Erkennen und Lernen neuer Wörter

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurswissenschaften
der Fakultät für Informatik
an der Universität Karlsruhe (Technische Hochschule)
genehmigte

Dissertation

von

Thomas Schaaf

aus Duisburg

Tag der mündlichen Prüfung: 22. Januar 2004
Erster Gutachter: Prof. Dr. A. Waibel
Zweiter Gutachter: Prof. Dr. K. Fellbaum

Zusammenfassung

Die zur Zeit besten Spracherkennungssysteme arbeiten mit einem Vokabular fester Größe. Hieraus ergibt sich bei hinreichend komplexen Domänen automatisch, daß niemals alle benötigten Wörter bei der Erkennung zur Verfügung stehen. Dies ist sogar bereits bei "kleinen" Domänen, wie beispielsweise der Terminabsprache oder Reiseplanung der Fall, für die kein geschlossenes Vokabular angegeben werden kann.

Begründet liegt dies in der Tatsache, daß es unmöglich ist, beispielsweise alle aktuellen Personennamen der Welt zu kennen, und dass es auch nicht möglich ist Wörter vorauszusehen, die zum gegenwärtigen Zeitpunkt noch gar nicht existent sind. Beispielsweise kommen in Nachrichtensendungen immer wieder neue Namen für Personen, Orte oder Organisationen zur Verwendung, die aus aller Welt stammen.

In dieser Arbeit wurde die GrOOV-Methode entwickelt, um in großen Domänen, wie beispielsweise der englischen Broadcast-News-Domäne (BN-Domäne), automatisch Klassen für die Modellierung von unbekannten Wörtern in einem statistischen Sprachmodell zu erzeugen. Die so erzeugten Klassen führen bei der Detektion unbekannter Wörter zu einer höheren Präzision im Vergleich zur Verwendung nur einer einzigen Klasse.

Für die akustische Modellierung unbekannter Wörter wurden verschiedene Varianten, sogenannte *Head-Tail-Modelle*, entworfen und untersucht. Die besten Modelle werden dabei unter Verwendung eines großen Hintergrundwörterbuches gewonnen und passen sich automatisch an die Größe des im Spracherkenner verwendeten Vokabulars an.

Mit den untersuchten Methoden wurde ein Prototyp realisiert, durch den ein sprachverarbeitendes System auf eigene Initiative den Benutzer auffordern kann, ihm neue Wörter beizubringen und somit seine Grenzen zu erweitern.

Für das Lernen eines unbekannten Wortes wurde festgestellt, dass die Wahrscheinlichkeit, mit der dieses Wort im Sprachmodell aufgenommen wird, höher zu schätzen ist, als die durchschnittliche Wahrscheinlichkeit eines OOV-Wortes. Hierzu wurden Untersuchungen zur Modellierung der Wahrscheinlichkeit unbekannter Wörter für Sprachmodelle durchgeführt.

Danksagung

Keine Dissertation ohne Danksagung, und ich bitte hiermit um Nachsicht, dass diese, angesichts meines nicht ganz so geradlinig verlaufenden Lebenslaufes, naturgemäß etwas länger ausfallen muss, da viele Menschen einen Einfluss darauf hatten, dass diese Arbeit nun vorliegt.

Entstanden ist diese Arbeit am Institut für Logik, Komplexität und Deduktionssysteme in der *Interactive Systems Labs* (ISL) Gruppe. Daher möchte ich zunächst meinem Doktorvater Professor Dr. Alexander Waibel für die gute Ausbildung danken, die ich an seinem Lehrstuhl erhalten habe. Danken möchte ich ihm auch für die Möglichkeit, an diesem internationalen, weltoffenen und interdisziplinären Institut mit seiner erstklassig ausgestatteten Forschungsumgebung arbeiten und promovieren zu können. Ganz besonders aber danken möchte ich ihm für die Betreuung dieser Arbeit, für sein Interesse, sein in mich gesetztes Vertrauen und den steten Ansporn diese Dissertation zu vollenden.

Ebenfalls zu Dank verpflichtet bin ich Herrn Professor Dr. Klaus Fellbaum für die Übernahme des Korreferats.

Aufgrund der vielen Jahre, die ich bereits als studentische Hilfskraft und Diplomand am ISL verbracht habe, gibt es viele ehemalige Kollegen und Kolleginnen in Karlsruhe und an der Carnegie Mellon University in Pittsburgh, denen ich für das meist angenehme und harmonische Arbeitsklima danken möchte. Aus den vielen Erfahrungen, die ich bei der Zusammenarbeit mit ihnen sammeln konnte, sowohl aus den positiven als auch aus den nicht so positiven, konnte ich einiges für meinen weiteren Lebensweg lernen. Dafür danken möchte ich hier Michael Bett, Susanne Burger, Matthias Denecke, Michael Finke, Christian Fügen, Jürgen Fritsch, Petra Geutner, Hermann Hild, Thomas Kemp, Detlef Koll, Kornell Laskowski, Victoria MacLaren, Rob Malkin, John McDonough, Uwe Meier, Florian Metze, Jürgen Reichert, Klaus Ries, Ivica Rogina, Tanja Schulz, Hagen Soltau, Rainer Stiefelhagen, Bernhard Suhm, Alica Tribble, Martin Westphal, Monika Woszczyna und Hua Yue.

Zu den ganz besonders positiven Erinnerungen zähle ich die Frühstücke montagmorgens um 8 Uhr mit Hermann Hild, Thomas Kemp, Martin Westphal und Monika Woszczyna, die nicht nur das Gemeinschaftsgefühl gestärkt, sondern auch viel zum wissenschaftlichen Austausch beigetragen haben.

Für diese Arbeit wichtige positive wissenschaftliche Impulse erhielt ich insbesondere von Jürgen Fritsch, Petra Geutner, John McDonough, Klaus Ries und Ivica Rogina, denen ich dafür danken möchte. Ivica Rogina gebührt mein ganz besonderer Dank, da er mich stets mit Rat und Tat fachlich und persönlich unterstützte und mir mit seinen Tipps half, die ersten Hürden beim Schreiben zu überwinden. Petra Geutner möchte ich danken, dass sie mir für einige Untersuchungen den von ihr entwickelten HDLA-Ansatz zur Verfügung gestellt hat und für ihre fernmündlichen Ratschläge und aufmunternden, motivierenden Gespräche. Matthias Honal möchte ich danken für die gute Zusammenarbeit bei der Verbesserung des Phonemerkenners.

Für die Betreuung des gesamten Rechnerbetriebes bedanken möchte ich mich bei Markus Baur, Frank Dreilich, Martin Klein und Norbert Berger. Ein ganz besonderes Dankeschön für den ganzen organisatorischen Ablauf im Sekretariat geht an Silke Dannenmaier, Ingrid Gemen, Sonja Seitz und Annette Römer.

Ein großes Dankeschön gebührt auch allen Korrekturlesern und Kommentatoren, die zur Fertigstellung dieser Arbeit einen wichtigen Beitrag geleistet haben. Nennen möchte ich hier insbesondere Petra Gieselmann, Hermann Hild, Jürgen Fritsch und Sylvia Ehlert.

An dieser Stelle möchte ich mich aber auch bei allen Lehrerinnen und Lehrern der Abendrealschule und des Abendgymnasiums der VHS Karlsruhe bedanken, die mir mit ihrem unermüdlichen Einsatz geholfen haben, auf dem zweiten Bildungsweg die Hochschulreife zu erlangen. Ein Dankeschön geht hierbei auch an meine damaligen Mitschüler, die wesentliches dazu beitrugen, die 5 1/2 Jahre durchzuhalten.
Großer Dank gebührt auch Ute Klein, die mich in meinem Vorhaben unterstützte, die mittlere Reife nachzuholen und es mir während meiner Ausbildungszeit zum Chemielaboranten ermöglichte, mich auf den Abendschulunterricht vorzubereiten.
Für ihren Ansporn und ihre Motivation den eingeschlagenen Weg weiter zu verfolgen, muss ich mich in besonderem Maße auch bei Herrn Feuerstein und Frau Schilli bedanken. Ohne ihre Anteilnahme wäre es mir unendlich schwerer gefallen, parallel zu einer Vollzeitbeschäftigung abends noch den Schulunterricht zu besuchen.

Ein großes Dankeschön geht auch an Dr. Thomas Klaus und Jürgen Lindemeyer, die mir zu Beginn meines Studiums hilfreich einen Weg in das Reich der Hochschulmathematik wiesen.

Schließlich möchte ich mich bei meiner Familie und bei meinen Freunden bedanken. Ein herzliches Danke geht an alle, die mich während meiner Promotion unterstützt und mich immer wieder ermuntert haben, weiter zu machen. Danke auch dafür, dass sie es mir immer wieder nachgesehen haben, dass mir für viele Unternehmungen einfach keine Zeit mehr geblieben ist.

Hierbei gilt natürlich mein besonderer Dank meiner Mutter Herta Eggl, die mich und

meine jüngere Schwester alleine grossziehen musste, was mit Sicherheit nicht immer einfach war. Danken möchte ich ihr dafür, was sie damals leisten musste und zuwege brachte.

Bedanken möchte ich mich auch bei meinem Schwager Martin Reihl, der, insbesondere im vergangenen Jahr, immer wieder zur Stelle war, um mir am Wochenende ungestörte Zeit zum Arbeiten zu verschaffen, einerlei ob es sich dabei um Probleme mit dem Auto, handwerklich-technische Arbeiten im Haushalt oder um Babysitten handelte.

An meine Schwester Michaela Ehlert geht ein besonderes Dankeschön dafür, dass sie mir vor 20 Jahren einen Weg zeigte, der nun bis hierher geführt hat. Danken möchte ich ihr aber auch für vieles mehr, beispielsweise dafür, dass sie eine wunderbare große Schwester und eine wunderbare Tante für meine Söhne ist.

Schließlich möchte ich mich ganz besonders bei meiner Lebensgefährtin und Ehefrau Christiane Reihl für die Liebe und Kraft, die sie mir gegeben hat, bedanken. In all den Jahren an meiner Seite hat sie den Stress und die Belastungen mit mir geteilt und getragen. Ihr Rückhalt und ihre Unterstützung bei der Promotion hat viel zum Gelingen dieser Arbeit beigetragen. Nicht zuletzt danken möchte ich ihr auch dafür, dass sie für unsere drei Söhne Jan, Philipp und Florian eine liebevolle und engagierte Mutter ist und einiges davon ausgeglichen hat, dass die drei oft auf ihren Vater verzichten und manches mal zurückstecken mussten.

Diese Arbeit widme ich meinen drei Söhnen und wünsche ihnen, dass sie niemals im Leben ihre Wissbegier und ihr Durchhaltevermögen verlieren werden.

Rheinstetten, 12. September 2003

Thomas Schaaf

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Kapitel 1

Einführung

"Wo Worte selten, haben sie Gewicht."

Shakespeare

"Guten Tag ich bin Herr Schaaf", lautete die Spracheingabe in ein State-of-the-Art Übersetzungssystem. Beim Gesprächspartner kam kurz darauf aus dem Lautsprecher die Ausgabe: "Hello I'm precise", womit dieser verständlicherweise nichts anfangen konnte.

Was ist geschehen, dass so eine unbrauchbare Übersetzung zustande gekommen ist?

Betrachtet man die Ausgabe des Spracherkenners, die als Grundlage für die Übersetzung diente: "Guten Tag ich bin haarscharf", erscheint die Qualität der Übersetzung doch gar nicht so schlecht.

Fehler beim Einsatz von Spracherkennungssystemen sind heute leider immer noch nichts Ungewöhnliches und gehören gewissermaßen zur Natur des verwendeten statistischen Ansatzes. Auch die Referenz, der Mensch, an der wir die Qualität eines Spracherkenners messen, macht Fehler, wenn auch selten. Wer kann von sich behaupten, noch nie ein Wort falsch verstanden zu haben?

Durch eine Wiederholung oder eine Reformulierung der Eingabe hätten die Erkennungsfehler[1] im obigen Beispiel nicht korrigiert werden können, da "Schaaf" für den Spracherkenner ein unbekanntes Wort ist, das *nicht* zu seinem Vokabular gehört.[2]

Die Spracherkennung hat sich in den letzten Jahren von der Erkennung weniger, präzise gesprochener Einzelwörter zu sprecherunabhängigen Systemen mit großen Wortschätzen entwickelt, die auch spontane, d.h. fehlerhaft gesprochene, Äußerungen robust verarbeiten können. Auch wenn das zur Verfügung stehende Vokabular sehr groß ist und die häufigsten Wörter enthält, gibt es immer wieder *seltene* Wörter, die nicht erkannt werden können, weil sie dem Spracherkenner unbekannt sind. Bei diesen Wörtern handelt es sich häufig um Wörter, die wichtige Informationen enthalten, die zum vollständigen Verständnis einer

[1] Es sind in der Tat zwei Fehler.

[2] Spracherkennungssysteme für große Vokabularien nach dem gegenwärtigen Stand der Technik arbeiten auf der Wortebene, da diese Modellierung bisher die besten Ergebnisse erzielte. Hieraus ergibt sich, dass nur bekannte, also eine endliche Menge, Wörter modelliert werden.

Äußerung notwendig sind, wie beispielsweise Eigennamen.

Wollte man 100% der Namen von Personen, die in einem deutschen Telefonbuch eingetragen sind, in einem Vokabular abdecken, müsste man eine Million Wörter aufnehmen. Mit einer halben Million Wörter deckt man jeden Namen ab, der mindestens zweimal im Telefonbuch enthalten sind. Um 90% der Einträge abzudecken, müsste man immer noch über 100000 Wörter in das Vokabular aufnehmen (vgl. Abbildung 1.1).

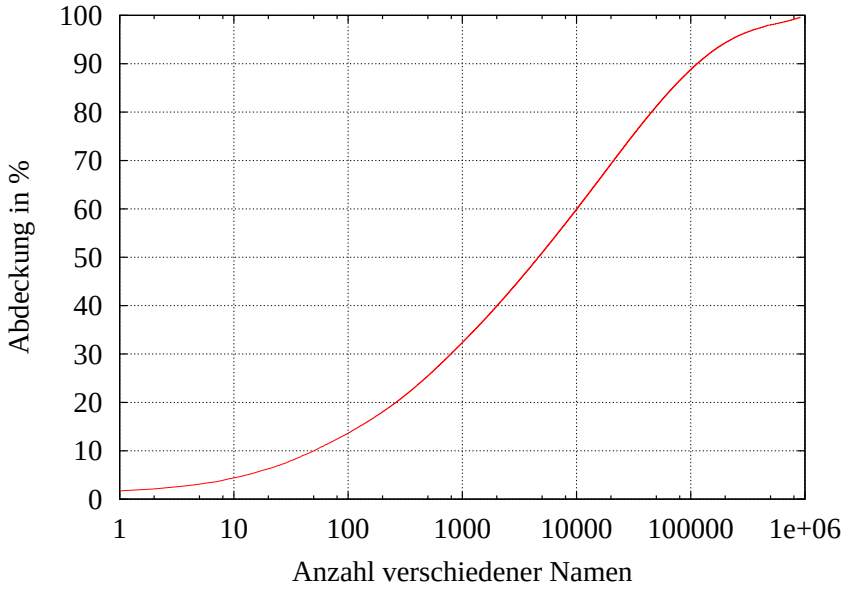

Abbildung 1.1: Abdeckung von Namenseinträgen im deutschen Telefonbuch

Mit dem beliebigen Anwachsen des Vokabulars steigt auch der Bedarf an Speicher- und Rechenkapazität. Zudem steigt die Verwechslungsgefahr von Wörtern, auch wenn diese seltenen Wörter in den allermeisten Fällen nie benutzt werden, da sie vorsorglich aufgenommen wurden. Selbst mit diesem Ansatz ist das Problem aber nicht bewältigt, da man immer noch mit unbekannten Wörtern rechnen muss. Denkt man dabei nur an die Neuschöpfung von Wörtern, liegt es auf der Hand, dass man prinzipiell zu dem Zeitpunkt, an dem ein Vokabular erstellt wird, nicht alle Wörter kennen *kann*.[3]

Der gegenwärtige Stand der Forschung, die sich mit dem Phänomen unbekannter Wörter befasst, konzentriert sich im Wesentlichen darauf, das Auftreten von unbekannten Wörtern zu erkennen und dann ihre Position zu markieren.

[3]Der Duden Rechtschreibung enthält in seiner 22., völlig neu bearbeiteten und erweiterten Auflage aus dem Jahr 2000 beispielsweise 5000 neue Wörter [21].

In der vorliegenden Arbeit werden die Methoden zur Modellierung und Detektion unbekannter Wörter in LVCSR-Systemen[4] aufgegriffen und weiterentwickelt. Es wird die Modellierung auf der akustischen Ebene und der Ebene des statistischen Sprachmodells untersucht und verbessert. Darüberhinausgehend wird ein neuer Ansatz vorgestellt, der das Wissen über das Auftreten unbekannter Wörter dazu benutzt, die Grenzen, die durch das Vokabular gesetzt sind, auf *Initiative des Sprachsystems* und in Kooperation mit dem Benutzer zu erweitern. Das System verfügt über die Möglichkeit, über seine ursprünglichen Grenzen hinauszuwachsen, indem es von sich aus einen Dialog mit dem Benutzer initiiert, um von ihm die zum Lernen notwendigen Informationen zu gewinnen. Ein gelerntes Wort steht danach zur weiteren Verwendung zur Verfügung.

1.1 Gliederung der Arbeit

Im Kapitel 2 werden grundlegende Verfahren der Spracherkennung vorgestellt und alle wesentlichen Begriffe definiert, die für diese Arbeit notwendig sind.

Die in dieser Arbeit verwendeten Sprachdatenbasen und Baseline-Sprach-erkennungssysteme werden in Kapitel 3 beschrieben.

Um festzustellen, welche Eigenschaften unbekannte Wörter besitzen und ob diese Gemeinsamkeiten aufweisen, wurden mehrere Untersuchungen auf unterschiedlichen Textkorpora durchgeführt und die Ergebnisse in Kapitel 4 miteinander verglichen.

Kapitel 5 gibt einen Überblick über die in der Forschung verwendeten Lösungsansätze und den Stand der Technik.

Im Kapitel 6 werden die im Rahmen dieser Arbeit neu entwickelten Verfahren zur Modellierung unbekannter Wörter vorgestellt.

Es handelt sich dabei zum einen um die unüberwachte Einteilung unbekannter Wörter in ein Klassensystem für unbekannte Wörter. Um dieses System mit anderen Verfahren zu vergleichen, wird ebenfalls beschrieben, wie andere statistische Sprachmodelle erzeugt werden, die in den Experimenten verwendet wurden.

Zum anderen wird die akustische Modellierung für unbekannte Wörter mit dem für diese Arbeit neu entwickelten Verfahren der Head-Tail-Modelle vorgestellt und untersucht. Abschließend wird gezeigt, wie diese Modelle genutzt werden können, um unbekannte Wörter zu detektieren.

Im Kapitel 7 werden die für diese Arbeit grundlegenden Experimente und Ergebnisse beschrieben.

[4] **L**arge **V**ocabulary **C**ontinuous **S**peech **R**ecognition

Es wird gezeigt, dass der Einsatz der im Rahmen dieser Arbeit entwickelten Techniken, wie Head-Tail-Modelle oder die unüberwacht gelernten Klassen, zu guten Ergebnissen bei der Erkennungsleistung führt.

Ferner wird untersucht, wie sich diese Modelle auf unterschiedlichen Domänen verhalten.

Im Kapitel 8 wird untersucht, wie man eine Aufteilung der Wahrscheinlichkeitsmasse zwischen der Detektion von unbekannten Wörtern und gelernten Wörtern durchführen kann und wie die nötigen Parameter dazu bestimmt werden können.

Es wird ein im Rahmen dieser Arbeit entwickeltes Verfahren vorgestellt, mit dem die Klasse (oder auch Klassen) bestimmt wird (werden), mit der (denen) ein unbekanntes Wort in das statistische Sprachmodell aufgenommen werden kann.

Einer der Prototypen zum interaktiven Lernen von unbekannten Wörtern auf Initiative des sprachverarbeitenden Systems wird in Kapitel 9 vorgestellt. Zunächst wird grob der Systemaufbau beschrieben. Anschließend werden die wesentlichen Komponenten und deren Zusammenspiel erläutert.

An einem konkreten Beispiel wird der Dialogablauf beim Erlernen eines unbekannten Wortes veranschaulicht.

Kapitel 10 bildet mit Zusammenfassung und Ausblick den Abschluss dieser Arbeit.

Kapitel 2

Grundlagen

Dieses Kapitel führt (kurz) in das Thema Spracherkennung ein. Terminologie und grundlegende Verfahren werden vorgestellt. Es werden Sprachdatenbasen beschrieben, der allgemeine Aufbau eines HMM-Spracherkenners wird dargestellt, verschiedene Spracherkennungssysteme und Evaluationskriterien werden erläutert. Ferner werden alle wesentlichen Begriffe definiert, die für diese Arbeit notwendig sind.

2.1 Sprachdaten

2.1.1 Einteilung der Sprachdatenbasen

Sprache dient der Kommunikation und kann in verschiedener Form auftreten. Man unterscheidet zwischen geschriebener Sprache, wie beispielsweise diesen Text, und gesprochener Sprache.

Große Korpora für geschriebene Sprache sind beispielsweise Zeitungstexte, wie die Süddeutsche Zeitung, die Frankfurter Allgemeine Zeitung, das Wallstreet Journal und das World Wide Web. Dabei sind bei Zeitungstexten die Sätze (meist) grammatikalisch korrekt und in einheitlicher Rechtschreibung formuliert, was bei Seiten aus dem WWW nicht unbedingt der Fall sein muss, da diese nicht von einer Redaktion bearbeitet worden sind.

Gesprochene Sprache besteht aus Klang und Phonemfolgen. Beim Menschen wird sie unter anderem durch die Erregung der Stimmbänder und die Stellung des Vokaltraktes erzeugt. Begleitet wird sie von Ausdrucksmitteln wie Tonhöhe, Lautstärke und Tonrhythmus und nichtverbalen Ausdrucksmitteln wie beispielsweise Mimik und Gestik.

Bei der maschinellen Spracherkennung kann Sprache gemäß verschiedener Kriterien eingeteilt werden, die am Schwierigkeitsgrad bei der Erkennung orientiert sind

und die sich in den Einschränkungen äußern, die dem Sprecher auferlegt werden. Mit zunehmendem Freiheitsgrad steigt die Schwierigkeit beim Finden der korrekten Hypothese.

Am einfachsten maschinell zu erkennen ist Sprache aus einzelnen isoliert gesprochenen Wörtern *(Einzelworterkennung)*. Jede Äußerung darf hierbei nur ein einziges Wort enthalten. Üblicherweise verfügt ein solches System nur über ein kleines Vokabular. Beispiel hierfür ist ein Ziffernerkenner, dessen Vokabular aus den zehn Ziffern und den Wörtern "ja" und "nein" besteht. Die Anwendungsmöglichkeiten für solche Systeme sind umfangreich, beispielsweise lässt sich ein menügeführtes Auskunftssystem realisieren.

Bei *semikontinuierlicher Sprache* muss vom Sprecher nach jedem Wort eine deutliche Pause eingefügt werden, um eine Segmentierung der Äußerung in einzelne Wörter zu unterstützen. Diese Systeme erfordern vom Anwender neben einer Anlernphase eine hohe Disziplin beim Sprechen und sind aufgrund dieser unnatürlichen Sprechweise anstrengend in der Benutzung. Systeme dieser Art wurden früher bei Diktiersystemen eingesetzt.

Bei *kontinuierlicher Sprache* entfällt die künstliche Pause zwischen den Wörtern. Unterschieden werden kann zwischen abgelesener und frei gesprochener Sprache.

Vorgelesene Sprache ist einfacher zu erkennen, da der Sprecher hier im allgemeinen flüssiger spricht. Nachrichtentexte im Radio oder Fernsehen werden außerdem meist von einem ausgebildeten Sprecher vorgelesen.

Spontane Sprache dagegen enthält oftmals Satzabbrüche und sogenannte 'Fehlstarts', bei denen mitten in einem Wort abgebrochen und ohne Übergang mit einem anderen Wort begonnen wird. Die Satzstrukturen unterscheiden sich unter Umständen deutlich von den Grammatikregeln für geschriebene Sprache und häufig werden vom Sprecher Pausenfüller (Hesitationen), wie beispielsweise <ähm> oder <eh>, verwendet. Der Satz «Ich hab' ähm am Mittw nein nächste Woche hab ich keine Zeit» enthält Pausenfüller und einen Satzabbruch. Aber auch hierbei gibt es Abstufungen in den Schwierigkeitsgraden bei der Erkennung. Ein Moderator mit Sprecherausbildung wird einfacher zu erkennen sein als eine Person, die beispielsweise interviewt wird und sich möglicherweise stark emotional äußert.

Die Schwierigkeit beim Finden der besten Hypothese korreliert auch stark mit der Qualität der Aufnahmen. Studioaufnahmen in einer optimalen Umgebung sind einfacher zu erkennen, als Aufnahmen, die in einer unkontrollierten realen Umgebung mit Hintergrundgeräuschen stattfinden. Im allgemeinen ist letzteres der Fall, wenn ein Spracherkennungsystem eingesetzt wird.

Typische Hintergrundgeräusche in einem Büro sind beispielsweise Telefonklingeln, Türenschlagen, Hintergrundgemurmel oder Computerrauschen. Außerhalb von Gebäuden kommt noch eine Vielzahl von ungünstigeren Faktoren hinzu, wie Geräusche in einem fahrenden Auto oder einem Hubschrauber, um nur einige Beispiele zu nennen.

Die Qualität der Aufnahmen hängt auch stark von der verwendeten Ausrüstung ab,

ob sie beispielsweise mit einem Nahbesprechungsmikrophon oder einem Tischmikrophon gemacht wurde, oder gar über eine Telefonleitung.

Es gibt Aufnahmen, die *speziell* für die Entwicklung von Spracherkennungsystemen gemacht wurden, wie beispielsweise für das BMBF-Projekt[1] "Verbmobil", bei dem Personen einen Terminkalender erhalten und die Aufgabe gestellt bekommen, eine Reise zu planen. Andererseits gibt es Aufnahmen, bei denen die Sprecher nicht eigens für die Entwicklung eines Spracherkennungssystems sprechen. Diese Sprachdaten werden beispielsweise aus dem Fernsehen, aus dem Radio oder bei Gesprächsrunden aufgezeichnet. Solche Sprache wird als "found speech" bezeichnet und stellt gegenwärtig in der Forschung eine der wesentlichen Herausforderungen dar, bei der sich Spracherkennungssysteme beweisen müssen.

Der Schwierigkeitsgrad steigt ferner mit der Anzahl der Sprecher und der Art der Gesprächspartner. Betrachtet man einen Dialog, der zwischen einem Menschen und einer Maschine geführt wird, beispielsweise bei einem Auskunftssystem, mit oder ohne Barge-in[2], birgt dieser andere Probleme als ein Dialog zweier Menschen, beispielsweise bei einer Terminabsprache, bei der es zu gleichzeitigem Sprechen der Gesprächspartner kommen kann (Crosstalk) oder einer Gesprächsrunde mit vielen Sprechern.

Eine weitere Herausforderung ist, ob das Spracherkennungssystem sprecherabhängig oder sprecherunabhängig sein soll. Ein sprecherunabhängiges System benötigt keine spezielle Trainingsphase in der das System sich in einer überwachten Form an einen Sprecher anpasst. *sprecheradaptive Systeme* sind Sprecherunabhängige Systeme die versuchen, alle nötigen Anpassungen an den aktuellen Sprecher von sich aus anhand der Hypothesen zu schätzen. Da hierbei stets Unsicherheiten auftreten, sind sprecherabhängige Systeme in der Regel besser, nachdem sie von einem Sprecher trainiert wurden. Das Training kann aber vom Anwender als störend empfunden werden, und in vielen Anwendungen ist dieser Mehraufwand einem Benutzer nicht zuzumuten.

Da spontan gesprochene Sprache aus einem nahezu unbegrenzten Wortschatz schöpfen kann, wird gegenwärtig bei der Anwendung das Gesprächsthema beschränkt. Es sind zwar weiterhin *alle* Wörter theoretisch erlaubt, aber die Beschränkung auf eine *Domäne* grenzt das zu erwartende Vokabular ein.

Es gibt Spracherkennungsaufgaben (Tasks), angefangen von einer eng begrenzten Domäne, die strenge Vorgaben hat, wie beispielsweise ein einziger Sprecher, ein sehr kleines Vokabular mit eingeschränkter Grammatik, bei denen nicht spontan gesprochen werden darf, wie beispielsweise dem Ressource-Management-Task, bei dem einfache Anfragen gestellt werden konnten (RM-Task). Es gibt Tasks, bei denen ein Sprecher nur eine Ja-/Nein-Eingabe, Ziffernfolgen oder Buchstabierungen benutzen darf, beispielsweise für Kreditkarten oder für das Ausfüllen von Formularen. Die Schwierigkeit bei den

[1] **B**undes**m**inisterium für **B**ildung und **F**orschung
[2] Hineinsprechen des Anwenders in die Ausgabe des Systems

Tasks steigt mit den Freiheiten im Vokabular und den Themen, wie beispielsweise bei der Domäne der Konferenzregistrierung (CR-Task), Diktieren von Zeitungsartikeln mit großem Wortschatz (WSJ-Task), Terminabsprache und Reiseplanung (ESST-Task, GSST-Task), dem Niederschreiben von Nachrichtensendungen (BN-Task, Hub-4), Vorlesungen und Präsentationen [63]. Zu den schwierigsten Tasks zählt der Switchboard-Task (SWB-Task), bei dem zwei Personen über Telefonleitung miteinander eine spontane Konversation zu einem zufällig ausgewählten Thema führen und die aktuell aufkommenden, nahezu unbegrenzten Tasks, wie allgemeine Besprechungen und Diskussionsrunden mit mehreren Sprechern in unterschiedlichen Sprachen (Meeting-Task [77, 69]).

2.2 Vokabular, Wörter und Wörterbücher

Erkennervokabular

Die Festlegung eines *Vokabulars* für LVCSR-Systeme wird im allgemeinen aufgrund der Häufigkeit (Frequenz) von Wörtern auf einem für die Domäne geeigneten Textkorpus durchgeführt. Wörter w, die zum Vokabular V eines Spracherkenners gehören sollen, werden auch als *Vokabular-Wörter* (*I*n *V*ocabulary, *IV-Wörter*) bezeichnet. Wörter, die nicht in diesem festgelegten Vokabular V enthalten sind, d.h. die im verwendeten Textkorpus nicht aufgetreten sind oder die die geforderte Mindesthäufigkeit nicht erfüllen, werden *OOV-Wörter* (*O*ut *O*f *V*ocabulary) genannt. Der prozentuale Anteil an OOV-Wörtern auf einem *Textkorpus* K wird als *OOV-Rate* bezeichnet. Sie ist stark abhängig vom verwendeten Vokabular und dem Textkorpus. Im optimalen Fall handelt es sich beim verwendeten Textkorpus um eine große Menge für die Ziel-Domäne transkribierter Daten, die dann auch für die Berechnung eines statistischen Sprachmodells benutzt werden.

Für die BN-Domäne ist es aber bei der Festlegung des Vokabulars beispielsweise üblich, zur Verfügung stehende Manuskripte mit Zeitungstexten zu untersuchen, und darüber ein Vokabular zu bestimmen.

Abbildung 2.1 zeigt eine mögliche Einteilung der Wörter eines Textkorpus anhand der Häufigkeit ihres Auftretens bei einer geforderten Mindesthäufigkeit von 31. Daraus ergibt sich, dass in diesem Fall im Vokabular eines Spracherkenners alle Wörter enthalten wären, die häufiger als 31 mal aufgetreten sind (oberes Dreieck im Bild). Alle Wörter, die diese Mindesthäufigkeit nicht erfüllen oder überhaupt nicht im Textkorpus enthalten sind, werden somit als OOV-Wörter bezeichnet.

Für das Verbmobil-Projekt wurden Transkripte erstellt und von Experten Vorgaben gemacht, welche Wörter außerdem noch zu erwarten sind, beispielsweise der Name "Bulmahn"[3].

[3]Frau Bulmahn war zu der Zeit des Verbmobil-Projekts Forschungsministerin

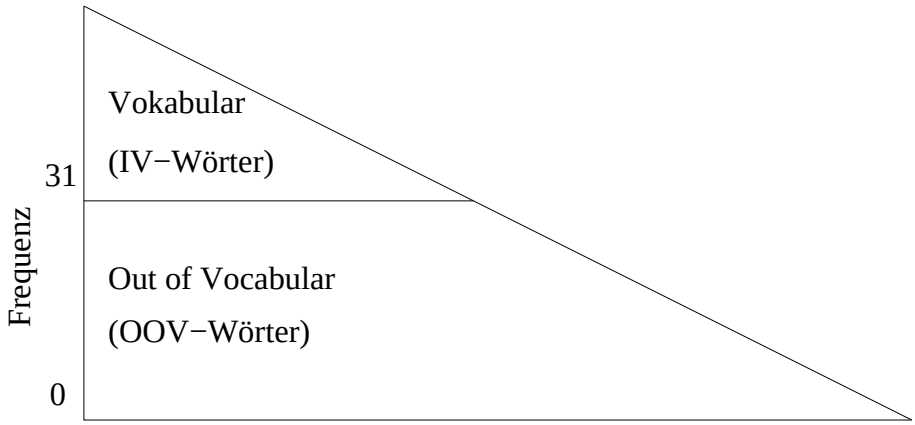

Abbildung 2.1: Einteilung der Wörter anhand einer geforderten Mindesthäufigkeit

Das Ziel beim Erstellen eines Vokabulars ist es, möglichst viele der benötigten Wörter zu beherrschen. Wächst das Vokabular, treten aber zwei negative Effekte auf. Zum einen ist es für das Training der akustischen Modelle schwierig, seltene Wörter und deren phonetischen Kontexte in ausreichender Zahl zu sehen, um so zuverlässige Modelle zu trainieren, weshalb diese akustischen Einheiten mit anderen Kontexten gemeinsam modelliert werden müssen. Zum anderen steigt mit der Anzahl an Wörtern auch die Wahrscheinlichkeit dafür, dass eines dieser Wörter akustisch mit einem anderen verwechselt werden kann. So kann es vorkommen, dass mit einem größeren Vokabular die Wortfehlerrate wieder steigt, obwohl die Zahl der unbekannten Wörter sinkt. Optimal wäre es, wenn der Spracherkenner immer nur die *tatsächlich benötigten* Wörter kennen würde.

Bei der Festlegung des Vokabulars wird also versucht abzuwägen zwischen der Anzahl an Fehlern, die dadurch verursacht werden, dass ein Wort im Vokabular unbekannt ist und der Anzahl an Fehlern, die dadurch entstehen, dass ein Vokabular so umfangreich ist, dass es vermehrt zu Verwechslungen von Wörtern kommen kann.

Um die OOV-Rate zu minimieren, die eine untere Schranke für die Fehlerrate darstellt, legt man ein Vokabular dadurch fest, dass alle Wörter $w \in V$ eine *Mindesthäufigkeit* min auf einem untersuchten Textkorpus K besitzen. Die Mindesthäufigkeit kann von verschiedenen Faktoren abhängen. Es wird beispielsweise eine bestimmte Menge von Beispielen benötigt, um ein Wort zu trainieren, oder die Mindesthäufigkeit hängt vom angestrebten Umfang des Vokabulars ab. Es gilt $V_{\min} := \{w : freq_K(w) \geq min\}$.

Da man zum gleichen Vokabular gelangen würde, wenn man die Wörter w nach ihrer Häufigkeit $freq_K(w)$ sortiert und die N häufigsten (TopN) davon auswählt, kann das gleiche Vokabular auch mit V_{TopN} bezeichnet werden.

So ergibt sich beispielsweise für den englischen BN-Task bei einer geforderten

Mindesthäufigkeit für ein Vokabular-Wort von 31 auf einem Trainingskorpus, dass $V_{\text{min31}} = V_{\text{Top39734}}$ ist. Dieses Vokabular enthält also die $|V_{\text{min31}}| = 39734$ häufigsten Wörter dieses Textkorpus.

Auf der GSST-Task von Verbmobil wurde für die Erzeugung eines Vokabulars beispielsweise eine Mindesthäufigkeit von vier benutzt [36].

Basierend auf einem so gewählten Vokabular werden dann die davon abhängigen Parameter, wie die *Aussprache* für das Wörterbuch des Spracherkenners und das *statistische Sprachmodell*, bestimmt.

Aussprachewörterbuch

Das Wörterbuch eines Spracherkenners definiert, aus welchen akustischen Einheiten sich ein Wort zusammensetzt, das im Vokabular des Spracherkenners enthalten ist und damit, wie es sich (korrekterweise) anhören sollte. Üblicherweise wird hierfür eine symbolische Beschreibung verwendet, die beispielsweise dem Zeichensystem der International Phonetic Association (IPA) folgt. Diese Lautschrift wird auch in Wörterbüchern, wie beispielsweise der Duden-Reihe [20] benutzt. Da sich der dabei verwendete Zeichensatz aber nur schwer mit einer Computertastatur eingeben lässt, wurden andere Darstellungsformen entwickelt. Eine weit verbreitete ist *Sampa* [80], die sich auf den ASCII-Zeichensatz beschränkt. In Sampa würde beispielsweise das Wort "Schaaf" als /S a: f / repräsentiert werden. In der Regel benutzen viele Spracherkennungssysteme eine eigene interne Darstellung, die sich nicht immer bijektiv auf eine Repräsentation in Sampa abbilden lässt.
In dieser Arbeit werden Phoneme in Sampa-Notation angegeben.

2.3 Statistischer Ansatz zur Spracherkennung

Heutige Spracherkenner basieren auf Methoden der statistischen Mustererkennung. Die Aufgabe dabei ist, eine Hypothese $\hat{W}$ zu einer Äußerung A zu finden, wobei die wahrscheinlichste Wortfolge zu bestimmen ist, also $\hat{W} = \arg\max_W p(W|A)$. Mit der Bayes-Formel

$$p(W|A) = \frac{p(A|W)p(W)}{p(A)} \tag{2.1}$$

lässt sich diese Aufgabe umformulieren woraus sich nun die Hypothese durch

$$\hat{W} = \arg\max_W \frac{p(A|W)p(W)}{p(A)} \tag{2.2}$$

bestimmen lässt.

