Universität Karlsruhe (TH)
Institut für Logik, Komplexität und Deduktionssysteme
Professor Dr. A. Waibel

Studienarbeit
Im Sommersemester 2003

Spontanisierung eines Spracherkenners durch Interpolation von Sprachmodellen

Betreuer: Ivica Rogina
Tanja Schulz

Referent: Bastian Hemminger
Wielandtstrasse 20
76137 Karlsruhe
Studiengang: Informatik
Matrikelnummer: 970245

Zusammenfassung

In dieser Arbeit wurden verschiedene Möglichkeiten untersucht, ein Sprachmodell aus Nachrichtentexten mit einem spontansprachlichen Sprachmodell zu interpolieren. Die Erkennungsleistung auf dem verwendeten Testdatensatz konnte nur unwesentlich verbessert werden. Die unterschiedlichen Ergebnisse werden dargestellt und analysiert. Aus der Analyse werden weitere Möglichkeiten zur Verbesserung der Erkennungsleistung abgeleitet.

Inhaltsverzeichnis

Tabellenverzeichnis

1 Einleitung

1.1 Motivation

Die Erkennung spontaner Sprache wird immer wichtiger. Bereits heute werden viele Spracherkennungssysteme zum Beispiel für telefonische Dienste wie Fahrplanauskunft oder Katalogbestellungen verwendet. Dabei ist die Erkennung spontaner Sprache von besonderer Bedeutung, da sich die Benutzer mit dem System wie mit einem Menschen unterhalten möchten.

1.2 Aufgabenstellung

Die Aufgabe dieser Arbeit war, einen vorhandenen Spracherkenner für diktierte spanische Sprache auf die Domäne spontaner Sprache anzupassen.

Dazu sollten zwei Sprachmodelle erstellt werden. Ein umfangreiches Sprachmodell, das mit Nachrichtentexten erstellt werden sollte, und ein Sprachmodell, das auf spontansprachlichen Texten basieren sollte. Anschließend wurden beide Sprachmodelle interpoliert, um dadurch die Erkennungsleistung auf einem Testdatensatz mit spontaner Sprache zu verbessern.

Ziel der Arbeit ist, die mögliche Verbesserung durch Interpolation zu untersuchen und die Ergebnisse zu dokumentieren.

1.3 Vorstellung des verwendeten Spracherkenners

Benutzt wurde ein Spracherkenner für Spanisch, der im Rahmen des GlobalPhone Projektes im Jahr 2000 in den Interactive System Labs (ISL) der Universität Karlsruhe entstand. Dieser Spracherkenner wurde mit vorgelesenen Nachrichtentexten trainiert. Für das akustische Training wurden 5384 Äußerungen von 80 verschiedenen Sprechern verwendet, die mit 16kHz, 16 Bit abgetastet wurden.

Das Aussprachewörterbuch enthält 23085 Einträge, Aussprachevarianten mit eingeschlossen. Das Vokabular umfasst 18538 Einträge.

Das Sprachmodell ist ein Trigramm-Modell, das aus einem Korpus mit 143844 Worten berechnet wurde, es enthält 9812 Unigramme. Hier galt es, zunächst ein größeres Sprachmodell zu erstellen, um das Vokabular zu vergrößern und die Schätzung der Wahrscheinlichkeiten zu verbessern.

Dieser Spracherkenner wurde auf einem Testdatensatz von 39 Äußerungen aus den GlobalPhone Daten getestet. Dabei wurde eine Wortakkuratheit von 78,9 Prozent erreicht.

Dieser Spracherkenner basiert auf dem *JANUS Recognition Toolkit* (*JRTk*), das von den Interactive System Labs der Universität Karlsruhe und der Carnegie Mellon University in Pittsburgh, USA, entwickelt wurde. Sämtliche in dieser Arbeit beschriebenen Versuche wurden mit dem *JRTk* Version 5.0 Revision P010 gemacht.

2 Entwurf

Es sollte in dieser Arbeit untersucht werden, wie die Erkennungsleistung des beschriebenen Spracherkenners auf spontaner Sprache verbessert werden kann, wenn das vorhandene Sprachmodell mit einem anderen Sprachmodell interpoliert wird.
Folgende Schritte waren dafür notwendig:

1. Erstellen einer Testumgebung
2. Erstellen der Sprachmodelle
3. Testen der Erkennungsleistung des nachrichtenbasierten Sprachmodells
4. Interpolation der Sprachmodelle
5. Testen Erkennungsleistung der durch Interpolation entstandenen Sprachmodelle

In diesem Kapitel wird beschrieben, welche Daten, Werkzeuge und Verfahren für diese Schritte ausgewählt wurden. Die Auswahl wird jeweils begründet.

2.1 Testumgebung

2.1.1 Anforderungen

Die Erkennungsleistung eines Spracherkenners kann nachvollziehbar nur anhand einer vorgegebenen Testumgebung bestimmt werden. Eine Testumgebung besteht im Allgemeinen aus einer Menge von Sprachaufnahmen und den dazugehörigen Transkriptionen. Um einen Spracherkenner zu testen, wird ein Erkennungsvorgang mit den Sprachaufnahmen durchgeführt und der ausgegebene Text mit den Transkriptionen verglichen. Um die Eignung eines Spracherkenners für eine bestimmte Aufgabe zu testen, muss eine Testumgebung erstellt werden, die möglichst genau die charakteristischen Anforderungen dieser Aufgabe beinhaltet.

Um die Erkennungsleistung auf spontaner Sprache beurteilen zu können, musste eine Testumgebung erstellt werden, die Aufnahmen spontaner Sprache enthält. Der ursprüngliche Wortschatz des Spracherkenners sollte beibehalten werden. Das heißt, die Testumgebung sollte ebenfalls Nachrichtentexte enthalten.

Der Umfang der Testdaten sollte so groß gewählt werden, dass ein statistisch relevantes Ergebnis erzielt werden kann. Kriterien dafür sind

- Anzahl der Sprecher
- Anzahl der Äußerungen
- Dauer der Äußerungen

In der Literatur werden diese Zahlen lediglich genannt, ihre Wahl aber nicht begründet. So finden sich in den Artikeln in [4] Werte zwischen 250 und 300 Äußerungen bei 12 bis 30 Sprechern. Diese Werte können als Anhaltspunkte dienen.

