Kategorie „Innovativste Applikation" ausgezeichnet. Das Kinoinformationssystem bietet Anrufern unter anderem Anfangszeiten, Filmtitel und Kinonamen zu aktuellen Filmen.

Bei BERTI können ohne einen langen Dialog gezielt die neuesten Informationen zur Bundesliga abgefragt werden; während der Spiele werden die aktuellen Zwischenstände mitgeteilt. Durch die Modellierung von Fachtermini ist auch bei einer Benutzung durch Fußball-Experten die Erkennung und Verarbeitung der Anfragen möglich („Wer hat die Rote Laterne?", „Wie steht es beim Club?"). Das Erkennervokabular umfasst ca. 1500 Wörter; eine Bestätigung erkannter Slotwerte erfolgt nicht.

Ein Beispiel für ein komplexeres System ist das Kinoinformationssystem, das Auskünfte zu Filmen einiger Kinos im Nürnberger Raum liefert. In Abbildung 1 sind die Vorteile des mixed-initiative Ansatzes dargestellt. Der Benutzer unterbricht das System und nennt sofort Werte zu allen für einen Datenbankzugriff relevanten Slotwerten. Nach einer expliziten Bestätigung erhält er die gewünschte Information. Mit Hilfe des Dialogkontexts ist es ihm danach möglich, eine weitere Information zu erhalten, ohne alle unveränderten Angaben noch einmal wiederholen zu müssen.

5 Zusammenfassung

Der vorliegende Beitrag versteht sich als ein Plädoyer für den Einsatz von mixed-initiative Dialogsystemen, um die noch immer niedrige Akzeptanz von automatischen Sprachanwendungen bei Firmen und Endbenutzern zu erhöhen. Es werden zum Einen theoretische Aspekte beleuchtet, die bei solchen Systemen zu beachten sind. Zum Anderen werden prak-

tisches Know-how und technische Werkzeuge beschrieben, die bei der Entwicklung von neuen Sprachapplikationen nötig oder wünschenswert sind. Anhand von kommerziell eingesetzten Systemen werden einige der in den vorigen Abschnitten aufgeführten Punkte veranschaulicht.

6 Telefonnummern der Beispielsysteme

- Bundesligaergebnis- und Tabellen-Informationssystem BERTI: 0900/3100748 (0,99 € pro Anruf aus dem Festnetz der deutschen Telekom)
- Kinoinformationssystem Filmtips: 01805/345684 (0,12 € pro Minute aus dem Festnetz der deutschen Telekom)

Literatur

[1] J. Fenn, A. Linden et al. Hype Cycle for Emerging Technologies. 2004.

[2] M. Peissner, J. Biesterfeldt, F. Heidmann. Akzeptanz und Usability von Sprachapplikationen in Deutschland. Technische Studie, Fraunhofer-Institut für Arbeitswirtschaft und Organisation (IAO), Stuttgart, 2004.

[3] G. Stemmer. Modeling Variability in Speech Recognition, Chair for Pattern Recognition, Universität Erlangen-Nürnberg, 2005.

[4] F. Gallwitz. Integrated Stochastic Models for Spontaneous Speech Recognition, Berlin 2002, Logos Verlag, Studien zur Mustererkennung, vol. 6, ISBN: 3-89722-907-2.

[5] A. Bouzid. The VUI View: Top 10 VUI No-Nos. Online-Newsletter von Angel.com http://www.angel.com/newsletter/12-04/vuiView.jsp, Dezember, 2004.

[6] E. Nöth, A. Horndasch, F. Gallwitz, J. Haas. Experiences with Commercial Telefone-based Dialogue Systems. In Information Technology, 6, S. 315-321, 2004.

Kontakt
Dipl.-Inf. Axel Horndasch, Dr.-Ing. Elmar Nöth
Lehrstuhl für Mustererkennung (Informatik 5)
Friedrich-Alexander-Universität Erlangen-Nürnberg
Martensstraße 3, 91058 Erlangen
Tel.: +49 (0)9131 8527297
Fax: +49 (0)9131 303811
Email: {horndasch,noeth}@informatik.uni-erlangen.de
www5.informatik.uni-erlangen.de

Dr.-Ing. Florian Gallwitz, Dr.-Ing. Jürgen Haas
Sympalog Voice Solutions GmbH
Karl-Zucker-Straße 10, 91052 Erlangen
Tel.: +49 (0)9131 61661-0
Fax: +49 (0)9131 61661-20
Email: {gallwitz, haas}@sympalog.de
www.sympalog.de

Axel Horndasch studierte an der Universität Erlangen-Nürnberg und am Rensselaer Polytechnique Institute in Troy, USA, Informatik. Nach seiner Diplomarbeit im DaimlerChrysler-Forschungszentrum in Palo Alto war er von 2000 bis 2004 bei der Firma Sympalog tätig. Seit Juli 2004 arbeitet er im Rahmen seiner Promotion als wissenschaftlicher Angestellter am Lehrstuhl für Mustererkennung der Uni Erlangen.

Florian Gallwitz ist Entwicklungsleiter der Sympalog Voice Solutions GmbH und beschäftigt sich seit 1994 mit automatischer Spracherkennung. Im Rahmen seiner Promotion entwickelte er Verfahren zur Erkennung und zum Verstehen von spontan gesprochener Sprache. Zur Präsentation seiner Forschungsergebnisse erhielt er mehrere Vortragseinladungen auf internationale Tagungen, u.a. nach Japan und Australien.

Jürgen Haas ist Leiter der Professional Services bei der Sympalog Voice Solutions GmbH und verantwortlich für die Umsetzung von Kundenprojekten. In Rahmen seiner Promotion entwickelte er einen neuen Ansatz zur Interpretation von Spracheingabe für automatische Dialogsysteme. Seine Arbeit wurde mit dem renommierten DAGM-Preis der Deutschen Arbeitsgemeinschaft für Mustererkennung ausgezeichnet.

Elmar Nöth erhielt sein Diplom in Informatik und seinen Doktortitel an der Friedrich-Alexander-Universität Erlangen-Nürnberg in 1985 bzw. in 1990. Von 1985 bis 1990 war er Mitglied der Forschungsgruppe am Lehrstuhl für Mustererkennung. Sein Arbeitsgebiet war die Verwendung von Prosodie für das automatische Sprachverstehen. Seit 1990 ist er akademischer Oberrat am gleichen Institut und Leiter der Sprachverarbeitungsgruppe.