Der Vorteil ist, dass nun für die Wortfolgen W statistische Modelle eingesetzt werden können, um $p(A|W)$ zu berechnen. Diese Modelle sind die akustischen Modelle eines Spracherkenners. Die a priori Wahrscheinlichkeit $p(W)$ wird dagegen von den Sprachmodellen eines Spracherkenners geschätzt. Da die a priori Wahrscheinlichkeit $p(A)$ für das Auftreten der Äußerung A zu einer gegebenen Äußerung unabhängig von der Wortfolge W ist, besitzt sie keinen Einfluss auf die Bildung des Maximums und wird für die Bestimmung der Hypothese $\hat{W}$ nicht benötigt.

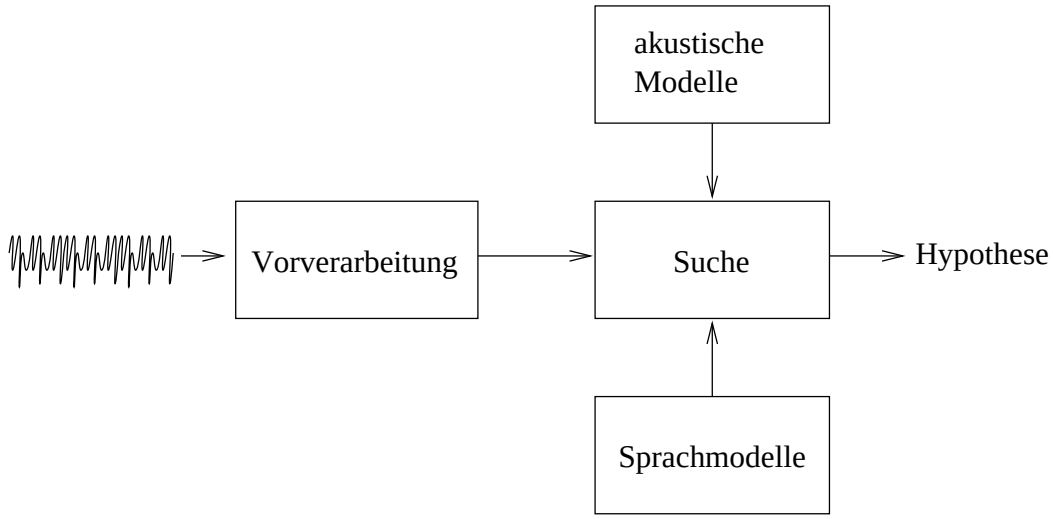

Abbildung 2.2: Aufbau eines Spracherkenners

Abbildung 2.2 zeigt den Grundaufbau eines Spracherkenners mit seinen Modellen. Die Vorverarbeitung dient dabei der Merkmalsberechnung für die akustischen Modelle und in der Suche wird $\hat{W} = \arg\max_W p(A|W)p(W)$ bestimmt. $\hat{W}$ ist dann die Hypothese des Erkenners.

Die Vorverarbeitung ist ein wichtiger Bestandteil eines Spracherkennungssystems und kann einen grossen Einfluss auf die Qualiät einer Hypothese haben. Sie dient dazu, die wesentlichen Merkmale der Sprache hervorzuheben, störendes Rauschen zu unterdrücken und das Sprachsignal in eine normalisierte Darstellung zu übertragen, in der sprecherbedingte und kanalabhängige Unterschiede ausgeglichen sind (CMN[4]). Beispielsweise besitzen verschiedene Sprecher üblicherweise einen verschieden langen Vokaltrakt, was das Sprachsignal beeinflusst. Zusätzlich wird eine Datenkompression angestrebt, um die Menge der zu verarbeitenden Daten gering zu halten. Die Vorverarbeitung liefert die *Observationen* (Merkmals-Vektoren), auf denen die statistischen Modelle dann operieren. Jedoch ist die Vorverarbeitung weitestgehend unabhängig vom verwendeten Vokabular des Spracherkenners und somit haben OOV-Wörter keinen relevanten Einfluss darauf. Ausführliche

[4]Cepstrale Mean Normalisation

Beschreibungen, wie Signalvorverarbeitung für die Spracherkennung durchgeführt werden kann, ist beispielsweise in [61] und [79] zu finden.

2.4 Aufbau eines HMM-Spracherkenners

Sprache wird normalerweise als eine zeitliche Abfolge von Lauten (Phonen) wahrgenommen. Ein Phonem stellt die kleinste akutische Einheit dar, die die Bedeutung eines Wortes unterscheidet und kann je nach Sprache eine Klasse von Lauten repräsentieren.[5] Auch ist die Reihenfolge einzelner Phoneme wichtig für die Bedeutung eines Wortes, wie beispielsweise bei den Wörtern "mit" und "Tim". Modelle der Mustererkennung, bei denen eine zeitliche Reihenfolge berücksichtigt wird, sind *Versteckte Markov-Modelle* (HMM = Hidden Markov-Models) [58] [70], bei denen in jedem Zustand Observationen mit zustandsabhängigen Verteilungen möglich sind. In der Spracherkennung modellieren Zustände häufig Teile eines Wortes, beispielsweise ein Phonem, wovon wir im folgenden ausgehen.

Ein HMM besteht aus fünf Mengen und lässt sich durch ein 5-Tupel $\lambda = (S, K, O, P, \pi)$ charakterisieren [58]:

1. Menge S von Zuständen $\{s_1 \dots s_N\}$
2. Menge K von gerichteten, gewichteten Kanten $k = (k_i, k_j, w_{i,j})$
3. Observationsalphabet O
4. Menge P von Emissionswahrscheinlichkeiten $P(O|j)$, $o \in O, 1 \leq j \leq N$
5. Initiale Zustandsbelegung $\pi = \pi_j, 1 \leq j \leq N$ der Zustände

Abbildung 2.3 zeigt ein einfaches HMM mit zwei Zuständen s_1 und s_2, in dem die Zeichen A und B des Observationsalphabets emittiert werden können.

Ein *HMM* wird in der Spracherkennung nach bestimmten Modelleigenschaften unterschieden. Eine Eigenschaft ist, wie viele Zustände verwendet werden, um ein Phonem zu modellieren. Möglich ist, für jedes Phonem genau einen Zustand zu verwenden, üblich sind aber drei oder mehr. Bei drei Zuständen kann der erste als Anlaufphase des Phonems angesehen werden, die noch durch die Nachbarschaft eines Vorgänger-Phonems beeinflusst ist. Auf die gleiche Weise lässt sich der dritte Zustand interpretieren, der das Ende des Phonems markiert, der bereits vom nachfolgenden Phonem beeinflusst wird. Der mittlere Zustand dagegen wird als vom Kontext wenig beeinflusst angesehen. Diese speziellen Positionen werden hier mit 'a', 'm', 'e' für Anfang, Mitte und Ende abgekürzt. Abbildung 2.4 zeigt die verschiedenen Möglichkeiten.

[5] Im Deutschen ist beispielsweise der Laut des Reibe-r und des Zungenspitzen-r nicht bedeutungsunterscheidend und in einer Phonemklasse /R/ zusammengefasst, im Spanischen gehören sie zu zwei verschiedenen Phonemklassen

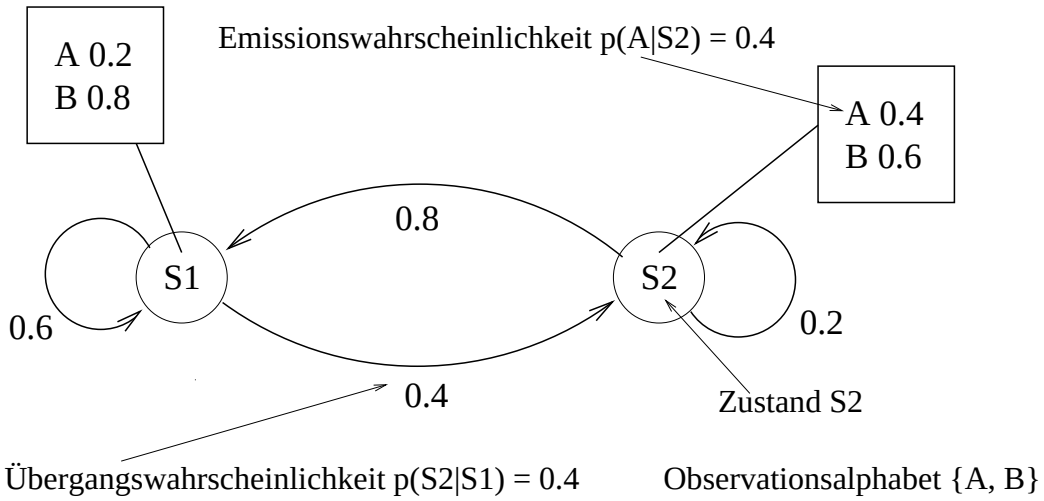

Abbildung 2.3: Einfaches HMM

Wenn es in einem HMM zu den Phonemen unterschiedliche, von den Nachbarphonemen abhängige Modelle gibt, so wird ein solches Modell *kontextabhängiges HMM* (CD = Context Dependent) genannt. Gibt es dagegen zu jedem Phonem nur ein einziges Modell, nennt man es ein *kontextunabhängiges HMM* (CI = Context Independent). Beispielsweise kann in einem kontextabhängigen HMM der Zustand 'a' des Phonems ein anderer sein, je nachdem ob der linke Kontext ein Plosiv- oder ein Frikativlaut ist. Der Effekt der Co-Artikulation tritt ebenfalls an Wortgrenzen auf und kann auch dort durch Wortgrenzen übergreifende kontextabhängige Modelle modelliert werden. Diese werden als Cross-Word-Triphone-Modelle (XWT-Modelle) bezeichnet.

Da bei den hier vorgestellten Modellen nur auf die akustischen Eigenschaften der Wörter eingegangen wird, werden diese Modelle *akustische Wortmodelle (akustische Modellierung)* genannt. Die Kombination von Wortmodellen wird mit Hilfe eines Sprachmodells durchgeführt (vgl. Abschnitt 2.7).

Im Zusammenhang mit einem HMM λ werden drei grundlegende Probleme unterschieden [58]:

- Das Optimierungsproblem:
  Wie können mit einer (hinreichend langen) Beobachtung O die Modellparameter so geschätzt werden, dass das Modell besser zu den Beobachtungen passt.

- Das Evaluationsproblem:
  Wie wahrscheinlich ist es, dass eine gegebene Merkmalsfolge O von Modell λ erzeugt wird.

- Das Dekodierungsproblem:
  Wie wird zu einer Beobachtungsfolge O und einem Modell λ die wahrscheinlichste Zustandsfolge für λ gefunden, die O produziert.

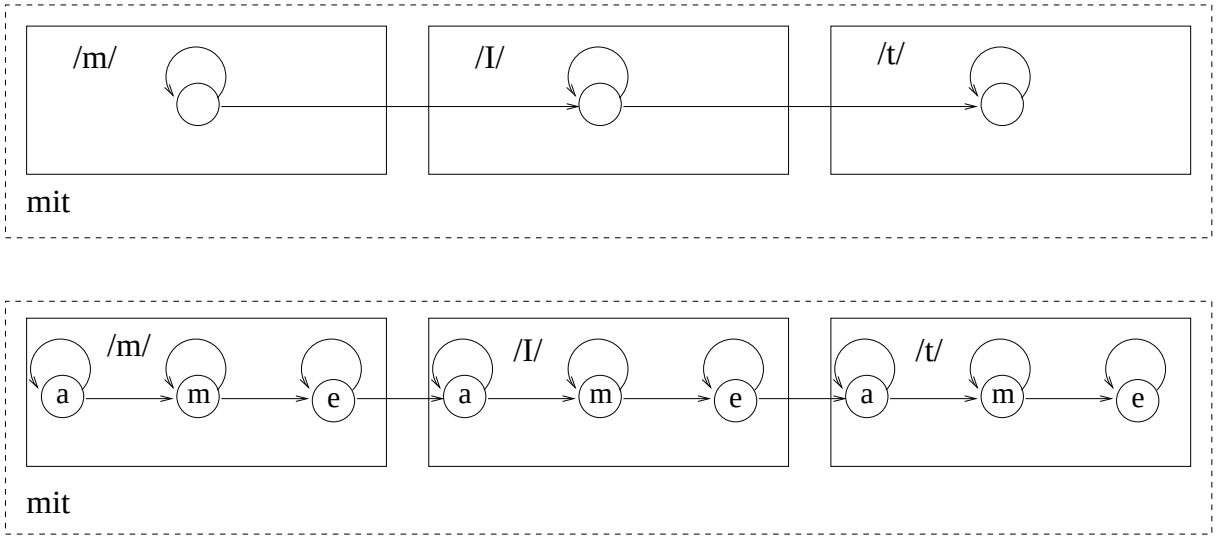

Abbildung 2.4: Mögliche HMM-Realisierungen für Wörter/Phoneme

Für die Lösung der Probleme gibt es verschiedene Algorithmen, die in [58] beschrieben sind.

2.5 Viterbi-Algorithmus

Die Aufgabe eines Spracherkenners ist es, herauszufinden, was in einer Äußerung gesagt wurde. Dies wird als *Dekodierung* und der Teil des Spracherkenners, der zu dieser Aufgabe eingesetzt wird, wird als *Dekoder* bezeichnet. Hierzu wird ein HMM λ verwendet, das aus Wortmodellen besteht, die untereinander so verbunden werden, dass die erlaubten Wortfolgen möglich sind.

Mit diesem Modell ist es nun anhand der Bayes-Formel 2.1 möglich, zu einer Observation O (Äußerung) für die möglichen Wortfolgen $W = w_1, \ldots, w_n$ die Wahrscheinlichkeit $p(O|W)$ zu bestimmen und dann die wahrscheinlichste Wortfolge $\hat{w}$ auszuwählen (vgl. Abschnitt 2.3).

Zur Berechnung der wahrscheinlichsten Wortfolge kann der *Viterbi-Algorithmus* verwendet werden. Er liefert als Ergebnis diejenige Zustandsfolge zum Modell λ, für die die Wahrscheinlichkeit der Observation maximal ist. Eine Zustandsfolge wird auch *Viterbi-Pfad* genannt, wenn sie durch den Viterbi-Algorithmus berechnet wurde. Die wahrscheinlichste Wortfolge kann gefunden werden, indem untersucht wird, zu welchen Wörtern die Zustände des Viterbi-Pfades gehören. Dabei lassen sich auch die Wortgrenzen bestimmen, das heißt, die Zeitpunkte, zu denen der Viterbi-Pfad von einem Wort in ein anderes wechselt.

Wenn die Äußerung bekannt ist, kann ein spezielles HMM nur für diese Äußerung erzeugt werden, mit dem eine Segmentierung durchgeführt werden kann. Dies wird als *Forced–Alignment* bezeichnet. Basierend auf diesem Verfahren können Spracherkenner trainiert (*Viterbi-Training*) werden.

Für die Bestimmung des global besten Viterbi-Pfades müssen häufig sehr viele Wahrscheinlichkeiten multipliziert werden. Aus numerischen und Effizienzgründen werden die Wahrscheinlichkeiten üblicherweise logarithmiert und negiert. Eine so umgeformte Wahrscheinlichkeit wird dann *Score* genannt. Die Multiplikationen der Wahrscheinlichkeiten werden durch das Logarithmieren zur Addition. Die umgerechnete Wahrscheinlichkeit einer Zustandsfolge wird als *akkumulierter Score* bezeichnet.

Die Integration eines Sprachmodells erfolgt, indem die Scores für Wortübergänge aus dem Sprachmodell mit einem *Sprachmodellgewicht* multipliziert werden. Damit wird der Einfluss des Sprachmodells gegenüber dem akustischen Modell gesteuert. Zusätzlich gibt es den Parameter *Wortübergangsstrafterm*, der auf die Scores addiert wird. Mit dem Wortübergangsstrafterm wird gesteuert, ob viele oder wenige Wörter in einer Hypothese sein sollen. Das Bestimmen der beiden Parameter erfolgt empirisch, und ihre Einstellung hat großen Einfluss auf die Leistungsfähigkeit eines Spracherkenners.

Da die Dekodierung eine sehr rechenintensive Aufgabe ist, wurden verschiedene Techniken entwickelt, um den Suchraum zu beschneiden (*Prunen*). Eine Technik ist die *Strahlsuche* (Beam Search), bei der Zustände aktiv oder inaktiv sein können. Ein Zustand wechselt von aktiv nach inaktiv wenn ein Viterbi-Pfad, der aktuell dort endet, eine sehr viel geringere Wahrscheinlichkeit als der aktuell beste Viterbi-Pfad besitzt. Technisch realisiert wird dies, indem die Differenz des Viterbi-Scores des Zustandes zum besten Viterbi-Score gebildet und dieser Wert mit einer Schwelle verglichen wird. Bei der Beschneidung des Suchraums kann es vorkommen, dass der Viterbi-Pfad beendet wird, der am Ende der Suche Teil des global besten Pfades gewesen wäre. Erkennungsfehler, die darauf beruhen, werden *Suchfehler* genannt [74].

Weitere Methoden, um die Decodierung einer Äusserung zu beschleunigen sind beispielsweise, die in der akustischen Modellierung verwendeten Gaußverteilungen abhängig von der Observation O nur teilweise auszuwerten (Bucked-Box-Intersection [27]), oder mit einfachen akustischen Modellen eine vorausschauende Selektion aktiver Zustände (Fast-Match, Lookahead [27]) durchzuführen.

2.6 Der Suchraum

Der Suchraum eines LVCSR-Systems erlaubt bei Verwendung eines statistischen Sprachmodells prinzipiell, dass *jedes* Wort aus dem Vokabular nach jedem anderen Wort aus dem Vokabular auftreten kann. Jedes Wort wird dabei durch eine Folge von HMM-Zuständen modelliert. Das Ziel für eine effiziente Suche ist es, möglichst wenig redundante Berechnungen durchzuführen.

Eine sehr einfache Methode, den Suchraum zu organisieren, ist in Abbildung 2.5 zu sehen, in dem für die Wörter "Wann", "Was" und "Wo" ein Zustandsnetzwerk dargestellt ist, das alle Wörter parallel modelliert.

Wie man leicht erkennen kann, beginnen alle Wörter mit dem Phonem /v/. Eine kompaktere Repräsentation dieses Suchraums kann durch das Zusammenfassen glei-

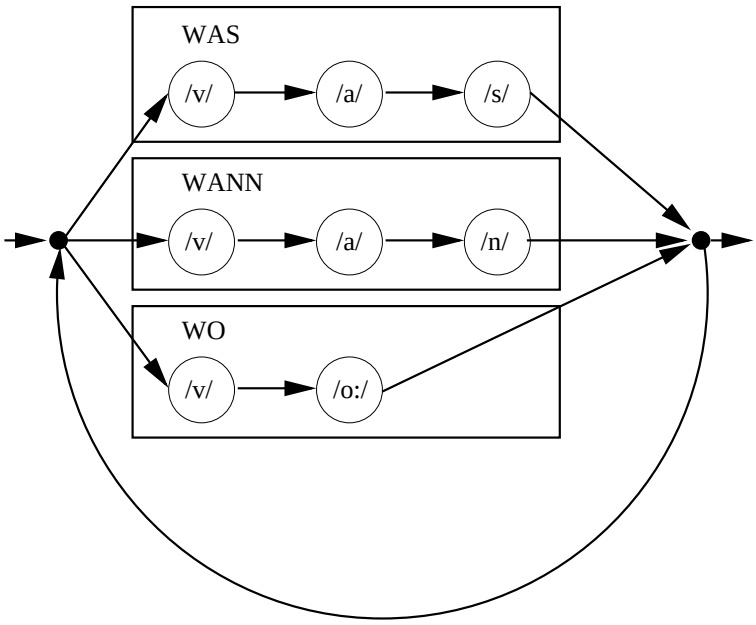

Abbildung 2.5: Flache Repräsentation des lexikalischen Suchraums

cher Aussprache-Präfixe erreicht werden. In Abbildung 2.6 ist eine Repräsentation als Aussprache-Präfix-Baum (*Pronunciation Prefix Tree, PPT*) für das Beispiel gegeben.

Diese Repräsentation benötigt nur fünf "Zustände" anstelle von acht. Wird kein Sprachmodell verwendet, sind beide Repräsentationen auch äquivalent. Bereits bei einem Monogramm ergeben sich durch die Struktur Unterschiede, die sich bei der Beschneidung des Suchraums auswirken können.

Der wesentliche Unterschied in der Aussprache-Präfix-Baum-Repräsentation ist, dass in unserem Beispiel für einen aktiven Viterbi-Pfad im Zustand /v/ noch unbekannt ist, ob er gerade zum Wort "WANN", "WAS" oder "WO" gehört. Die flache Repräsentation aus Abbildung 2.5 kann dies bereits im ersten Zustand entscheiden. Dies erlaubt es, die aus einem N-Gramm-Sprachmodell stammende Wahrscheinlichkeit zu diesem Zeitpunkt gleich in vollem Umfang in den Viterbi-Pfad einzurechnen. Bei der Aussprache-Präfix-Baum-Darstellung ist dies nur schrittweise möglich, bis die Wortidentität eindeutig ist. Ohne Beschneiden des Suchraums kommen beide Repräsentationen zum gleichen Ergebnis, aber die Aussprache-Präfix-Baum-Repräsentation ist effizienter im Speicherbedarf und in der Rechenzeit [62][72].

Wird eine Beschneidung des Suchraums durchgeführt, ist es möglich, dass sich, aufgrund von Suchfehlern, die an unterschiedlichen Stellen auftreten können, auch unterschiedliche Hypothesen ergeben. Bei sehr großen Vokabularien ist bei Verwendung von Aussprache-Präfix-Bäumen eine deutliche Reduktion der Anzahl von Zuständen möglich, was insgesamt zu einer beschleunigten Spracherkennung führt und somit auch dazu, dass gegebenenfalls

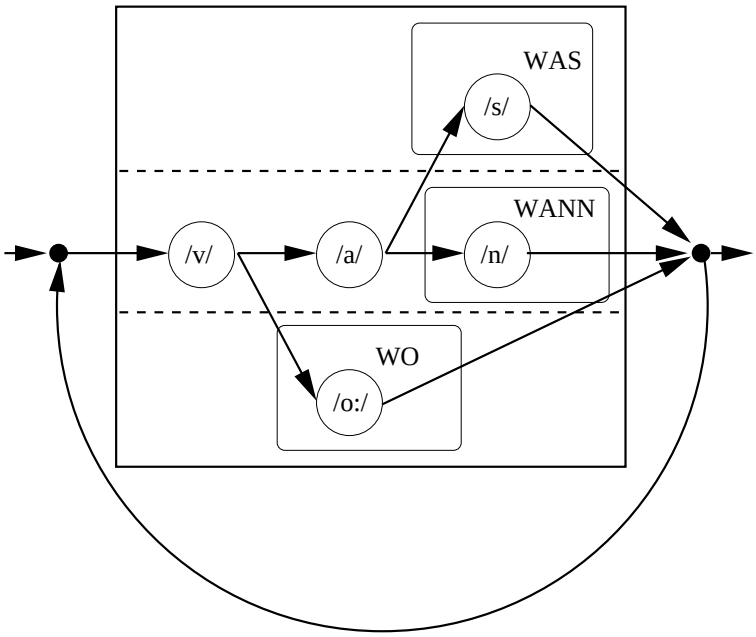

Abbildung 2.6: Aussprache-Präfix-Baum-Repräsentation des lexikalischen Suchraums

der Suchraum weniger stark beschnitten werden muss und insgesamt weniger Suchfehler auftreten.

In dieser Ausarbeitung wird häufig anstelle des Wortes Aussprache-Präfix der kürzere Begriff Präfix benutzt, womit eine initiale Phonemfolge eines Wortes gemeint ist.

2.7 Statistische Sprachmodelle

Die Wörter, die ein Spracherkenner erkennen kann, bilden sein *Vokabular* und sind in einem *Wörterbuch* zusammengefaßt. Dieses Wörterbuch besitzt zusätzlich die Beschreibung des Aufbaus der Wörter aus Phonemen. Für die Modellierung spontansprachlicher Effekte wie Stottern, Stammeln, Atmen enthält dieses Wörterbuch meist spezielle Wörter. Um den Spracherkenner zu unterstützen, wird ein Modell für die erwarteten Äußerungen (Sprachmodell) verwendet.

Ein Typ von Sprachmodell schreibt genau vor, welche Äußerungen möglich sind, indem als Grammatik ein *endlicher Automat* (reguläre Sprache) [59] verwendet wird. Ein solches Sprachmodell schränkt die Verbindungsstruktur der *Wortmodelle* eines HMM ein. Dieser Ansatz ist für spontane Sprache aber wenig geeignet, da fast jede Wortkombination möglich sein kann, obwohl sie eventuell sehr unwahrscheinlich ist. Ein statistisches Sprachmodell besitzt diese Einschränkung nicht. Es liefert die Wahrscheinlichkeit, dass ein Wort w_m auftritt, basierend auf seiner Historie, also $p(w_m|w_{m-1}w_{m-2}\dots w_1)$. Diese Wahrscheinlichkeit kann in der Bayes-Formel 2.1 für die Berechnung von $p(W)$ verwendet werden. Dazu wird für die Wortfolge $W = w_1w_2\dots w_n$ die a priori Wahrscheinlichkeit $p(W)$ durch $\prod_{m=1}^{n} p(w_m|w_{m-1}\dots w_1)$ berechnet.

Wie gelangt man nun zu einem statistischen Sprachmodell? Gehen wir davon aus, dass es im Deutschen ungefähr 300000 Wörter[6] gibt und die meisten davon sehr selten auftreten. Für die Schätzung der Wahrscheinlichkeiten wird daher eine immense Menge an Wortfolgen benötigt. Selbst wenn nur eine beschränkte Historie von zwei Wörtern (Trigramm-Sprachmodell) oder einem Wort (Bigramm-Sprachmodell) berücksichtigt wird, sind die benötigten Datenmengen für eine sichere Schätzung enorm groß. Aus diesem Grund wurden verschiedene Verfahren entwickelt, ungesehene Tri- oder Bigramme zu schätzen, indem beispielsweise eine *interpolierte Wahrscheinlichkeit* verwendet oder eine sogenannte *Back-off-Wahrscheinlichkeit* [46][70] eingesetzt wird. Ein statistisches Sprachmodell kann dann als Übergangswahrscheinlichkeit in einem HMM von einem Wortmodell zu einem anderen Wortmodell (Bigramm) interpretiert werden.

Wie schwer Sprachmodellierung an sich ist, sei an folgendem Beispiel illustriert. Im Louvre ist der Satz «Hier sehen Sie die Mona Lisa von Leonardo da Vinci» in der Nähe des Bildes sehr viel wahrscheinlicher als auf dem Fischmarkt in Hamburg. Der Satz «Wir schließen in einer Stunde» dagegen ist von der Zeit abhängig und daher an beiden Orten eine Stunde vor dem Schließen wahrscheinlicher als zwei Stunden davor. Mögliche Äußerungen sind von so vielen Faktoren abhängig und besitzen eine so große Vielfalt, dass Sprache nur schwer in Modelle zu fassen ist.

2.7.1 Berechnung eines N-Gramm-Sprachmodells

Für die Berechnung der a priori Wahrscheinlichkeit $p(W)$ aus Formel 2.1 kann diese folgendermaßen zerlegt werden:

$$\begin{aligned} p(W) &= p(w_1, ..., w_n) \\ &= p(w_1)p(w_2|w_1)p(w_3(w_1, w_2) \dots p(w_n|w_1, w_2, \dots, w_{n-1})) \\ &= \Pi_{i=1}^{n} p(w_i|w_1, w_2, \dots, w_{i-1}) \end{aligned} \tag{2.3}$$

Dabei ist $p(w_i|w_1, w_2, \dots, w_{i-1})$ die Wahrscheinlichkeit, dass das Wort w_i der Wortfolge $w_1, w_2, \dots, w_{i-1}$ folgt. Die Wortfolge $w_1, w_2, \dots, w_{i-1}$ wird auch als *Historie* h_{i-1} für das Wort w_i bezeichnet. In der Formulierung der Gleichung 2.3 ist die Wahrscheinlichkeit für das Wort w_i abhängig von jedem Wort aus der Historie h_{i-1}.

Für ein Vokabular mit $|V|$ verschiedenen Wörtern müsste für die vollständige Bestimmung der Wahrscheinlichkeiten $p(w_i|w_1, w_2, \dots, w_{i-1})$ $|V|^i$ Werte bestimmt werden. Selbst für kurze Historien ist die Zahl der zu schätzenden Parameter immens. Daher ist es praktisch unmöglich alle Werte zu bestimmen, da die meisten Historien nur wenige

[6] Diese Zahl ist eine konservative Schätzung, in der Tat sind es sehr viel mehr. Der Duden Band 1 Rechtschreibung [21] beispielsweise hat 120000 Stichwörter mit über 500000 Einträgen; in Kapitel 1 wurde eine Million als Anzahl der Namenseinträge in einem deutschen Telefonbuch genannt.

Male in einer Stichprobe zu beobachten sind. Aus diesem Grund werden üblicherweise die verschiedenen Historien zu Klassen zusammengefasst.

Eine Methode, die sich in der Spracherkennung bisher bewährt hat, fasst alle Historien zusammen, die sich in den letzten Wörtern nicht unterscheiden. Beispielsweise würden die Historien "es ist klar" und "mir ist klar", die beide vor dem Wort "dass" stehen könnten, bei $N = 3$ in eine einzige Klasse fallen. Das bedeutet, dass die Wahrscheinlichkeit, die für das Wort "dass" geschätzt wird, für die beiden Historien gleich ist, also

$p(\text{dass}|\text{es ist klar}) = p(\text{dass}|\text{mir ist klar})$.

Sprachmodelle mit $N = 3$ werden als *Trigramm-Sprachmodelle*, mit $N = 2$ als *Bigramm-Sprachmodelle* und mit $N = 1$ als *Monogramm-Sprachmodelle* bezeichnet.

Wenn die Wahrscheinlichkeit eines Wortes w_i nur von zwei Vorgängerwörtern abhängt (Trigamm-LM), wird die Wahrscheinlichkeit auch als $p(w_i|w_{i-2}, w_{i-1})$ geschrieben.

Die Wahrscheinlichkeit $p(w_i|w_{i-2}, w_{i-1})$ wird auf üblicherweise umfangreichen Textkorpora bestimmt, indem die Häufigkeiten von 3-Grammen und 2-Grammen bestimmt wird.

$$p(w_i|w_{i-2}w_{i-1}) = \frac{freq(w_{i-2}, w_{i-1}, w_i)}{freq(w_{i-2}, w_{i-1})} \tag{2.4}$$

Der Korpus, auf dem diese Wahrscheinlichkeiten bestimmt wird, heißt auch *Trainingskorpus*.

Die Festlegung der Wahrscheinlichkeiten nach Formel 2.4 hat die Schwäche, dass für das Training die zur Verfügung stehenden Textkorpora zu klein sind, und eine nicht beobachtete Wortfolge die Wahrscheinlichkeit 0 erhalten würde, also als *unmöglich* angesehen wird. Da aber jede auch noch so unsinnige Wortfolge theoretisch gesprochen werden kann, und sei es nur, um zu zeigen, dass es möglich ist, ist das Auffinden von unmöglichen Ereignissen bei einem LVCSR-System nicht plausibel. Aus diesem Grund wurden verschiedene Verfahren entwickelt, die auch für Ereignisse eine positive Wahrscheinlichkeit bestimmen, die nicht auf dem Trainingskorpus beobachtet wurden. Diese Verfahren glätten die Wahrscheinlichkeitsverteilung (Smoothing). Hierzu muss von den beobachteten Ereignissen ein Teil der Wahrscheinlichkeitsmasse reserviert werden (discounting), beispielsweise durch Zurückfallen auf generellere Modelle (Back-off), die eine kürzere Historie besitzen.

Eine Methode einen Teil der verfügbaren Wahrscheinlichkeitsmasse für ungesehene Ereignisse zu reservieren, ist das *absolute discounting*. Dabei berechnet sich die Wahrscheinlichkeit für gesehene N-Gramme

$$p(w_i|w_{i-N+1}, \ldots, w_{i-1}) = \frac{freq(w_{i-N+1}, \ldots, w_{i-1}, w_i) - D}{freq(w_{i-N+1}, \ldots, w_i)} \tag{2.5}$$

Die Konstante D wird dabei von der Häufigkeit der aufgetretenen N-Gramme subtrahiert und steht damit für die Modellierung der Wörter zur Verfügung, die nicht nach der Historie $w_{i-N+1}, \ldots, w_i$ beobachtet wurden. Ein üblicher Wert für D ist 0,5, da es mehr oder weniger zufällig ist, ob ein N-Gramm in einem Korpus M-mal aufgetreten ist, oder es auch $M-1$-mal hätte auftreten können.

Eine andere Methode, die tatsächliche Häufigkeit für ein gesehenes Ereignis und dabei auch die ungesehenen Ereignisse mit zu schätzen, wurde von Good in [37] veröffentlicht und wird als *Good-Turing-Reestimation* bezeichnet. Dabei wird keine Konstante abgezogen sondern ein von der Häufigkeit des Auftretens abhängiger Wert.

Steht nun eine Wahrscheinlichkeitsmasse (α) zur Verfügung, die für die Modellierung ungesehener Ereignisse reserviert ist, muss noch ein Verfahren angegeben werden, wie nun die Wahrscheinlichkeit für ein konkretes ungesehenes Ereignis bestimmt wird. Ein Verfahren hierfür ist das *Back-off*, bei dem gilt:

$$\hat{p}(w_i|w_{i-N+1}, \ldots, w_{i-1}) = \begin{cases} p(w_i|w_{i-N+1}, \ldots, w_{i-1}) : \text{freq}(w_{i-N+1}, \ldots, w_i) \geq 1 \\ \alpha(w_{i-N+1}, \ldots, w_{i-1}) p(w_i|w_{i-N+2}, \ldots, w_{i-1}) : \text{sonst} \end{cases} \tag{2.6}$$

Wird eine Wahrscheinlichkeit für ein auf dem Trainingskorpus ungesehenes Ereignis benötigt, wird eine Statistik über einer verkürzten Historie verwendet und mit der zuvor reservierten Wahrscheinlichkeismasse multipliziert.

Dabei hat sich gezeigt, dass für $p(w_i|w_{i-N+2}, ..., w_{i-1})$ nicht die Wahrscheinlichkeiten verwendet werden sollten, die sich durch die Häufigkeitsverteilung auf dem Trainingskorpus ergeben [54]. Ein Grund hierfür ist, wenn beispielsweise bei einem Bigramm-Sprachmodell w_1 und w_2 in vielen verschiedenen Kontexten beide sehr häufig aufgetreten sind, aber *nicht* gemeinsam beobachtet wurden, müsste dann die Wahrscheinlichkeit für ein gemeinsames Auftreten klein sein. $p(w_2)\alpha(w_2)$ würde aber wegen der hohen Wahrscheinlichkeit für $p(w_2)$ und der großen Back-off-Wahrscheinlichkeit $\alpha(w_2)$ insgesamt einen hohen Wert ergeben. Ein Verfahren, das diesen Effekt reduziert, ist das Glätten mit einer von Kneser und Ney [54] entwickelten Methode, bei dem anstelle der Häufigkeiten auf dem Trainingskorpus die Anzahl der verschiedenen Kontexte verwendet wird, in denen das Wort auftritt. Die Wahrscheinlichkeit für den Fall, dass ein ungesehenes Ereignis auftritt, ist dann $\alpha(w_1)\hat{p}(w_2)$. $\hat{p}(w_2)$ bezeichnet das Verhältnis der verschiedenen Kontexte, in denen w_2 aufgetreten ist zu der Gesamtzahl der aufgetretenen Kontexte.

Für die vorliegende Arbeit ist es relevant, was geschieht, wenn beispielsweise in einem Trigramm-LM ein Wort im Vokabular enthalten ist, dass *nicht* im Trainingskorpus auftritt und dementsprechend nur über die Back-off-Wahrscheinlichkeit modelliert werden kann.

Sei w_n das Wort und es ist $p(w_n|w_{n-2}w_{n-1})$ zu berechnen, dann ist

$$
\begin{aligned}
p(w_n|w_{n-2}w_{n-1}) &= p(w_n|w_{n-1}) * \alpha_2(w_{n-2}) \\
&= p(w_n)\alpha_1(w_{n-1}) * \alpha_2(w_{n-1}w_{n-2}) \\
&= \alpha_0 * \alpha_1(w_{n-1}) * \alpha_2(w_{n-1}w_{n-2})
\end{aligned}
$$

Das heißt, die Wahrscheinlichkeit für das Wort w_n wird allein von den Back-off-Wahrscheinlichkeiten der Historie und dem Wert von α_0 bestimmt.

2.7.2 Perplexität

Außer der Wortfehlerrate für ein Spracherkennungssystem, die das anerkannte Evaluierungsmaß für die Qualität eine Spracherkenners ist, gibt es noch die *Testset-Perplexität* (PP), die als Zwischenmaß für die Bewertung von statistischen Sprachmodellen verwendet wird.

$$
\begin{aligned}
PP &= 2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 p(w_i|w_1 \ldots w_{i-1})} \\
&= 2^{-\sum_{i=1}^{N}\log_2 \sqrt[N]{p(w_i|w_1 \ldots w_{i-1})}} \\
&= \prod_{i=1}^{N} \frac{1}{\sqrt[N]{p(w_i|w_1 \ldots w_{i-1})}} \\
&= \frac{1}{\sqrt[N]{\prod_{i=1}^{N} p(w_i|w_1 \ldots w_{i-1})}}
\end{aligned}
\tag{2.7}
$$

Mit ihr wird die mittlere Vorhersagegenauigkeit eines Modells gemessen. Dabei wird, wie aus Formel 2.7 zu ersehen ist, der Kehrwert des geometrischen Mittels über die auf einem Trainingskorpus bestimmten Wahrscheinlichkeiten $p(w_i|w_1, \ldots w_{i-1})$ auf einem neuen ungesehenen Textkorpus bestimmt. Die Perplexität kann dabei auch als mittlerer Verzweigungsgrad betrachtet werden, so dass nach jedem Wort im Mittel zwischen PP Wörtern ausgewählt werden muss.

Auf dem Testkorpus können natürlich auch Ereignisse auftreten, die auf dem Trainingskorpus nicht beobachtet wurden. Würde also kein Back-off oder Glätten durchgeführt und würde diesen Ereignissen eine Wahrscheinlichkeit von 0 zugeordnet, würde die Perplexität sofort unendlich groß sein. Um die Perplexität berechnen zu können, muss also auch den auf dem Trainingskorpus nicht gesehenen Ereignissen eine positive Wahrscheinlichkeit zugeordnet werden, da im voraus ja nicht bekannt sein kann, welche ungesehenen Ereignisse nie auftreten werden.

Der Wert, den die Perplexität annimmt, hängt im wesentlichen vom verwendeten Sprachmodell, das die Wahrscheinlichkeiten $p(w_i|w_1, \ldots w_{i-1})$ zur Verfügung stellt, und dem Testkorpus ab, auf den diese Wahrscheinlichkeiten angewendet werden.