2.1.2 Auswahl der Testdaten

Da das Erstellen einer Datenbank mit Testmaterial zu aufwendig gewesen wäre, musste auf bestehendes Material zurückgegriffen werden. Die Wahl fiel auf den Testdatensatz zur Evaluierung des Hub-4NE Benchmark Tests, den das Linguistic Data Consortium 1997 veröffentlichte. Der Hub-4NE Benchmark Test dient der Evaluierung der Erkennung von kontinuierlicher Sprache in Nachrichtensendungen. Neben vorgelesenen Nachrichtentexten enthält er auch Werbespots, Jingles, Berichte von Korrespondenten und Interviews. Die beiden zuletzt genannten Inhalte bringen die gewünschte spontane Sprache.

2.1.3 Testkriterium

Als Maß für die Erkennungsleistung dient die Wortakkuratheit WA (siehe [8]).

$$WA = 100\% - \frac{\text{minimale Editierdistanz}}{\text{Anzahl zu erkennender Worte}} \quad (1)$$

Wobei die minimale Editierdistanz als Summe der Vertauschungen, Auslassungen und Einfügungen durch dynamisches Programmieren berechnet wird.

2.2 Sprachmodelle

Sprachmodelle werden benutzt, um die Wahrscheinlichkeit von Wortfolgen zu schätzen. Dafür werden in der Literatur einige unterschiedliche Ansätze aufgezeigt.

2.2.1 Form der Sprachmodelle

In dieser Arbeit stand die Interpolation von zwei Sprachmodellen im Vordergrund. Darum wurden die allgemein üblichen Trigramm Modelle mit backing-off verwendet (siehe [3]). Das verwendete *JRTk* erwartet ein Trigrammodell im NIST Format (siehe [1]).

2.2.2 Software

Als Software zur Erstellung der Sprachmodelle wurde das *Clausi Toolkit* von Klaus Ries (siehe [6]) verwendet. Das *Clausi Toolkit* ist eine Sammlung von Programmen zur Sprachmodellierung. Der Hauptgrund, warum ich diese Software gewählt habe, war, dass sie frei verfügbar ist und die Kollegen am Institut bereits Erfahrung im Umgang mit dieser Software hatten.

2.2.3 Auswahl der Textkorpora

Das Basissprachmodell sollte aus einer möglichst großen Menge an Text erstellt werden. Dafür wurde der Spanish Language News Corpus ausgewählt, der 1995

vom Linguistic Data Consortium (LDC) veröffentlicht wurde. Diese Sammlung enthält spanischsprachige Nachrichtentexte von fünf verschiedenen Nachrichtenagenturen:

1. Agence France Presse
2. AP WorldStream
3. Infosel
4. Reuters Latin American Business Report
5. Reuters Spanish Language News Service

Der Korpus umfasst insgesamt über 200 Millionen (206234884) Wörter. Der Korpus auf dem das GlobalPhone Sprachmodell geschätzt wurde enthält dagegen nur 143844 Wörter. Es konnte somit mit einer Vergrößerung des Vokabulars als auch mit einer Verbesserung der Schätzung für die Wahrscheinlichkeiten gerechnet werden.

Die einzelnen Nachrichtenartikel sind durch *SGML* Tags getrennt und gekennzeichnet. Außer den *SGML* Tags enthalten die Texte noch viele Unsauberkeiten, die entfernt werden mussten. Das genaue Vorgehen wird im Abschnitt 3.2.3 beschrieben.

Für das spontansprachliche Sprachmodell sollte ein Textkorpus gefunden werden, der Transkriptionen spontaner Sprache enthält. Darum wurden die Transkriptionen des CallHome Spanish Datensatzes (LDC, 1997) verwendet. Dieser Datensatz umfasst 120 Transkriptionen von Ausschnitten aus Telefongesprächen. Diese Transkriptionen enthalten 282861 Wörter, wobei wiederum Markierungen im Text einen großen Teil davon ausmachen. Da es sich um Mitschnitte von Telefongesprächen zwischen Privatpersonen handelt, sind auch die vorkommenden Themen privater Natur. Ein Datensatz, der spontane Sprache mit Themen aus den Nachrichten enthält, wäre wünschenswert gewesen. Zudem ist dieser Korpus

immer noch sehr klein. Es stand aber kein anderer Korpus mit spontaner Sprache zur Verfügung. Darum mussten diese Einschränkungen in Kauf genommen werden.

2.2.4 Abschätzung der Güte eines Sprachmodells

Das *Clausi Toolkit* bietet die Möglichkeit, beim Erstellen eines Sprachmodells gleichzeitig die Qualität des Sprachmodells abzuschätzen. Dazu muss dem Programm zusätzlich zu Vokabular und Trainingskorpus ein Testkorpus übergeben werden, der vom Trainingskorpus unabhängig sein soll. Es berechnet dann die Perplexität PP des erzeugten Sprachmodells S auf dem Testkorpus W. Die Perplexität ist definiert über die Entropie H von S auf W. Es ist

$$H \approx \frac{1}{n} \log P(w_1, w_2, \ldots, w_n) \tag{2}$$

und

$$PP = 2^H. \tag{3}$$

Die Perplexität ist das geometrische Mittel der Anzahl der Wörter, aus denen der Spracherkenner in jedem Schritt auswählen muss (siehe [3]). Sie stellt damit ein Maß für die Schwierigkeit eines Textkorpus aus Sicht des Sprachmodells dar. Darum kann die Perplexität zum Vergleich von verschiedenen Sprachmodellen in einem Spracherkenner mit gleichbleibender Akustik verwendet werden.

Diese Voraussetzung war in dieser Arbeit gegeben. Darum wurden die beiden verwendeten Korpora nach der Bereinigung in disjunkte Mengen aufgetrennt. Jeweils zehn Prozent des Textes wurde für den Testkorpus zur Berechnung der Perplexität verwendet, die restlichen 90 Prozent für den Trainingskorpus.