Hält man den Testkorpus konstant, lassen sich anhand der Perplexität zwei unterschiedliche Sprachmodelle vergleichen. Das Sprachmodell mit der geringeren Perplexität passt besser zu dem Textkorpus, und es besteht die Hoffnung, dass dieses Modell dann auch zu einer geringeren Wortfehlerrate führt. Dies muss aber nicht so sein. Zum einen können die Verbesserungen des Modells an Stellen erfolgt sein, die für den Spracherkenner als Ganzes keine Schwierigkeit darstellen, zum anderen besteht die Gefahr, dass zwischen dem Testkorpus und den Daten, mit denen der Spracherkenner ausgewertet wird, ein Mismatch besteht, so dass in die falsche Richtung optimiert wird. Ist letzteres nicht der Fall, bedeutet eine Reduktion der Perplexität, dass im Mittel das Modell bessere Vorhersagen macht. Selbst wenn dies keine Auswirkungen auf die Fehlerrate haben sollte, so kann es doch sein, dass der Spracherkenner beispielsweise einen geringeren Suchaufwand erbringen muss, um zur gleichen Hypothese zu gelangen, dass also das Ergebnis schneller gefunden wird, wenn der Suchraum beschnitten wird.

2.7.3 Trigger-Sprachmodelle

Bei einem Trigger-Sprachmodell wird die Wahrscheinlichkeit für ein vorhergesagtes Wort in Abhängigkeit davon geschätzt, ob in der Historie h bestimmte Triggerwörter t_i aufgetreten sind. Hierzu werden Indikatorfunktionen $b_i(h)$ eingeführt, die die Werte 0 oder 1 annehmen, wenn sich das Triggerwort t_i in der verwendeten Historie befindet. Die verwendete Wahrscheinlichkeit ist dann $p(w_n|w_{n-1}, w_{n-2}, b_1(h), \ldots, b_m(h))$.

Bei der Erzeugung eines Trigger-Sprachmodells müssen zunächst Triggerwörter bestimmt werden und es muss bestimmt werden, welche Menge an Wörtern sie beeinflussen. Hierzu wird ermittelt, wie stark das Auftreten oder das Nicht-Auftreten eines Triggerwortes die Häufigkeitsverteilung eines bestimmten Wortes im Mittel beeinflusst. Wörter, die einen großen Einfluss besitzen, eignen sich als Triggerwörter.

Die Wahrscheinlichkeit eines Wortes w_n ist dann abhängig davon, ob es von bestimmten Triggerwort-Kombinationen beeinflusst wird und kann für diese Wörter eine von den Indikatorfunktionen abhängige Wahrscheinlichkeit bestimmen.

So könnte beispielsweise das Wort "PRESIDENT" ein Triggerwort für die Wörter "CLINTON" und "ELECTION" sein. Die Wahrscheinlichkeit für die Wörter "CLINTON" und "ELECTION" wären dann abhängig von der Indikatorfunktion

$b_{PRESIDENT}(h)$ also $p(w_n = CLINTON|w_{n-2}, w_{n-1}, b_{PRESIDENT}(h))$.

Triggerwort-Sprachmodelle nutzen also aus, dass bestimmte Wörter mit anderen Wörtern in enger Beziehung stehen und darüber die a priori Wahrscheinlichkeit für eine Äußerung beeinflussen. Sie stellen eine Methode dar, die Einschränkungen der kurzen

Historie abzuschwächen, die sich von N-Gramm-Sprachmodellen ergibt.

2.7.4 Cache-Sprachmodelle

Bei einem Cache-Sprachmodell wird versucht, die benötigten Kontexte und deren Verteilung aufgrund der Historie zu schätzen. Dazu kann aus den erkannten dynamischen Hypothesen eines Spracherkenners ein dynamischer Korpus geformt werden, auf dem ein neues Sprachmodell bestimmt wird, das dann mit einem normal auf einem großen Trainingskorpus berechneten N-Gramm-Sprachmodell interpoliert wird.

Cache-Sprachmodelle besitzen den Vorteil, dass sie die Möglichkeit besitzen, die Wahrscheinlichkeiten für auf dem Trainingskorpus ungesehene Wortfolgen anzupassen. Dies erlaubt auch das Schätzen oder Anpassen von Wahrscheinlichkeiten für gelernte Wörter. Dies ist beispielsweise in Diktieranwendungen gut möglich, bei denen fehlerhafte Erkennungen von einem Benutzer per Tastatur korrigiert werden müssen. In Spracherkennungssystemen, die keine Korrektur der Eingabe erlauben, kann es aber zu unerwünschten Effekten kommen, in der einmal gelernte Fehlerkennungen sich verstärken. So konnte von Clarkson und Robinson [12] keine signifikante Verbesserung durch ein Cache-Sprachmodell beobachtet werden, wenn dieses in einer überwachten Form angewendet wurde und keine signifikante Verschlechterung, wenn diese unüberwacht stattfand. Von Woodland [81] wurde auf der BN-Domäne eine nicht signifikante Verbesserung beobachtet, wenn ein unüberwachtes Cache-Sprachmodell eingesetzt wurde.

Ein Cache-Sprachmodell ist nicht dafür gedacht, neue Wörter zu finden sondern besitzt die Möglichkeit, die Wahrscheinlichkeitsverteilungen neu gelernter Wörter zu lernen, wenn deren Erkennung genau genug funktioniert.

2.7.5 Sprachmodelle mit Klassen

Die Verwendung von Klassen ist eine Möglichkeit, mit der Schwierigkeit umzugehen, nicht genügend Daten für die Schätzung der Sprachmodellparameter zur Verfügung zu haben. Teilt man beispielsweise ein Vokabular der Größe 40000 in 10000 Klassen ein, ist bei einem 3-Gramm-Sprachmodell die maximale Zahl an Parametern ungefähr 40000^3, dagegen bei dem Klassensystem ungefähr $30000 + 10000^3$, also ungefähr $\frac{1}{64} \approx 2\%$ der Parameter.

Der richtige Einsatz von Klassen führt zu einer Verbesserung der Modellierung, da gerade seltene Wörter von der Klassenbildung profitieren können. Durch das gemeinsame Modellieren von Wörtern w_1 und w_2 in einer Klasse wird erreicht, dass w_1 die Kontexte von w_2 kennt und w_2 die von w_1 und dementsprechend besser in diesen Kontexten erkannt werden kann.

Es gibt verschiedene Verfahren, wie Wörter in Klassen eingeteilt werden können. Die naheliegendste ist, die Wörter manuell von Experten in Klassen einteilen zu lassen, wie es beispielsweise bei semantischen Klassensystemen gemacht wird. Es ist eine sehr aufwendige

Methode, bei der vorausgesetzt wird, dass auch ein geeignetes Klassensystem vorhanden ist. Der Vorteil liegt aber darin, dass diese Art von Einteilung mit besonders wenig Daten möglich ist und die Klassen einen "Sinn" ergeben.

Es gibt aber auch datengetriebene Verfahren, wie das Clustern von Kneser und Ney, das in Kapitel 6 ausführlich beschrieben wird und sich besonders gut für die Bestimmung von Klassen für die Modellierung in Sprachmodellen eignet.

Bei einem klassenbasierten 3-Gramm-Sprachmodell berechnet sich die Wahrscheinlichkeit

$$\begin{aligned} p(w_n|w_{n-2}, w_{n-3}) &= p(w_n|\phi(w_n))p(\phi(w_n)|\phi(w_{n-2})\phi(w_{n-3})) \\ &= p(w_n|g_n)p(g_n|g_{n-2}g_{n-1}) \end{aligned}$$

Hierbei wird eine disjunktive Klasseneinteilung vorausgesetzt, wie sie in dieser Arbeit verwendet wird. Die Wahrscheinlichkeit $p(w_n|\phi(w_1))$ ist die klassenbedingte Wahrscheinlichkeit für das Wort w_n in der Klasse $\phi(w_n) = g_n$. Je nach Anwendung wird diese klassenbedingte Wahrscheinlichkeit auf einem Trainingskorpus bestimmt oder aufgrund bestimmter Modellannahmen festgelegt. Beispielsweise kann so bei einer semantischen Klasse für "Straßennamen" jeder Straße die gleiche Wahrscheinlichkeit zugeordnet werden, wenn keine weiteren Informationen vorliegen. So können dann auch Straßennamen mit einer Wahrscheinlichkeit versehen werden, die *nicht* im Trainingskorpus enthalten waren.

Durch das Anwenden der Abbildung ϕ auf einem Korpus K kann mit den Standardverfahren zur Erzeugung von statistischen Sprachmodellen ein klassenbasiertes Sprachmodell erzeugt werden. Die Bestimmung der klassenbedingten Wahrscheinlichkeit erfolgt dann durch

$$p(w|\phi(w)) = \frac{freq_K(w)}{freq_{\phi(K)}(\phi(w))} \tag{2.8}$$

2.8 Training der akustischen Modelle

Das akustische Training dient dazu, die akustischen HMM-Modelle (Wörter, Phoneme, Subphone) so anzupassen, dass sie einerseits die vorhandene (akustische) Trainingsmenge von Äußerungen gut repräsentieren und andererseits auf ungesehene Äußerungen generalisieren können. Für jedes Modell, das trainiert werden soll, muss eine ausreichende Menge an Trainingsbeispielen vorhanden sein. Üblicherweise ist der erste Schritt das Segmentieren der Trainingsdaten und die Zuordnung, welche Modelle damit initialisiert werden sollen. Dieser Arbeitsschritt wird *labeln* genannt. Für das *Labeln* gibt es verschiedene Verfahren. Die einfachste Methode ist die Annahme, dass die Zuordnung der Eingabe linear

mit der Zahl der Zustände zusammenhängt, die sich aus der Transkription ergeben. Wesentlich besser ist es, die Daten von einem Menschen *labeln* zu lassen. Hierbei ergibt sich aber der Nachteil, dass, je feiner die Segmentierung sein soll, beispielsweise Wörter oder Phoneme, nur gut ausgebildete Experten die Segmentierung durchführen können. Zusätzlich ist bei einer feineren Segmentierung der Arbeitsaufwand größer, was dazu führt, dass verhältnismäßig wenig Daten bearbeitet werden können. Ein anderes Problem ist, dass die Segmentierung durch einen Menschen für eine *maschinelle Klassifikation* nicht optimal zu sein braucht. Wegen des großen Aufwands ist es sehr teuer, die Daten von Menschen labeln zu lassen. Ist bereits ein (einfaches) Spracherkennungssystem vorhanden, kann dieses verwendet werden, um initiale Labels zu erzeugen. Hierzu wird lediglich eine Transkription benötigt. Da wesentlich mehr Daten verarbeitet werden können und somit mehr Daten für die Initialisierung der Modelle zur Verfügung stehen, ist dies ein bedeutender Vorteil.

Nachdem nun die vorverarbeiteten Sprachdaten segmentiert vorliegen, wird eine *Ballungsanalyse* der Daten für jedes Modell durchgeführt, und für die gefundenen Ballungen eine repräsentative Darstellung erzeugt. Dies ist in den meisten Fällen eine multimodale Gauß-Verteilung. Das Ergebnis ist ein initiales *Kodebuch* und eine initiale *Gewichtung*. In einem kontinuierlichen oder semikontinuierlichen HMM ist dies eine parametrische Darstellung aus Mittelwert und Varianz der Ballung (*Kodebuch*) und der a priori Wahrscheinlichkeit der Ballungen in einem gegebenen Zustand des Modells (*Gewichtung*). Beides zusammen nennen wir das *akustische Modell* (λ) des Spracherkenners. In einem semikontinuierlichen HMM besitzen verschiedene Modellzustände gemeinsame Einträge im *Kodebuch*, bei denen sich aber die a priori Verteilungen unterscheiden.

Besitzen nun alle Modelle ihre initialen *Kodebücher* und *Gewichtungen*, können die Modelle weiter angepasst werden. Dazu existieren verschiedene Trainingsalgorithmen. Einer ist der *Baum–Welch-Algorithmus* [58][70], der das Optimierungsproblem aus Abschnitt 2.4 löst. Dieser Algorithmus ändert die Modellparameter so ab, dass die modellbedingte Wahrscheinlichkeit $p(O|\lambda)$ aller im Training verwendeten Daten maximiert wird. Es ist nicht garantiert, dass ein globales Maximum gefunden wird. Der Rechenaufwand des *Baum-Welch-Algorithmus* ist im Vergleich zum *Viterbi-Training* groß, da als Ergebnis eine Wahrscheinlichkeit berechnet wird, die die Summe der Wahrscheinlichkeiten aller Pfade ist. Beim Viterbi-Training hingegen wird nur die Wahrscheinlichkeit des besten Pfades berechnet. Dies geschieht, indem beim Zusammentreffen mehrerer Pfade nicht die Summe sondern das Maximum gebildet wird. Wird der wahrscheinlichste Pfad zurückverfolgt, ergibt sich eine Zuordnung einzelner Sprachdatenvektoren zu bestimmten Zuständen. Das Ergebnis ist eine Segmentierung der Daten, mit der erneut eine Ballungsanalyse durchgeführt werden kann.

Mit den gefundenen Verteilungen (Kodebücher und Gewichtung) wird im Dekoder des Spracherkenners die klassenbedingte Wahrscheinlichkeit einer Observation zu einem bestimmten Modellzustand berechnet. Die resultierende Wahrscheinlichkeit kann als Maß für die akustische Ähnlichkeit der Observation zu einem Modell angesehen werden.

2.9 Evaluationskriterien

Um festzustellen, ob ein bestimmtes Verfahren zur Verbesserung eines Spracherkenners führt, bedarf es Kriterien (*Evaluationskriterien*), die es ermöglichen, die Qualität eines Spracherkenners zu beurteilen.

2.9.1 Der Align-Algorithmus

Der Align-Algorithmus ist in der Spracherkennung von großer Bedeutung, da mit ihm die *Levensthein-Distanz* auch bekannt als *String Editier-Distanz*, berechnet werden kann, die ein anerkanntes Maß zur Beurteilung der Qualität eines Spracherkenners ist. Mit dem Align-Algorithmus wird eine Zuordnung der Wörter zwischen der tatsächlich gesagten Äußerung (Referenzsatz) und dem vom Spracherkenner erzeugten Hypothesensatz durchgeführt. Die Editier-Distanz ist dabei die *minimale* Anzahl der Operationen *Wort-Einfügen*, *Wort-Löschen* und *Wort-Ersetzen*, die nötig ist, um den Referenzsatz in den Hypothesensatz zu überführen.

Anhand der Operationen ist es möglich, Erkennungsfehler weiter zu unterscheiden.[7] Ein Wort, das in den Referenzsatz eingefügt wird, gehört zur Klasse der *Einfügefehler*, eines, das aus dem Referenzsatz gelöscht wird, zur Klasse der *Löschfehler*. Einfügefehler treten zwischen Wörtern des Referenzsatzes, Löschfehler zwischen den Wörtern der Hypothese auf. Ein Wortpaar aus Referenz- und Hypothesensatz, das durch die Operation Wort-Ersetzen gebildet wird, gehört dementsprechend zur Klasse der *Ersetzungsfehler*.

Sei beispielsweise der Referenzsatz ≪Das hier ist der erste Satz≫ und der Hypothesensatz ≪Das ist aber der zweite Satz≫, so gibt es mehrere Möglichkeiten wie die minimale Anzahl von drei Operationen erreicht werden kann (vgl. Tabelle 2.1 und 2.2).

Referenz	Hypothese	Fehlertyp
Das	Das	korrekt
hier	ist	Ersetzung
ist	aber	Ersetzung
der	der	korrekt
erste	zweite	Ersetzung
Satz	Satz	korrekt

Tabelle 2.1: Fehlerzuordnung 1

Es ist dabei auch von großer Bedeutung, wie die Fehler zugeordnet werden. Im angeführten Beispiel werden bei der Zuordnung nach Tabelle 2.2 zwei Wörter der Hypothese als falsch angesehen, nach der Zuordnung mit Tabelle 2.1 drei. Welche der Zuordnungen den tatsächlichen (zeitlichen) Gegebenheiten entspricht, kann a priori nicht entschieden

[7] Aufgrund der Minimalität können sich die Operationen nicht aufheben.

Referenz	Hypothese	Fehlertyp
Das	Das	korrekt
hier	*	Löschung
ist	ist	korrekt
*	aber	Einfügung
der	der	korrekt
erste	zweite	Ersetzung
Satz	Satz	korrekt

Tabelle 2.2: Fehlerzuordnung 2

werden. Eine Zuordnung jedoch, die möglichst wenig fehlerhafte Wörter im Hypothesesatz erzeugt, scheint im allgemeinen die plausiblere zu sein.

Der Align-Algorithmus in obigem Beispiel würde die Lösung aus Tabelle 2.2 erzeugen. Die Operationen Wort-Einfügen und Wort-Löschen werden bevorzugt, da beide mit geringeren Kosten bewertet sind als die Operation Wort-Ersetzen (vgl. Tabelle 2.3).

Operation	Kosten
Wort-Einfügen	75
Wort-Löschen	75
Wort-Ersetzen	100

Tabelle 2.3: Kosten der Operationen bei Editier-Distanz

Die Arbeitsweise des Align-Algorithmus ist in Abbildung 2.7 dargestellt, sie entspricht etwa dem Viterbi-Algorithmus. Dabei muss ein Pfad mit minimalen Kosten in der dargestellten Matrix berechnet werden. Treffen mehrere Pfade zusammen wird nur der kostengünstigste weiter verfolgt. Wird der Pfad, der die beiden Satzenden erreicht (Matrix oben, rechts), zurückverfolgt, können die benötigten Operationen (gestrichelten Pfeile) bestimmt werden. In der Abbildung ist nur der günstigste Pfad eingezeichnet.

Die *Wortfehlerrate* (WER) eines Spracherkenners ist definiert als

$$\text{WER} = \frac{\#Ersetzungen + \#Einfuegungen + \#Loeschungen}{\#Worte\_im\_Referenzsatz} \tag{2.9}$$

Auf der GSST-Domäne werden *vor* dem Durchführen der Zuordnung beispielsweise alle "Müllworte" (#noise#, #atmen#) aus der Hypothese und der Transkription entfernt. Dies ist wichtig, da diese häufig nicht in der Transkription enthalten sind und somit die Zahl der Fehler steigt.

Von der NIST[8] wird das Evaluierungswerkzeug *sclite* zur Verfügung gestellt, das wei-

[8]National Institute of Standards and Technology

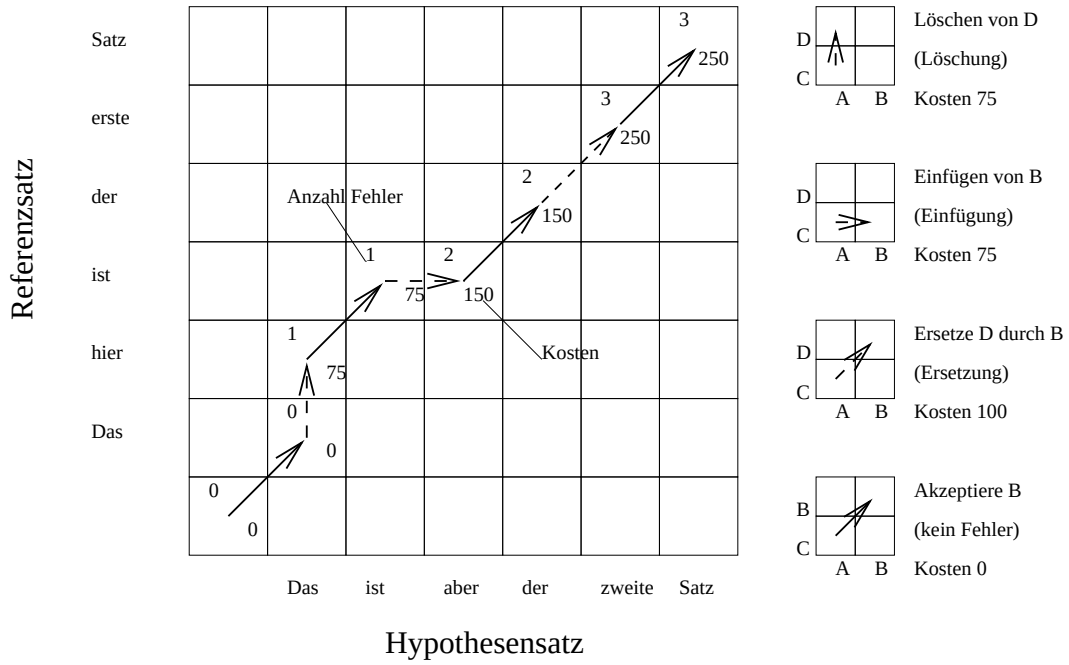

Abbildung 2.7: Arbeitsweise des Align-Algorithmus

tere Fähigkeiten besitzt und auf der HUB-4-Domäne das offizielle Evaluierungswerkzeug darstellt. Es erlaubt Alternativen in den Referenzsätzen und das Durchführen einer Normalisierung der Referenzen und Hypothesen. So werden beispielsweise Komposita wie "MAILBAG" in "MAIL" und "BAG" zerlegt [57].

2.9.2 Präzision und Recall

Für eine genauere Aussage über die Qualität, mit der ein Spracherkenner unbekannte Wörter findet, ist die Wortfehlerrate allein nicht geeignet.

Wird von einem Spracherkenner eine Menge an Äußerungen untersucht, die eine bestimmte Anzahl an OOV-Wörtern enthält und sind in den Hypothesen mögliche Positionen von OOV-Wörtern dadurch markiert, dass an deren Stelle das Symbol <OOV> ausgegeben wird, so lässt sich ermitteln, wie viele der enthaltenen unbekannten Wörter gefunden wurden und wie viele der OOV-Hypothesen korrekt sind.

Ob ein unbekanntes Wort korrekt gefunden wurde, wird dabei durch den Align-Algorithmus bestimmt, in dem die OOV-Wörter in den Referenzen durch das Symbol <OOV> ersetzt werden. Wird ein <OOV>-Symbol aus der Hypothese einem <OOV>-Symbol aus der Referenz zugeordnet, wird dies als korrekte OOV-Hypothese gezählt, ansonsten nicht.

$\#\text{OOV}_{\text{Kor}}$	Anzahl korrekt gefundener OOV-Symbole
$\#\text{OOV}_{\text{Ref}}$	Anzahl OOV-Symbole in den Referenzen
$\#\text{OOV}_{\text{Hyp}}$	Anzahl OOV-Symbole in den Hypothesen

$$PRC_{\text{OOV}} = \frac{\#\text{OOV}_{\text{Kor}}}{\#\text{OOV}_{\text{Ref}}} \tag{2.10}$$

$$REC_{\text{OOV}} = \frac{\#\text{OOV}_{\text{Kor}}}{\#\text{OOV}_{\text{Hyp}}} \tag{2.11}$$

Mit PRC_{OOV} wird die Präzision oder auch Zuverlässigkeit einer OOV-Hypothese bezeichnet. Die Ausschöpfung, wie viele der OOV-Wörter gefunden werden, wird als *Recall* bezeichnet und mit REC_{OOV} abgekürzt. Da in dieser Arbeit nur OOV-Wörter detektiert werden, wird das Subscript "OOV" zur Vereinfachung weggelassen.

Präzision und Recall haben einen Wertebereich von 0% bis 100%. Das Optimum ist ein System, das alle unbekannten Wörter findet ($REC = 100\%$) und dabei keine fehlerhaften Hypothesen erzeugt ($PRC = 100\%$). Präzision und Recall sind nicht unabhängig von einander, weshalb in der Regel beide Werte angegeben werden müssen. So ist es beispielsweise möglich, alle unbekannten Wörter zu finden, indem einfach alle Wörter in der Hypothese des Spracherkenners durch das Symbol <OOV> ersetzt werden, was aber dazu führt, dass die Präzision sehr schlecht ausfällt.

Da zwei Werte die Qualität eines Systems beschreiben, ist der Vergleich zweier Systeme miteinander schwierig. Einfach ist eine Bewertung, wenn System A im Vergleich zu System B eine höhere Präzision bei einem höheren Recall als System B besitzt.

Häufig lässt sich der Recall beziehungsweise die Präzision durch verschiedene Parameter beeinflussen. Dann kann entweder die Präzision und der Recall über diesem Parameter aufgetragen werden, oder es wird die Präzision über dem Recall aufgetragen. Ist die Präzision über dem Recall aufgetragen, wird dies auch als *Receiver-Operator-Characteristic-Kurve* (ROC-Kurve) bezeichnet.

2.9.3 Klassenfehlerrate

Die *Klassenfehlerrate* (Class Error Rate (CER)) ergibt sich, wenn der Align-Algorithmus wie bei der Bestimmung der Präzision und des Recalls angewendet wird, also nachdem in den Referenzen die Wörter, die nicht im Vokabular des Spracherkenners enthalten sind, auf das Symbol <OOV> abgebildet wurden.

Besitzt ein Spracherkenner *nicht* die Möglichkeit, OOV-Wörter zu detektieren, sind die Wortfehlerrate WER und die Klassenfehlerrate CER identisch, da jedes OOV-Wort

automatisch zu einem Fehler führt. Verfügt der Spracherkenner über die Fähigkeit, OOV-Wörter zu detektieren, reduziert jedes korrekt gefundene OOV-Wort die Fehlerrate.

$$\begin{aligned} \text{WER} &= \frac{\#Ersetzungen + \#Einfuegungen + \#Loeschungen + \#OOV_{KOR}}{\#Worte\_im\_Referenzsatz} \\ &= CER + \frac{\#OOV_{KOR}}{\#Worte\_im\_Referenzsatz} * \frac{\#OOV_{REF}}{\#OOV_{REF}} \\ &= CER + \text{OOV-Rate} * REC \end{aligned} \tag{2.12}$$

Wie aus Gleichung 2.12 zu ersehen ist, muss auf die CER nur das Produkt aus OOV-Rate und Recall addiert werden, um die Wortfehlerrate zu erhalten.

Der Wert für die Klassenfehlerrate ist kleiner als der Wert für die Wortfehlerrate, wenn durch eine Detektion von OOV-Wörtern mehr OOV-Wörter korrekt gefunden werden als zusätzliche Ersetzungen, Einfügungen und Löschungen produziert werden. In der Regel wird in dieser Arbeit die Klassenfehlerrate CER angegeben.

2.10 Zipf'sche Verteilung

Untersucht man Textkorpora, so kann man für jedes Wort seine Häufigkeit (*Frequenz*) bestimmen und die Wörter nach ihrer Frequenz sortieren. Die sich daraus ergebende Liste kann dann numeriert werden, indem dem häufigsten Wort der Rang 1 zugeordnet wird, dem zweit häufigsten Wort der Rang 2, und so weiter.

Von Zipf [84] wurden verschiedene Texte[9] untersucht und die Frequenz f über dem Rang r dargestellt. Dabei wurde von ihm festgestellt, dass für die häufigen Wörter das Produkt $r \times f = c$ annähernd einer Konstanten entspricht, was sich bei Verwendung einer doppelt logarithmischen Skala für die Achsen darin zeigt, dass die Datenpunkte entlang einer gedachten Linie liegen. Eine Gerade $a + (-b)\log(r) = \log(f)$, die die Datenpunkte repräsentiert, kann durch die Methode der kleinsten Fehlerquadrate als Regressionsgerade [2] bestimmt werden und beschreibt damit die Beziehung $e^a * r^{-b} = f$.

Diese Beziehung zwischen dem Rang und der Frequenz wird als *Zipf'sche Verteilung* bezeichnet und beschreibt die häufigen Wörter eines Textkorpus. Für den SZ-Korpus, die Transkripte von zwei Talkshow- und drei Tagesschau-Sendungen ist in Abbildung 2.8 die Häufigkeit der Wörter über dem Rang aufgetragen.

Für diese Arbeit wichtiger ist aber eine weitere von Zipf untersuchte Eigenschaft seltener Wörter in Texten. Hierbei hat er die Frequenz seltener Wörter betrachtet und festgestellt, dass mit abnehmender Häufigkeit die Anzahl *verschiedener* Wörter mit gleicher Frequenz zunimmt. D.h. je seltener Wörter in einem Text vorkommen, desto grösser ist die Anzahl der Wörter mit gleich (niedriger) Frequenz in einem Text. Zipf hat bestimmt, wie viele

[9] beispielsweise Ulysses von James Joyce, Zeitungstexte (Eldridge data), und vieles mehr

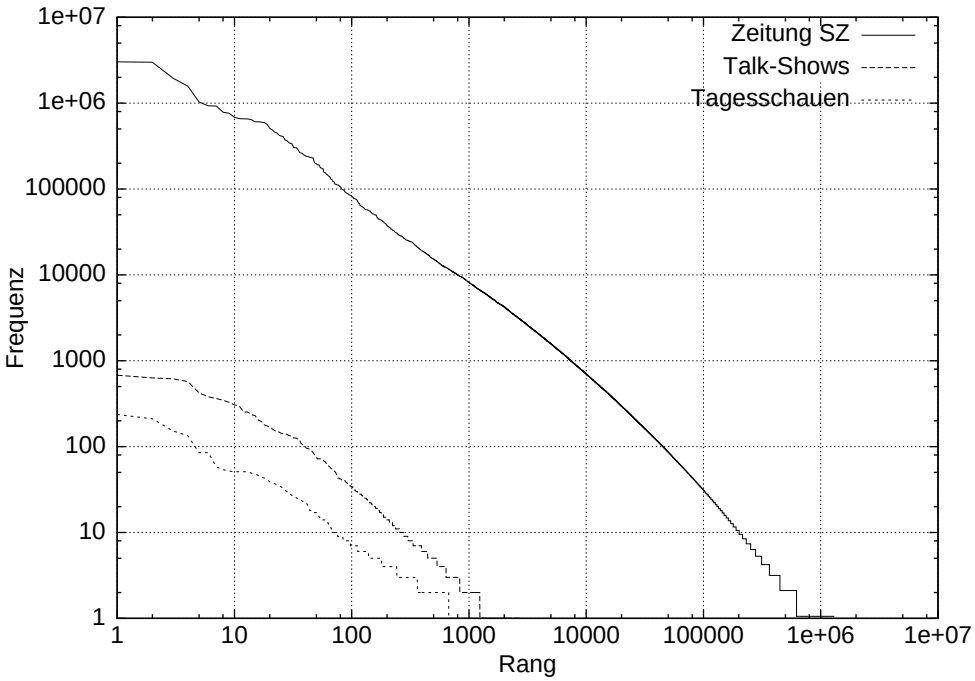

Abbildung 2.8: Häufigkeit der Wörter in verschiedenen Korpora

verschiedene Wörter mit der gleichen Häufigkeit in Textkorpora enthalten sind, also die Häufigkeit gleich häufiger Wörter. Dies hat er ebenfalls auf doppelt logarithmischem Papier dargestellt. Dabei entdeckte er wieder, dass sich die Datenpunkte auf einer gedachten Geraden befinden, die sich mit einer Regressionsgeraden $a - b\log(f) = \log(freq(f))$ beschreiben lässt. Diese Beziehung wird ebenfalls als *Zipf'sche Verteilung* bezeichnet und beschreibt die Verteilung der seltenen Wörter in einem Textkorpus. In vorliegender Arbeit ist mit *Zipf'sche Verteilung* stets diese Verteilung gemeint. Für den SZ-Korpus, die Transliterationen von zwei Talkshow- und drei Tagesschau-Sendungen ist diese Verteilung in Abbildung 2.9 dargestellt. Man erkennt, dass die Häufigkeit der seltenen Wörter in der Tat gut auf einer gedachten Geraden liegt.

Zur Veranschaulichung sind in den beiden folgenden Abbildungen die Ausgleichsgeraden für den SZ-Korpus eingezeichnet. In Abbildung 2.10 ist die Frequenz über dem Rang und in Abbildung 2.11 die Häufigkeit der verschiedenen gleich häufigen Wörter über der Frequenz dargestellt.

Zipf hat noch viele weitere interessante Untersuchungen durchgeführt, beispielsweise wie viele verschiedene Bedeutungen ein Wort in Abhängigkeit seiner Frequenz besitzt. Besonders bemerkenswert hierbei ist, dass diese ganzen Arbeiten manuell ausgeführt wurden, da Computer im heutigen Sinne damals noch nicht zur Verfügung standen.

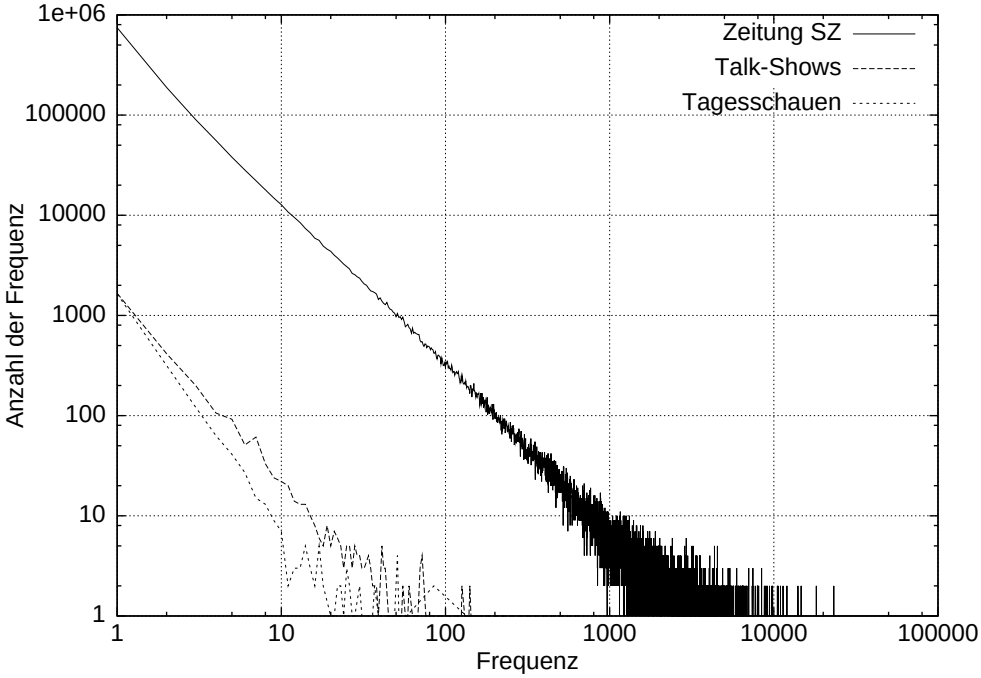

Abbildung 2.9: Häufigkeit der Frequenz der Wörter in verschiedenen Korpora

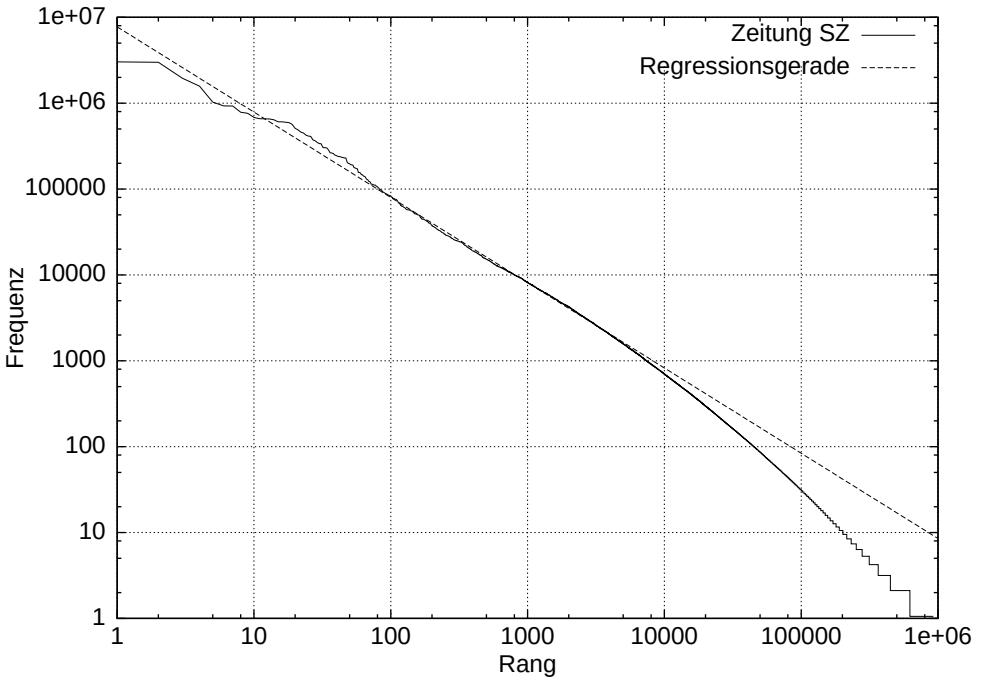

Abbildung 2.10: Ausgleichsgeraden über dem Rang des SZ-Korpus

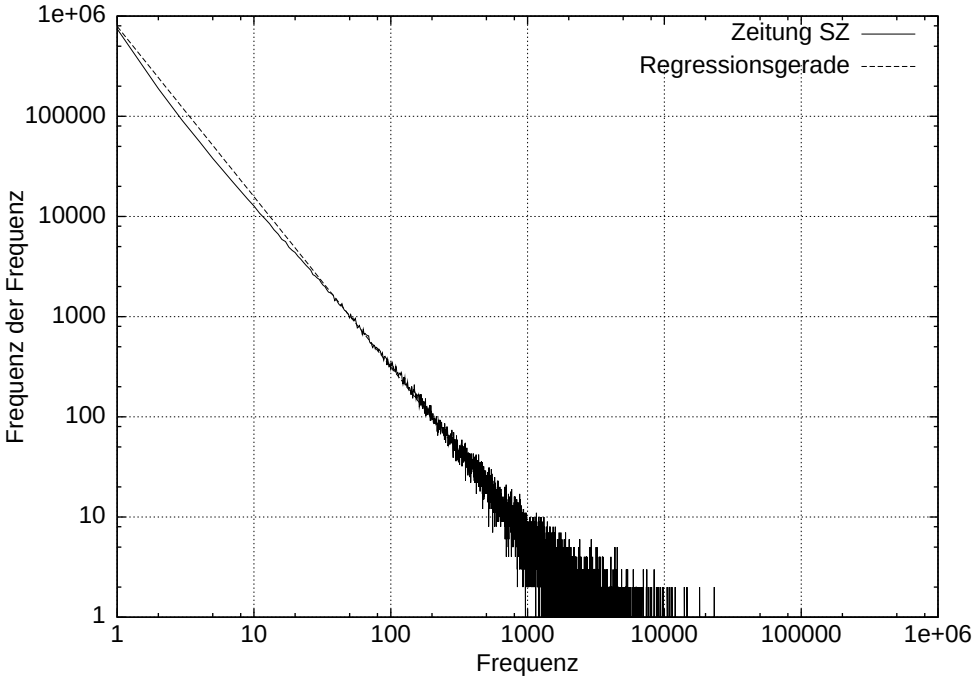

Abbildung 2.11: Ausgleichsgeraden über der Frequenz des SZ-Korpus

Kapitel 3

Verwendete Sprachdatenbasen

Die Untersuchungen in dieser Arbeit wurden auf den deutschen Verbmobil-Daten (Reiseplanung und Terminabsprache - GSST-Domäne) und englischen sowie deutschen Nachrichtensendungen (Broadcast News (BN-Domäne, HUB-4) und Tagesschau) durchgeführt. Eine ausführliche Beschreibung der verwendeten Sprachdatenbasen und der Baseline-Spracherkennungssysteme wird in diesem Kapitel gegeben.

3.1 GSST-Daten

Für das deutsche GSST-Spracherkennungssystem wurden die im Rahmen von Verbmobil gesammelten Daten für das Training der akustischen Modelle und des Trigramm-Sprachmodells benutzt. Wie bereits erwähnt, handelt es sich bei dieser Domäne um Terminabsprache und Reiseplanung, in der zwei Personen eine Konversation führen. In Phase II von Verbmobil wurden die Daten ohne Unterbrechung aufgenommen, in Phase I musste der Sprecher einen Knopf drücken um die Aufnahme zu starten und diesen während seiner Äußerung gedrückt halten. Wenn er den Knopf losgelassen hat, wurde sein Mikrophon ausgeschaltet (Push While Talking). Der Wegfall dieses Mechanismus führte dazu, dass die Daten aus Phase II sehr viel mehr spontansprachliche Effekte aufweisen als die Daten aus Phase I, in der sich die Sprecher viel mehr Zeit für die Planung ihrer Äußerung genommen haben.

Das Sprachmodell wurde aus den Transkripten der Trainingsdaten erstellt und Klassen für die semantischen Kategorien "Nachname", "Ortsname" und "Straßenname" eingesetzt. Die Einteilung der Wörter in Klassen wurde anhand von Wörterlisten durchgeführt und partiell durch manuelle Inspektion des Trainingskorpus korrigiert.

Das Baseline-System wurde auf den Testdaten Eval99.1 getestet, die für die erste Evaluierung 1999 des Verbmobil-Systems benutzt wurden. Diese Daten wurden nicht im Training verwendet. In Tabelle 3.1 sind die wesentlichen Merkmale der verwendeten Daten zusammengefasst.

Umfang	Training	Testdaten Eval99.1
Dauer	$\approx$ 63 Stunden	54 Minuten
Äußerungen	35683	538
Wörter	$\approx$ 0.68 Mio.	8577
Sprecher	889	8

Tabelle 3.1: Zusammensetzung der verwendeten GSST-Daten

Der GSST-Baseline-Spracherkenner besitzt ein Vokabular mit 10212 Wörtern und 31764 Gauss-Verteilungen in 2647 Codebüchern. Die verwendete Vorverarbeitung besteht aus 13 MFCC (Mel-Frequency Cepstrum Coefficients), die durch eine CMS (Cepstral Mean Substraction) normalisiert wird. Danach werden noch Deltas und Delta-Deltas der MFCC zu dem Merkmalsvektor bestimmt und durch eine LDA-Matrix (Linear Discriminant Ananlysis) von 39 auf 32 Dimensionen reduziert.

3.2 Top30HH-Daten

Bei den *Top30HH-Daten*, die eigens für die Untersuchungen in dieser Arbeit aufgenommen wurden, handelt es sich um vorgelesene Sätze aus der GSST-Domäne, von denen jeder Satz einen unbekannten Nachnamen enthält. Tabelle 3.2 gibt einen Überblick über die wesentlichen Merkmale dieser Daten.

Umfang	Test
Dauer	57 Minuten
Äußerungen	1200
Wörter	5760
Sprecher	8

Tabelle 3.2: Zusammensetzung der Top30HH-Daten

3.3 Broadcast News-Daten

Das englische Baseline-Spracherkennungssystem wurde auf den amerikanischen Broadcast News-Daten (BN-Daten) trainiert. Die wesentlichen Merkmale dieser Datenbasis sind in Tabelle 3.3 zusammengefasst. Das Sprachmodell wurde aus den Transkripten und sehr umfangreichen Nachrichtentexten erstellt, die von verschiedenen Nachrichtensendern beziehungsweise Zeitungen zur Verfügung gestellt wurden und vom Linguistic Data Consortium (LDC) vertrieben werden. Eine ausführliche Beschreibung dieser Datenbasis wird von Graff in [39] gegeben.

Das Spracherkennungssystem wurde entsprechend der Vorgaben für die HUB-4-Evaluationen der DARPA[1] [11] entwickelt und ist von Hua Yue in [82] und in [83] ausführlicher beschrieben.

Das Vokabular des englischen BN-Baseline-Spracherkenners enthält die 39732 häufigsten Wörter (V_{Top39734}), die, wie in Abschnitt 2.2 beschrieben, auf dem Textkorpus bestimmt wurden, mit dem auch das Sprachmodell trainiert wird. Das Vokabular V_{Top39734} wird in dieser Arbeit zur Vereinfachung auch als V_{Top40k} bezeichnet.

Der Baseline-Spracherkenner verfügt ein semi-kontinuierliches akustisches Modell. Es besteht aus 7739 Distributionen über 2143 Codebüchern, die insgesamt 104746 Gauss-Verteilungen beinhalten. Die Signalvorverarbeitung berechnet auf dem Spektrum eine VTLN (Vocal Tract Lenght Normalisation) und danach 13 MFCC (Mel-Frequency Cepstrum Coefficients), auf die eine Mittelwert- und eine Varianznormalisierung angewendet wird. Die normalisierten MFCC-Vektoren werden mit einer Kontextbreite von $\pm$ 7 Frames zusammengefasst und durch eine LDA-Transformation auf 42 Dimensionen reduziert. Abschließend wird eine Semi-Tied-Covariance-Transformation [30] verwendet. Die akustischen Modelle sind kontextabhängig geklustert, wie in [25] beschrieben, und verfügen über eine maximale Kontextbreite von $\pm$ 2 Phonemen.

Auf den DARPA-Evaluierungsdaten 1996 Testset 1 erreichte der BN-Baseline-Erkenner eine Wortfehlerrate von 19.2%. Tabelle 3.4 schlüsselt dies nochmal nach den verschiedenen Fokus-Konditionen auf.

Umfang	Training	Test
Dauer	$\approx$ 201 Stunden	84 Minuten
Äußerungen	22161	399
Wörter	1.57 Mio.	15556
Sprecher	2878	79

Tabelle 3.3: Zusammensetzung der verwendeten BN-Daten

Für die Untersuchung von *unbekannten Wörtern* wird eine große Zahl unbekannter Wörter benötigt, um signifikante Ergebnisse zu erhalten.

Erreichen kann man dies beispielsweise dadurch, dass man das Vokabular künstlich verkleinert und somit die OOV-Rate künstlich erhöht. Eine andere Möglichkeit ist es, möglichst viele Sätze zusammenzustellen, die im Baseline-Vokabular des Baseline-Spracherkennersystems unbekannte Wörter darstellen. Der Vorteil ist, dass es sich so um für die Domäne möglichst *authentische* unbekannte Wörter handelt.

Für die Experimente zur Detektion unbekannter Wörter auf der BN-Domäne wurden deshalb Sätze zusammengestellt, die *nicht* im Training verwendet wurden und die Wörter enthalten, die nicht im Vokabular V_{Top40k} enthalten sind. Dabei handelt es sich um 247

[1] Defense Advanced Research Projects Agency der Vereinigten Staaten

Fokus	Beschreibung	WER
F0	geplant	11.0%
F1	spontan	19.4%
F2	über Telefon	24.0%
F3	von Musik überlagert	28.8%
F4	schlechtere akustische Bedingungen	15.0%
F5	nicht Muttersprachler	32.1%
FX	Kombinationen und sonstiges	26.9%

Tabelle 3.4: Wortfehlerraten des BN-Baseline-Spracherkenner nach Fokus-Konditionen

Äußerungen aus den Fokus-Konditionen F0, F1 und F4 der BN-Daten, die insgesamt 373 unbekannte Wörter enthalten. Die wesentlichen Merkmale dieser Datenbasis sind in Tabelle 3.5 zusammengefasst. Wie man sieht, ist die OOV-Rate mit 1,86% relativ groß, sie liegt aber, aufgrund der Länge der Segmente, auch nicht außerhalb des üblichen Wertebereichs für die englische BN-Domäne.

Umfang	Test
Dauer	106 Minuten
Äußerungen	247
Wörter	20089
Sprecher	76
OOV-Rate für V_{Top40k}	1.86%

Tabelle 3.5: Zusammensetzung der verwendeten BN-OOV-Daten

3.4 Tagesschau-Daten

Für die deutsche Sprache wurde eine detaillierte Analyse unbekannter Wörter mit Transliterationen von drei Tagesschau-Sendungen (TGS) durchgeführt (Kapitel 4). Die Tagesschau-Sendungen, die hierbei verwendet wurden, sind in Tabelle 3.6 angegeben.

Name	Sendedatum	Anzahl Wörter
TGS-1	10. Oktober 2001	2012
TGS-2	11. Oktober 2001	2100
TGS-3	12. Oktober 2001	2096

Tabelle 3.6: Untersuchte Tagesschau-Sendungen

Als Grundlage für die Erstellung der Vokabularien wurde dabei der Korpus der Süddeutschen Zeitung (SZ-Korpus) benutzt, indem die Wörter des Korpus nach ihrer Frequenz sortiert und die N häufigsten Wörter (TopN) als Vokabular ausgewählt wurden.

Zahlen und Zahlwörter, die in den untersuchten Texten aufgetreten sind, wurden kanonisiert und nicht zu den unbekannten Wörtern gezählt. Kommen in einer Äußerung Zahlen vor, werden diese bei der Kanonisierung in ihre kleinstmöglichen Teile aufgespalten. Beispielsweise wird die Altersangabe "21-jährige" oder "einundzwanzigjährige" umgeformt in "ein und zwanzig jährige'.

Kapitel 4

Analyse des OOV-Problems

Um festzustellen, welche Eigenschaften unbekannte Wörter besitzen, wurden Experimente mit verschiedenen Textkorpora durchgeführt und die Ergebnisse miteinander verglichen.

Es wurde untersucht, wie sich die Häufigkeit unbekannter Wörter (OOV-Rate) bei wachsendem Vokabular auf ungesehenen Testdaten verhält. Ferner wurde eine detaillierte Analyse der unbekannten Wörter auf Transliterationen von Tagesschau-Sendungen (vgl. 3.4) durchgeführt. Es wurde ein HDLA-Ansatz[1] verwendet (vgl. Geutner [35]), um zu untersuchen, welche Art von unbekannten Wörtern nach seiner Anwendung übrig bleibt. Schließlich wurde die Häufigkeit unbekannter Wörter in englischen Nachrichtensendungen (HUB-4-Domäne, BN-Evaluationsdaten) berechnet, um die OOV-Raten der beiden Domänen zu vergleichen. Eine genauere Analyse der unbekannten Wörter wurde hier ebenfalls vorgenommen.

4.1 Häufigkeit unbekannter Wörter bei deutschen Nachrichtensendungen

Zunächst wurde eine Untersuchung der OOV-Rate auf Transliterationen von drei Tagesschau-Sendungen (TGS) durchgeführt, bei denen die OOV-Rate mit unterschiedlich großem Vokabularumfang bestimmt wird. Die Daten zu den Tagesschau-Sendungen, die hierbei verwendet wurden, sind in Tabelle 3.6 angegeben.

In Abbildung 4.1 ist die Entwicklung der OOV-Rate auf der Menge dieser Tagesschau-Sendungen in Abhängigkeit von der Vokabulargröße dargestellt. Zu beachten ist hierbei, dass die y-Achse logarithmisch zur Basis 2 und die x-Achse logarithmisch zur Basis 10 dargestellt wurde. Diese Darstellung zeigt einen deutlich geraden Verlauf über einen weiten Bereich der Kurve, bekommt aber gegen Ende einen leichten Knick, der darauf zurückgeführt werden kann, dass auch das Vokabular aus einem endlichen Korpus erstellt

[1] Hypthesis Driven Lexical Adaptation

wurde und gegen Ende eine gewisse Menge an unbekannten Wörtern verbleibt, die niemals im SZ-Korpus aufgetreten sind. Diese OOV-Rate kann mit keinem der Vokabularien, die aus den Wortlisten des SZ-Korpus gewonnen werden können, unterschritten werden.

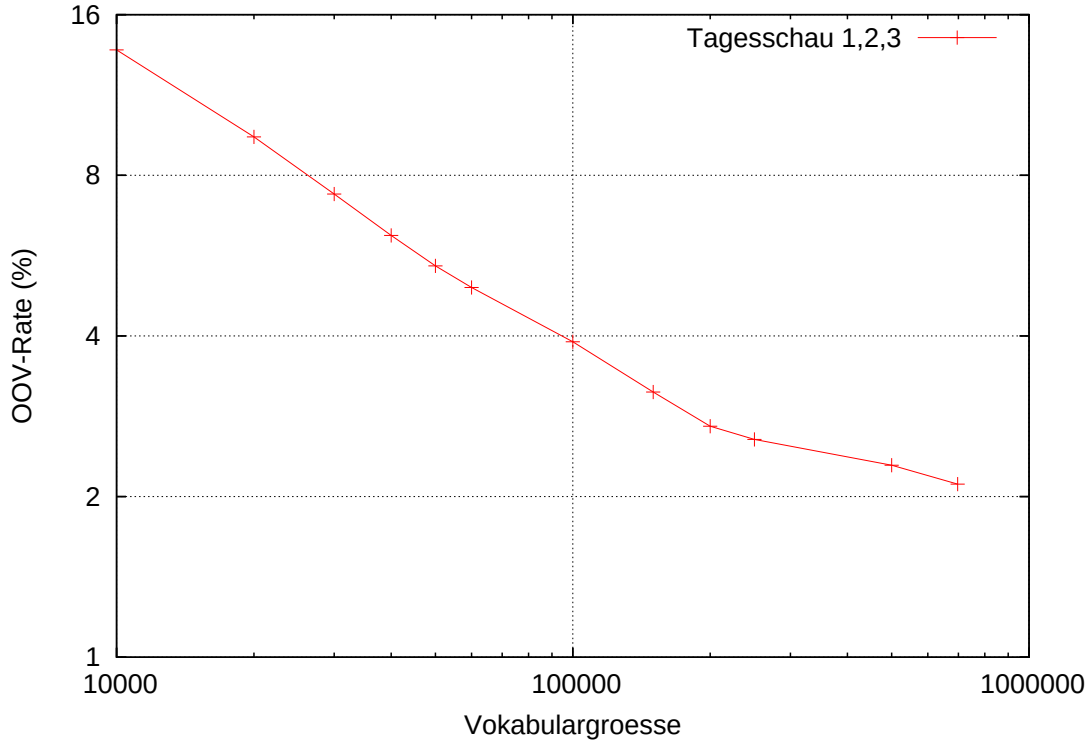

Abbildung 4.1: OOV-Rate über der Vokabulargröße

Wie man sieht, ist die OOV-Rate bei einem kleinen Vokabular der 10000 häufigsten Wörter des SZ-Korpus mit fast 15% sehr hoch, was bedeutet, dass fast jedes siebte Wort unbekannt ist. Selbst bei einer Vervielfachung des Vokabular um den Faktor 50 auf eine halben Million Wörter sinkt die OOV-Rate nur auf knapp unter 3%. In diesem Fall wird jedes Wort aus dem SZ-Korpus in das Vokabular aufgenommen, das mindestens zweimal aufgetreten ist. Selbst mit einem so großen Vokabular ist die Häufigkeit unbekannter Wörter also noch relativ hoch.

Auf ungesehenen Daten *muss* immer damit gerechnet werden, dass es selbst beim Verwenden eines beliebig großen aber endlichen Korpus für die Erzeugung des Vokabulars stets Wörter gibt, die noch nicht in diesem Korpus aufgetreten sind. Hierbei handelt es sich beispielsweise um Neuschöpfungen von inzwischen gängigen Wörtern wie "Pentium" oder "Handy", oder um Eigennamen, insbesondere wenn diese aus dem nicht-deutschen Sprachraum stammen (vgl. Kapitel 1).

In Tabelle 4.1 sind für die drei untersuchten Tagesschau-Sendungen die OOV-Raten zu einem Top60000-Vokabular angegeben.

Sendung	OOV-Rate
TGS-1	5,9%
TGS-2	4,4%
TGS-3	4,6%

Tabelle 4.1: OOV-Raten bei einem Vokabular der Größe Top60000

Detaillierte Analyse unbekannter Wörter

Um die Eigenschaften von OOV-Wörtern genauer zu untersuchen, wurden zunächst deren grammatische Kategorien betrachtet. Hierzu wurden die Transliterationen der Tagesschau-Sendungen TGS-1 und TGS-2 verwendet. Das Ergebnis ist in Tabelle 4.2 angegeben.

Kategorie	Anteil an OOV-Wörtern
Nomen	80%
Adjektive	14%
Verben	6%

Tabelle 4.2: Grammatische Kategorien TGS-1 und TGS-2 (Vokabular Top60000)

Wie erwartet stellt die Menge der Nomen die größte Anzahl der 209 unbekannten Wörter dar. Eine Analyse dieser unbekannten Wörter ergab, dass es sich bei über der Hälfte der unbekannten Nomen um Komposita handelt.

In Anlehnung an den von Geutner [35] entwickelten HDLA-Ansatz ließen sich 42% aller OOV-Wörter auf Wörter des Top60000-Vokabulars zurückführen. Durch die Verwendung des HDLA-Ansatzes kann effektiv über ein sehr viel größeres Vokabular verfügt werden, da Flexionen, Deklinationen und Komposita von Wörtern eines Basisvokabulars (z.B. Top60000-Vokabular) als bekannt vorausgesetzt werden können. Es verblieben aber immer noch 2,9% der Wörter als unbekannt.

Diese restlichen unbekannten Wörter wurden manuell daraufhin untersucht, wie viele von ihnen Eigennamen sind. Das Ergebnis ist in Tabelle 4.3 angegeben.

Es zeigt sich, dass der Anteil an Eigennamen bei den OOV-Wörtern mit erweitertem Vokabular steigt und Namen somit einen großen Anteil an den OOV-Wörtern ausmachen. Im allgemeinen steigt der Anteil an Namen an den OOV-Wörtern auch bei wachsendem Vokabular, wenn das Vokabular mit einem TopN-Kriterium festgelegt wird.

Ansatz	OOV-Rate	Eigennamen
ohne HDLA	5,1%	38%
mit HDLA	2,9%	76%

Tabelle 4.3: Anteil von Eigennamen bei unbekannten Wörtern

Mit einem HDLA-Ansatz lässt sich dieser verbleibende harte Kern unbekannter Wörter nicht erkennen, da er primär für Komposita, Deklinationen und Flexionen gedacht ist, bei denen auch eine deutliche Senkung der OOV-Rate erreicht wird. Besteht ein Name beispielsweise aus einem Kompositum wie "Baumstark" oder "Hellmann", gibt es die Möglichkeit, dass diese richtig erkannt werden. Bei vielen Namen wie beispielsweise "Reihl", "Clinton", "Putin" oder "Angola" ist dies aber so nicht möglich. Bei dem Wort "Kurskbergung" kann eine morphologische Zerlegung in die Bestandteile "Kursk" und das im Vokabular enthaltene Wort "Bergung" durchgeführt werden. Da aber der Eigenname nicht im Vokabular enthalten ist, kann das gesamte Wort so nicht erkannt werden.

4.2 Untersuchung unbekannter Wörter bei englischen Nachrichtensendungen

Ebenfalls für die Domäne der Nachrichtensendungen, aber für die englische Sprache wurde untersucht, welche unbekannten Wörter es auf dem Evaluierungsset BN96 Testset-1 gibt. Als Vokabular für die Bestimmung der unbekannten Wörter wurde das Vokabular V_{Top40k} zugrunde gelegt, das auch in den späteren Experimenten auf dem BN-Task verwendet wird.

Die untersuchten Daten haben im Vergleich zum Deutschen mit 0,55% eine sehr viel geringere OOV-Rate aufgrund der fehlenden Kompositabildung und der nur schwach vorhandenen Deklination und Konjugation im Englischen. Insgesamt waren 83 Wörter nicht im Vokabular des Spracherkenners enthalten. Der größte Teil davon mit 90,8% fällt in die grammatische Kategorie "Nomen", wie aus Tabelle 4.4 zu ersehen ist.

Kategorie	Anteil an OOV-Wörtern
Nomen	90,8%
Adjektive	10,8%
Verben	8,4%

Tabelle 4.4: Grammatische Kategorien

Von den unbekannten Wörtern stellen 53% Eigennamen dar, wobei Personennamen mit 41,0% den größten Anteil der unbekannten Wörter ausmachen (Nachnamen 38,6%, Vornamen 2,4%). Die genaue Aufschlüsselung der Namen nach Kategorien, wie sie im

Named-Entitiy-Task (NE-Task) der NIST [11] eingeteilt sind, ist aus Tabelle 4.5 zu ersehen.

Namensklasse	Anteil an OOV-Wörtern
Person	41,0%
Lokation	10,8%
Organisation	1,2%

Tabelle 4.5: Verteilung der Namen

Um eine Aussage darüber zu erhalten, wie groß die OOV-Rate auf den englischen BN-Evaluationsdaten ist, wurden die Evaluationsdaten aus den Jahren 1995 bis 1999 zusammengefasst, was 122k fortlaufende Wörter ergab. Es wurde dabei, bezogen auf das Vokabular V_{Top40k}, eine OOV-Rate von 0,96% gemessen.

Komposita stellen im englischen BN-Task im Vergleich zum Deutschen kein so bedeutendes Problem dar. Der Grund dafür ist, dass in der englischen Sprache im allgemeinen keine so häufige Kompositabildung stattfindet. Ferner wird auf diesem Task bei der Evaluierung eine regelbasierte Textnormalisierung durchgeführt, die neben Tippfehlern in den Referenzen auch englische Komposita zerlegt. Beispielsweise wird der Zeitungsname "WORLDVIEW" in "WORLD VIEW" aufgetrennt. Eine entsprechende Hypothese wird dann als korrekte Erkennung betrachtet. Eine solche Zerlegung ist prinzipiell auch für Tasks in anderen Sprachen möglich. In Verbmobil wird aber beispielsweise eine Zerlegung der Komposita nicht vorgenommen, da die Domäne sich mit dem verwendeten Vokabular gut abdecken läßt. Ob eine Zerlegung durchgeführt wird hängt in der Regel von der Anwendung des Tasks ab.

4.3 Zusammenfassung

Die Analyse ergab, dass im Deutschen ein großer Anteil der unbekannten Wörter durch die Bildung von Komposita, Deklination oder Flexionen aus bekannten Wörtern eines gegebenen Vokabulars abgeleitet werden können. Diese Wörter können beispielsweise durch einen HDLA-Ansatz (vgl. Geutner [35]) verarbeitet werden, wodurch das effektive Vokabular des Spracherkenners vergrößert wird und damit die OOV-Rate, so wie auch die Wortfehlerrate gesenkt werden kann.
Wie aus dem deutlichen Ansteigen des Anteils an Eigennamen in der verbleibenden Menge unbekannter Wörter zu sehen ist, stellen Namen eine Gruppe von Wörtern dar, die sich mit diesem Ansatz nicht verarbeiten lassen.

Die Untersuchung zeigte, dass auch in englischen Nachrichtensendungen bei großem Vokabular Namen einen wesentlichen Anteil an unbekannten Wörtern ausmachen. Auch die Verteilung unbekannter Wörter in grammatische Kategorien ist ähnlich wie im

Deutschen. In beiden Sprachen stellen erwartungsgemäß Nomen die größte Menge der OOV-Wörter dar.

Insgesamt lässt sich schließen, dass gerade der große Anteil an Eigennamen bei den OOV-Wörtern eine besondere Rolle spielt und wegen dem ihnen innewohnenden Informationsgehalt eine besondere Beachtung verdienen.

Kapitel 5

Stand der Forschung

In diesem Kapitel sind die relevanten Aufgaben für das Erkennen und Lernen unbekannter Wörter zusammengestellt. Es wird ferner ein Überblick über die in der Forschung untersuchten Lösungsansätze und den Stand der Technik gegeben.

Es gibt eine umfangreiche Menge an Publikationen, die sich mit dem OOV-Phänomen auseinandersetzen, insbesondere im Bereich der Einzelworterkennung. Hier wird der Schwerpunkt auf Untersuchungen gelegt, die sich hauptsächlich mit kontinuierlich gesprochener Sprache auseinandersetzen.

Die verfolgten Ansätze mit den Problemen umzugehen, die unbekannte Wörter in der Spracherkennung verursachen, sind einerseits diese Wörter zu detektieren oder anderseits die OOV-Rate ganz zu eliminieren. Um unbekannte Wörter zu detektieren, werden im Wesentlichen Methoden eingesetzt, die das verwendete Modell des Spracherkenners entweder erweitern (*direkte Methode*) oder versuchen, Erkennungsfehler zu detektieren und diese einem OOV-Wort zuzuordnen (*indirekte Methode*).

5.1 Direkte Methoden

Die Modelle des Spracherkenners, die bei den direkten Methoden in der Regel erweitert werden müssen, sind das Vokabular und in Abhängigkeit davon das Sprachmodell und die akustischen Modelle.

Aus diesem Ansatz heraus ergeben sich für die aktuelle Forschung zwei wesentliche Fragen:

- Welche akustische Modellierung ist für eine im Prinzip unendlich große Menge an unbekannten Wörtern zu verwenden?
- Wie sind die Modelle für unbekannte Wörter im Sprachmodell zu integrieren?

Das Ziel der akustischen Modellierung für OOV-Wörter ist, eine Modellierung zu finden, für die die Gleichungen 5.1 und 5.2 erfüllt sind, je nachdem ob Äußerung A ein *OOV-Wort* oder ein *Vokabular-Wort* ist. Dabei sollte Gleichung 5.1 gegenüber allen Wörtern des Vokabulars V gelten; Gleichung 5.2 muss nur gegenüber dem korrekten Wort gelten.

$$p(A|w_{\mathrm{OOV}}) > p(A|w) : \text{für alle } w \in V \text{ wenn } A \text{ ein OOV-Wort ist} \quad (5.1)$$
$$p(A|w_{\mathrm{OOV}}) < p(A|w) : \text{wenn } A \text{ das Vokabular-Wort } w \text{ ist} \quad (5.2)$$

Diese wünschenswerte Eigenschaft allein reicht jedoch nicht aus, um ein OOV-Wort zu detektieren, da die Wahrscheinlichkeit des Sprachmodells ebenfalls eine sehr wichtige Rolle spielt.

Wird davon ausgegangen, dass im Abschnitt A_j der Äußerung A ein OOV-Wort enthalten ist, wird die Gesamtwahrscheinlichkeit der Hypothese W beispielsweise für ein Tri-Gramm-Sprachmodell (Gleichung 5.3) beeinflusst von der Wahrscheinlichkeit $p(w_j|w_{j-2}w_{j-1})$, die für den Fall $w_j = w_{\mathrm{OOV}}$ ein OOV-Wort vorhersagt sowie von den Wahrscheinlichkeiten $p(w_{j+1}|w_{j-1}w_j)$ und $p(w_{j+2}|w_jw_{j+1})$:

$$\begin{aligned} p(W|A)p(W) = & \sum_{i=1}^{j-1} p(w_i|A_i)p(w_i|w_{i-2}w_{i-1}) \\ & + p(w_j|A_j)p(w_j|w_{j-2}w_{j-1}) \\ & + p(w_{j+1}|A_{j+1})p(w_{j+1}|w_{j-1}w_j) \\ & + p(w_{j+2}|A_{j+2})p(w_{j+2}|w_jw_{j+1}) \\ & + \sum_{i=j+3}^{N} p(w_i|A_i)p(w_i|w_{i-2}w_{i-1}) \end{aligned} \quad (5.3)$$

Diese Wahrscheinlichkeiten modellieren, ob das Auftreten eines OOV-Wortes für den Kontext der Hypothese plausibel, beziehungsweise eher wahrscheinlich oder unwahrscheinlich ist.

Darum lautet eine weitere wichtige Frage:

- Wie wird die Modellierung unbekannter Wörter am besten durch das Sprachmodell unterstützt?

Die Veröffentlichungen konzentrieren sich oft auf einen der beiden Aspekte: akustische Modellierung *oder* Sprachmodellierung. Es wird meist eine einfache akustische Modellierung verwendet, um die Stärken der Sprachmodellierung aufzuzeigen oder eine einfache Modellierung im Sprachmodell, um die Stärken der akustischen Modellierung hervorzuheben.

5.1.1 Akustische Modellierung

Asadi [1] war einer der ersten, der sich mit dem Problem von unbekannten Wörtern auseinandergesetzt hat, und er legte dabei die wesentlichen Grundsteine für die späteren Arbeiten auf diesem Gebiet. Er untersucht grundlegende Ansätze zur akustischen Modellierung unbekannter Wörter. Eines der Modelle für unbekannte Wörter besteht aus einer Folge eines generischen Phonemmodells. Dieses Modell wurde mit allen Phonemen trainiert (P*). Asadi nennt dieses Modell "Flat New-Word-Modell" (flaches Modell). In Abbildung 5.1 sind zwei Varianten eines Flat New-Word-Modells dargestellt. Im unteren Fall wird durch das Wiederholen des generischen Phonem-Modells eine Mindestlänge für ein unbekanntes Wort erzwungen.

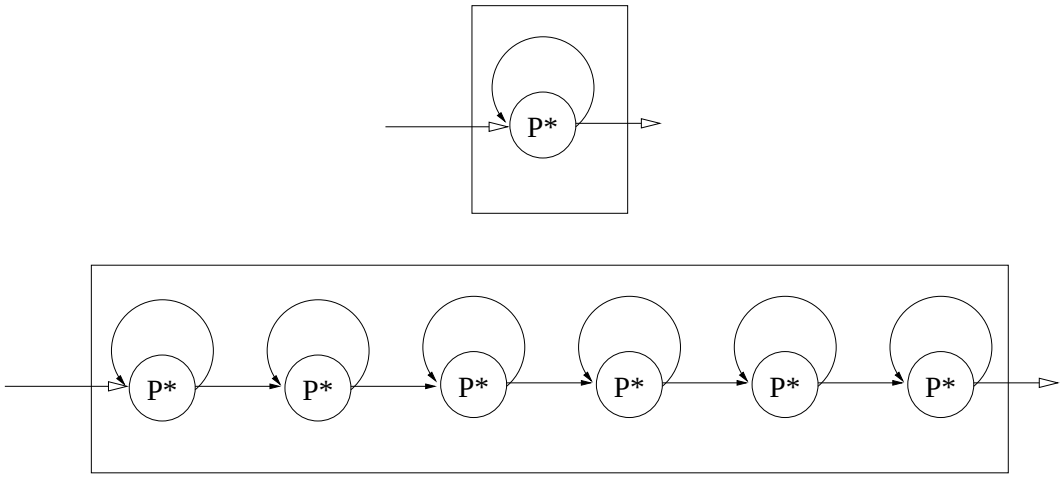

Abbildung 5.1: Zwei Realisierungen eines Flat New-Word-Modell

Dieses Modell vergleicht Asadi mit einem Modell, das eine Phonemschleife realisiert und beliebige Phonemfolgen erlaubt (Abbildung 5.2).

Er führte Experimente auf dem RM-Task durch, in denen er das flache Modell mit verschiedenen Realisierungsformen der Phonemschleife verglich. Um eine Detektion durch die flachen Modelle zu erleichtern, wurde ein Bias zugunsten des Modells eingeführt. Dagegen wurde für die Phonemschleifen ein Bias gegen das Modell verwendet. Der Spracherkenner verwendet eine kontextabhängige Modellierung für die bekannten Wörter aus dem Vokabular. Das beste Ergebnis wurde mit der Phonemschleife und einer Modellierung der Mindestlänge von zwei Phonemen unter Verwendung kontextunabhängiger akustischer Modelle erzielt.

Werden die gleichen, kontextabhängigen akustischen Modelle wie für die bekannten Wörter eingesetzt, steigt die Zahl der Fehldetektionen stark. Asadi führt dies darauf zurück, dass durch die Phonemschleife bekannte Wörter ebenfalls gut modelliert werden, die Phonemschleife selbst aber die akustischen Modelle freier kombinieren darf, als dies bei bekannten Wörtern der Fall ist. Insbesondere das Phänomen, dass Folgen von bekannten

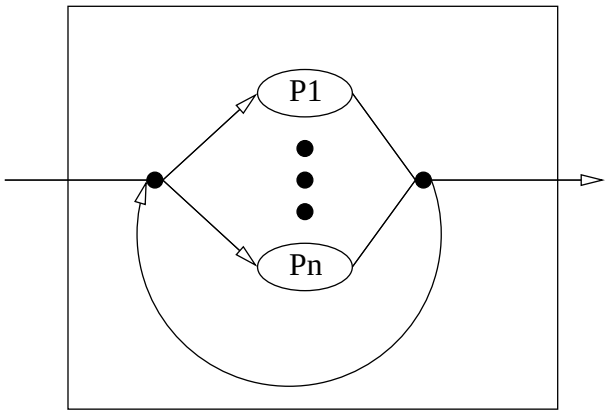

Abbildung 5.2: Schleife mit allen Phonemen

Wörtern von den Phonemschleifen "aufgefressen" werden, stellt eine Schwäche dieses Modellierungsverfahrens dar.

Die Modelle für OOV-Wörter wurden vom verwendeten Sprachmodell, einer Grammatik, wie bekannte Wörter modelliert. In der Grammatik ließen sich offene Klassen, in denen unbekannte Wörter zu erwarten sind, auf einfache Art manuell bestimmen, und ein Übergang in ein akustisches Modell für unbekannte Wörter realisieren. Es wurden offene Klassen von Namen, beispielsweise für Schiffe, um die Möglichkeit erweitert, ein Symbol für ein OOV-Wort in der Hypothese auszugeben. Dabei konnte auch gleichzeitig eine Klasse, beispielsweise Schiffsnamen, für das unbekannte Wort hypothetisiert werden [1].

Phonemschleifen

Suhm [75] verwendet für die Detektion auf dem englischsprachigen *CR-Task* wie Asadi ebenfalls eine Phonemschleife, benutzt aber bei der Modellierung eine Mindestlänge von drei Phonemen, da davon ausgegangen wird, dass alle kurzen Wörter mit weniger als drei Phonemen bereits im Erkennervokabular enthalten sind. Um zu verhindern, dass das Modell der unbekannten Wörter zu gut auf bekannte Wörter passt, werden zum einen kontextunabhängige akustische Modelle verwendet, im Gegensatz zu den bekannten Wörtern, die kontextabhängig modelliert sind. Zum anderen werden die Phonemübergänge mit einem Bias versehen, der aus einem Wörterbuch für den WSJ-Task mit 30k Wörtern trainiert wurde (Bigramm-Sprachmodell) und es wird noch ein von der durchlaufenen Phonemanzahl anhängiger Strafterm eingeführt.

Bazzi [6] [5] benutzt in seinen Untersuchungen ebenfalls Phonemschleifen, vergleicht diese aber mit einer Schleife, die auch längere Phonemfolgen als Grundeinheit benutzt. Die Phonemschleifen verfügen über ein Bigramm-Sprachmodell. Er vergleicht dabei ein auf dem Trainingskorpus trainiertes Bigramm-Sprachmodell mit einem, das ähnlich wie bei Suhm

[75], auf einem großen Wörterbuch trainiert ist. Das auf dem Wörterbuch trainierte Sprachmodell liefert dabei erwartungsgemäß bessere Ergebnisse, da das auf dem Trainingskorpus trainierte Sprachmodell die häufigen Wörter des Korpus bevorzugt.

Die längeren Einheiten, die anstelle der Phoneme benutzt werden, sind Phonemfolgen, die datengetrieben gefunden werden, indem Paare zusammengefügt werden, deren gemeinsame Auftreten (co-occurancy) mehr als zufällig ist (Mutial Information). Dies ist das gleiche Verfahren, wie es von Brown [9] für die Bestimmung von Phrasen benutzt wird.

Bazzi bestimmte 1977 von diesen Phonemphrasen mit einem großen Wörterbuch und trainiert ein Bigramm-Sprachmodell für die verwendete Schleife. Er stellt fest, dass 2/3 der so gelernten Einheiten korrekte englische Silben sind. Es ist zu bemerken, dass durch die größere Anzahl an längeren Einheiten dieses Sprachmodell auch eine längere Historie modelliert.

Die datengetrieben gefundenen Einheiten ergeben in einer ROC-Darstellung bei hohem Recall bessere Ergebnisse als die Phonemschleife mit dem Bigramm-Sprachmodell, das mit dem Wörterbuch gelernt wurde.

Ergebnisse werden für zwei unterschiedlichen Testmengen angegeben. So wird ein Recall mit einem Testset angegeben, das mindestens ein unbekanntes Wort enthält, die Wortfehlerrate und die Rate der Fehlalarme mit einem anderen Testset, das nur bekannte Wörter enthält. Durchgeführt wurden diese Messungen für die Jupiter-Domäne, einem Wetterauskunftsdienst. Der Erkenner verfügt über ein kleines Vokabular von 2k Wörtern.

Für das Testset ohne unbekannte Wörter erzielte der Erkenner eine niedrige Wortfehlerrate von 10.4%. Durch den Einsatz der Detektion unbekannter Wörter verschlechtert sich die Wortfehlerrate um 0.4%, wenn kein Bias im Sprachmodell benutzt wird, um 0.5%, wenn ein Bias eingesetzt wird. Bei diesen Einstellungen werden auf dem Testset mit unbekannten Wörtern ungefähr 50% der unbekannten Wörter gefunden. Leider wurden keine Angaben zu der Rate der unbekannten Wörter auf der Testmenge angegeben, Angaben über die Wortfehlerrate fehlen ebenfalls. Diese Angaben wären notwendig, um einschätzen zu können, ob diese Modelle eine Verbesserung bei der Wortfehlerrate bewirken.

Jusek [50] [49] vergleicht die Phonemschleife mit einer Modellierung von unbekannten Wörtern durch einen Silbenautomat, einer Schleife, die Silben als Einheiten modelliert. Als akustische Modelle wurden kontextunabhängig trainierte Phonemmodelle eingesetzt. Nachdem der Wortkontext, in dem ein unbekanntes Wort hypothetisiert werden kann, und die maximal mögliche Anzahl der Silben-, beziehungsweise Phonemmodelle eingeschränkt wird, die durchlaufen werden können, liefert der Silbenautomat bei vergleichbarem Recall eine deutlich höhere Präzision.