2.3 Interpolation

In der Literatur werden unterschiedliche Möglichkeiten zur Interpolation von Sprachmodellen vorgestellt. Aus praktischer Sicht lassen sie sich in zwei Klas-

sen einteilen:

1. Interpolation zur Laufzeit (online)

2. das interpolierte Sprachmodell wird extern berechnet (offline)

Im ersten Fall werden vom Spracherkennungssystem zwei oder mehrere Sprachmodelle geladen. Erst zur Laufzeit werden aus den geladenen Sprachmodellen die endgültigen Wahrscheinlichkeiten berechnet. Wird das interpolierte Sprachmodell dagegen zuvor berechnet, muss zur Laufzeit nur dieses eine, neue, Sprachmodell geladen werden. Beide Ansätze haben Vor- und Nachteile, die im folgenden Kapitel untersucht werden.

2.3.1 Vergleich online - offline

Wenn die Interpolation erst zur Laufzeit stattfindet, können die verwendeten Sprachmodelle leicht ausgetauscht werden. Parameter, zum Beispiel die Gewichtung der Modelle, können nachträglich verändert werden. Es ist möglich, die Interpolationsgewichte während der Laufzeit situationsbedingt zu wählen. Dafür muss man in Kauf nehmen, dass das System langsamer wird weil es zusätzliche Berechnungen für die Interpolation ausführen muss. Außerdem lässt sich das Ergebnis schwer auf ein anderes Spracherkennungssystem übertragen. Man muss die Sprachmodelle, die verwendeten Parameter und möglicherweise auch den verwendeten Algorithmus auf das andere System übertragen.

Wird dagegen das interpolierte Sprachmodell extern berechnet, entfällt der zusätzliche Rechenaufwand zur Laufzeit des Spracherkenners. Allerdings muss für jede Modifikation ein neues Sprachmodell berechnet werden. Das braucht wiederum viel Zeit und Speicherplatz. Ein solches Sprachmodell kann aber auch leichter wiederverwendet werden. Es kann in jedem Spracherkennungssystem, das das verwendete Datenformat lesen kann, verwendet werden.

2.3.2 Wahl der Interpolationsmethoden

Das *Janus Recognition Toolkit* bietet die Möglichkeit, Sprachmodelle zur Laufzeit zu interpolieren. Nach obigen Überlegungen erschien mir das die erfolgversprechendere Variante, da leicht verschiedene Parameter ausprobiert werden können. Dass die offline Variante keinen zusätzlichen Rechenaufwand zur Laufzeit bringt und das Ergebnis leichter wiederverwendbar ist, erschien mir aber ein guter Grund, auch diese Möglichkeit zu testen.

2.3.3 Software

Für die online Interpolation wurde das *JRTk* und das Programm *interpol* (siehe 3.3.2) benutzt. Für die offline Interpolation wurde keine weitere Software benötigt.

3 Ausführung

3.1 Testumgebung

Das verwendete Testmaterial verfügt über Aufnahmen von 48 verschiedenen Sprechern. Diese Aufnahmen dauern insgesamt 30 Minuten und sind in Äußerungen unterteilt. Eine Äußerung umfasst meist einen oder wenige Sätze, selten nur Satzfragmente oder einzelne Worte.

3.1.1 Anpassung

Das Testmaterial war in der vorliegenden Form dazu gedacht, die Erkennungsleistung eines Spracherkenners zu evaluierten. Dabei ist zu beachten, dass das Testmaterial bereits 1997 veröffentlicht wurde. Damals war generell weniger Rechenleistung vorhanden als heute. Man musste also davon ausgehen, dass ein entsprechender Testdatensatz aktuellen Datums größer wäre.

Da bereits die Auswertung dieser relativ geringe Menge an Testmaterial mit den vorhandenen Computern manchmal bis zu drei Tage dauerte, musste das Testmaterial verkürzt werden. Dazu wurden von jedem der 48 Sprecher maximal 3 Äußerungen verwendet. Das daraus entstandene Testmaterial umfasst 85 Sätze und ist zehn Minuten lang. Das ist, verglichen mit den Beispielen aus Abschnitt 2.1, sehr wenig. Aber da nur beschränkte Rechenleistung vorhanden war, musste diese Einschränkung in Kauf genommen werden. Dieser Testdatensatz wird im Folgenden mit hub4-ne.test bezeichnet.

Der hub4-ne.test Testdatensatz enthielt hauptsächlich (80 Prozent) vorgelesene Nachrichtentexte. Um eine bessere Aussage über die Erkennungsleistung auf spontaner Sprache treffen zu können, wurde eine weitere Teilmenge des Hub4-NE Testkorpus erstellt, die nur spontane Sprache enthielt. Dazu mussten die vorhandenen Aufnahmen zuerst von Hand klassifiziert werden. Es fanden sich 33 spontansprachliche Äußerungen von 25 verschiedenen Sprechern. Die Äuße-

rungen dauern insgesamt 4 Minuten. Dieser Datensatz wird als hub4-ne.spontan bezeichnet.

3.1.2 Algorithmen

Um die Ergebnisse dieser Arbeit mit den Ergebnissen des vorhandenen Systems vergleichen zu können, wurden die ebenfalls vorhandenen Testskripten verwendet. Lediglich die Sprachaufnahmen und die dazugehörigen Transkriptionen wurden ausgetauscht.

3.2 Sprachmodelle

Ein Sprachmodell berechnet die Wahrscheinlichkeit für das Auftreten einer Wortfolge. Diese Wahrscheinlichkeit wird dazu verwendet, unter mehreren gefundenen Hypothesen für eine Äußerung die wahrscheinlichste auszuwählen.

Eng mit dem Sprachmodell in Zusammenhang stehen das Aussprachewörterbuch und das Vokabular des Spracherkenners. Das Aussprachewörterbuch bildet die von der Akustik gefundenen Lautfolgen auf Worte ab. Das Vokabular legt fest, welche Worte der Spracherkenner überhaupt kennt.

Um aus den ausgewählten Textkorpora Sprachmodelle zu erstellen, mussten für die beiden Textkorpora folgende Schritte durchgeführt werden:

1. Text normalisieren
2. Vokabular auswählen
3. Sprachmodell erstellen
4. Aussprachewörterbuch erstellen

3.2.1 Normalisierungsregeln

Den Text zu normalisieren bedeutet, alle Zeichenfolgen aus dem Korpus zu entfernen, die nicht als Wort berücksichtigt werden sollen. Um mit dem *Clausi Toolkit* ein Sprachmodell zu erstellen, muss der benutzte Textkorpus in einem festgelegten Format vorliegen:

- jedes Wort steht in einer eigenen Zeile
- Sätze sind durch eine Leerzeile getrennt

Da die beiden verwendeten Korpora sehr unterschiedliche Qualität aufwiesen, war es zudem notwendig, ein einheitliches Format für die Daten zu finden. Damit sollte sicher gestellt werden, dass beim Interpolieren der Wahrscheinlichkeiten

auch tatsächlich die Wahrscheinlichkeiten einander beeinflussen, die zum selben gesprochenen Wort gehören. Das heißt, es sollten zum Beispiel die Abkürzung „km" und das ausgeschriebene Wort „kilometros" als das selbe gesprochene Wort behandelt werden.

Als Maßstab wurde der CallHome Korpus gewählt. Dieser ist, im Gegensatz zum Spanish Language News Korpus, schon für die Verwendung in Spracherkennern aufbereitet. Das bedeutet unter anderem:

- Zahlen und Abkürzungen sind ausgeschrieben
- buchstabierte Buchstaben sind ausgeschrieben

Diese Vorgaben sollten auch für den Spanisch Language News Korpus umgesetzt werden.

3.2.2 Vorgehen

Um diese Vorgaben umzusetzen, wurden die Korpora mit *Perl* Skripten bearbeitet. Alle Textersetzungen wurden mit dem Substitutionsoperator durchgeführt, der auf regulären Ausdrücken basiert (siehe [9]).

Als weiteres Hilfsmittel stand das Programm *normalize* zur Verfügung. Dieses Programm wurde am Institut E.T.S.I Telecomunicacion der Polytechnischen Universität von Barcelona entwickelt. Mit *normalize* kann Text normalisiert werden. Es bearbeitet Text zeilenweise und ersetzt dabei in dieser Reihenfolge:

- Email- und Internetadressen
- Datumsangaben
- Telefonnummern
- Uhrzeiten
- Jahreszahlen (im Spanischen mit römischen Ziffern)

- Adressen
- Währungseinheiten
- aus Ziffern und Buchstaben zusammengesetzte Wörter
- Akronyme
- spezielle Eigennamen
- Prozentzeichen
- Mengenangaben
- Ziffernfolgen
- zusammengesetzte Wörter mit Bindestrich
- katalanische Präfixe
- Satzzeichen

Da das Programm leider einige Fehler aufwies und viele Dinge, die in den benutzen Korpora vorkamen, nicht berücksichtigt, wurde es lediglich dazu benutzt, Zahlen und römische Zahlen umzusetzen.

3.2.3 Bereinigung der Textkorpora

In diesem Abschnitt wird beschrieben, welche Textersetzungen bei den einzelnen Korpora vorgenommen wurden.

Im Spanish Language News Korpus wurden zuerst mit *Perl* Skripten folgende Dinge ersetzt:

- durch Paragraphentags isolierte, einzelne Zeilen gelöscht
- Uhrzeiten umgesetzt

- Abkürzungen buchstabiert ausgeschrieben
- Abkürzungen mit fünf und mehr Buchstaben beibehalten
- Großschreibung am Satzanfang beibehalten
- Zahlen ausgeschrieben, mitsamt den Trennzeichen
- alle Sportartikel gelöscht
- Text in Klammern gelöscht
- Bindestriche zwischen Zahlen durch a ersetzt, sonst durch Leerzeichen
- Kommata zwischen Zahlen durch `coma` ersetzt
- katalanische c's durch c ersetzt
- Abkürzungen für Längen, Weiten und Geschwindigkeiten ausgeschrieben
- Prozentzeichen durch `porcientos` ersetzt
- 0 in Zahlen durch 0 ersetzt
- Autorenkürzel am Ende von Artikeln gelöscht
- Header der Artikel gelöscht
- *HTML* Tags gelöscht

Um diese Textersetzungen vorzunehmen, musste für jede der fünf Nachrichtenagenturen ein individuelles Skript geschrieben werden. Die Verwendung von Tags, Kommentaren und Sonderzeichen war so unterschiedlich, dass die Textersetzung basierend auf regulären Ausdrücken nicht einheitlich möglich war.

Während dieses Arbeitsschrittes stellte sich heraus, dass die Artikel der Agentur Infosel sich nicht auf diese Weise bereinigen ließen. Sie enthalten zu viele

unterschiedliche und unsaubere Formatierungen. Darum wurden die Artikel von Infosel aus dem Korpus entfernt und nicht verwendet.

Die so vorbereiteten Artikel der anderen Nachrichtenagenturen wurden nun mit dem Programm *normalize* bearbeitet um Ziffern und römische Zahlen umzusetzen. Danach wurden nacheinander Doppelpunkte, Punkte, mehrfache Leerzeichen und Leerzeilen gelöscht. Zuletzt wurden falsch geschriebene Akzente korrigiert und alle verbliebenen Sonderzeichen, die nicht im spanischen Alphabet vorkommen, aus den Artikeln entfernt.

Im CallHome Korpus gab es keine Fehler oder Sonderzeichen wie im Spanish Language News Korpus. Nachdem der eigentliche Text zwischen den Tags extrahiert war, mussten nur noch Markierungen im Text ersetzt werden. Im Folgenden werden in Stichworten die im Text auftretenden Markierungen genannt und beschrieben, wie sie ersetzt wurden.