Das Baseline-System ohne die Möglichkeit der Detektion unbekannter Wörter erreicht eine Wortfehlerrate von 17,5%. Das beste Ergebnis, das mit dem Silbenautomat erreicht wurde, lag jedoch bei 26.9% und stellt damit einen deutlichen Einbruch in der Erkennungsleistung dar, der bei Verwendung einer Phonemschleife mit 31.5% noch stärker ausfällt. So gesehen reduziert der Silbenautomat die Fehler der Phonemschleife bei der Wortfehlerrate um 15% relativ.

Bei einer späteren Untersuchung setzte Jusek [48] nicht mehr kontextunabhängige Modelle zur Modellierung der unbekannten Wörter ein sondern Triphonmodelle, bei denen als Kontext nur zwischen Vokalen und Konsonanten unterschieden wurde, das heißt, jedes Phonem kann in vier verschiedenen Kontexten modelliert werden. Die bekannten Wörter wurden mit Triphonmodellen realisiert, die über einen präzisen Kontext verfügen.

Bei den Experimenten mit 84 Äußerungen von 42 Sprechern aus der Verbmobil-Domäne mit einer hohen OOV-Rate von 9.5%, ergeben die kontextabhängigen Modelle unter Verwendung deterministischer Regeln für den Wortkontext, in denen die unbekannten Wörter auftreten können, bei einer um zirka 6% höheren Präzision von 75% einen doppelt so hohen Recall von 45%.

Modellierung unbekannter Wörter durch Morphem-Graphem-Einheiten

Chung [13] [15] [14] benutzt zur Modellierung von unbekannten Wörtern Einheiten, die Morpheme darstellen, aber eine Abbildung auf Buchstabenfolgen besitzen. Wird ein unbekanntes Wort hypothetisiert, kann dadurch gleich eine Verschriftung mit angegeben werden. Diese Einheiten wurden automatisch in einem iterativen Prozess gelernt. Da dies auch die grundlegenden Einheiten für die Spracherkennung sind, musste die Rechtschreibung der bekannten Wörter angepasst werden. So wird beispielsweise "London" als "Lundon" geschrieben. Trainiert ist der Erkenner für die Jupiter-Domäne. Das wortbasierte Baseline-System besitzt ein Vokabular von zirka 1341 Wörtern. Unbekannte Wörter, die in der Domäne auftreten, sind hauptsächlich Namen von Städten und Regionen.

In den Experimenten wurden leider keine Angaben gemacht, wie oft ein unbekanntes Wort vorkam und mit welcher Präzision es detektiert wurde.

Auf einem Testset, das unbekannte Städtenamen enthält, wurde eine Reduktion der Wortfehlerrate von 24.6% (Baseline) auf 15.6% erreicht. Dabei wurden 167 Städtenamen gefunden auf denen zusätzlich eine Fehlerrate auf Buchstabenfolgen von 57% gemessen wurde. Es wurde ferner eine Auswahl der Hypothese aus einer N-Bestenliste durchgeführt, bei der eine NLP-Komponente die erste parsebare Hypothese selektierte. Dabei wurde in Kauf genommen, dass die Wortfehlerrate wieder anstieg, aber insgesamt mehr Einträge in die semantischen Rahmen getätigt wurden. Das dort definierte Fehlermaß für Nicht-Verstehen wurde von 67% (Baseline) bis auf 21.8% gesenkt, dabei lag dann eine Wortfehlerrate von 17.4% vor.

Gemischte lexikalische Einheiten

Die bisher beschriebenen Modelle zur Modellierung unbekannter Wörter benutzen Grundeinheiten, die in einem HMM zusammengefasst sind und in das Wörterbuch aufgenommen werden. Damit ist eine wortgleiche Modellierung im Sprachmodell einfach möglich. Das bedeutet, dass in einem Sprachmodell ein Eintrag für ein *OOV-Wort* benutzt werden kann. Eine andere Modellierung benutzt im Vokabular zusätzlich kleinere Einheiten als ganze Wörter. Diese modellieren beispielsweise Teile von Wörtern und werden als

eigenständige Erkennungseinheiten im Wörterbuch interpretiert. Dementsprechend muss auch das Sprachmodell erzeugt werden. Es ist beispielsweise möglich, Silben für die Modellierung unbekannter Wörter zusammen mit den bekannten Wörtern zu benutzen. In diesem Fall enthalten die Hypothesen eines Spracherkenners bekannte Wörter; an den Stellen, an denen ein unbekanntes Wort aufgetreten ist, wird eine Folge von Silben ausgegeben. Dabei zeigt sich der wesentliche Unterschied erst bei der Kombination mit dem statistischen Sprachmodell, da nun diese Einheiten auch im Sprachmodell auf der gleichen Abstraktionsebene modelliert werden müssen [51] [52].

Der Vorteil, dass die Einheiten mit Wörtern gemischt werden, liegt darin, dass dies prinzipiell mit jedem Sprachdekoder ohne großen Aufwand durchzuführen ist, wenn dafür der Nachteil einer verringerten Historie im statistischen N-Gramm-Sprachmodell in Kauf genommen wird.

Wie wir von Suhm [75] wissen, stellt das statistische Sprachmodell eine wichtige Wissensquelle für die Vorhersage von unbekannten Wörtern dar, weshalb Ansätze vorzuziehen sind, die den vollen Kontext auf Wörterebene erhalten.

Modellierung unbekannter Wörter durch einen Silbenautomaten

Von Kemp und Jusek [51] wurden zirka 10000 Silben in das Vokabular und somit auch in das statistische Sprachmodell aufgenommen.

Es wurde festgestellt, dass durch die Verwendung eines 3-Gramm-Sprachmodells, also eines Sprachmodells mit einer festen Länge, zwar eine gute Vorhersage in den so realisierten Silbenautomaten möglich ist, das Sprachmodell jedoch Schwierigkeiten hat, ein Nachfolgewort vorzuschlagen, da die Historie hierfür nicht mehr ausreicht. Für die Übergänge im Sprachmodell zwischen den Silben-"Wörtern" wurde eine Statistik erstellt, die auf den seltenen Wörtern des Trainingskorpus bestimmt wurden. Mit diesem Ansatz, der einen Silbenautomaten nachbildet, wurden Experimente auf der Verbmobil-Domäne mit einem 3k Vokabular auf einer Teilmenge der offiziellen Verbmobil-Evaluationsdaten durchgeführt, die eine OOV-Rate von 3.19% aufweist und 3823 Wörter umfasst. Für die Silbenwörter wurden dabei kontextabhängige akustische Modelle verwendet.

Durch diese Modellierung von OOV-Wörtern konnte die Fehlerrate von 27.9% um 0.5% gesenkt werden. Es wurde dabei ein Recall von 23.8% bei einer Präzision von 68.9% erreicht. Trotz der Schwierigkeiten, die sich durch die Verkürzung der Historie im Sprachmodell ergeben, ist dies ein gutes Ergebnis.

Modellierung unbekannter Wörter durch Kernvokabular und Phonemphrasen

Einen ähnlichen Ansatz wie Jusek und Kemp benutzt auch Klakow [52] in seiner Untersuchung, bei der er bei einem 64k-Spracherkenner nur die 5k häufigsten Wörter als Kern-

vokabular lässt und die restlichen Wörter auf dem Trainingskorpus in ihre Phonemfolgen zerlegte. Dabei wurde der Phonemsatz verdoppelt, um die Information der Wortgrenzen zu erhalten. Die so im Korpus des Sprachmodells entstandenen Phonemfolgen werden dann datengetrieben zu Phrasen zusammengefasst. Als Kriterium dient dabei die Häufigkeit ihres Auftretens. Dadurch ergeben sich Phrasen, die die Phonemfolgen aus den häufigeren der seltenen Wörter gut modellieren. Als akustische Modellierung wurden kontextunabhängige Modelle für die gefundenen Fragmente eingesetzt. Experimente wurden mit einem 5.5k Vokabular durchgeführt, zu dem jeweils die gefundenen Phonemphrasen hinzugefügt wurden. Die Experimente wurden auf dem männlichen Teil der ARPA NAB'94-Entwicklungsdaten durchgeführt.

Durch das kleine Vokabular für diesen Task ergibt sich eine für das Englische sehr große OOV-Rate von 10%. Das beste Ergebnis wurde durch das Hinzunehmen von nur 1259 Phonemphrasen erreicht. Es wurden 47% der unbekannten Wörter (REC) mit einer Präzision von fast 100% gefunden. Das Baseline-System mit 3-Gramm-Sprachmodell ohne Cross-Word-Modelle besitzt eine Fehlerrate von 28.9%. Bei Verwendung von Cross-Word-Modellen, die aber nur nach dem Löschen der Phonemphrasen aus den Hypothesen an den Grenzen der Phonemphrasen angewendet werden, wird eine Fehlerrate von 25.0% erreicht.

Es ist nicht sicher, ob durch die künstlich hohe OOV-Rate eine Aussage darüber gemacht werden kann, wie die Erkennung beeinflusst wird, wenn man diese Modelle mit dem Vokabular des Baseline-Erkenners von 64k Wörtern einsetzt, bei dem der Erkenner eine Wortfehlerrate von 14.8% besitzt. Ob ein so hoher Recall und eine so hohe Präzision zu erwarten sind, ist ungewiß.

Flache akustische Modelle

Die von Asadi untersuchte flache akustische Modellierung wird von Fetter [23] [22] erweitert, indem anstelle eines einzigen Wortmodells, das aus einer Hintereinanderfügung der gleichen akustischen Modelle besteht, fünfzehn Ganzwortmodelle trainiert werden. Diese fünfzehn Wortmodelle unterscheiden sich in der Anzahl der verwendeten Zustände und wurden mit entsprechend unterschiedlich langen Wörtern aus dem akustischen Trainingskorpus trainiert.

Jusek [48] bemerkt hierzu, dass diese Modelle der mittleren Struktur der häufigsten Wörter aus dem Trainingskorpus entsprechen müssten.

Fetter [22] Seite 116 erzielte in der Verbmobil-Domäne mit diesen Modellen bei einer OOV-Rate von 1.96% einen Recall von 21% mit einer Präzision von 49%. Dabei wurde die Wortfehlerrate nicht negativ beeinflusst sondern gar um 0.1% verbessert. Er verwendete dabei ein statistisches Sprachmodell, das eine einzige Klasse für die Vorhersage von unbekannten Wörtern besitzt, zu denen Fetter auch Wortfragmente zählt, die rund 1% der OOV-Wörter ausmachen.

Während in den Untersuchungen von Fetter auch Wortfragmente zu den unbekannten Wörtern gezählt werden, werden sie in dieser Arbeit zu den Spontaneffekten gerechnet, die der Spracherkenner von sich aus beherrschen sollte, da diese Wörter nie vom Spracherkenner gelernt werden. Aufgrund der unterschiedlichen Behandlung der Wortfragmente

sind die erzielten Ergebnisse darum nicht direkt zu vergleichen.

In den Arbeiten von Gallwitz [31] [32] [33] wurde ebenfalls ein flaches Modell für unbekannte Wörter benutzt. Dabei wurde ein Bias an den Transitionen zwischen den verschiedenen HMM-Zuständen verwendet. Bei jedem Merkmalsvektor wird so die Bewertung durch die flachen Modelle angehoben. Experimente mit dieser akustischen Modellierung wurden auf einem Zugauskunftsystem (EVAR-Domäne) durchgeführt, das per Telefon erreichbar ist. Dabei konnte ein Ansteigen der Fehlerrate beobachtet werden, das auf die geringe Präzision von 13% bei der Vorhersage unbekannter Wörter zurückgeführt wird.

Der Einsatz dieses Modells in der Verbmobil-Domäne war dagegen sehr erfolgreich. Dort konnte die Fehlerrate der Baseline von 22.5% auf 22.4% bei einem Recall von 30% und einer Präzision von 34% gesenkt werden. Auch bei dieser Untersuchung wurden, wie bei Fetter, spontane Effekte, wie beispielsweise Wortabbrüche, als unbekanntes Wort gezählt. Es wurden mehrere Klassen von unbekannten Wörtern modelliert und zusätzlich Ergebnisse für die Klasse der Nachnamen angegeben, die in Verbmobil die meisten unbekannten Wörter enthält. Für die Klasse der Nachnamen wurde eine Präzision von 68% und ein Recall von 35% erreicht.

Modellierung unbekannter Wörter mit OOV-Wörtern

Ein interessanter Ansatz wurde von Bouilianne und Dumouchel [8] untersucht, bei dem zu einem französischen Diktiererkenner mit einem Vokabular von 20k Umfang die nächsten 40k (beziehungsweise 600k) häufigsten Wörter aufgenommen werden. Als unbestreitbarer Vorteil dieser Modelle wird genannt, dass sie nicht die Möglichkeit besitzen, Wörter aus dem Vokabular nachzubilden und damit nicht so leicht mit Wörtern aus dem Vokabular verwechselt werden können. Im verwendeten Erkenner können die hinzugefügten Wörter in ein einziges HMM-Wort zusammengefasst werden (Abbildung 5.3), wodurch die Wortidentität verlorengehen kann. Besonders bemerkenswert ist, dass dieses OOV-Wort-Modell im Sprachmodell ohne Kontextinformation modelliert wird, im wesentlichen also nur die akustischen Modelle für die Detektion verantwortlich sind.

Experimente zeigen eine gute Leistung. Bei einer OOV-Rate von 3.78% erreicht die Baseline eine Wortfehlerrate von 24.7%. Mit den hinzugefügten 600k Wörtern wird ein Recall von 41.0% und bei einer beachtlichen Präzision von 85.2% und eine Fehlerrate von 22.8% erreicht. Werden nur die 40k Wörter verwendet, so wird eine geringfügig höhere Fehlerrate von 23.0% erzielt.

Da Verwechslungen zwischen verschiedenen OOV-Wörtern nicht zu den Erkennungsfehlern zählen und insbesondere das OOV-Modell besonders gut passt, wenn eines der OOV-Wörter, das auch für die Konstruktion des OOV-Modells benutzt wurde, ließe sich auch die hohe Präzision erklären.

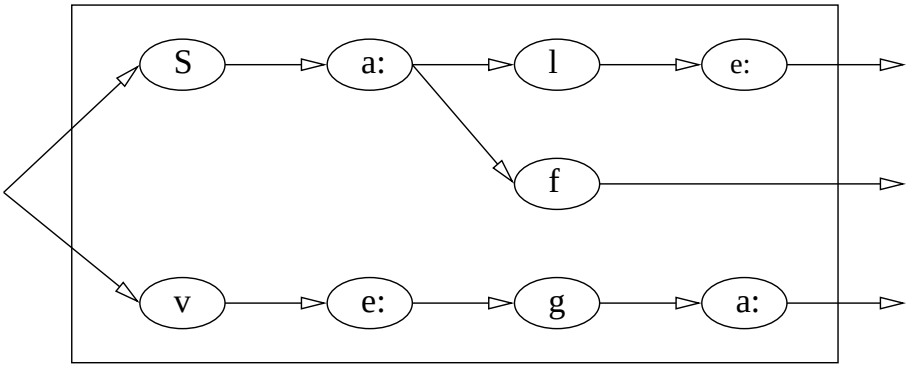

Abbildung 5.3: Zusammenfassen von unbekannten Wörtern

5.1.2 Sprachmodellierung unbekannter Wörter

Bei der Sprachmodellierung unbekannter Wörter wird untersucht, wie die Wahrscheinlichkeit und die Position für das Auftreten dieser Wörter bestimmt werden kann.

Modellierung mit deterministischen Grammatiken

Die ersten Spracherkennungssysteme hatten kleine fest umrissene Tasks, bei denen nicht statistische sondern grammatische Sprachmodelle eingesetzt wurden. In diesen Modellen ließen sich offene Klassen, in denen unbekannte Wörter zu erwarten sind, auf einfache Art bestimmen, und ein Übergang in ein akustisches Modell für unbekannte Wörter realisieren. Dies gilt beispielsweise für die bereits genannte grundlegende Arbeit von Asadi [1], die auf dem RM-Task durchgeführt wurde, in dem neben anderen beispielsweise die Klasse der Schiffsnamen als offene Klasse angesehen wurde.

Modellierung ohne Kontextwissen als Monogramm

Eine einfache Methode unbekannte Wörter im Sprachmodell zu modellieren ist, sie an jeder beliebigen Stelle mit der gleichen Wahrscheinlichkeit vorherzusagen. Dies entspricht der Modellierung von sogenannten Fillerworten. Mit dieser Art der Integration lässt sich beispielsweise gut zeigen, dass sich die akustischen Modelle der unbekannten Wörter von denen der bekannten Wörter gut unterscheiden. Außerdem wird der Umfang eines Sprachmodells durch dieses sehr einfache Verfahren nicht vergrößert. Benutzt wurde dieses Verfahren beispielsweise in der bereits genannten Arbeit von Boulianne und Dumouchel [8].

Modellierung in statistischen Sprachmodellen

Die Annahme, dass unbekannte Wörter an jeder Stelle gleich wahrscheinlich sind, ist im allgemeinen nicht richtig. So ist es beispielsweise nach den Worten "Guten Tag Herr" sehr viel wahrscheinlicher ein unbekanntes Wort, in diesem Falle einen Personennamen,

anzutreffen, als nach der Wortfolge "der Zug geht um", wo eine Zeitangabe zu erwarten ist. Der einfachste Ansatz eine Unterstützung durch das Sprachmodell zu erreichen ist es, die OOV-Wörter in eine einzige Klasse abzubilden, wie beispielsweise Suhm [75], und zusätzlich die Größe der Klasse zu schätzen. Eine Schwierigkeit hierbei ist, dass die Häufigkeit und die Größe der Klasse unterschätzt wird, da das Vokabular meist ebenfalls auf dem Trainingskorpus bestimmt wird.

5.1.3 Schätzen der Häufigkeit unbekannter Wörter

Good-Turing Schätzung

Good [37] hat eine Methode vorgestellt, mit der sich die Anzahl unbekannter Wörter aus der Häufigkeit der Wörter bis zu einem bestimmten Mindestauftreten schätzen lässt. Die Idee, die dabei zugrunde liegt ist, dass die Wahrscheinlichkeit der ungesehenen Ereignisse durch das Verhältnis der Ereignisse, die höchstens r-mal vorkommen und der Gesamtanzahl N der Ereignisse zu schätzen. Die Formel, die daraus abgeleitet wird, ist als "Good-Turing reestimation" bekannt, und wird beispielsweise neben dem "absolut discounting" benutzt, um die Wahrscheinlichkeiten ungesehener N-Gramme in statistischen Sprachmodellen vorherzusagen.

Erzeugen von OOV-Wörtern für die Modellierung im Sprachmodell "Iteratives Substitutionsverfahren"

Fetter [22] hat eine weitere Methode entwickelt, um die Wahrscheinlichkeit für das Auftreten von unbekannten Wörtern zu schätzen und zugleich mögliche Kontexte von unbekannten Wörtern zu bestimmen.
Das Verfahren nennt Fetter "Iterative Substitution". Dazu wird der Trainingskorpus aufgeteilt in einen Teil, auf dem ein Vokabular bestimmt wird und in einen Teil, auf dem dann die unbekannten Wörter markiert werden. Der Trainingskorpus wird dabei N-mal hintereinander geschrieben und dementsprechend die gewonnenen Statistiken über die Wortkontexte durch N geteilt. Damit wird dann ein statistisches Sprachmodell berechnet. Das Verfahren eignet sich besonders gut, wenn *jedes* Wort des Sprachmodellkorpus im endgültigen Vokabular enthalten sein soll und damit die OOV-Rate auf dem Trainingskorpus 0% betragen würde.

5.1.4 Verfahren zur Bildung von Klassen

Manuelle Klassenbildung durch Domänenwissen

Anstelle einer gemeinsamen Klasse für unbekannte Wörter lassen sich auch mehrere Klassen definieren. Dabei werden im allgemeinen semantische Klassen benutzt, die sich direkt aus der Domäne ergeben. So werden von Gallwitz [31] für die GSST-Domäne in Verbmobil

die drei semantischen Klassen "Vornamen", "Nachnamen" und "Ortsnamen" verwendet. In der EVAR-Domäne werden zusätzlich noch die semantische Klassen "Ländernamen" und "Fluchworte" modelliert. Weiter wird eine Klasse für seltene Wörter und eine Klasse für spontane Effekte benutzt, wie beispielsweise Wortabbrüche. Gallwitz schätzt durch eine Simulation des Vokabularwachstums (VocSim) für die semantischen Klassen die Wahrscheinlichkeit für das Auftreten eines unbekannten Wortes als die Ableitung (Steigung) des Vokabularwachstums innerhalb dieser Klasse. Die so geschätzte Wahrscheinlichkeit für das Auftreten eines unbekannten Wortes wird dann in der betreffenden Klasse für die Detektion von unbekannten Wörtern reserviert, in dem sie gleichmäßig von allen anderen Klassenelementen abgezogen wird.

Die Einteilung der Wörter in die semantischen Klassen war manuell vorgegeben. Das Sprachmodell besitzt durch diese Modellierung die Möglichkeit, aufgrund des Kontextes eine semantische Klasse für ein unbekanntes Wort vorzuschlagen, analog zur Grammatik, die von Asadi verwendet wurde.

Manuelle Klassenbildung durch Wortlisten

Der einfachste Ansatz zur Bildung von Klassen ist Wortlisten aufzustellen und diese manuell Klassen zuzuordnen. In der Praxis lässt sich das jedoch nur sinnvoll bei kleinen Wortschätzen in kleinen Domänen durchführen. Der manuelle Aufwand ist sonst zu groß, außerdem sind bei großen Domänen immer mehr Wörter ambig, so dass Listen nicht mehr anwendbar sind.

Der Begriff "Essen" beispielsweise kann, je nach Kontext in dem er verwendet wird, sowohl für den Name einer Stadt im Ruhrgebiet (Deutschland) stehen als auch als Bezeichnung für ein Treffen, bei dem gegessen wird. In der GSST-Domäne kann also das Wort "Essen" der Start- oder Zielort einer Reise sein oder auch bedeuten, dass man einen Termin *zum Essen* vereinbaren will.

Bei Verbmobil wurde "Essen"in der Tat aufgrund der Verwendung von Wortlisten mit Ortsnamen als Ortsangabe betrachtet, aber eine manuelle Überprüfung ergab, dass dieser Begriff *nicht ein einziges Mal* mit dieser Bedeutung verwendet wurde sondern immer nur im oben genannten Zusammenhang. Ein weiteres Beispiel für die Schwierigkeit bei der Einteilung von Wörtern in Klassen anhand von Wortlisten ist der Namen "Washington", der entweder einen Ort oder eine Person bezeichnen kann.

Klasseneinteilung mit einem Named-Entity-Tagger

Um diese Ambiguitäten nicht manuell auflösen zu müssen, gibt es die Möglichkeit, auf einem kleineren manuell eingeteilten Text eine Abbildung von Wörtern abhängig vom Kontext in die *vorgegebenen* Klassen zu lernen. Dies wird beispielsweise beim Named-Entity-Task (NE-Task) gemacht. Mit einem so trainierten Tagger (NE-Tagger) ist es dann möglich, eine genauere Einteilung in die trainierten Klassen auch auf großen Korpora durchzuführen.

Die von der DARPA durchgeführten Evaluationen von NE-Taggern konzentrieren sich im Wesentlichen auf die drei hier relevanten Klassen "LOCATION", "PERSON" und "ORGANISATION" [11].

Die restlichen Klassen sind auch deshalb nicht von Interesse, da von Miller und Schwartz [56] festgestellt wurde, dass mit einem Vokabular von 40k in der englischen BN-Domäne alle Angaben der Klassen "PERCENT", "TIME" oder "MONEY" zu 100% auf den Testdaten abgedeckt sind. Miller beobachtete zudem, dass die OOV-Rate auf den Namensklassen beim NE-Task um das zehnfache größer sein kann als im Mittel für "Nicht-Namen" und dass Personennamen 50% der Gesamtmenge der Namen ausmachen.

Named-Entity-Klassen

Von Gotoh et al [38] wurden Sprachmodelle um die Klassen "PERSON", "LOCATION", "ORGANISATION" und die Klasse "NAME" für nicht genauer spezifizierte Namen sowie der Restklasse "UNKNOWN" erweitert. Experimente mit jeweils einem 20k umfassenden Basisvokabular wurden auf dem WSJ-Task und dem BN-Task durchgeführt. Das Sprachmodell wurde mit dem entsprechenden Basisvokabular und einer Einteilung der entsprechenden Wörter in diese Klassen erstellt, die durch einen NE-Tagger durchgeführt wurde. Eine analoge Vorgehensweise wurde auch in vorliegender Arbeit für die Einteilung der unbekannten Wörter untersucht.

Gotoh erweiterte dann das Basisvokabular um Wörter, die in die oben genannten Klassen eingefügt werden. Das Erkennervokabular auf dem WSJ-Task umfasste danach 124k, auf dem BN-Task 109k. Die OOV-Rate sank dadurch auf dem WSJ-Task von 4.1% auf 1.3%. Auf dem BN-Task sank sie von 2.1% auf 0.6%. Verblieben sind damit noch immer zirka 30% unbekannte Wörter, obwohl das Vokabular um mehr als das fünffache vergrößert wurde.

Verglichen wurde dieser Ansatz mit einem Baseline-System, das nur über das Basisvokabular verfügt und einem System, das nur eine einzige Klasse "UNKNOWN" anstelle der fünf Klassen verwendet.

Durch die Erweiterung des Vokabulars konnte die Wortfehlerrate auf dem WSJ-Task von 20.5% auf 17.7% gesenkt werden und zwar für beide Verfahren der Vokabularerweiterung, also sowohl mit einer Klasse als auch mit fünf Klassen. Das entspricht der Größenordnung, um die die OOV-Rate gesenkt werden konnte.

Aus Zeitgründen wurden auf dem BN-Task nur Systeme benutzt, die den Suchraum stark beschneiden. Das Ergebnis für das System mit den engen Beams ist überraschend, da hier die Fehlerrate von 44.2% auf 38.4% (= 5.8%) gesenkt werden konnte. Das entspricht fast dem vierfachen Wert, um den die OOV-Rate reduziert werden konnte. Dieses Ergebnis deutet darauf hin, dass hier noch ein anderer Effekt als nur die Reduktion der OOV-Rate auftritt.

Auf dem BN-Task ist sogar die Modellierung durch nur eine einzige Klasse im Sprachmodell geringfügig besser. Das lässt darauf schließen, dass das Sprachmodell nur

eine untergeordnete Bedeutung besitzt, wenn die benötigten Aussprachen im Vokabular enthalten sind. Diese Vermutung legt auch die Arbeit von Boulianne [8] nahe, in dessen Untersuchung, wie bereits erwähnt, gar keine Unterstützung durch das Sprachmodell verwendet wird.

Klassenbildung mittels Part-of-Speech-Informationen

Anstelle von semantischen Kategorien können Wörter auch anhand von grammatischen Informationen wie Part-of-Speech-Zuordnungen (POS) in Klassen eingeteilt werden.

Sammelsson [64] hat hierzu für den WSJ-Task zu einem Vokabular von 20k die Verteilungen der POS-Tags zu jedem Wort des Vokabulars bestimmt. Diese Statistiken werden benutzt, um die POS-Tags zu ordnen. Dabei werden alle POS-Tags hintereinander geschrieben bis 90% der Wahrscheinlichkeitsmasse abgedeckt ist oder aber maximal die ersten vier POS-Tags. Diese *geordneten* POS-Tag-Folgen ergeben die Klassen.

Das Hinzufügen von neuen Wörtern in diese Klassen soll sehr einfach sein, da lediglich die Statistik der POS-Tags für das unbekannte Wort bestimmt werden muss. Jedoch liegt die Perplexität mit dieser Art des Sprachmodells bei 570 und somit deutlich höher als im Vergleich mit einem wortbasierten Sprachmodell, das eine Perplexität von 187 besitzt. Durch die Interpolation der beiden Sprachmodelle konnte die Perplexität auf 179 gesenkt und dabei auch die Wortfehlerrate reduziert werden.

Geringfügig bessere Ergebnisse wurden erreicht, wenn man die Klassen in das Backoff-Schema einbezog, obwohl die Perplexität bei diesem Verfahren mit 180 geringfügig höher lag. Wie sich ein unbekanntes Wort verhält, das nur über das Backoff-Schema integriert wird, wurde von Sammelsson nicht untersucht.

Wegen der Interpolation mit einem wortbasierten Sprachmodell, das die besten Ergebnisse erzielte, wird aber auch weiterhin eine geeignete Möglichkeit der Integration des neuen Wortes in ein wortbasiertes Sprachmodell benötigt.

5.2 Indirekte Methode

Bei der indirekten Methode wird das Modell des Spracherkenners nicht erweitert. Es wird versucht, die Detektion eines OOV-Wortes anhand "indirekter" Beobachtungen durchzuführen, ohne den Erkenner zu verändern. Darum lautet eine der wesentlichen Fragen:

- An welchen Merkmalen kann erkannt werden, dass ein OOV-Wort benutzt wird?

5.2.1 Regelbasierte Detektion

In der Verbmobil-Domäne ist durch eine einfache Heuristik, bei der nach allen Hypothesen von "Herr", "Frau" oder "Doktor" ein Wort auftritt, dass nicht in einer Ausschlussliste eingetragen ist, darauf zu schließen, dass ein unbekannter Nachname genannt wurde. Dies

ist eine recht einfache Möglichkeit mit spezifischem Wissen über die Sprache und die Zieldomäne der Terminabsprache und Reiseplanung eine Detektion von unbekannten Wörtern durchzuführen oder eine existierende Detektion nochmals zu verbessern.

In der Tat wurde mit diesem Verfahren der erste Prototyp für das Projekt Verbmobil realisiert, der in der Lage war, unbekannte Nachnamen zu erkennen und zu lernen [78].

Der Nachteil dieses Verfahren liegt darin, dass diese Regeln gerade dann versagen, wenn durch unbekannte Wörter bedingte Folgefehler in den Nachbarworten auftreten, da die Triggerworte dann nicht mehr erscheinen. Ferner ist es notwendig, die Sprache und die Domäne *genau* zu kennen, um solche Regeln aufstellen zu können. Gallwitz hat diese Methode ebenfalls verwendet und in [31] genauer untersucht.

5.2.2 Konfidenzmaße

Konfidenzmaße oder Merkmale, die für die Bestimmung von Konfidenzmaßen benutzt werden, wurden oft zur Detektion von unbekannten Wörtern eingesetzt. Die zugrundeliegende Idee ist, dass ein unbekanntes Wort zu einem Erkennungsfehler führt und es darum möglich ist, beim Finden eines Erkennungsfehlers auch unbekannte Wörter zu detektieren. Diese Annahme ist zwar grundsätzlich nicht falsch, aber die meisten Spracherkenner machen aktuell die größte Anzahl an Fehlern bei bekannten Wörtern. So ist es nicht ungewöhnlich, dass die Wortfehlerrate nur zu 10% oder weniger durch unbekannte Wörter verursacht wird. Unter der Annahme, dass man jedes falsche Wort in einer Hypothese absolut korrekt identifizieren könnte, wäre die Präzision doch sehr gering.

Anwendbar ist ein Konfidenzmaß aber für den Fall, dass wirklich die Erkennergenauigkeit sehr hoch ist, wie beispielsweise bei der Erkennung von Ziffernfolgen oder anderen kleinen Vokabularien. Allerdings wird in Anwendungen, die über so kleine Vokabularien verfügen, bei einer geringen Konfidenz meist eine Ablehnung der Hypothese sinnvoller sein.

Der Vorteil, dass ein Konfidenzmaß für die Hypothese eines optimierten Spracherkenners bestimmt wird und es dadurch die Leistung des Spracherkenners nicht nachteilig beeinträchtigt, ist nur bedingt als Argument anzusehen, da es ja auch nicht möglich ist, weniger Fehler dadurch zu machen.

Eine umfangreiche Untersuchung wichtiger Merkmale für die Bestimmung von Konfidenzmaßen wurde in [67] und [68] durchgeführt. Dort wurde nicht nur die Korrelation der Merkmale zu Erkennungsfehlern sondern auch zu Erkennungsfehlern durch OOV-Wörter bestimmt. Diese Korrelationen sind im Anhang C angegeben und zeigen, dass nur wenige der Merkmale stärker mit dem Auftreten von OOV-Wörtern korrelieren als mit einem Erkennungsfehler im allgemeinen.

Auch Chase [10] untersuchte auf der Switchboard-Domäne und WSJ-Task Merkmale für Konfidenzmaße unter dem Aspekt, ob damit die Detektion von unbekannten Wörtern möglich ist und kam zu dem Schluss, dass die Konfidenzmaße ein "poor man's" Ansatz für die Detektion von unbekannten Wörtern darstellen. Ein einmal ohne OOV-Detektion

entwickeltes System muss nicht verändert werden, da die gleichen Modelle für die Erkennung verwendet werden und somit kein zusätzlicher Aufwand entsteht.

Hazen und Bazzi [41] untersuchten ebenfalls Konfidenzmaße für die Detektion von unbekannten Wörtern auf der Jupiter-Domäne und verglichen diese mit der Erweiterung des Spracherkenners um Modelle zu Detektion von OOV-Wörtern. Sie kamen ebenfalls zu dem Ergebnis, dass sich Konfidenzmaße nicht gut für die Detektion unbekannter Wörter eignen und im Gegenzug sich die Modelle für unbekannte Wörter nicht gut für Konfidenzmaße eignen. Durch die Kombination von Konfidenzmaßen und Erweiterung des Spracherkenners konnte aber die Qualität der Detektion, also die Präzision, verbessert werden.

Eine interessante Anwendung für ein Konfidenzmaß ist von Decadt und Duchauteau [17] veröffentlicht worden. Dort wird die Bewertung benutzt, um den Teil der Hypothese, der nur eine geringe Konfidenz hat, durch eine Phonemfolge zu ersetzen, die dann in eine lesbare Buchstabenfolge umgewandelt wird. Das primäre Ziel hierbei ist es, die Lesbarkeit des Textes zu verbessern und die kognitive Leistung des Lesers zu benutzen, um herauszufinden, was gesagt wurde.

Die Autoren haben zwar festgestellt, dass die Buchstabenfolge gerade für unbekannte Wörter nicht sehr gut ist, da es sich dabei oft um Fremdwörter handelt. Die Bereiche, in denen Erkennungsfehler auf bekannten Wörtern aufgetreten sind, wurden aber verständlicher.

5.3 Einfügen von neuen Wörtern in das Sprachmodell

Für das Einfügen von neuen Wörtern in ein bereits bestehendes Sprachmodell hat Jelinek et al in [47] einen Ansatz von Khazatsky untersucht und erweitert. Dabei fasst er jedes Wort als eine (eigene) Klasse auf und berechnet aufgrund vorhandener Kontextinformationen über ein neues Wort, welches Wort aus dem Vokabular des Sprachmodells einen ähnlichen Kontext modelliert (Synonym). Dabei wird der Ansatz von Khazatsky, der einen Entropy-basierten Ansatz benutzt hat, mit dem Maximum-Likelihood-Ansatz verglichen. Die neuen Wörter werden nicht in eine einzige Klasse sondern in eine Menge von Klassen aufgenommen. Die Wahrscheinlichkeit für das Auftreten eines Wortes ist dabei abhängig von der Zahl der bis dahin gelernten Wörter und der Häufigkeit ihres Auftretens. Weil ein gelerntes Wort in mehr als einer Klasse enthalten sein kann, wird für die Berechnung der Sprachmodellwahrscheinlichkeit, die Summe aus den Wahrscheinlichkeiten über alle Klassen berechnet, in denen das Wort eingefügt wurde. Um stabile Synonyme für ein unbekanntes Wort zu finden, reichen bereits zehn bis zwanzig Textbeispiele.

5.4 Zusammenfassung

Ein wesentlicher Schwerpunkt bei der Modellierung unbekannter Wörter ist es einerseits eine akustische Beschreibung zu finden, die eine im Prinzip unendlich große Menge an Wörtern modelliert, andererseits sollen diese Modelle aber nicht gut auf die Wörter des bekannten Vokabulars passen, um Verwechslungen zu vermeiden.

Der Ansatz der "flachen Neue-Worte-Modelle" verfolgt dies durch akustische Modelle, die auf vielen verschiedenen Phonemen trainiert werden.

Ein Vorteil dieses Verfahrens besteht darin, dass nur eine geringe Anzahl zusätzlicher Zustände für die Modellierung aufgenommen werden muss und diese akustischen Modelle im Mittel sehr viel schlechter auf die bekannten Wörter passen, da die akustischen Modelle für bekannte Wörter schärfer sind. Da auch OOV-Wörter im Mittel eine schlechte Bewertung erhalten, kann es leicht dazu kommen, dass ein bekanntes Wort, das nur teilweise gut passt, im Mittel eine günstigere Bewertung erhält. Somit besteht die Gefahr, dass diese Modelle ganz aus dem Suchraum entfernt werden, bevor sie zum Zuge kommen oder leicht mit einem hinreichend ähnlichen bekannten Wort verwechselt werden können.

Die Verwendung einer Phonemschleife erlaubt eine genauere akustische Modellierung, besitzt aber den Nachteil, dass für die Modellierung eine große Zahl zusätzlicher Zustände benötigt wird und diese Modelle auch sehr gut auf bekannte Wörter passen. Das trifft insbesondere zu, wenn Modelle verwendet werden, die die gleich Genauigkeit wie die bekannten Wörter aufweisen. Aus diesem Grunde werden meistens für die Phonemschleife weniger genaue akustische Modelle verwendet.

Das hat zwar den Vorteil, dass weniger zusätzliche Zustände benötigt werden, aber diese akustischen Modelle extra trainiert und bei der Suche auch zusätzlich ausgewertet werden müssen, da die Phonemschleife wegen ihrer Rückwärtssprünge nicht zusammen mit den bekannten Wörtern integriert werden kann. Durch viele zusätzliche Zustände mit unterschiedlichen Modellen, die aktiv sein können, kommt es vor, dass der mittlere Aufwand während der Suche deutlich stärker ansteigen kann als bei der flachen akustischen Modellierung.

Eine weitere Methode, den Abstand einer Phonemschleife zum bekannten Vokabular zu vergrößern, ist es, Wahrscheinlichkeiten für Phonemübergänge zu verwenden, die auf großen Hintergrundwörterbüchern trainiert sind.

Die Modellierung von Phonemschleifen durch das Aufheben der Modellgrenzen zwischen den Wörtern und kleineren Worteinheiten hat in den bisher untersuchten Systemen den Vorteil, dass keine Modifikation der verwendeten Decoder notwendig war, aber durch die Verwendung von N-Gramm-Sprachmodellen dieser Kontext, in dem ein unbekanntes Wort auftritt, nicht mehr genau modelliert werden kann. Der Kontext ist aber wichtig, um eine präzise Vorhersage über die möglichen Orte zu machen, an denen ein unbekanntes Wort auftreten kann.