- # bezeichnet simultan gesprochenen Text, das Zeichen # löschen - Text beibehalten
- unvollständige Wörter enden mit Bindestrich - ersetzt durch `UNK`
- menschliche Laute, die nicht zur Sprache gehören, sind in geschwungenen Klammern geschrieben - ersetzt durch `#noise#`
- Hintergrundgeräusche stehen in eckigen Klammern - ersetzt durch `#noise#`
- an andere gerichtete Sprache ist in // eingeschlossen - gelöscht
- unklare Worte in doppelter runder Klammer - Klammern gelöscht, Text beibehalten
- Kommentare des Transkriptors in doppelter eckiger Klammer - gelöscht
- Zahlen sind ausgeschrieben - keine Ersetzung notwendig

- Abkürzungen sind in einzelnen Großbuchstaben geschrieben - als Worte ausgeschrieben (a, be, ce, de, e, efe, ...)
- Hesitationen aaa ,eee ,iii , mmm , emm , amm und imm - ersetzt durch #fragment#
- Worte in einer anderen Sprache stehen in spitzen Klammern, der erste Eintrag in der Klammer ist die Sprache - Klammern und erster Eintrag gelöscht

Nachdem nun beide Korpora aufbereitet waren wurde die Größe der beiden Korpora nochmals verglichen. Das Ergebnis ist in Tabelle 1 zu sehen. Obwohl der Spanish Language News Korpus über 200 mal so groß ist wie der CallHome Korpus, enthält er nur etwa 15 mal so viele verschiedene Wörter. Der CallHome Korpus enthält 1596 Wörter, die nicht im Spanisch Language News Korpus vorkommen. Bemerkenswert ist hier, dass nur etwa die Hälfte der ursprünglichen Datenmenge tatsächlich verwendbar war (vergleiche Kapitel 2.2.3 auf Seite 9).

3.2.4 Erstellen eines gemeinsamen Vokabulars

Bevor aus einem Textkorpus mit dem *Clausi Toolkit* ein Sprachmodell erstellt werden kann, muss das Vokabular dafür festgelegt werden. Das Vokabular ist eine Liste derjenigen Wörter aus dem Textkorpus, die bei der Erstellung des Sprachmodells berücksichtigt werden. Beim Erkennungsvorgang legt das Vokabular fest, welche Wörter in der erstellten Transkription vorkommen können.

	Spanish Language News	CallHome
Wortanzahl	91289189	162649
verschiedene Wörter	424480	11098

Tabelle 1: Größe der normalisierten Korpora

Da man mit dem *JRTk* nur Sprachmodelle interpolieren kann, die das selbe Vokabular haben, wurde ein gemeinsames Vokabular für alle Sprachmodelle erstellt. Dabei musste berücksichtigt werden, dass die Größe des Vokabulars durch das *JRTk* auf maximal 65534 Einträge beschränkt ist.

Aus theoretischer Sicht sind aber andere Kriterien für die Auswahl des Vokabulars wichtig. Die Größe des Vokabulars beeinflusst direkt die Größe des Sprachmodells. Mit der Größe des Sprachmodells wiederum wächst und sinkt

1. der Speicherbedarf des Spracherkenners
2. die Dauer des Erkennungsvorgangs.

Um sowohl den Speicherbedarf als auch die Dauer des Erkennungsvorgangs möglichst klein zu halten, wünscht man sich ein möglichst kleines Vokabular. Andererseits ist ein großes Vokabular wichtig, um möglichst viele Wörter erkennen zu können (siehe [3]).

Die Größe des Vokabulars wurde festgelegt durch einen Kreuzüberdeckungstest. Das bedeutet, dass ein Testkorpus festgelegt wird, der der Domäne des Spracherkenners entspricht. Aus dem Trainingskorpus werden die n häufigsten Wörter als Vokabular benutzt. Die Zahl n wird dabei so gewählt, dass mit dem Vokabular ein vorgegebener Prozentsatz der Wörter des Testkorpus überdeckt wird.

Um 98 Prozent Überdeckung auf dem Testkorpus (hub4-ne.test) zu erreichen, wären 81412 Wörter aus dem Spanish Language News Korpus notwendig gewesen. Ein so großes Vokabular ist, wie oben beschrieben, nicht möglich. Es wurde darum ein Vokabular mit 60000 Wörtern benutzt, was eine Überdeckung von 97.5 Prozent auf dem hub4-ne.test Korpus bringt.

3.2.5 Erstellen des Aussprachewörterbuchs

Die Akustik des Spracherkenners erzeugt Hypothesen in Form von Phonemfolgen. Das Sprachmodell dagegen berechnet Wahrscheinlichkeiten für Wortfolgen. Das Bindeglied zwischen beiden ist das Aussprachewörterbuch. Das Aussprachewörterbuch bildet die Einträge des Vokabulars auf Phonemfolgen ab, wie sie die Akustik erzeugt.

Der im Spracherkenner verwendete Phonemsatz war durch die vorhandene Akustik bereits vorgegeben. Darum wurde das ebenfalls vorhandene Skript *spanish_dictmaker* von Marsal Gavalda (Version 1994.2) verwendet, um aus dem Vokabular ein Aussprachewörterbuch zu erstellen. Das Programm *spanish_dictmaker* unterscheidet 39 Phoneme und drei Positionen innerhalb eines Wortes: Wortanfang, Wortmitte, Wortende.

Anschließend wurden die Namen der Phoneme im Aussprachewörterbuch mit einem *Perl* Skript an die im GlobalPhone Projekt verwendeten Phonemnamen angepasst.

3.2.6 Erzeugen der Sprachmodelle

Zum Erzeugen der Sprachmodelle sollte das *Clausi Toolkit* von Klaus Ries verwendet werden. Dieses Werkzeug erfordert, dass die Texte in folgender Form übergeben werden: jedes Wort steht in einer eigenen Zeile, Sätze werden durch eine Leerzeile getrennt. Diese Anpassung wurde mit einem *Perl* Skript durchgeführt.

Um die Perplexität der erzeugten Sprachmodelle berechnen zu können, wurden die beiden Korpora wie in Kapitel 2.2.4 beschrieben in Trainings- und Testkorpora aufgetrennt.

Aus den beiden Korpora wurde dann mit dem *Clausi Toolkit* Sprachmodelle berechnet. Dabei wurden nur Trigramme berücksichtigt, die mindestens zweimal vorkamen. Die Perplexität auf den Testkorpora betrug 85 beim Spanish Lan-

guage News Korpus und 210 beim CallHome Korpus. Hier wird die Bedeutung der Bereinigung der Textkorpora klar. Auf einer Testmenge die zehn Prozent der unbereinigten Daten des Spanish Language News Korpus enthielt, war die Perplexität des Spanish Language News Sprachmodells 122. Die Perplexität des ursprünglichen GlobalPhone Sprachmodells auf der unbereinigten Testmenge betrug 184.

3.3 Interpolation

In Kapitel 2.3 wurden schon die praktischen Vorüberlegungen zur Wahl der Interpolationsmethode gemacht. In diesem Kapitel wird beschrieben, welche Algorithmen für die jeweilige Methode benutzt wurden.

3.3.1 Offline: Mischen der Korpora

Sprachmodelle offline zu interpolieren geht am einfachsten, indem man die benutzten Korpora zusammenfügt und ein neues Sprachmodell berechnet. Dieser Ansatz wird in [8] beschrieben. Eine Gewichtung der Modelle kann dadurch vorgenommen werden, dass ein Korpus mehrfach zum gemeinsamen Korpus hinzugefügt wird.

Da der CallHome Korpus sehr viel kleiner ist als der Spanish Language News Korpus (siehe Tabelle 1), war zu erwarten, dass der Einfluss des CallHome Korpus bei einem Mischungsverhältnis von 1:1 kaum merklich sein würde. Es wurden darum fünf verschiedene Korpora mit unterschiedlichen Mischungsverhältnissen erstellt. Tabelle 2 zeigt, wie oft die CallHome und Spanish Language News jeweils zum neuen Korpus hinzugefügt wurden.

Aus jedem der erstellten Korpora wurde mit dem Programm *ngrammodel* ein Sprachmodell erstellt. Dabei wurden nur Trigramme berücksichtigt, die mindestens dreimal im Korpus vorkamen.

gemischter Korpus	Spanish Language News	CallHome
SLN1_CH1	1	1
SLN1_CH10	1	10
SLN1_CH100	1	100
SLN1_CH200	1	200
SLN1_CH300	1	300

Tabelle 2: Mischungsverhältnisse der Korpora

3.3.2 Online Interpolation

In der Janus Lanuage Modelling Dokumentation von Klaus Ries ([1]) wird beschrieben, wie Sprachmodelle mit dem *JRTk* online interpoliert werden können. Es gibt die Möglichkeit, mehrere Sprachmodelle zur Laufzeit zu laden und daraus ein interpoliertes Sprachmodell zu berechnen. Dabei werden die Wahrscheinlichkeiten der einzelnen Sprachmodelle LM_i jeweils mit einem Faktor λ_i gewichtet. Bei der Wahl der λ_i werden zwei Kategorien unterschieden:

1. die λ_i sind kontextunabängig
2. die λ_i sind kontextabhängig

Werden die Sprachmodelle kontextunabhängig gewichtet, so kann die Gewichtung als Parameter beim Laden der Sprachmodelle übergeben werden.

Im kontextabhängigen Fall muss eine zusätzliche Datei geladen werden, die die Gewichtungsfaktoren $\lambda_i(\mathrm{h})$ für jedes Sprachmodell LM_i bezüglich einer Geschichte h enthält. Die Geschichte h kann dabei maximal zwei Worte lang sein. Die Summe der $\lambda_i(\mathrm{h})$ über alle Sprachmodelle muss dabei jeweils 1 ergeben.

$$\sum_i \lambda_i(\mathrm{h}) = 1 \tag{4}$$

Das *Clausi Toolkit* stellt das Programm *interpol* zur Verfügung, mit dem die Gewichtungsfaktoren λ_i automatisch berechnet werden können. *interpol* benötigt einen Trainingskorpus, einen Testkorpus und einen Kreuzvalidierungskorpus um die Faktoren zu berechnen. Dabei werden die Faktoren bezüglich der Perplexität auf dem Kreuzvalidierungskorpus optimiert. Mit diesem Programm lassen sich sowohl globale Faktoren (kontextunabhängig) als auch kontextabhängige Faktoren berechnen. Für die Berechnung kontextabhängiger Faktoren kann die Länge der verwendeten Geschichte als Parameter übergeben werden. Es erzeugt als Ausgabe ein Skript, dass zusätzlich zu den interpolierten Sprachmodellen vom *JRTk*

geladen werden muss. Das Skript enthält die berechneten Gewichtungsfaktoren und nimmt automatisch notwendige Einstellungen des Spracherkenners vor.

Für den kontextunabhängigen Fall wurden zunächst neun Versuche gemacht, bei denen die Gewichtungen der beiden Sprachmodelle in Schritten von zehn Prozent variiert wurden. Das bedeutet, dass für das Paar $(\lambda_1;\lambda_2)$ die Werte $(0,1;0,9), (0,2;0,8), (0,3;0,7) \ldots (0,9;0,1)$ getestet wurden.

Um das Programm *interpol* verwenden zu können, wurden neue Korpora erstellt. Als Grundlage dienten die zum Training der Sprachmodelle verwendeten Korpora, wobei größeres Gewicht auf den Anteil des CallHome Korpus gelegt wurde, um genügend spontane Sprache in den Korpus zu bringen. Der neue Trainingskorpus bestand aus sechs Millionen Wörtern des Spanish Language News Korpus und dreimal dem CallHome Korpus. Test- und Kreuzvalidierungskorpus bestanden aus jeweils der Hälfte der Wörter der beiden Testkorpora die in Kapitel 2.2.4 beschrieben wurden.

Unter Verwendung dieser neuen Korpora wurden dann mit dem Programm *interpol* die Gewichtungsfaktoren für eine kontextunabhängige Interpolation berechnet.

Um die kontextabhängigen Gewichtungsfaktoren zu berechnen, wurden die selben Korpora verwendet wie für die kontextunabhängigen. Als Kontext wurde das vorangehende Wort benutzt, also eine Geschichte der Länge 1.