Das Sprachmodell hat einen Einfluß auf die Qualität der Vorhersage wo unbekannte

Wörter auftreten können. Häufig wird nur eine einzige Klasse zur Modellierung unbekannter Wörter verwendet. Genauere Vorhersagen sind aber durch die Verwendung von Klassensystemen möglich.

Dieses Klassensystem wird meist durch Experten festgelegt, indem sie eine Menge semantischer Klassen definieren, die (offensichtlich) ein offenes Vokabular besitzen. Diese Klassendefinitionen sind nicht notwendigerweise geeignet, um auch gute Klassen in einem Sprachmodell zu ergeben, erlauben aber dafür eine Interpretation des unbekannten Wortes. Ein gut definiertes Klassensystem benötigt eine exakte Kenntnis der untersuchten Domäne und häufig auch eine manuelle Annotation der Trainingsdaten, was die Verwendung großer Datenmenge unmöglich macht.

Ziel sollte es also sein, eine hinreichend genaue akustische Modellierung unbekannter Wörter zu entwickeln, die nicht gut auf die bekannten Wörter des Vokabulars passt. Dabei soll sie effizient in bestehende Spracherkennerungstechnologie integriert werden können, ohne dabei die Anzahl der notwendigen Zustände deutlich zu erhöhen oder die mittlere Laufzeit negativ zu beeinträchtigen. Weiter ist durch geeignete Klassensysteme in den Sprachmodellen die Detektion zu unterstützen um präzisere Vorhersagen über das Auftreten unbekannter Wörter machen zu können. Dabei sollte ein Verfahren diese Klassensysteme automatisch für die jeweilige Domäne erstellen um somit insbesondere die Bearbeitung großer komplexer Domänen mit umfangreichen Textkorpora zu ermöglichen.

Kapitel 6

Neu entwickelte Verfahren zur Modellierung unbekannter Wörter

Um eine optimale Modellierung der OOV-Wörter zu erreichen, sind alle relevanten Modelle des Spracherkenners für diese Aufgabe zu erweitern. Hierzu gehört zunächst das akustische Modell, das bei Verwendung eines OOV-Wortes die Möglichkeit besitzen muss, diesen akustischen Abschnitt einer Äußerung nicht auf ein bekanntes Wort abzubilden, sondern auf ein *generisches* Wort, d.h. ein Wort, das für viele Wörter steht. Hierbei ist es wichtig, dass die Modellierung der generischen Wörter genau genug ist, um mit den Wörtern des Vokabulars konkurrieren zu können, dass sich aber die generischen Wörter von den Wörtern hinreichend des Vokabulars unterscheiden, um zu verhindern, dass diese Wörter an Stelle von bekannten Wörtern ausgegeben werden.

Geht man bei der Modellierung so vor, dass das Auftreten eines generischen Wortes unabhängig von den umgebenden Wörter ist, spricht man in diesem Fall auch von sogenannten "Filler-Modellen", da sie keine Unterstützung durch das a priori-Wissen des Sprachmodells erfahren und nur "Lücken" in den akustischen Daten auffüllen. Das a-priori-Wissen des Sprachmodells stellt aber in der Spracherkennung eine wichtige Informationsquelle dar, die beispielsweise hilft, bei ähnlich bewerteten Wörtern anhand der möglichen Kontexte sich für das a-priori höher wahrscheinliche Wort zu entscheiden. Das Auftreten von OOV-Wörtern ist aber nicht in jedem Kontext gleichwahrscheinlich, weshalb das Sprachmodell ebenfalls für die Modellierung unbekannter Wörter erweitert werden muss. Da die OOV-Wörter eine Menge an Wörtern darstellen, werden diese im Sprachmodell üblicherweise durch eine einzige Klasse für OOV-Wörter modelliert. Dies bietet den Vorteil, dass die OOV-Klasse nicht nur die Kontexte eines bestimmten OOV-Wortes kennt, und es besteht die Hoffnung, dass zukünftige unbekannte Wörter im gleichen Kontext auftreten werden.

Da die Verwendung von Klassensystemen dabei die Genauigkeit der Modellierung im Vergleich zu nur einer einzigen Klasse erhöhen kann, ist es Ziel dieser Arbeit, ein Klassensystem für die Modellierung unbekannter Wörter automatisch zu erzeugen.

Um dieses System in den Experimenten von Kapitel 7 mit anderen Verfahren zu vergleichen, wird ebenfalls beschrieben, wie mit diesen Verfahren statistische Sprachmodelle erzeugt werden.

Abschließend wird die akustische Modellierung für unbekannte Wörter mit dem für diese Arbeit neu entwickelten Verfahren der optimierten Head-Tail-Modelle vorgestellt und analysiert.

6.1 Beschreibung der untersuchten Sprachmodelle

Die Untersuchungen in dieser Arbeit wurden mit den Daten der deutschen GSST-Domäne (Verbmobil-Daten) und der englischen BN-Domäne (HUB-4) durchgeführt, die in Kapitel 3 beschrieben sind.

Das Sprachmodell für die deutsche GSST-Domäne wird in Kapitel 7.1 als semantisches Klassensprachmodell für die Detektion unbekannter Wörter benutzt. Bei den Sprachmodellen, die in der englischen BN-Domäne benutzt werden, handelt es sich um Sprachmodelle, die Klassen für die Modellierung von unbekannten Wörtern besitzen. Es werden vier Ansätze vorgestellt, mit denen man Wörter auf Klassen verteilen kann, die wegen des großen Datenaufkommens im BN-Korpus alle eine automatische Verarbeitung des Korpus voraussetzen. Bei dem neu entwickelten Verwahren wird *ohne* explizites Domänenwissen ein Klassensystem zur Modellierung von unbekannten Wörtern konstruiert.

6.1.1 Das Sprachmodell für die GSST-Domäne

Für dieses Sprachmodell wurden die Wörter unter Verwendungen von Wortlisten in das Klassensystem aus [66] eingeteilt. Für die Modellierung unbekannter Wörter wurde dann die Klasse "<UNK:Surname>" ausgewählt, die Nachnamen modelliert, da diese Klasse den größten Anteil an OOV-Wörtern in der GSST-Domäne darstellt. Da die Verwendung von Tabellen aufgrund ambiger Wörter ungenau ist, wurden die häufigen Tri-Gramme manuell auf Plausibilität überprüft und korrigiert, wenn die Klasseneinteilung nicht zutreffend war. Genauso wurde auch mit den Klassen "<UNK:Town>" und "<UNK:Street>" verfahren, die aber in den späteren Experimenten nicht verwendet wurden, da zu wenig OOV-Wörter dieser Klassen für die Untersuchungen vorlagen. Diese Klassen wurden aber im Verbmobil-Prototypen benutzt, um zu demonstrieren, dass die Detektion mehrerer Klassen möglich ist.

6.1.2 Das BN-Baseline-Sprachmodell

Für das Sprachmodell des englischen Baseline-Erkenners wurde der BN-Sprachmodellkorpus nicht speziell aufbereitet. Das Vokabular zur Erstellung des Sprachmodells umfasst 39734 Wörter (V_{Top40k}), die nach dem Häufigkeitskriterium ausgewählt wurden. Unbekannte Wörter des Korpus werden auf das Symbol "<UNK>" abgebildet. Ihre Verteilung oder ihre Anzahl wird aber nicht bestimmt. Weil das

Sprachmodell ohne die Verwendung von Mindesthäufigkeiten für Tri- und Bigramme sehr groß werden würde, wird die Mindesthäufigkeit bei 3-Grammen auf eins und bei 2-Grammen auf zwei gesetzt. Dies ist die Standardeinstellung, die bei allen erzeugten Sprachmodellen verwendet wird. Mit diesem Korpus und diesen Einstellungen wurde ein 3-Gramm-Sprachmodell erzeugt, das als BASELINE-LM bezeichnet wird.

6.1.3 Einklassensprachmodell

Um ein Sprachmodell mit einer gemeinsamen Klasse für alle unbekannten Wörter zu erzeugen, wurden alle Wörter des BN-Korpus, die nicht im Vokabular V_{Top40k} liegen, auf das Klassensymbol "<OOV>" abgebildet. Über die abgebildeten Wörter wurde zusätzlich eine Statistik erstellt, die aus einer Liste der Wörter mit ihren Häufigkeiten besteht. Aus dieser Statistik kann die klassenbedingte Wahrscheinlichkeit $p(w_{\text{UNK}}|\text{<OOV>})$ eines unbekannten Wortes w_{UNK} geschätzt werden.

Die mittlere Wahrscheinlichkeit eines unbekannten Wortes kann durch Formel 6.1 angenähert werden.

$$p(w_{\text{UNK}}|\text{<OOV>}) = \frac{1}{M} \tag{6.1}$$

Dabei bezeichnet M die Anzahl verschiedener Wörter, die auf "<OOV>" abgebildet wurden. Die Menge aller verschiedenen Wörter, die im Textkorpus K aufgetreten sind, wird mit wordsOf(K) bezeichnet. Die OOV-Wörter eines Korpus K bezüglich eines Vokabulars V lassen sich damit als $\text{OOV} = \{w|w \in \text{wordsOf}(K) \wedge w \notin V\}$ beschreiben und es gilt die Beziehung $M = |\text{OOV}|$.

6.1.4 Das Named-Entity-Sprachmodell

Für die Erstellung eines Sprachmodells, das über die semantischen Klassen "PERSON", "ORGANISATION" und "LOCATION" sowie die allgemeine Restklasse "REST" verfügt, müsste eigentlich jedem (OOV-)Wort des gesamten BN-Korpus manuell ein Tag angehängt werden, das die Klasse des Wortes in seinem konkret verwendeten Kontext bezeichnet. Da dies aber bei einem so umfangreichen Korpus (>140 Mio. Wörter) so arbeitsaufwendig ist, dass eine manuelle Bearbeitung praktisch unmöglich ist, wurde ein NE-Tagger eingesetzt, der von Fei Huang zur Verfügung gestellt wurde [44].

Der NE-Tagger kann, wie in Kapitel 5.1.4 erklärt, die sieben Klassen "PERSON", "LOCATION", "ORGANISATION", "TIME", "DATE", "MONEY" und "PERCENT" des NE-Task annotieren. Miller und Schwartz haben in [56] festgestellt, dass ab einem Vokabular, das die häufigsten 40581 Wörtern enthält, in den Klassen "TIME", "DATE", "MO-

NEY" und "PERCENT" keine unbekannten Wörter mehr auftreten und sie sich ab dieser Vokabulargröße vollständig formulieren lassen. Die Klassen "PERSON", "LOCATION" und "ORGANISATION" hingegen lassen sich aber auch mit sehr großem Vokabular nicht vollständig abdecken und können somit als offene Klassen angesehen werden.

Da wir bei unseren Untersuchungen ein vergleichbar großes Vokabular V_{Top40k} verwenden, sind die anderen Klassen nicht relevant für die Modellierung von unbekannten Wörtern. Vielmehr sollte aufgrund dieser Beobachtung kein kleineres Vokabular als V_{Top40k} benutzt werden.

Der verwendete NE-Tagger ist auf den manuell annotierten Trainingsdaten für den NE-Task des NIST trainiert worden. Er basiert auf Hidden-Markov-Modellen und entspricht der Arbeitsweise des von BBN entwickelten NE-Taggers "IdentiFinder" [7]. Er arbeitet auch auf den BN-Daten im SNOR-Format[1] [55], in dem es weder Interpunktion noch Groß- und Kleinschreibung gibt. Dabei werden vergleichbar gute Ergebnisse wie die des "IdentiFinder" von BBN erzielt.

Beim NE-Task geht es darum, aus einem Text Namen als Ganzes zu identifizieren. Dabei kann eine Named-Entity auch aus mehreren Wörtern bestehen, die durch die SGML-Tags[2] mit <b_enamex> vor dem Entity-Namen und mit <e_namex> nach dem Entity-Namen markiert sind. Ein SGML-Attribut mit dem Namen "TYPE" spezifiziert die Klasse dann genauer. Eine ausführliche Beschreibung der Syntax ist in [11] zu finden. Beispielsweise enthält der Satz "MY NAME IS THOMAS SCHAAF" die Named-Entity "THOMAS SCHAAF", der durch den NE-Tagger korrekt mit "MY NAME IS <b_enamex TYPE="PERSON"> THOMAS SCHAAF <e_namex>" als Personenname annotiert würde.

Beim Vorbereiten des Textes zum Erzeugen des statistischen Sprachmodells werden alle unbekannten Wörter durch ein Symbol ersetzt, das der zugewiesenen NE-Klasse entspricht. Sei im Beispiel nun nur das Wort "SCHAAF" im Vokabular unbekannt, so ergebe dies den Satz "MY NAME IS THOMAS <OOV:PERSON>". Dabei stellt "<OOV:PERSON> das Symbol für die Klasse "PERSON" dar, die nicht im Vokabular enthalten ist. Es werden nur die unbekannten Wörter durch ein Klassentag ersetzt, nicht aber die im Vokabular enthaltenen Wörter. Das ist sinnvoll, da sich die durch den NE-Task vorgegebenen Klassen nur eingeschränkt für eine Modellierung in einem statistischen Sprachmodell eignen. Durch die geringe Zahl an Klassen werden die grammatikalischen Strukturen kaum berücksichtigt, die (unbekannte) Wörter besitzen. Auch die Semantik ist sehr stark zusammengefasst. So ist im Deutschen ein Straßenname, ein Städtename oder ein Land als "LOCATION" anzusehen. Beispielsweise heißt es "man trifft sich *in der* XYZ-Straße", aber nicht "man trifft sich *in der* XYZ-Platz". Die korrekte grammatische Form hierfür lautet "man trifft sich *auf dem* XYZ-Platz", was für alle Plätze aber wiederum nicht

[1] Speech Normalized Orthograpic Representation
[2] *Standard* Generalized *Markup* Language

für Straßen gilt. Solche Feinheiten des Kontextes gehen beim Zusammenfassen zu einer einzigen Klasse verloren. Insbesondere trifft dies auf die Klasse "REST" zu, in der alle verbleibenden unbekannten Wörter zusammengefasst werden.

Eine größere Zahl an Klassen, die die Semantik der Wörter feiner modellieren und die auch grammatische Eigenschaften in Betracht ziehen würde, wäre wünschenswert, da dies bei einem statistischen Sprachmodell präzisere Vorhersagen erlaubt. Feinere Klassensysteme bezogen auf die semantischen Kategorien sind beispielsweise von Sheremeteyeva [71] gegeben, die auch einen hierarchischen Aufbau besitzen, wie beispielsweise auch das domänenspezifische Klassensystem für Verbmobil [66]. Für die Erstellung eines BN-Sprachmodells braucht man aber automatische Werkzeuge zum Annotieren, die für diese Klassensysteme fehlen, da ausreichende Mengen an manuell annotierten Daten nicht vorliegen.

Eine Schwierigkeit, die ein NE-Tagger zu meistern hat, sind unbekannte oder seltene Wörter, das heißt Wörter, die nicht in seinem Vokabular enthalten sind oder auf denen er nur unzureichend trainiert werden konnten. Der Grad der Schwierigkeit hängt natürlich mit der Menge des verfügbaren Trainingsmaterials zusammen. Die Menge an verfügbarem Trainingsmaterial ist wegen der großen Anzahl seltener Wörter und dem enormen Aufwand für die manuelle Annotation wohl stets als zu klein anzusehen. Darum müssen Glättungstechniken eingesetzt werden, um auch Wahrscheinlichkeiten für auf den Trainingsdaten ungesehene Ereignisse liefern zu können. Gerade bei den unbekannten Wörtern, die ja besonders selten sind, lässt die Qualität der Einteilung in Klassen nach, und es besteht die Tendenz, diese Wörter der Klasse "PERSON" zuzuordnen.

Der große Vorteil, den eine Einteilung der unbekannten Wörter durch einen NE-Tagger bietet ist, dass die Wörter abhängig vom benutzten semantischen Zusammenhang eingeteilt werden. Wenn beispielsweise der Satz

"THE SCHAAF CORPORATION INVENTED A NEW TECHNOLOGIE" annotiert würde, ergibt sich korrekterweise

"THE <b_enamex TYPE="ORGANISATION"> SCHAAF CORPORATION <e_enamex> INVENTED A NEW TECHNOLOGIE",

und dementsprechend der Satz

"THE <OOV:ORGANISATION> CORPORATION INVENTED A NEW TECHNOLOGIE"

für das Training des statistischen Sprachmodells. Im Gegensatz zum obigen Beispiel, in dem "SCHAAF" eine Person bezeichnet, wurde nun "SCHAAF" durch ein anderes Klassensymbol ersetzt.

Dies erlaubt insbesondere die Auflösung von ambigen Wörtern in der Hoffnung, dass ein statistisches Sprachmodell unbekannte Wörter aus diesen Klassen dann besser vorhersagen kann, weil die möglichen Kontexte im Modell präziser modelliert sind, was aber aufgrund der automatischen Annotation nur eingeschränkt gilt.

Nach dem oben beschriebenen Verfahren wurde der gesamte BN-Korpus mit dem NE-Tagger annotiert und alle $w \notin V$ (OOV-Wörter) durch die Symbole "<OOV:PERSON>", "<OOV:ORGANISATION>", "<OOV:LOKATION>" und "<OOV:REST>" ersetzt. Die ersetzten Wörter wurden gezählt, um eine Statistik für die Schätzung der klassenbedingte Wahrscheinlichkeit $p(w|\text{<OOV:CLASS>})$ für die betrachteten Klassen zu erhalten.

Für das Erstellen des Sprachmodells wurden die Symbole, die die Klassen repräsentieren, in das verwendete Vokabular V aufgenommen und ein statistisches Sprachmodell mit den gleichen Werten für die Parameter bei der Mindesthäufigkeit erstellt, wie für das BASELINE-Sprachmodell. Das Sprachmodell wird in den Experimenten mit NE-LM bezeichnet.

6.1.5 Sprachmodell mit unüberwacht gefundenen Klassen - Gruppierungsverfahren für unbekannte Wörter (GrOOV)

Um auf automatischem Wege aus einem Sprachmodellkorpus eine für eine Domäne optimierte Menge an Klassen für die Modellierung unbekannter Wörter zu erhalten, die eine feinere Modellierung der Eigenschaften dieser Wörter erlauben und zusätzlich gut für statistische Sprachmodelle geeignet sind, wurde in dieser Arbeit ein neuer Ansatz entwickelt:

Die zugrundeliegende Idee ist, eine *kleine* Menge an Repräsentanten von unbekannten Wörtern zu beobachten, die durch ein geeignetes datengetriebenes Clustern zusammen mit den Wörtern aus dem Vokabular V in Gruppen geballt werden, um anschließend eine kleine Anzahl Klassen zur Modellierung unbekannter Wörter auszuwählen.

Ein Grund, warum das Clustern zusammen mit den Wörtern aus dem Vokabular V durchgeführt wird ist, dass normalerweise auch diese Menge Wörter enthält, die nur zufällig häufig genug waren, um in das Vokabular aufgenommen zu werden. Insbesondere Klassen wie beispielsweise Namen umfassen sowohl seltene als auch häufigere Wörter (beispielsweise die Namen "Schaaf" und "Schröder"), die gemeinsame Eigenschaften aufweisen, so dass sie in einer Klasse zusammengefasst werden könnten.

Ein weiterer Grund ist, dass die Bildung von Klassen auch für die häufigen Wörter des Vokabulars bei der geringen Menge an statistischem Material für ein *einzelnes* der seltenen Wörter helfen kann, Gemeinsamkeiten bei seltenen Wörtern aufzuzeigen. Beispielsweise kommen Nachnamen im Deutschen häufig in Verbindung mit "Herr" oder "Frau" vor. Kommen Nachnamen nun aufgrund ihres seltenen Auftretens nur im Zusammenhang mit "Herr"(oder nur mit "Frau") vor, könnte es sein, dass eine eigene Klasse mit diesen geschlechtsspezifischen Nachnamen gebildet würde. Da im Deutschen Nachnamen nicht an ein Geschlecht gebunden sind, würde sich diese Klasse unter Umständen überspezialisieren.

Enthält das Vokabular V bereits alle Wörter des Korpus K, so kann für das hier beschriebene Verfahren eine Menge an potentiell unbekannten Wörtern durch das Heraufsetzen der Mindesthäufigkeit oder auch durch das iterative Substitutionsverfahren von Fetter [22] bestimmt werden. Hierbei können auch Wörter zu potentiell unbekannten Wörtern (pUNK-Wörter) werden, die nur lokal gehäuft auftreten.

Für die hier beschriebenen Untersuchungen wurde der BN-Korpus mit dem Vokabular V_{Top40k} benutzt. Die Wörter, die seltener als 31 mal aufgetreten sind, sind somit nicht im Vokabular V_{Top40k} enthalten. Daraus ergibt sich, dass Wörter aus dem Korpus, die nicht zum Vokabular gehören, bis zu 30 mal auftreten können. Für die Klasseneinteilung ist es auch hilfreich, wenn genügend Beispiele zu den Wörtern vorliegen. Eine Mindesthäufigkeit von zehn ist dabei erfahrungsgemäß ausreichend aber nicht notwendig. Die Anzahl der OOV-Wörter, die mindestens zehn mal und maximal 30 mal aufgetreten sind, beträgt 24124 und stellt eine ausreichend große Menge an potentiell unbekannten Wörtern dar, so dass auf eine Anwendung des iterativen Substitutionsverfahrens von Fetter in dieser Arbeit verzichtete wurde.

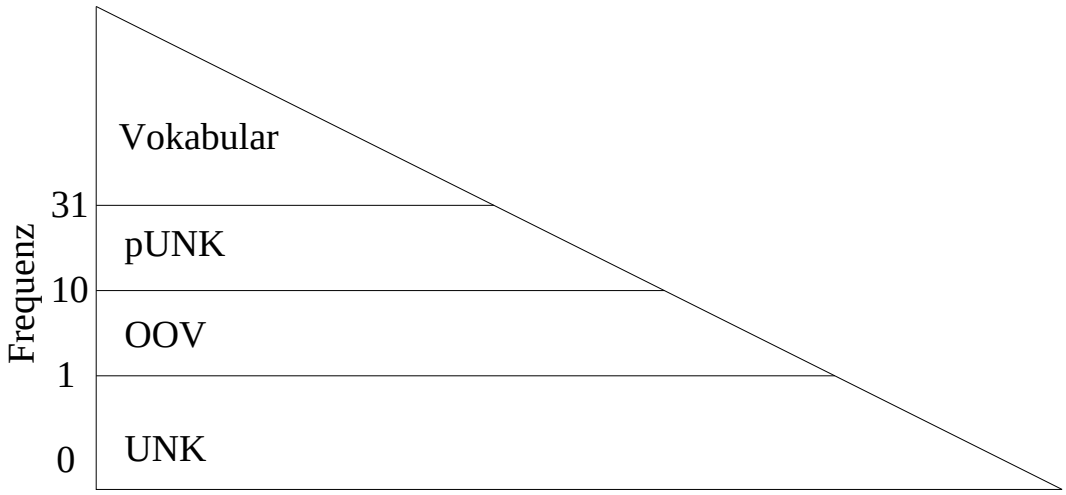

Abbildung 6.1: Einteilung der Wörter für die Klassenfindung

Das Abbildung 6.1 zeigt eine mögliche Einteilung der Wörter eines Korpus anhand der Häufigkeit ihres Auftretens. Dabei wurden als Werte für die Unterteilung zwischen Vokabular, pUNK-Wörtern und den restlichen OOV-Wörtern die Werte eingetragen, wie sie für den BN-Task benutzt wurden. Wörter, die auf dem Korpus seltener als einmal, also niemals aufgetreten sind, werden dabei in der Menge "UNK" zusammengefasst, da sie, im Gegensatz zu den potentiell unbekannten Wörtern, die Menge der völlig unbekannten Wörter darstellen.

Der verwendete Cluster-Algorithmus

Welches Wort zu einer bestimmten Klasse gehört, wird durch die Funktion $\phi : V \mapsto C$ festgelegt, die Wörter aus dem Vokabular V auf Klassen aus $C = \{g_1 ... g_N\}$ abbildet. Ein statistisches Sprachmodell, das mit einem guten Klassensystem ϕ erzeugt wurde, besitzt für beliebige (domänenabhängige) Texte eine geringe Perplexität, oder anders formuliert, der Text würde mit einer hohen Wahrscheinlichkeit zufällig erzeugt. Werden für die Schätzung der Wahrscheinlichkeiten nur Bigramme verwendet, führt dies zur Likelihood-Funktion L (Gleichung 6.2), die es zu maximieren gilt. Dabei bezeichnet p_0 die klassenbedingte Wahrscheinlichkeit und p_1 die Wahrscheinlichkeit für das Aufeinanderfolgen von Klassen. Wenn aus dem Kontext klar ist, welche Wahrscheinlichkeiten gemeint sind, werden die Indizes zum einfacheren Lesen weggelassen.

$$L = \prod_{i=1}^{N} p_\phi(w_i|w_{i-1}) = \prod_{i=1}^{N} p_0(w_i|\phi(w_i)) p_1(\phi(w_i)|\phi(w_{i-1})) \tag{6.2}$$

Es wird ein Maximum-Likelihood-Ansatz für die Bestimmung der benötigten Wahrscheinlichkeiten p_ϕ benutzt, bei dem die relativen Häufigkeiten der Wortfolgen beziehungsweise Klassenfolgen auf dem Korpus $K = (w_1 w_2 ... w_N)$ zur Schätzung verwendet werden. Wird sowohl die Bestimmung der Likelihood-Funktion L als auch die Schätzung der Wahrscheinlichkeiten p_ϕ auf dem gleichen Korpus K durchgeführt, folgt daraus, dass es keine ungesehenen Ereignisse, das heißt Wort- beziehungsweise Klassenfolgen gibt. Ferner gilt, je mehr Freiheitsgrade, das heißt Klassen in $C = \{g_1 ... g_N\}$ verwendet werden, desto besser kann der Wert für L ausfallen. Den optimalen Wert für L würde man erhalten, wenn man genauso viele Klassen wie Wörter im Vokabular verwendet. Dieser Fall stellt jedoch ein Problem dar, da die Anzahl an Klassen ein wichtiger Parameter ist, den es ebenfalls zu optimieren gilt. Schließlich sollen auch die Klassen eine Generalisierung für die enthaltenen Wörter darstellen. Die geeignete Anzahl an Klassen müsste in diesem Fall auf einer Kreuzvalidierungsmenge K_H bestimmt werden, indem man die Funktion ϕ mit der Vorgabe für verschieden viele Klassen bestimmt, und jeweils L auf K_H berechnet wird.

Dieser Ansatz hat mehrere Nachteile. Zum einen kann er zu einer Fragmentierung der verfügbaren Daten führen, zum anderen liegen die gefundenen Abbildungen ϕ meist in einem lokalen Optimum, das vom Startpunkt des verwendeten Verfahrens abhängt, wodurch es erschwert wird, eine geeignete Anzahl an Klassen festzustellen. Ferner ist auf der Kreuzvalidierungsmenge mit dem in der Spracherkennung sehr wahrscheinlichen Fall zu rechnen, dass auch ungesehene Klassenfolgen auftreten können, weshalb die Wahrscheinlichkeiten für die Berechnung von L auf K_H angepasst werden müssen.

Das Cluster-Verfahren von Kneser und Ney [53] überwindet die beschriebenen Nachteile, indem es eine Kreuzvalidierungsmenge nach einem "Leaving-One-Out"-Schema in die zu optimierende Likelihood-Funktion integriert und nun sowohl ungesehene Ereig-

nisse berücksichtigt als auch die Optimierung der Anzahl der verwendeten Klassen erlaubt.

Die neue Likelihood-Funktion L_{LO} (Gleichung 6.3) benutzt nun die Wahrscheinlichkeiten $p_{\phi,K_i}(w_i|w_{i-1})$.

$$L_{\mathrm{LO}} = \prod_{i=1}^{N} p_{\phi,K_i}(w_i|w_{i-1}) = \prod_{i=1}^{N} p_{K_i}(w_i|\phi(w_i))p_{K_i}(\phi(w_i)|\phi(w_{i-1})) \tag{6.3}$$

Hierbei bedeutet der Index i bei K_i, dass die Wahrscheinlichkeit für das Auftreten der Wortfolge $(w_{i-1}w_i)$ auf dem Korpus K ohne das Ereignis $(w_{i-1}w_i)$ geschätzt wird. Das hat zur Folge, dass nun Ereignisse, die im durch ϕ abgebildeten Korpus genau einmal auftreten, nach dem Maximum-Likelihood-Ansatz mit der Wahrscheinlichkeit 0 geschätzt würden, und die Likelihood-Funktion L_{LO} nicht optimiert werden kann. Für ungesehene Ereignisse wird darum ein Teil der Wahrscheinlichkeitsmasse reserviert, der über ein Backing-Off-Schema nach Formel 6.4 bestimmt wird (absolute discounting).

$$p_{\phi,K}(g_1,g_2) = \begin{cases} \frac{N_K(g_1,g_2)-b}{N_K} & : \quad N_K(g_1,g_2) > 0 \\ \frac{n_{+,K}b}{n_{0,K}N_K} & : \quad N_K(g_1,g_2) = 0 \end{cases} \tag{6.4}$$

Dabei bezeichnen g_1 und g_2 zwei der Klassen, in die die Wörter des Vokabulars eingeteilt werden und b ist der Wert für das absolute discounting, der reserviert wird, um ungesehene Ereignisse durch das Backing-Off-Schema zu modellieren. Mit $n_{+,K}$ ist die Anzahl der gesehenen Klassenbigramme auf dem gesamten Korpus K und mit $n_{0,K}$ die Anzahl der ungesehenen Klassenbigramme bezeichnet. $N_K(.,.)$ gibt die Häufigkeiten eines Ereignisses an. Formt man nun die zu maximierende Likelihood-Funktion für L_{LO} um, und lässt alle von ϕ unabhängigen Terme weg, so ergibt sich daraus das zu maximierende Kriterium F_{LO} (Gleichung 6.5).

$$F_{\mathrm{LO}} = \sum_{g_1,g_2} N(g_1,g_2)\log\left[N(g_1,g_2)-1-b\right] + n_1 \log\left[\frac{(n_+-1)\,b}{n_0+1}\right] - 2\sum_{g} N(g)\log\left[N(g)-1\right] \tag{6.5}$$

Dabei bezeichnet n_1 die Anzahl der Klassenbigramme, die genau einmal aufgetreten sind.

Um F_{LO} zu optimieren, verschiebt der von Kneser und Ney entwickelte Algorithmus die Wörter w aus dem Vokabular V zwischen den Klassen $C = \{g_1 ... g_N\}$ so, dass F_{LO} monoton wächst. Wegen der Komplexität wird das Verschieben mit einer Greedy-Strategie durchgeführt und es ist nicht garantiert, ein globales Optimum zu finden. Dennoch liefert das Verfahren gute Klassensysteme, die gut für statistische Sprachmodelle geeignet sind.

Verteilung der Klassengrößen

Formt man die zu optimierende Funktion F_{LO} um in

$$\begin{aligned}
\hat{F}_{\text{LO}} &= \frac{n_1}{N_K} \log \left[\frac{(n_+ - 1)\, b}{(n_0 + 1) N_K} \right] \\
&+ \sum_{g_1, g_2} p_K(g_1, g_2) \log \underbrace{\left[\frac{N(g_1, g_2) - 1 - b)}{N_K} \right]}_{\approx p_K(g_1, g_2)} \\
&- 2 \sum_{g} p_K(g) \log \underbrace{\left[\frac{N(g) - 1}{N_K} \right]}_{\approx p_K(g)} \qquad (6.6) \\
&\approx \frac{n_1}{N_K} \log \left[\frac{(n_+ - 1)\, b}{(n_0 + 1) N_K} \right] - H(p_K(g_1, g_2)) + 2H(p_K(g)) \qquad (6.7)
\end{aligned}$$

ist zu erkennen, dass beim Maximieren der Funktion annäherungsweise die Entropie $H(p(g_1, g_2))$ minimiert und die Entropie $H(p(g))$ über die Verteilung der Klassenwahrscheinlichkeiten maximiert wird. Dies gilt, wenn hinreichend große Mengen an Daten zur Verfügung stehen, da dann die geschätzten Wahrscheinlichkeiten auch den Wahrscheinlichkeiten auf der Kreuzvalidierungsmenge entsprechen.

Durch das separate Ausweisen des Anteils für die im Korpus K einmal aufgetretenen Klassenbigramme fehlt dieser im Term $H(p(g_1, g_2))$. Der Term für diese auf K_i ungesehenen Klassenbigramme ist der gewichtete Logarithmus einer Wahrscheinlichkeit und ist damit niemals positiv; sein Maximum ist somit null. Dieses Maximum kann durch die Funktion ϕ dadurch erreicht werden, dass die Zahl der nur einmal gesehenen Klassenbigramme n_1 minimiert, oder die Anzahl der ungesehenen Klassenbigramme n_0 minimiert wird, da für n_1 der gesamte Term zu null wird. Der Wert von n_+ wird groß, wenn der Wert für n_0 klein wird, der Logarithmus aus Formel 6.7 wird damit maximal, wenn n_0 minimal ist. Dies tritt beispielsweise dann ein, wenn die Zahl der möglichen Klassen klein gehalten wird. Es ergibt sich dann automatisch, dass n_0 und n_1 klein sind. Dieser Term bildet einen wesentlichen Unterschied zwischen der Liklihood-Funktion L und L_{LO}, was einer der Gründe dafür ist, dass auch die Optimierung der Klassenanzahl möglich ist.

Der Term $H(p(g_1, g_2))$ kann durch die Funktion ϕ minimiert werden, indem er die Wahrscheinlichkeitsmasse auf wenige Klassenbigramme konzentriert. Dies kann beispielsweise ebenfalls dadurch erreicht werden, dass die Zahl der Klassen gering gehalten wird.

Dagegen ist der zu maximierende Term $H(p(g))$ dann am größten, wenn die Wahrscheinlichkeitsmasse über alle Klassen möglichst gleichverteilt ist und dazu noch möglichst viele Klassen verwendet werden. Dieser Term ist somit mitverantwortlich, dass es eine

Tendenz gibt, die Summe der Häufigkeiten der Klassenelemente über die Klassen hinweg anzugleichen.

Das führt dazu, dass gerade die seltenen Wörter in wenigen Klassen zusammengefasst werden. Besonders häufige Wörter lassen sich dabei nicht so einfach verschieben, da dies einen größeren Einfluss auf die angestrebte Gleichverteilung der Klassenhäufigkeiten hat.

Für die Analyse der seltenen Wörter hat dies zur Folge, dass sich beispielsweise auf dem BN-Korpus die für die Modellierung der unbekannten Wörter hinzugenommenen pUNK-Wörter im wesentlichen in einer einzigen Klasse wiedergefunden haben.

Für ein Klassensystem, das mit der Zielvorgabe von ungefähr 2500 Klassen trainiert wurde, wird die Gesamthäufigkeit der Klasse (cN) auf dem Korpus BN und die Anzahl Wörter der Klasse (cS) über die Klasse angegeben. Die Klassen wurden nach der Häufigkeit sortiert und die Gesamthäufigkeit logarithmisch dargestellt. Abbildung 6.2 zeigt die 50 häufigsten Klassen. Auffällig in dieser Abbildung tritt die Klasse hervor, die auf Position 29 steht. Sie besitzt einen Umfang von 24834 Wörtern, von denen 19487 (81%) zu den pUNK-Wörtern gehören. Diese 19487 pUNK-Wörter machen dabei 77% des Auftretens von pUNK-Wörtern innerhalb des BN-Korpus aus. Somit können sich die häufigeren Wörter auf mehr Klassen verteilen, während die seltenen Wörter (aufgrund ihrer geringen Häufigkeit) weniger Gewicht in der zu optimierenden Funktion L besitzen und tendenziell zum Auffüllen der Klassen benutzt werden, wenn dies möglich ist.

Um dies zu vermeiden, kann die Anzahl der Klassen erhöht werden, was ebenso zu einer Erhöhung der Rechenzeit führt und das Phänomen nicht beseitigt, die seltenen Wörter in einer Klasse zusammenzufassen. So wurde ein Versuch mit 10000 Klassen als Zielvorgabe durchgeführt. Es zeigte sich, dass auch hier 17181 (69 %) der pUNK-Wörter, die 68% der pUNK-Masse repräsentieren, in einer einzigen Klasse konzentriert wurden.

Es ist zu erwarten, dass mit einer weiteren Erhöhung der Zielklassenanzahl dieser Effekt nach und nach verschwindet, aber die Generalisierung, die die Klassen bieten sollen, wird dadurch ebenfalls abgeschwächt.

Aus dieser Beobachtung entwickelte sich der Ansatz, die potentiell unbekannten Wörter stärker zu berücksichtigen, indem aus dem Korpus die Sätze beziehungsweise Segmente extrahiert werden, in denen potentiell unbekannte Wörter auftreten, und daraus einen Korpus K_{Cluster} für das Clustern geformt wird. Es ergibt sich so ein deutlich kleinerer Korpus, in dem die potentiell unbekannten Wörter deutlich höher gewichtet sind. Für den englischen BN-Korpus ergab sich nach der Selektion des Cluster-Korpus noch ein Umfang von 7% (10% bei Mindesthäufigkeit für pUNK-Wörter von 120).[3]

[3]Ist der neu gewonnene Korpus durch die Selektion nicht kleiner geworden, liegt dies möglicherweise an der Segmentierung, die dem Basiskorpus zugrundeliegt. Beispielsweise trat dies beim SZ-Korpus auf, der nicht nach Sätzen sondern nach ganzen Artikeln segmentiert ist. In solch einem Fall gibt es mehrere Möglichkeiten die relevanten Wörter höher zu gewichten. Man kann beispielsweise die Segmentierung für das Clustern ändern, oder die Menge der potentiell unbekannten Wörter anders wählen, beispielsweise nur eine einzige Schicht von Wörtern mit der gleichen Häufigkeit. Mit Schicht ist die Menge aller pUNK-Wörter gemeint, die gleichhäufig auftreten.

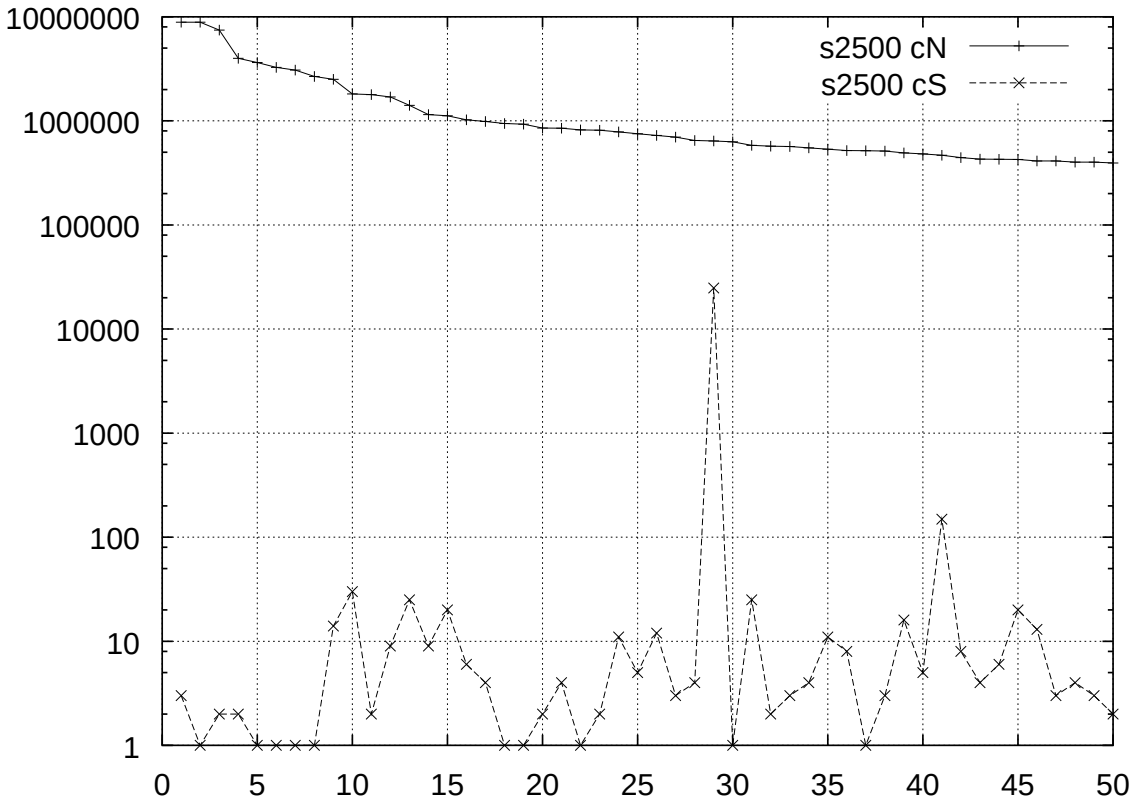

Abbildung 6.2: Klassenfrequenz und Klassengröße über Klassen

Basierend auf dem so gewonnenen Korpus K_{Cluster} wird nun ein neues Vokabular V_{Cluster} für das Clustern bestimmt. Dabei wird die gleiche Mindesthäufigkeit verwendet, die auch für das Bestimmen der potentiell unbekannten Wörter benutzt wurde. Damit ist sichergestellt, dass alle potentiell unbekannten Wörter in V_{Cluster} enthalten sind.

Nachdem nun das Vokabular und der Korpus für das Clustern vorliegen, wird das Clustern, wie von Kneser und Ney [53] beschrieben, durchgeführt. Das Ergebnis ist eine Menge disjunkter Mengen von Wörtern, die die Klasseneinteilung von V_{Cluster} auf K_{Cluster} beschreibt und bei der sich nun nicht mehr fast alle pUNK-Wörter in einer einzigen Klasse konzentrieren.

Dies zeigt sich in den Abbildungen 6.3 und 6.4 deutlich, in denen die Anzahl der enthaltenen pUNK-Wörter je Klasse dargestellt sind. Zur übersichtlicheren Darstellung wurden die Klassen nach der pUNK-Häufigkeit sortiert. Das Klassensystem, das sich durch das Verwenden des gesamten Korpus ergeben hat, wird dabei mit "gesamter Korpus" und das Klassensystem, bei dem der Korpus K_{Cluster} benutzt wurde, wird mit "selektierter Korpus" bezeichnet. Für Abbildung 6.3 wurde als Zielgröße ein System mit 250 Klassen und in Abbildung 6.4 ein System mit 2500 Klasse vorgegeben. Beide Achsen sind in logarithmischer Darstellung.

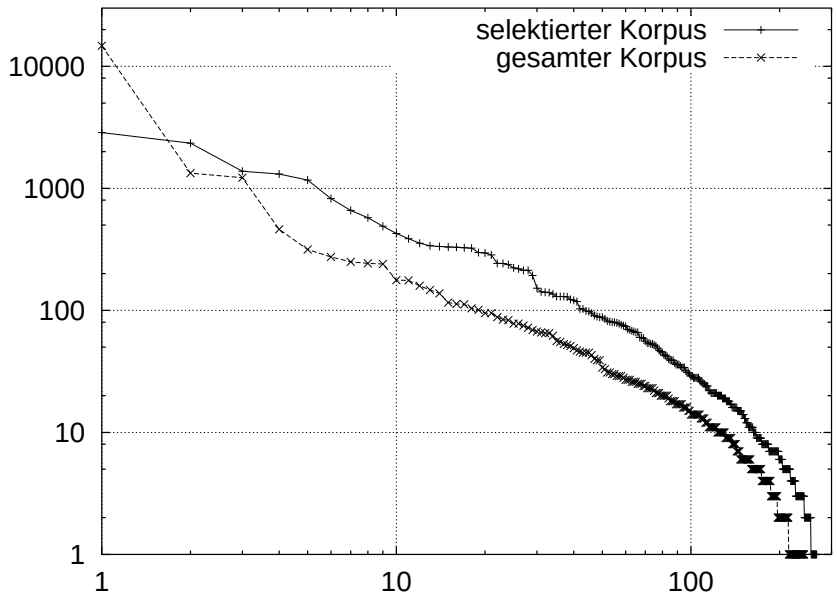

Abbildung 6.3: Anzahl pUNK-Wörter über 250 Klassen

In beiden Abbildungen ist deutlich zu erkennen, dass es bei Verwendung des gesamten Korpus eine Klasse gibt, die den Großteil der pUNK-Wörter aufnimmt. Bei Verwendung

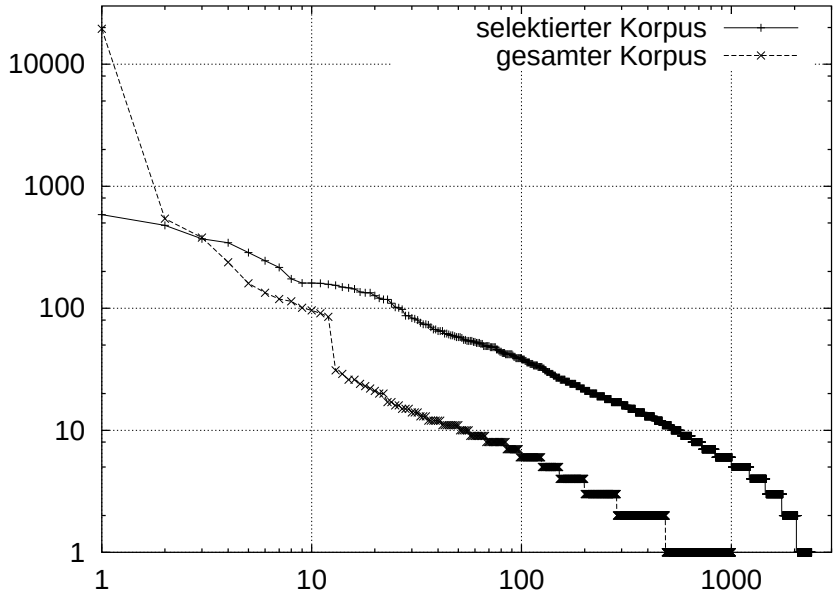

Abbildung 6.4: Anzahl pUNK-Wörter über 2500 Klassen

des selektierten Korpus dagegen konzentrieren sich die pUNK-Wörter nicht in einer Klasse sondern verteilen sich über mehr Klassen. Ein wesentliche Vorteil dieses Verfahrens liegt darin, dass nun die Eigenschaften der pUNK-Wörter oder die der seltenen Wörter insgesamt durch mehr Klassen besser modelliert werden.

Ein so gewonnenes Klassensystem eignet sich aber nur bedingt zur direkten Verwendung in einem statistischen Sprachmodell, da durch die Textselektion die in einem Sprachmodell sehr wichtigen, *häufigen* Wörter zu wenig berücksichtigt sind. Da das Klassensystem eigens zur Gewinnung für Klassen von unbekannten Wörtern erstellt wurde, benötigt man nun noch ein Auswahlverfahren zur Bestimmung der Klassen, die sich für die Modellierung unbekannter Wörter eignen. Diese dann ausgewählten Klassen haben die wichtigsten Strukturen der unbekannten Wörter zu repräsentieren.

Auswahl geeigneter Klassen

Nach dem Abschluss des Clusterns von initialen Klassen muss eine Teilmenge $C_{\mathrm{OOV}} \subset C$ bestimmt werden, die für die Modellierung von unbekannten Wörtern geeignet ist. Für die Selektion der Klassen wird ein Kriterium benötigt, anhand dessen bestimmt wird, ob sich eine Klasse für die Modellierung unbekannter Wörter eignet oder nicht. Hierzu wird für jede Klasse g aus C eine Statistik über die darin enthaltenen potentiell unbekannten Wörter erstellt.

Zur Klassenselektion bieten sich folgende Merkmale an:

- Anzahl der pUNK-Wörter in einer Klasse
- Anteil der pUNK-Masse der Klasse g an der Gesamtmasse der pUNK-Wörter, $p(g|w) : w \in V_{\text{pUNK}}$
- Verhältnis von pUNK-Wörtern zu Vokabular-Wörtern in der Klasse, $p(w|g) : w \in V_{\text{pUNK}}$
- Parameter der Ausgleichsgerade nach Zipf (vgl. Kapitel 2.10).

Als ein geeignetes Kriterium für die Klassenselektion hat sich die absolute Anzahl an pUNK-Wörtern der Klasse herausgestellt. Mit diesem Kriterium ist sichergestellt, dass genügend unbekannte Wörter in der Klasse enthalten sind, damit diese sich nicht überspezialisiert und auch trainiert werden kann. Dabei bietet dieses Kriterium die Möglichkeit, die Klassen so zu ordnen, dass es auch möglich ist, eine maximale Anzahl von N Klassen vorzugeben, wenn dies gewünscht ist.

Die Möglichkeit die Klassen zu ordnen, ist auch für das Kriterium des Anteils an der Gesamtmasse von pUNK-Wörtern gegeben. Damit können so viele Klassen ausgewählt werden, bis ein bestimmter Anteil der Masse der pUNK-Wörter abgedeckt ist. Bei diesem Ansatz werden aber Klassen bevorzugt, die pUNK-Wörter enthalten, die häufiger auftreten als andere pUNK-Wörter.

Da das Bilden der Cluster auf dem vereinigten Vokabular $V \cup V_{\text{pUNK}}$ durchgeführt wird, ist es möglich, dass die Klassen sowohl Vokabular-Wörter als auch pUNK-Wörter enthalten. Das Verhältnis zwischen Vokabular-Wörtern und pUNK-Wörtern ist, wie oben beschrieben, ein mögliches Kriterium für die Selektion einer Klasse. Es ist sogar die Regel, dass Vokabular-Wörter und pUNK-Wörter gemeinsam in einer Klasse auftreten.

Nun stellt sich die Frage, wie die selektierten Klassen für die Erzeugung eines statistischen Sprachmodells verwendet werden. In Abbildung 6.5 seien beispielsweise in den Basisklassen (in Abbildung 6.5 oben) die Klassen C2 und C3 zur Modellierung unbekannter Wörter ausgewählt worden.

Es gibt die Möglichkeit, die selektierten Klassen so zu belassen, wie sie durch das Clustern erzeugt wurden (Umformung A in Abbildung 6.5). Die pUNK-Wörter der selektierten Klassen werden innerhalb dieser Klasse zu einer Klasse zusammengefasst. In Abbildung 6.5 sind dies die Unterklassen C2:UNK und C3:UNK. Die Elemente der Klassen C2 und C3 gehören dann alle zum Vokabular, wobei die Elemente der Klassen C2:UNK und C3:UNK nur durch diese Klassen als Ganzes repräsentiert sind und dann zur Modellierung unbekannter Wörter verwendet werden. Klassen, die nicht selektiert werden, weil sie beispielsweise

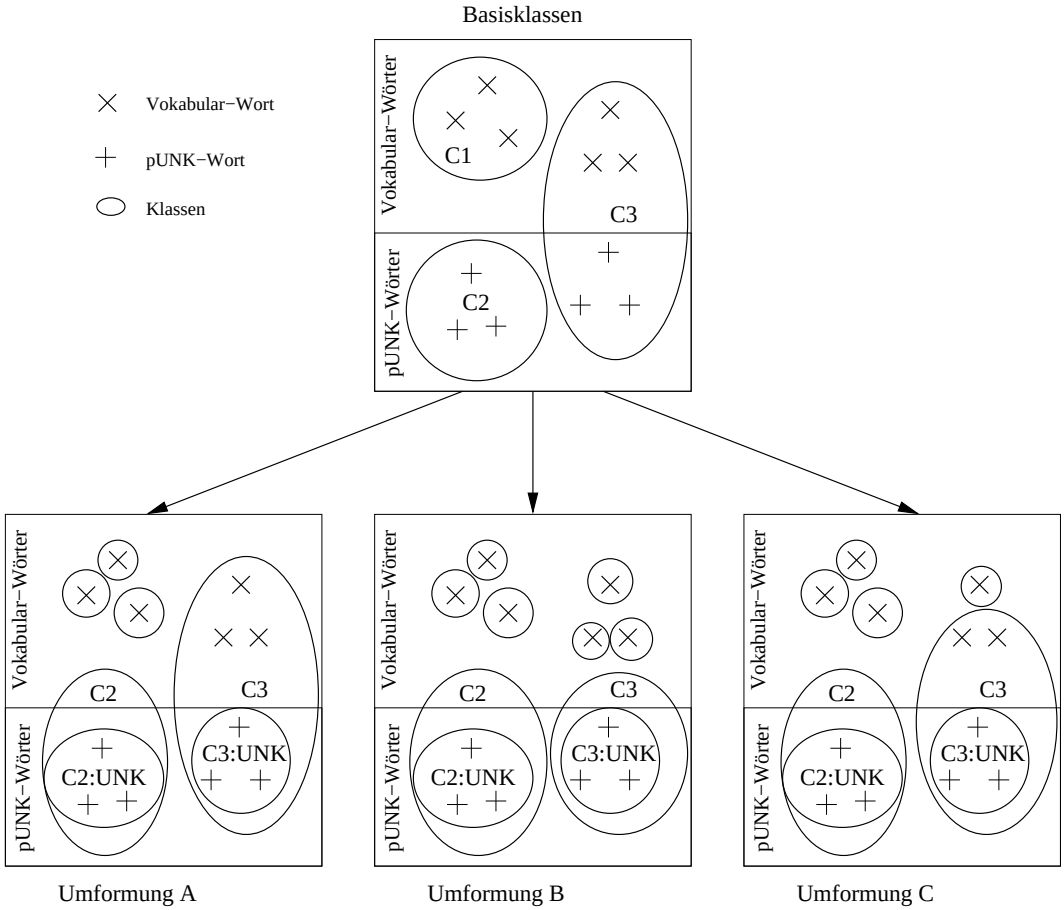

Abbildung 6.5: Klassenbildung mit pUNK-Wörtern

wie Klasse C1 keine pUNK-Wörter enthalten, werden aufgelöst und jedes Vokabular-Wort bildet seine eigene Klasse.

Die Verwendung der Klassen genau so, wie sie während des Clusterns erzeugt und anschließend selektiert werden, ist aber nur sinnvoll, wenn hinreichen viele Klassen während des Clusterns erzeugt werden. Sollte dies nicht der Fall sein, ist einer der anderen Ansätze zu wählen.

Beim zweiten Ansatz werden stets alle Vokabular-Wörter aus den selektierten Klassen entfernt und somit Klassen erzeugt, die nur durch pUNK-Wörter modelliert werden (Umformung B in Abbildung 6.5).

Eine dritte Möglichkeit ist, Wörter aus dem Vokabular V in den selektierten Klassen verbleiben zu lassen (Umformung C in Abbildung 6.5). Dies ist aber nur dann sinnvoll, wenn hinreichend viele Klassen erzeugt werden. So ist beispielsweise auf dem BN-Korpus mit der Zielvorgabe von 50 oder 250 Klassen die Zahl der Klassen zu klein, um daraus ein eigenständiges Sprachmodell zu erzeugen. In diesen Fällen ist es notwendig, mit einer Stopp-Wortliste Wörter wieder aus den Klassen zu entfernen. Eine solche Liste enthält dann beispielsweise die N häufigsten Wörter.

Aufgrund des Korpus-Selektionsschrittes ist nicht für alle Wörter die gesamte Statistik über das jeweilige Wort in die Klasseneinteilung eingeflossen. Dies ist beabsichtigt, führt aber dazu, dass den Wörtern, die die Klassenselektion bewirkt haben, ein größeres Vertrauen entgegengebracht werden kann, da für sie mit Sicherheit die vollständige Statistik extrahiert wurde.

Um die Beziehung zwischen den seltenen Wörtern aus dem Vokabular und den pUNK-Wörtern zu verstärken, ist lediglich die Mindesthäufigkeit für die Erzeugung des Vokabulars zu erhöhen, das für die Selektion benutzt wird.

Für den BN-Korpus ergibt sich so beispielsweise, dass bei einer Mindesthäufigkeit von 120 noch 22287 Wörter im Vokabular enthalten sind und für die Korpus-Selektion 42418 Wörter (V_{select}) verwendet würden. In diesem Fall würden Wörter aus den Klassen entfernt, die in $V \setminus V_{\text{select}}$ liegen.

Diese Vorgehensweise entspricht Umformung C und ist besonders sinnvoll, da auch die seltenen Wörter, die zum Vokabular gehören, von einer Klassenbildung profitieren können.

Für die Experimente in Kapitel 7.2.1 wurde das Clustern auf dem selektierten Korpus mit einer Zielvorgabe von 2500 Klassen durchgeführt. Für die Modellierung unbekannter Wörter wurden alle Klassen ausgewählt, die mindestens 50 der pUNK-Wörter enthalten. Das ergab 72 Klassen, die für die Untersuchungen in den Experimenten verwendet werden. Im Anhang A sind die Tabellen vollständig aufgeführt. Im folgenden werden sieben ausgewählte Klassen näher betrachtet.

Da die Klassen datengetrieben entstanden sind, ist es nicht immer möglich den "Sinn"

einer Klasse zu ergründen. Darum wird hier für jede der ausgewählten Klassen eine Liste der fünf häufigsten Vokabular-Wörter (Tabelle 6.1) und der fünf häufigsten OOV-Wörter (Tabelle 6.2) angegeben.

Klasse	Die fünf häufigsten Vokabular-Wörter der ausgewählten Klassen
0	ROSEANNE CATERPILLAR SEQUESTRATION SUNLIGHT DASH
6	STALIN CHERNOMYRDIN DURAN NORPLANT WHITEHURST
29	TUBERCULOSIS DIABETES ASTHMA CHOLERA ALCOHOLISM
38	KEVORKIAN LAKSHMANAN GERDES BLAKE BADEN
42	SMITH DAVIS THOMPSON ROBERTS CHAPMAN
45	FREEDOM DESTRUCTION SILENCE INNOCENCE SEPARATION
62	MICROSOFT PARAMOUNT INTEL VIACOM BLOCKBUSTER

Tabelle 6.1: Beschreibung der ausgewählten Klassen zur Modellierung von OOV-Wörtern

Klasse	Die fünf häufigsten OOV-Wörter der ausgewählten Klassen
0	BLANCO HOBBES RHYMING TRISTAN DVORAK
6	AVRAHAM JELKS CARRANZA ACCION AILICH
29	GLAUCOMA PHARMACIA PERTUSSIS DCIS JAUNDICE
38	DESILVA TENNANT BRAZELTON HIMMELSTEIN TARNOWER
42	LASSITER CANO NAVA PADILLA FREDRICK
45	DISINFLATION STEADINESS PROBITY ZEALOTRY CORDIALITY
62	CIGNA SAMSUNG MEDRAD BULLOCK'S GEMSTAR

Tabelle 6.2: Beschreibung der ausgewählten Klassen zur Modellierung von OOV-Wörtern

Die aufgeführten Wörter von Klasse 62 zeigen, dass diese Klasse anscheinend überwiegend Namen von Firmen modelliert, die durch einen NE-Tagger als "ORGANISATION" markiert worden wären. Klassen 42, 38 und 6 modellieren anscheinend überwiegend Namen von Personen, die ein NE-Tagger mit "PERSON" markiert hätte.

Die Klassen 38 und 6 besitzen dabei eine große Steigung $-b$ für die Zipf'sche Ausgleichsgerade für die Häufigkeit der Frequenzen der Wörter (Tabelle 6.3), die diese Verteilung auch gut repräsentiert (Abbildungen 6.6). Dies legt die Vermutung nahe, dass es sich bei diesen Klassen eher um offene Mengen von Wörtern handelt als bei Klasse 42, bei der die Ausgleichsgerade in Abbildung 6.8 die Verteilung der Häufigkeiten anscheinend schlecht repräsentiert.

Die Klasse 0 lässt sich nicht so einfach charakterisieren. Die Häufigkeiten der Wörter sind durch die Ausgleichsgerade gut repräsentiert (Abbildung 6.7), was, wie bei Klasse 6 und 38 nahelegt, dass es sich um eine offene Klasse handelt. Die Wörter sind aber eher eine Art Restklasse zuzusordnen, was auch die Anzahl verschiedener Wörter in der Klasse insgesamt zeigt (Tabelle 6.3).

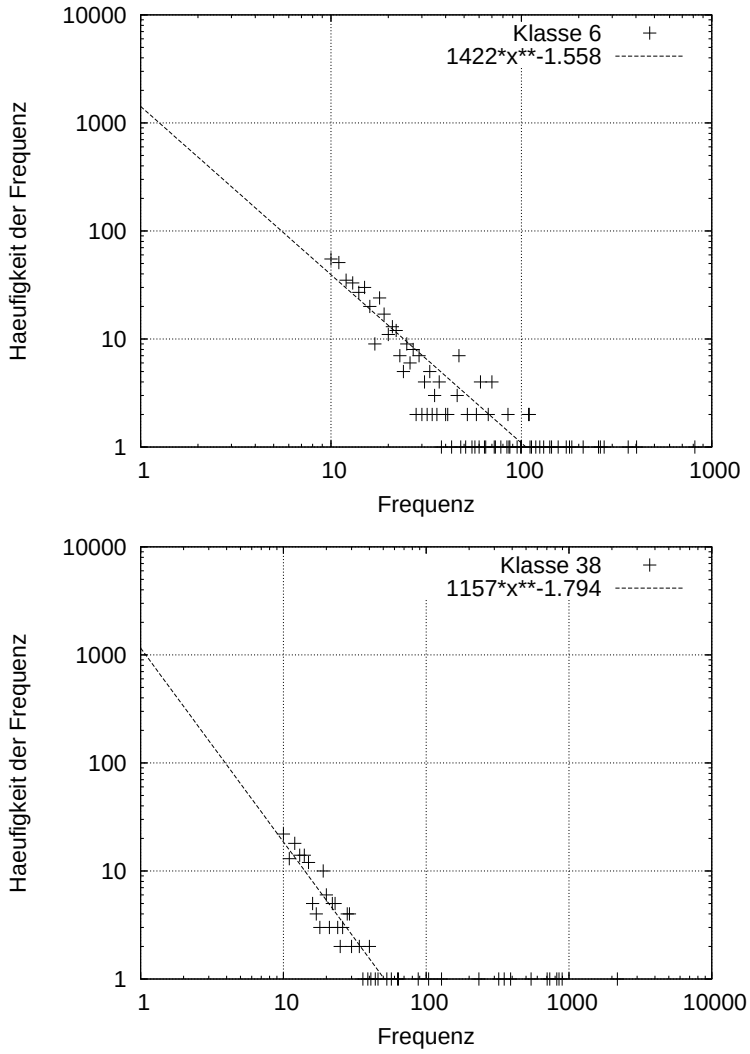

Abbildung 6.6: Die Klassen 6 und 38 mit einem großen Wert für Steigung $-b$ (Häufigkeit der Frequenz über Frequenz)

Klasse	Anzahl der OOV-Wörter	Wörter insgesamt	OOV-Rate	Zipf-Ausgleichsgerade	
				$-a$	$-b$
0	600	1169	15.51	647.2	1.134
6	383	475	41.92	1422.2	1.558
29	62	126	8.36	9.0	0.261
38	152	179	21.34	1157.8	1.794
42	50	126	2.09	8.4	0.269
45	50	92	2.52	5.0	0.123
62	65	97	8.39	34.2	0.718

Tabelle 6.3: Beschreibung der Klassen zur Modellierung von OOV-Wörter

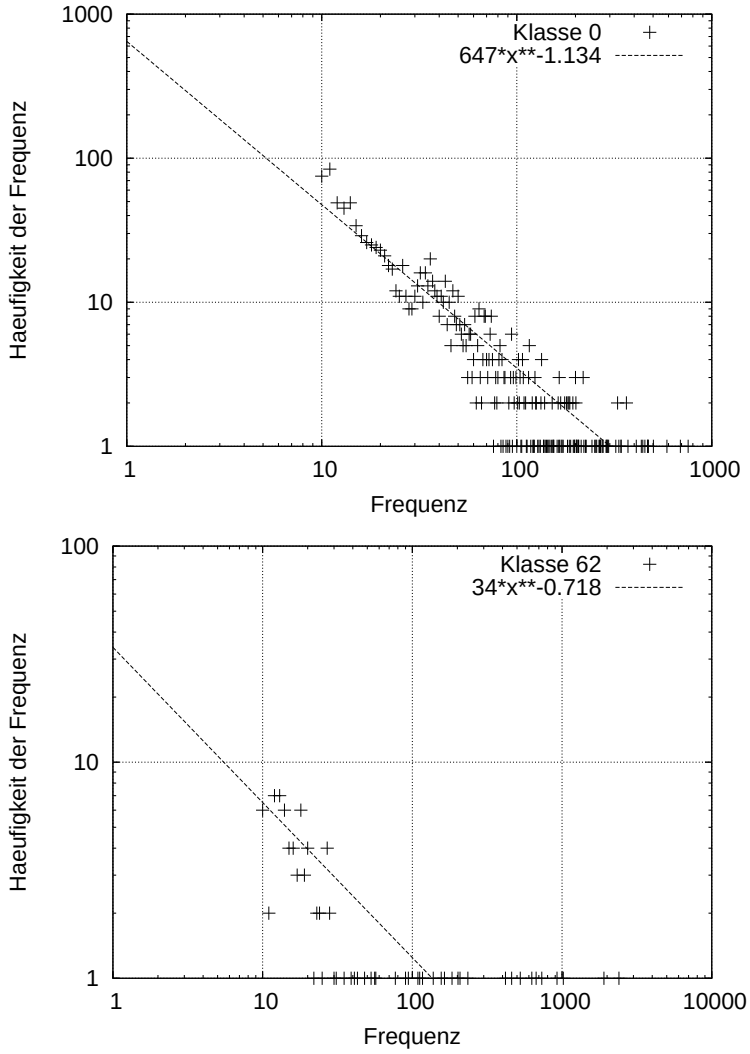

Abbildung 6.7: Die Klassen 0 und 62 mit einem mittleren Wert für Steigung $-b$

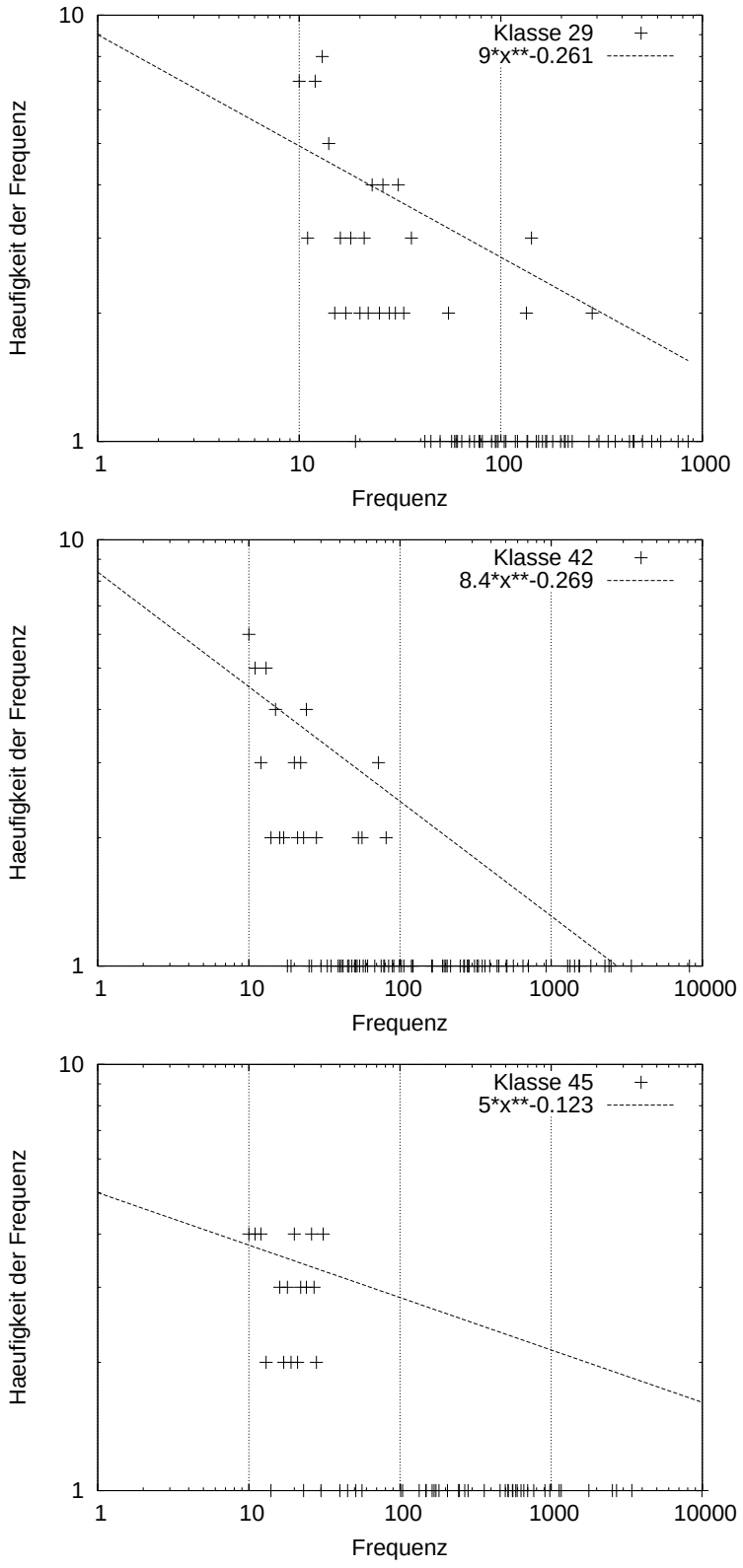

Abbildung 6.8: Die Klassen 29, 42 und 45 mit einem kleinen Wert für Steigung $-b$

Besonders hervorzuheben ist aber Klasse 29, bei der über 8% der Wahrscheinlichkeitsmasse auf die OOV-Wörter fällt und in der sich anscheinend Namen für Krankheiten und Begriffe aus dem medizinischen Fachbereich gesammelt haben. Eine solche Klasse ist beim NE-Task *nicht* vorgesehen, ist aber für die Modellierung von unbekannten Wörtern sinnvoll, da in der BN-Domäne unbekannte Wörter aus diesem Bereich häufig genug aufgetreten sind, so dass datengetrieben eine eigene Klasse dafür entsteht.

Klasse 45 enthält Adjektive und ist somit nach grammatischen Eigenschaften zu deuten, ebenso wie weitere Klassen, die im Anhang A aufgeführt sind. Beispielsweise enden die meisten Wörter, die in Klasse 26 enthalten sind, mit der Silbe '-ING'.

Die mit diesem Verfahren erzeugten Klassen teilen die OOV-Wörter also sowohl nach semantischen als auch nach syntaktischen "Kategorien" ein.

Mit dem so erzeugten Klassensystem wird der ursprüngliche Korpus K bearbeitet, indem die OOV-Wörter, die in den Klassen zur Modellierung von unbekannten Wörtern enthalten sind, durch ein entsprechendes Symbol (<OOV:Class-x>) im ursprünglichen Korpus K ersetzt werden. Diese Symbole für diese OOV-Wörter werden dann ebenfalls in das Vokabular für die Erzeugung des Sprachmodells aufgenommen. Mit diesem abgebildeten Korpus wurde ein statistisches Sprachmodell mit den gleichen Parametern für die Mindesthäufigkeiten wie für das BASELINE-LM erzeugt, das mit LM-72 bezeichnet wird.

Die Wahrscheinlichkeit für das Auftreten eines unbekannten Wortes w_{UNK} wird am Beispiel eines Trigramm-Sprachmodells nach Gleichung 6.8 bestimmt.

$$\begin{aligned} p(w_{\text{UNK}}|w_{i-2}w_{i-1}) \;=\; & p(w_{\text{UNK}}|\text{<OOV:Class-x>}) \\ & p(\text{<OOV:Class-x>}|\text{<Class-x>}) \\ & p(\text{<Class-x>}|w_{i-2}w_{i-1}) \end{aligned} \tag{6.8}$$

Dabei ist $p(\text{<Class-x>}|w_{i-2}w_{i-1})$ die Wahrscheinlichkeit, dass ein Wort aus der Klasse $Class{-}x$ den Worten $(w_{i-2}w_{i-1})$ folgt und $p(\text{<OOV:Class-x>}|\text{<Class-x>})$ die Wahrscheinlichkeit für das Auftreten eines beliebigen OOV-Wortes innerhalb der Klasse $Class - x$.

Wenn die Klasse nur OOV-Wörter enthält, ist $p(\text{<OOV:Class-x>}|\text{<Class-x>}) = 1.0$. Handelt es sich um eine gemischte Klasse aus OOV-Wörtern und Vokabular-Wörtern, besteht die Möglichkeit, die Wahrscheinlichkeit für das Auftreten eines OOV-Wortes zu erhöhen, indem die Wahrscheinlichkeit für die Vokabular-Wörter abgesenkt wird, wie es beispielsweise von Gallwitz [31] beschrieben wurde.

Bei Klassen, in denen sich die Häufigkeitsverteilung gut durch die Zipf'sche Ausgleichsgerade approximieren lässt, kann diese auch verwendet werden, um die Zahl der fehlenden

OOV-Wörter zu schätzen, die sich durch die Mindesthäufigkeit von 10 ergeben. Dies ist beispielsweise für die Klassen 0, 6 und 38 möglich und erlaubt so auch das Anheben der Wahrscheinlichkeit $p(<\text{OOV:Class-x}>|<\text{Class-x}>)$, *wenn* auch Vokabular-Wörter in den Klassen enthalten sind. Dafür steigt aber im Gegenzug die Zahl verschiedener OOV-Wörter, die die Klasse nun enthält. Effektiv wird durch das Verschieben der Wahrscheinlichkeitsmassen zwischen Vokabular- und OOV-Wörtern den Klassen *selbst* keine höhere Wahrscheinlichkeit oder keine neuen Kontexte zugeordnet, weshalb auf diese Modifikationen verzichtet wurde.

Aus diesem Grunde werden im Prinzip die Wahrscheinlichkeiten für $p(<\text{Class-x}>|w_{i-2}w_{i-1})$ und $p(<\text{OOV:Class-x}>|<\text{Class-x}>)$ mit den üblichen Methoden für die Erzeugung eines Sprachmodells auf dem umgeformten Korpus bestimmt. Für die Wahrscheinlichkeit $p(w_{\text{UNK}}|<\text{OOV:Class-x}>)$ kann, wie im Einklassenfall, die mittlere Wahrscheinlichkeit für ein OOV-Wort benutzt werden. Hierzu wird bestimmt, wie viele verschiedene Wörter durch das Symbol <OOV:Class-x> repräsentiert sind, also welchen Umfang die Klasse OOV:Class-x besitzt.

$$p(w_{\text{UNK}}|<\text{OOV:Class-x}>) = \frac{1}{|\text{OOV:Class-x}|} \tag{6.9}$$

6.1.6 Zusammenfassung

Das Verfahren zur Bestimmung von Klassen für die Modellierung unbekannter Wörter führt eine datengetriebene Gruppierung von OOV-Wörtern durch. Die Untersuchung zeigt, dass ohne den Korpus-Selektionsschritt die pUNK-Wörter beim Clustern im wesentlichen in einer einzigen Klasse zusammengedrängt werden, da die häufig auftretenden Wörter die Klassen besetzen. Der Schritt der Korpusselektion ist wichtig, damit sich die pUNK-Wörter stärker auf mehrere Klassen verteilen können, ohne dass dabei die Zahl der Klassen für das Clustern besonders gross gewählt werden muss.[4]

Nach dem Clustern werden die Klassen ausgewählt, die für die Modellierung der unbekannten Wörter benutzt werden. Alle anderen Klassen werden aufgelöst. Mit den ausgewählten Klassen wird ein Sprachmodell erzeugt, das dann Klassensymbole bereitstellt, die für die Modellierung unbekannter Wörter benutzt werden.

Das Verfahren ermöglicht die Definition eines Klassensystems für die Modellierung unbekannter Wörter ohne manuelle Vorgaben und bedarf keines Experten. Es findet automatisch ein an die Domäne angepasstes Klassensystem, wie sich an Klasse 29 zeigt, in der sich anscheinend Begriffe aus dem medizinischen Fachbereich gesammelt haben.

[4]Ein positver Nebeneffekt hiervon ist, dass das Clustern wenig rechenaufwendig ist.

6.2 Akustische Modellierung unbekannter Wörter mit Head-Tail-Modellen

Die für diese Arbeit entwickelte Modellierung von unbekannten Wörtern kombiniert die Eigenschaften der exakten Phonemschleife mit denen der vergröberten Modelle, wie sie bei den flachen akustischen Modellen verwendet werden [1]. Die Kombination erfolgt durch einen Kopf-Teil (Head-Modell), der exakte phonetische Modelle benutzt, und einem Schwanz-Teil (Tail-Modell), der weniger präzise akustische Modelle verwendet, die in Abbildung 6.9 durch einen Stern dargestellt sind und Gruppen von Phonemen repräsentieren können.