4 Ergebnisse

Für jedes der in den Kapiteln 3.2 und 3.3 beschriebenen Sprachmodelle wurde ein Erkennungsvorgang mit beiden Testdatensätzen durchgeführt. Dazu wurden die in Kapitel 3.1 beschriebene Testumgebungen unverändert benutzt. Anschließend wurden der Parameter für die Gewichtung des Sprachmodells und der Multiplikator für die Bestrafung von Wortübergängen angepasst. Es wurden jeweils zwei bis drei Anpassungsschritte durchgeführt, bis sich die Ergebnisse nicht mehr verbesserten.

In diesem Kapitel aufgelistet sind die besten Ergebnisse für jedes der Sprachmodelle. Angegeben wird jeweils die Wortakkuratheit in Prozent.

4.1 CallHome und Spanish Language News

Die beiden nicht interpolierten Korpora erzeugten bereits sehr schlechte Ergebnisse, siehe Tabelle 3.

Sprachmodell	hub4-ne.test	hub4ne.spontan
CallHome	22,5%	23,5%
Spanish Language News	45,1%	35,7%

Tabelle 3: Wortakkuratheiten der nicht interpolierten Sprachmodelle

4.2 Interpolierte Sprachmodelle

überraschenderweise konnte auch mit den interpolierten Sprachmodellen nur eine sehr geringe Verbesserung der obigen Ergebnisse erreicht werden.

4.2.1 Offline

In Tabelle 4 sind die Ergebnisse aufgelistet, die mit den offline interpolierten Korpora erzielt wurden. Offensichtlich hat hier der hinzugefügte CallHome Korpus

keine Verbesserung, sondern eine Verschlechterung der Ergebnisse bewirkt.

Sprachmodell	hub4-ne.test	hub4ne.spontan
SLN1_CH1	43,4%	k.A.
SLN1_CH10	43,6%	35,4%
SLN1_CH100	42,5%	35,0%
SLN1_CH200	41,7%	33,2%
SLN1_CH300	41,9%	33,5%

Tabelle 4: Ergebnisse der offline Interpolation

4.2.2 Online

Die in [1] beschriebene Möglichkeit, Sprachmodelle mit festen Gewichtungsfaktoren zur Laufzeit zu interpolieren, war nicht durchführbar, da die Software nicht wie dokumentiert funktionierte.

Die online Interpolation mit den von *interpol* erstellten Skripten erzielte die in Tabelle 5 dargestellten Wortakkuratheiten. Lediglich mit dem online interpolierten Sprachmodell mit kontextabhängig berechneten Interpolationsgewichten konnte eine geringe Verbesserung gegenüber dem Spanish Language News Sprachmodell erreicht werden.

Sprachmodell	hub4-ne.test	hub4-ne.spontan
kontextunabhängig	44,9%	35,7%
Geschichte Länge 1	43,7%	36,1%

Tabelle 5: Ergebnisse der online Interpolation

5 Analyse

Bei der Untersuchung der im Kapitel 4 dargestellten Versuchsergebnisse fallen hauptsächlich zwei Dinge auf. Erstens sind alle Ergebnisse vergleichsweise schlecht. In [2] wird für den Hub-4NE Broadcast News Benchmark Test eine Wortakkuratheit von 77,6 Prozent berichtet, in [5] sogar 78,5 Prozent. Zweitens wurden durch die verschiedenen Interpolationsansätze kaum bessere, sondern schlechtere Ergebnisse erreicht. In diesem Kapitel soll untersucht werden, warum diese schlechten Ergebnisse erhalten wurden.

5.1 OOV-Rate

Eine erste mögliche Fehlerquelle sind OOV-Worte (Out Of Vocabulary). Das sind Worte in den Testäußerungen, die im Vokabular des Spracherkenners nicht vorkommen. Solche OOV-Worte führen zwingend zu Fehlern in den Hypothesen. Wird ein Wort vom Spracherkenner falsch erkannt, werden auch zur Schätzung der nächsten Wörter falsche Trigramme herangezogen. Das führt oft zu Folgefehlern. Ein OOV-Wort verursacht darum laut [7] durchschnittlich etwa 1,5 bis 2 Fehler.

Da alle verwendeten Sprachmodelle mit dem selben gemeinsamen Vokabular erstellt wurden (siehe 3.2.4) war dieser Wert nicht aussagekräftig. Darum wurde die OOV-Rate für CallHome und Spanish Language News auf dem Vokabular berechnet, das vom jeweiligen Korpus in das gemeinsame Vokabular einging. Tabelle 6 zeigt die OOV-Raten der verwendeten Sprachmodelle. Die interpolierten

Sprachmodell	hub4-ne.test	hub4-ne.spontan
Spanish Language News	2,8%	10,5%
CallHome	20,9%	13,2%
interpolierte Modelle	2,2%	2,3%

Tabelle 6: OOV Raten der Sprachmodelle auf den Testkorpora

Modelle besitzen alle die selbe OOV-Rate, da sie mit dem selben Vokabular erstellt wurden. Aufgrund der geringen OOV-Rate von 2,2 Prozent können OOV-Wörter also nur für eine Fehlerrate von etwa vier Prozent verantwortlich gemacht werden.

5.2 Perplexität

Ein weiterer Anhaltspunkt für die Qualität eines Sprachmodells gemessen an einem bestimmten Testkorpus ist die Perplexität. Bei der Erstellung der Sprachmodelle wurde die Perplexität jeweils auf einer Teilmenge des Trainingskorpus berechnet. Nun wurde zusätzlich die Perplexität auf den Testkorpora berechnet. Die drastischen Unterschiede in der Perplexität auf den Teilmengen der Trainingskorpora und der Perplexität auf den Testdaten sind sehr auffällig. Das ist ein Hinweis darauf, dass die Sprache des Hub-4NE Korpus sich sehr unterscheidet von den beiden Trainingskorpora CallHome und Spanish Language News. Und damit eine mögliche Erklärung für die schlechten Ergebnisse.

Sprachmodell	eigne Testmenge	hub4-ne.test	hub4-ne.spontan
CallHome	160	1848	895
Spanish Language News	85	186	266
SLN1_CH1	84	246	406
SLN1_CH10	88	260	458
SLN1_CH100	135	290	595
SLN1_CH200	58	308	676
SLN1_CH300	60	318	719
kontextunabhängig	106	k.A.	k.A.
Geschichte Länge 1	104	k.A.	k.A.

Tabelle 7: Perplexität der Sprachmodelle auf den verwendeten Testdaten

5.3 Unbekannte Worte

Eine Ursache für schlechte Erkennungsraten kann auch eine hohe Wahrscheinlichkeit für unbekannte Worte im Sprachmodell sein. Wie Tabelle 8 zeigt, konnte

Sprachmodell	Wahrscheinlichkeit
CallHome	0,3274
Spanish Language News	0,0236
offline interpolierte	0,0012
online interpolierte	k.A.

Tabelle 8: Wahrscheinlichkeiten für unbekannte Wörter

die Wahrscheinlichkeit für unbekannte Worte durch die Interpolation deutlich gesenkt werden. Sie kann darum nicht für die schlechten Ergebnisse verantwortlich gemacht werden.

5.4 Akustik

Ein weiterer Grund für die schlechten Ergebnisse könnte die Akustik des Spracherkenners sein. Das akustische Modell wurde mit vorgelesenen Zeitungsartikeln trainiert. Die Aufnahmen fanden unter Studiobedingungen statt. Die Sprache in den Hub-4NE Aufnahmen ist dagegen spontan und teilweise durch Hintergrundgeräusche gestört.

Ebenso können unterschiedliche Dialekte der Sprecher von Trainings- und Testdaten die Erkennungsergebnisse verschlechtern. Von den 84 Sprechern der GlobalPhone Aufnahmen sprechen 82 Tico (der spanische Dialekt in Costa Rica) und 2 Castellano. Die Dialekte in den Hub-4NE Aufnahmen sind unterteilt in die Kategorien coastal, interior und other. Die Transkriptionen lassen vermuten, dass es sich ebenfalls um Sprecher aus Costa Rica handelt. Eine Dokumentation der Dialekte gibt es aber nicht, weshalb diese Frage nicht abschließend geklärt werden konnte.

6 Fazit

Die anfangs gestellte Frage, ob sich eine Spontanisierung durch die Interpolation erreichen lässt, kann nur mit Einschränkungen bejaht werden. Zumindest, wenn man die hier verwendeten Daten und Verfahren betrachtet. Bereits das vorangehende Kapitel 5 deutet auf Möglichkeiten hin, wie eine Verbesserung erreicht werden könnte.

Die hohe Perplexität auf den Testdaten lässt darauf schließen, dass sich Trainings- und Testdaten sehr unterscheiden. Dafür könnte die unterschiedliche Größe der beiden Korpora verantwortlich sein. Da der CallHome Korpus nur ein halbes Prozent des Umfangs des Spanish Language News Korpus hat, ist auch sein Informationsgehalt entsprechend gering. Es wäre also zu untersuchen, ob ein größerer spontansprachlicher Korpus die Ergebnisse verbessern würde. Ein solcher Korpus war für diese Arbeit nicht vorhanden. Der starke Anstieg der Perplexität der interpolierten Korpora gegenüber dem Spanish Language News Korpus deutet außerdem auf eine Inkompatibilität zwischen Trainings- und Testdaten hin. Darum wäre es wünschenswert, einen passenden Testdatensatz mit spontaner Sprache zu haben.

Ein zweiter möglicher Ansatzpunkt ist die unterschiedliche Akustik von Trainingsmaterial und Testmaterial. Die Aufnahmen des Hub-4NE Testdatensatzes enthalten zusätzlich zur spontanen Sprache auch unterschiedliche Aufnahmequalitäten und Hintergrundgeräusche. Beides war in der Aufgabenstellung des Spracherkenners nicht vorgesehen, erschwert die Erkennung aber sehr. Darum sollte eine Testumgebung erstellt werden, die genau dieser Anforderung entspricht.

Gleichzeitig muss die Idee in frage gestellt werden, einen Spracherkenner für spontane Sprache mit einem akustischen Modell das auf vorgelesenen Texten trainiert wurde zu erstellen. Da die Aussprache sich hier stark unterscheidet, sollte auch ein entsprechendes akustisches Modell trainiert werden.

Literatur

[1] Christian Fuegen, Florian Metze und Hagen Soltau: Online JRTk documentation `http://isl.ira.uka.de/ jrtk/janus-doku.html`, in der Fassung vom 13.07.2002.

[2] Juan M. Huerta, Stanley Chen und Richard M. Stern: The 1998 Carnegie Mellon University SPHINX-3 Spanish Broadcast News Transcription System. Proceedings of the DARPA Broadcast News Workshop, Herndon, Virginia, 1999.

[3] Frederick Jelinek: Statistical methods for speech recognition. The MIT Press, Cambridge, Massachusetts, 1997.

[4] Pietro Laface und Renato De Mori (Editoren): Speech Recognition and Understanding. Springer, Berlin/Heidelberg, 1992.

[5] David S. Pallett, Jonathan G. Fiscus, John S. Garofolo, Alvin Martin und Mark Przybocki: 1998 Broadcast News Benchmark Test Results: English and non-english word error rate performance measures. Proceedings of the DARPA Broadcast News Workshop, Herndon, Virginia, 1999.

[6] Klaus Ries, Bernhard Suhm und Petra Geutner: Language Modelling in JANUS. `http://isl.ira.uka.de/ jrtk/doc.LM/janus-lm.doku.html`, in der Fassung vom 10.10.1997.

[7] Ivica Rogina, Alex Waibel: Vorlesung Sprachliche Mensch-Maschine-Kommunikation. Universität Karlsruhe, Sommersemester 2003.

[8] Ernst Günter Schukat-Talamazzini: Automatische Spracherkennung. Vieweg, Braunschweig/Wiesbaden, 1995.

[9] Toni Stubblebine: Regular Expression Pocket Reference. O'Reilly, 2003.