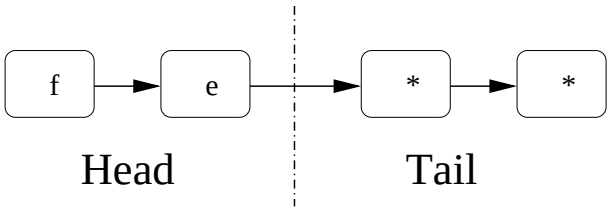

Abbildung 6.9: Generisches Wort als Head-Tail-Modell

Eine Analyse des Wörterbuches zeigt, dass bereits nach drei bis vier Phonemen die Wortidentität für die Mehrzahl der Wörter eindeutig ist. Bis sich abzeichnet, dass der Dekoder nun an einem Punkt angelangt ist, an dem die bekannten Wörter anfangen, nicht mehr auf die vorgeschlagene Akustik zu passen, sollte eine genauere Modellierung im OOV-Wort verwendet werden. Danach können auch Modelle eingesetzt werden, die weniger exakt sind.

Jedes generische Head-Tail-Wort repräsentiert eine Menge an Wörtern, die das gleiche Präfix an Phonemen besitzt. Durch die weniger scharfe akustische Modellierung des Tail-Modells unterscheiden sich diese Wörter deutlich von bekannten Wörtern aus dem Vokabular. Dabei kann durch die Länge des Tail-Modells zum einem eine Mindestdauer für ein unbekanntes Wort realisiert werden, zum anderen vergrößert ein Tail-Modell mit mehr Zuständen auch den Abstand zu bekannten Wörtern.

In diesem Abschnitt sind die für diese Arbeit entwickelten Methoden zur akustischen Modellierung von unbekannten Wörtern beschrieben, die dann in Kapitel 7 experimentell untersucht werden.

6.2.1 Allgemeine Überlegungen

Im Gegensatz zu einer Phonemschleife ist das akustische Modell insgesamt ungenauer, da die Tail-Modelle weniger gut passen als exakte Phonemmodelle. Andererseits wird ein Teil

des Wortes mit exakten Modellen realisiert, wodurch eine präzisere Modellierung als mit einer flachen akustischen Modellierung erreicht wird.

Die exakten Head-Modelle befinden sich am Anfang des generischen Wortes, da der für diese Arbeit verwendete Dekoder [24][72] eine zeitsynchrone Suche in Richtung der Zeitachse durchführt. Bei Verwendung einer Präfix-Baum-Struktur wird versucht, gleiche Wortanfänge gemeinsam zu nutzen, um den Suchraum kompakt zu halten. Für die Head-Modelle besteht in diesem Fall ebenfalls die Möglichkeit von dieser kompakten Suchraum-Repräsentation zu profitieren.

Der verwendete Viterbi-Algorithmus garantiert zwar den global optimalen, d.h. den wahrscheinlichsten Zustandspfad zu finden, dies gilt aber nicht mehr, wenn Methoden zum Beschneiden des Suchraums eingesetzt werden. Das kann zu den in Kapitel 2.5 beschriebenen Suchfehlern führen, in denen ein Wort aus den Hypothesen entfernt wird, weil es lokal eine schlechte Bewertung erhalten hat.

Offensichtlich führen ungenauere akustische Modelle im Mittel zu schlechteren Bewertungen. Da gerade der Beginn eines Modells besonderes rechenintensiv ist, weil dieser für jeden möglichen Wortübergang aktiv ist, muss dort der Suchraum am aggressivsten beschnitten werden. Würden die Tail-Modelle dort eingesetzt, ist zu erwarten, dass die Zahl der Suchfehler ansteigt, in denen ein generisches Wort fälschlicherweise aus dem Suchraum entfernt wird, es sei denn, die Parameter, die die Beschneidung des Suchraums steuern, werden so gewählt, dass durch sie keine generischen Wörter entfernt werden, also beispielsweise mit einem weiten Beam.

Da eine effektive Beschneidung des Suchraumes gewünscht wird, in der möglichst wenig Suchfehler erzeugt werden, ist in dem hier untersuchten Fall das Head-Modell am Wortanfang zu verwenden. Weiter lassen sich, wie bereits erwähnt, bei Verwendung eines Aussprache-Präfix-Baumes für den Suchraum die Head-Modelle der generischen Wörter gut in diese Struktur integrieren, was die Anzahl zusätzlicher Zustände im Suchraum verringert.

Möglichkeiten, Head-Tail-Modelle zu implementieren

Eine Möglichkeit, die generischen Wörter zu modellieren ist, die Tail-Modelle zusammenzufassen, wie es beispielsweise in Abbildung 6.10 dargestellt ist. Mit einer solchen Modellierung ist der zu erwartende Mehraufwand für die generischen Wörter gering. Das ist insbesondere dann der Fall, wenn bei gleichen Historien am Ende des Modells nur das Modell mit dem am besten passenden Aussprache-Präfix in die möglichen nachfolge Worte expandiert wird.

Es ist aber auch möglich, die Head-Tail-Wörter als eigenständige Wörter aufzunehmen. Das bietet zusätzlich noch die Möglichkeit, verschiedene Gewichtungen im Sprachmodell vorzunehmen, oder bestimmte generalisierte Wörter einfach auch aus der Suche zu entfernen.

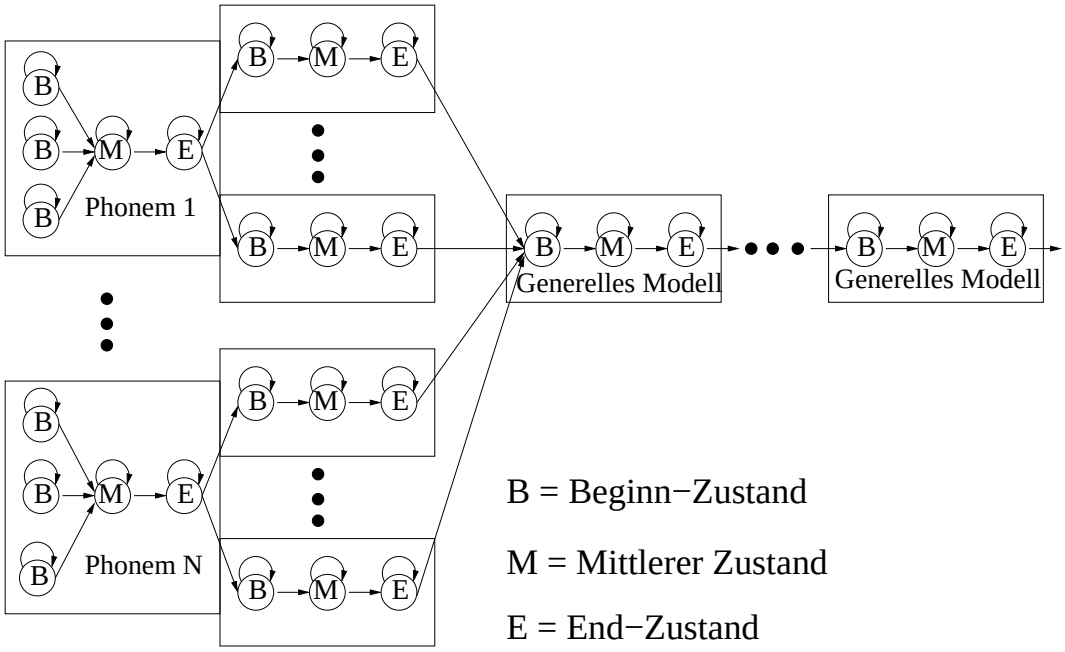

Abbildung 6.10: Mögliche Implementierung der generalisierten Wörter

Aufgabe des Tail-Modells

Bevor nun verschiedene Methoden zur Erzeugung von Head-Modellen beschrieben und analysiert werden, soll kurz die Rolle der Tail-Modelle dargelegt werden.

Die primäre Aufgabe des Tail-Modells ist es, den Teil eines unbekannten Wortes zu modellieren, der nicht vom Head-Modell abgedeckt ist. Das heißt, er dient auch zum Modellieren der Länge von unbekannten Wörtern. Dabei soll der Abstand zu bekannten Wörtern aus dem Vokabular vergrößert und damit die Zahl fehlerhafter Hypothesen für unbekannte Wörter reduziert werden. Die Länge des verwendeten Tail-Modells hat einen Einfluss darauf, wie leicht ein unbekanntes Wort hypothetisiert wird. Je kürzer desto einfacher. Durch ein längeres Tail-Modell ist zu erwarten, dass beispielsweise die Präzision höher ist, der Recall aber dafür niedriger ausfällt.

In dieser Arbeit wurde als Tail-Modell ein generisches Modell für Sprache verwendet. Es ist natürlich auch möglich, stufenweise von feineren Modellen zu den gröberen überzugehen.

6.2.2 Erzeugung von Head-Modellen für generische Wörter

In diesem Abschnitt wird beschrieben, wie die Head-Modelle für generische Wörter erzeugt werden können. Die hier beschriebenen Head-Modelle werden dann in den Experimenten in Kapitel 7 nochmals eingehender untersucht.

Erzeugen von Head-Tail-Modelle aus Vokal-Konsonanten-Modellen

Die Vokal-Konsonanten-Head-Modelle (VC) sind die ersten Head-Modelle, die entwickelt wurden [65]. Dabei wird der im Spracherkenner verwendete Phonemsatz in die Teilmengen Vokale V und Konsonanten C unterteilt. Zu einer vorgegebenen Länge l werden dann alle Vokal-Konsonanten-Folgen erzeugt. Für $l = 2$ würde sich die Menge der Head-Modelle $H = V \times C \cup C \times V$ ergeben. Das sind beispielsweise für den verwendeten deutschen Phonemsatz 780 verschiedene Head-Modelle. Ohne die Einschränkung auf Vokal-Konsonanten-Folgen würde eine ungefähr doppelt so große Menge entstehen ($39^2 = 1521$). Darin würden auch phonotaktisch unmögliche Phonemkombinationen enthalten sein, wie beispielsweise "|p k|" oder "|p p|".[5] Dennoch wächst die Menge der Head-Modelle in Abhängigkeit der vorgegebenen Länge exponentiell an.

Der Vorteil der Vokal-Konsonanten-Head-Modelle liegt darin, dass sie keine weiteren Informationen benötigen als den Phonemsatz und eine allgemeine grammatische Regel, wie sich Vokale und Konsonanten in einer Sprache abwechseln.

Mit der Länge steigt auch die Zahl an Head-Modellen, die keinen sinnvollen Präfix eines unbekannten Wortes darstellen, da Worte mit solch einem Präfix nicht existieren und darum nicht sinnvoll für einen Spracherkenner trainiert werden können.

Erzeugen von Head-Tail-Modelle aus Vokabularpräfixen

Dieser Ansatz setzt voraus, dass für die Konstruktion der Head-Modelle das Vokabular des verwendeten Spracherkenners bekannt ist (IVP).

Die zugrundeliegende Idee ist, den Suchraum des Spracherkenners so zu erweitern, dass ab einer bestimmten Tiefe[6] in ein Tail-Modell verzweigt werden kann.

Die hierfür benötigten Head-Modelle lassen sich aus dem verwendeten Vokabular erzeugen, indem von den in Frage kommenden Wörtern die ersten h Phoneme als ein Head-Modell benutzt werden.

In den untersuchten Modellen in Kapitel 7 wurde die Menge der verwendeten Wörter aus dem Vokabular eingeschränkt, indem die Wörter aus mindestens $h + t = min_l$ Phonemen bestehen müssen. Ist beispielsweise $t = 4$ und $h = 2$ gewählt, werden nur Wörter aus dem Vokabular in Betracht gezogen, die aus mindestens sechs Phonemen bestehen.

Mit den folgenden Abbildungen wird das Verfahren für die Parameter $h = 2$, $t = 2$, also $min_l = 4$, beispielhaft illustriert.

[5] Vokal-Vokal-Folgen werden nicht modelliert, können aber durch die Modelle abgedeckt werden, wenn keine bessere, konkurrierende Modellierung vorhanden ist.

[6] Anzahl Phoneme von Wörtern aus dem Vokabular

In Abbildung 6.11 ist ein fiktives Vokabular bestehend aus acht Wörtern $w1 - w8$ in einer vereinfachten Präfix-Baumstruktur dargestellt.

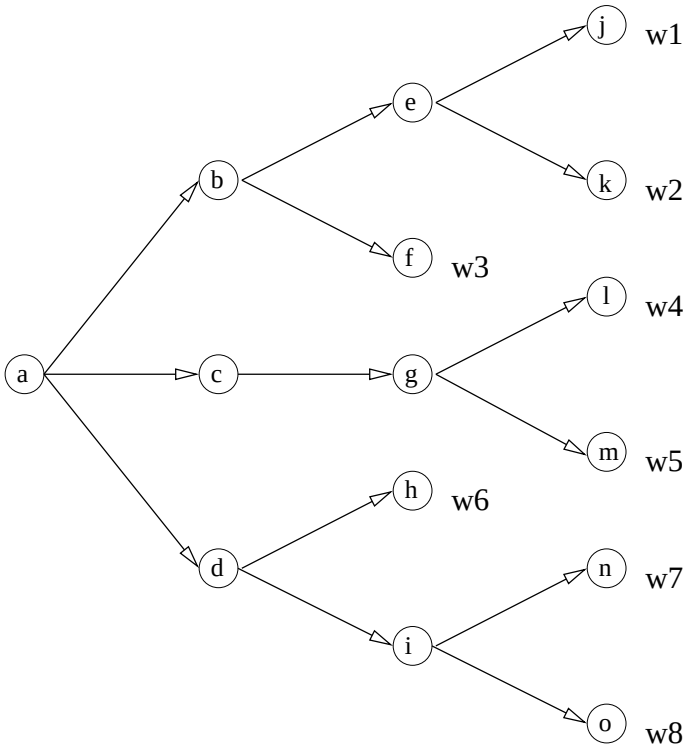

Abbildung 6.11: Als Präfix-Baum organisiertes Basisvokabular

Die für die Erzeugung der Head-Modelle zu kurzen Wörter $w3$ und $w6$ sind in Abbildung 6.12 entfernt und die Schnittkante für $h = 2$ als gestrichelte Linie eingezeichnet. Der Teil des Präfix-Baumes, der links von der Schnittkante liegt wird für die Head-Modelle verwendet, der Teil der rechts der gestrichelten Linie liegt, wird durch zwei generische Phoneme ersetzt, die in Abbildung 6.13 durch Quadrate mit abgerundeten Ecken dargestellt sind.

Diese generischen akustischen Modelle für Sprache bilden das Tail-Modell der Länge zwei und sind in der Abbildung links für jedes Head-Modell separat vorhanden und rechts zusammengefasst, was zu einer kompakteren Repräsentation des generischen Wortes (GW) führt.

Diese Struktur wird in den ursprünglichen Suchraum aus Abbildung 6.12 eingefügt, was in Abbildung 6.14 dargestellt ist. Die gestrichelten Pfeile deuten dabei an, an welcher Stelle das bekannte Vokabular verlassen wird.

Um mehrere Klassen für die Modellierung unbekannter Wörter zu benutzen, kann die

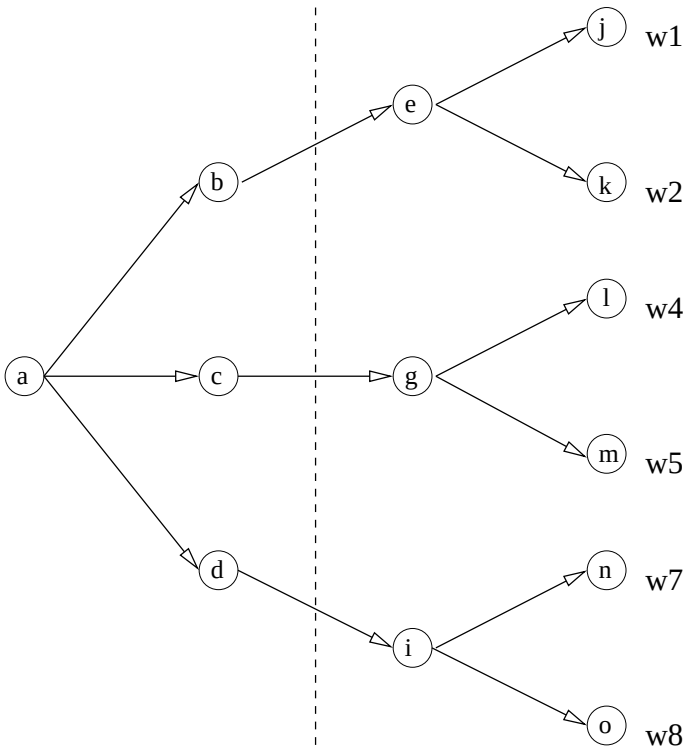

Abbildung 6.12: Reduziertes Vokabular

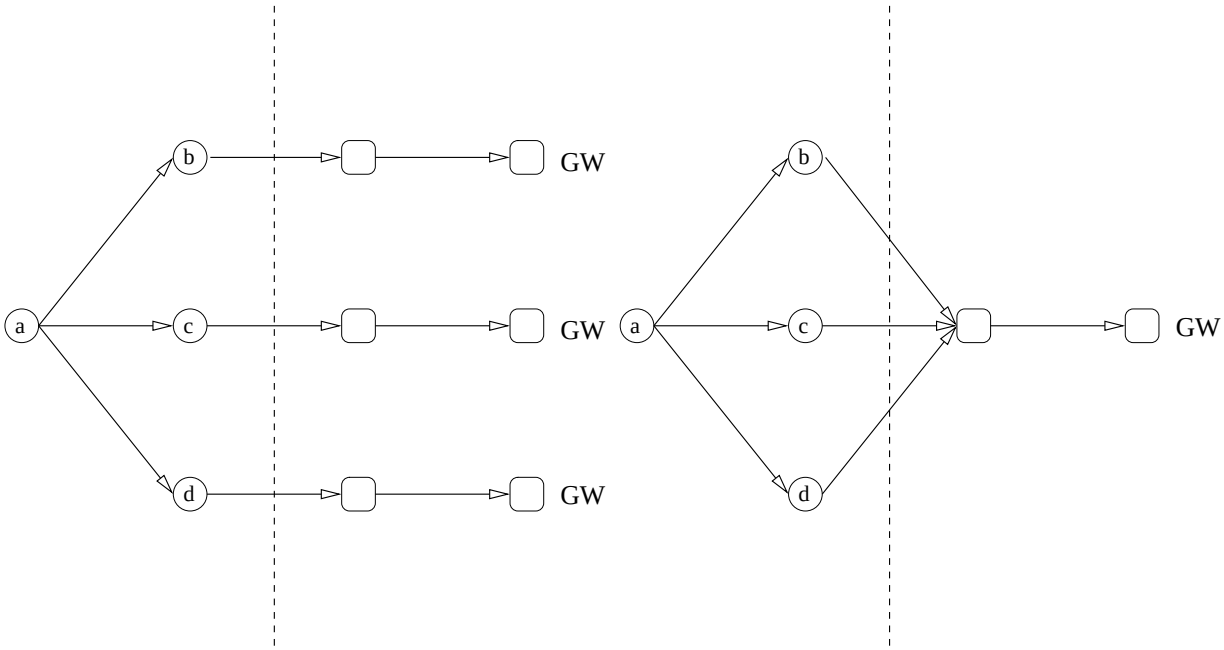

Abbildung 6.13: Head-Tail-Modelle mit separatem (links) und gemeinsamem (rechts) Tail

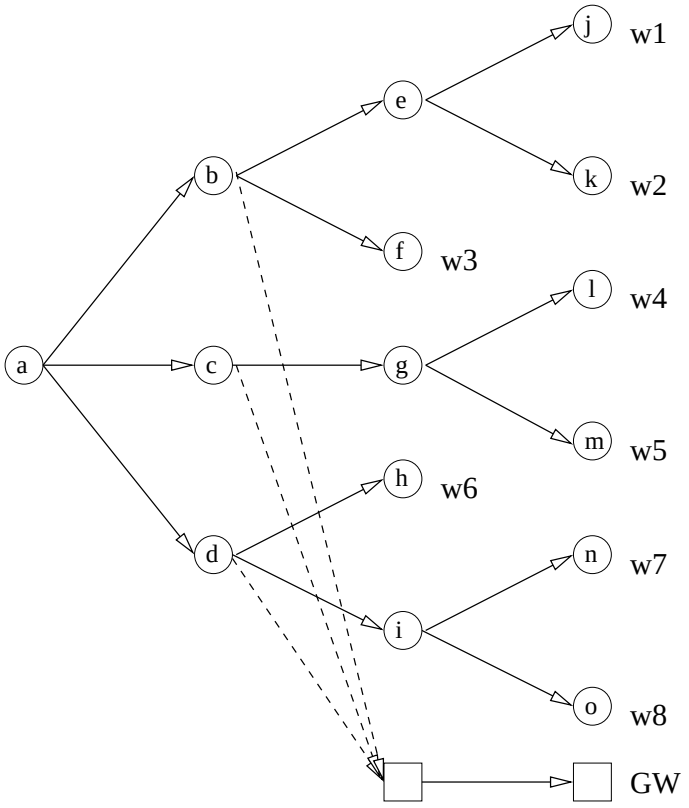

Abbildung 6.14: Erweiterter Suchraum mit gemeinsamem Tail

gleiche Struktur als "unterschiedliche" Wörter mehrfach eingefügt werden. Dabei muss dann beispielsweise nur der Blattknoten vervielfältigt werden, wie das in Abbildung 6.15 dargestellt ist. Hier könnte beispielsweise GW-Class1 im Sprachmodell Personennamen und GW-Class2 Organisationen modellieren.

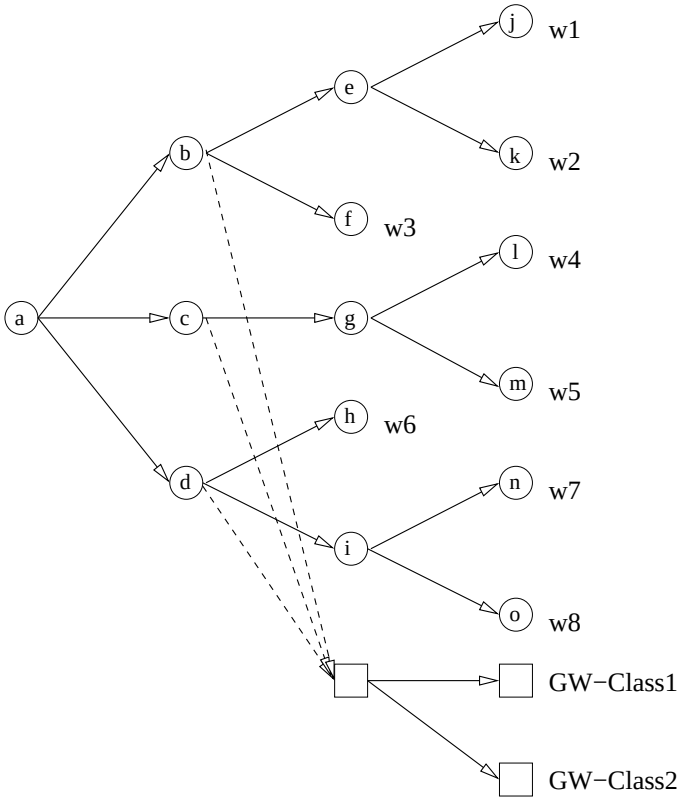

Abbildung 6.15: Modellierung mehrerer Klassen

Im Gegensatz zu den Vokal-Konsonanten-Folgen sind *alle* Head-Modelle Präfix eines Vokabularwortes. Die akustischen Modelle, die eingesetzt werden können, sind demnach genauso gut trainiert, wie die akustischen Modelle von Wörtern aus dem Vokabular. Die Zahl der zusätzlichen Zustände für die Modellierung generischer Wörter in der Suchstruktur kann beim Zusammenfassen der Tail-Modelle sehr gering ausfallen. Insgesamt stellt dies eine sehr kompakte Erweiterung des Suchraumes dar. Die Menge der Head-Modelle ist durch die Größe des Vokabulars beschränkt. Wird beispielsweise das Vokabular um ein Wort größer gewählt, wächst die Menge der Head-Modelle höchstens um eins an. Dazu muss das hinzugefügte Wort die Mindestlänge an Phonemen besitzen, was für seltene Wörter eher zutrifft, und ein Präfix der Länge h besitzen, der noch nicht in den Head-Modellen enthalten ist.

Es ist nochmals hervorzuheben, dass in diesem Fall nur Head-Modelle erzeugt werden, die Präfix eines "echten" Wortes sind. Präfixe, die sich vom Vokabular unterscheiden,

können mit dieser Methode nicht erzeugt werden.

Erzeugen von Head-Modellen mit Hintergrundwörterbuch

Verfügt man über ein größeres Hintergrundwörterbuch als das Vokabular, können mit dem eben beschriebenen Verfahren aus diesem Wörterbuch ebenfalls die Head-Modelle bestimmt werden (**D***ictionary*).

Dabei kann beispielsweise das Vokabular des Spracherkenners aus dem Wörterbuch entfernt, und dann die Präfixe der Länge h aus den Wörtern mit mindestens $h + t$ (min_l) Phonemen erzeugt werden. Dadurch werden dann nur Präfixe für die Head-Modelle erzeugt, die aus OOV-Wörtern stammen.

Für die späteren Experimente auf englischen BN-Daten wurde aus der Liste aller Wörter des Sprachmodellkorpus, über dem auch das Vokabular definiert ist, mit dem Graphem-zu-Phonem-Konverter von Fisher [26] ein solches Wörterbuch erstellt.

Als Bedingung für die Verwendung eines Wortes bei der Erzeugung der Head-Modelle wurde noch eine Mindest- (min_f) und Maximalhäufigkeit (max_f) im Textkorpus verwendet. Wurden beispielsweise für die BN-Domäne nur die Wörter verwendet, die mindestens zweimal und maximal 30mal aufgetreten sind, so wurden die Präfixe nur aus OOV-Wörtern erzeugt. Dies hat den Vorteil, dass nun auch Head-Modelle generiert werden, die Anfänge von unbekannten Wörtern modellieren, die *kein* Präfix von Wörtern aus dem Vokabular des Spracherkenners darstellen.

Wird die maximale Häufigkeit auf 120 gesetzt, werden ungefähr 20000 Wörter aus dem Vokabular V_{Top40k} des Spracherkenners beim Erzeugen der Head-Modelle mitberücksichtigt.

Der zuvor beschriebene Ansatz, der *allein* das Vokabular des Spracherkenners benutzt, stellt somit einen Spezialfall dieser Methode dar.

Vokabular-optimierte Head-Modelle

Das nun folgende Verfahren passt die Menge der Head-Modelle an das verwendete Vokabular so an, dass diese Head-Modelle genau das längste Präfix von OOV-Wörtern aus dem Hintergrundwörterbuch repräsentieren, das mit einem Präfix aus dem Vokabular übereinstimmt (OPT).

Das hat zur Folge, dass die so gewonnenen Head-Modelle im Gegensatz zu den zuvor beschriebenen Verfahren unterschiedliche Längen besitzen. Die zugrundeliegende Idee dabei ist, dass unbekannte Wörter, die sich erst nach m Phonemen vom Vokabular unterscheiden, bis zu dieser Länge exakt modelliert werden sollen. Sonst kann dies zu Suchfehlern führen, weil das generische Wort aus dem Suchraum entfernt wird, obwohl die

Bewertungen für die Wörter aus dem Vokabular zu einem späteren Zeitpunkt schlechter ausfallen. Außerdem wird dadurch ein großer Teil eines unbekannten Wortes durch exakte Phonem-Modelle repräsentiert.

Das für die Erzeugung der Head-Modelle verwendete Verfahren, die in den Experimenten in Kapitel 7 untersucht werden, stellt zusätzlich die folgenden Bedingungen an die erzeugten Head-Modelle: Die Länge eines Head-Modells beträgt mindestens min_p und maximal max_p Phoneme. Dabei werden zu lange Phonemfolgen auf max_p gekürzt. Bei zu kurzen Phonemfolgen werden diese, wenn die Phonemfolge des Wortes nicht zu kurz ist, auf min_p verlängert. Ist das nicht möglich, wird das Wort nicht für die Erzeugung eines Head-Modells benutzt. Weiter gilt, dass die erzeugte Phonemfolge mindestens t Phoneme kürzer sein muss, als die Phonemfolge des gesamten Wortes. Diese Bedingung *muss* gelten, um sicherzustellen, dass nicht versucht wird, ganze OOV-Wörter zu modellieren. In dieser Arbeit ist $t = 2$ gesetzt. Bei Phonemfolgen, die diese Bedingung nicht erfüllen, wird die Phonemfolge entsprechend gekürzt. Wird dabei die Mindestlänge unterschritten, wird das Wort nicht für die Erzeugung eines Head-Modells verwendet.

Für die so gewonnene Menge an Phonemfolgen wird nun noch eine Mindesthäufigkeit min_f angewendet, bei der nur Phonemfolgen für die Head-Modelle herangezogen werden, die sich aus mindestens min_f verschiedenen Wörtern ergeben haben. Generische Wörter, die mit diesem Verfahren erzeugt wurden, werden im Experiment Kapitel 7.1.4 als OPT-1 bezeichnet.

Phonemfolgen der so erzeugten Head-Modelle ($min_p = 0$) sind in der Regel Präfix eines Wortes aus dem Vokabular des Spracherkenners und passen somit optimal in einen bestehenden Aussprache-Präfix-Baum für das verwendete Vokabular. Wird eine Mindestlänge für die Head-Modelle $min_p > 0$ verwendet, stellt dies eine Kombination mit dem zuvor beschriebenen Wörterbuchverfahren dar, in dem auch Head-Modelle entstehen, die kein Präfix eines Wortes aus dem Vokabular sind.

Das Ziel, die Head-Modelle so kompakt zu gestalten, dass sie die bestehenden Suchraumstrukturen möglichst mit Wörtern aus dem Vokabular teilen, ist zwar gewünscht, da sich die Zahl der Zustände nur geringfügig erhöht, für die Detektion unbekannter Wörter ist es aber wünschenswert, den Abstand zum bekannten Wort zu erhöhen und dafür die Bewertung eines generischen Wortmodells bei Auftreten eines unbekannten Wortes zu verbessern. Dies kann dadurch erreicht werden, dass die Head-Modelle in einem Teil des erzeugenden OOV-Wortes, der nicht mehr zum Vokabular des Spracherkenners passt, verlängert werden. Durch diese Verlängerung der Präfixe unterscheiden sie sich stärker vom verwendeten Vokabular. Der Parameter oov_p gibt an, um wie viel Phoneme ein Präfix in die OOV-Wörter verlängert werden darf.

Die Zahl der verschiedenen Präfixe steigt dadurch natürlich weiter an. Bei Verlängerung der Präfixe des oben beschriebenen Verfahrens um ein Phonem ($oov_p = 1$), würde sich die Zahl der sich daraus ergebenden neuen Präfixe um maximal einen Faktor von $|P| - 1$ erhöhen, und mit jedem weiteren Phonem um den Faktor $|P|$. Da sich das Präfix

vom bekannten Vokabular auch in einem exakt modellierten Phonem unterscheidet, wird erwartet, dass diese Modelle in den Experimenten die höchste Präzision aufweisen werden.

Bei diesem erweiterten Verfahren zur Erstellung für vokabularoptimierte Head-Modelle werden die bereits beschriebenen gleichen Randbedingungen an die Präfixe gestellt. So haben die mit OPT-2 bezeichneten Präfixmodelle in Kapitel 7 wieder eine Mindestlänge von drei und eine Maximallänge von sechs. Das Wort muss mindestens um zwei Phoneme länger sein als das erzeugte Präfix und es wurden nur die Präfixe in die Head-Modelle aufgenommen, die aus mindestens zwei verschiedenen Wörtern hervorgegangen sind.

Auch hier sollen mit den folgenden Abbildungen die Verfahren beispielhaft illustriert werden.

In Abbildung 6.16 ist ein Aussprache-Präfix-Baum dargestellt, der auf einem großen Wörterbuch beruht. Wörter, die zum Vokabular des Spracherkenners gehören, sind mit *wx*, Wörter, die zu den OOV-Wörtern zählen, sind mit *ox* bezeichnet. Die Knoten (Phoneme), die allein zu einem OOV-Wort gehören, sind durch einen Kreis in einem Quadrat dargestellt.

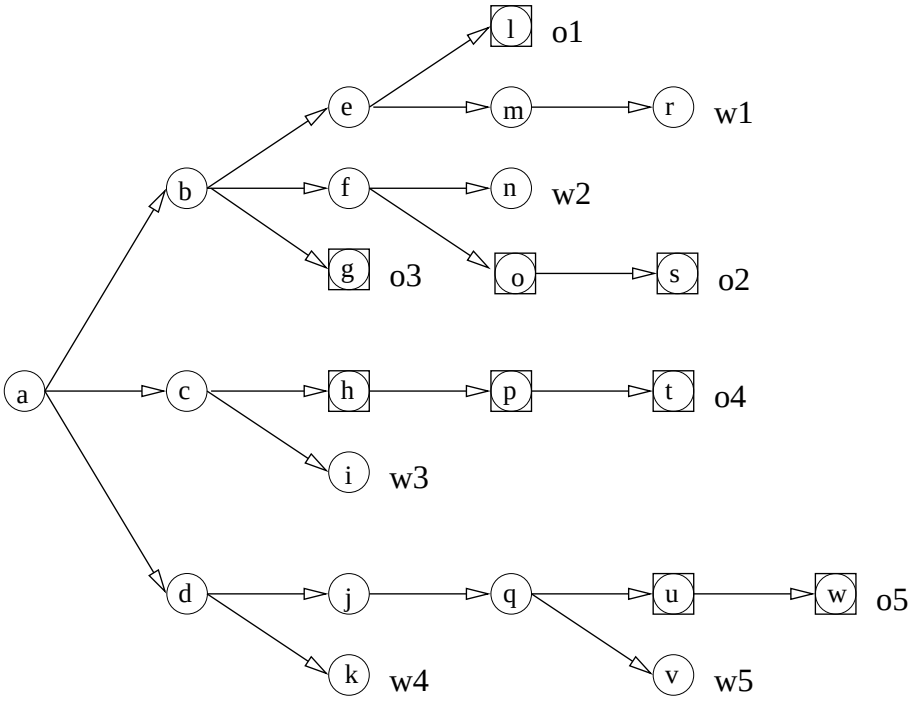

Abbildung 6.16: Als Aussprache-Präfix-Baum organisiertes Basisvokabular mit zusätzlichem OOV-Vokabular

Für die Erzeugung der optimierten HT-Modelle wird eine Präfixlänge von $min_p = 2$

vorausgesetzt. Das entstehende Präfix muss außerdem mindestens zwei Phoneme kürzer sein als das erzeugende Wort ($t = 2$). Aus diesem Grund wird in Abbildung 6.17 das Wort $o3$ nicht für die Erzeugung eines Head-Modells benutzt. Der Präfix aus dem Wort $o1$ wurde gekürzt, weshalb ein gestrichelter Pfeil von Knoten "b" anstelle von Knoten "e" in die generischen akustischen Modelle führt. Für dieses Beispiel (OPT-1) wurde die Mindestanzahl der Worte, die den Präfix erzeugen zur übersichtlicheren Darstellung als eins angenommen. Außerdem wurde der Präfix **nicht** in die OOV-Wörter verlängert ($oov_p = 0$, $min_f = 1$).

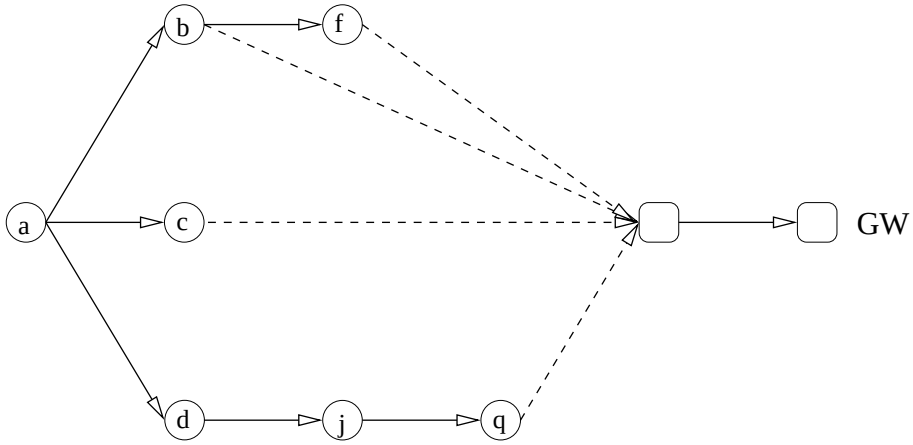

Abbildung 6.17: Optimiertes Head-Tail-Modell ohne verlängertem Präfix (OPT-1)

Dies wurde für die Abbildung 6.18 (OPT-2) gemacht. Hier wurde eine Verlängerung um ein Phonem benutzt ($oov_p = 1$), was dazu führt, dass der Zustand "h", der eine exakte akustische Modellierung eines bestimmten Phonems darstellt, in das generische Modell mit aufgenommen wird.

Die Abbildungen 6.19 und 6.20 zeigen dann die resultierenden Suchstrukturen, wenn man das Vokabular des Spracherkenner um die generischen Modelle OPT-1 oder OPT-2 erweitert.

6.2.3 Analyse der Präfixe in Abhängigkeit vom verwendeten Vokabular

In diesem Abschnitt wird die Anzahl verschiedener Präfixe untersucht, die sich ergibt, wenn bei den optimierten Verfahren die Größe des Vokabulars des Spracherkenners variiert wird. Ferner wird untersucht, wie beim Vokabular V_{Top40k} sich die Präfixe auf die Zahl der Ursprungswörter auf die verschiedenen Präfixlängen verteilen.

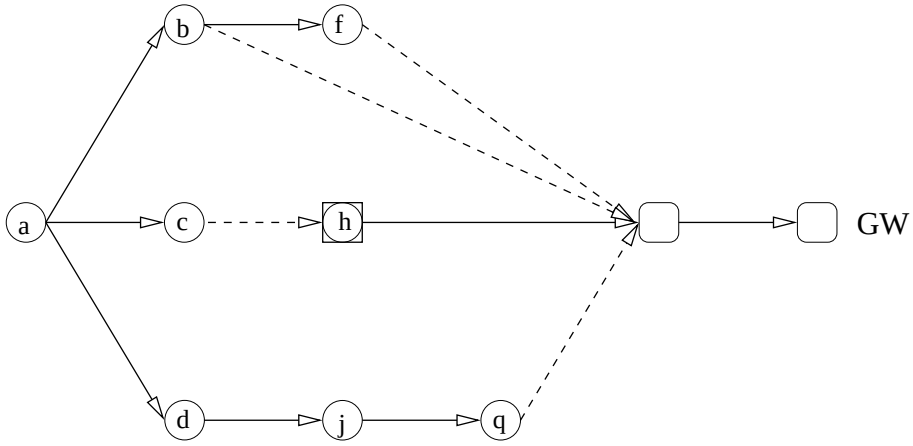

Abbildung 6.18: Optimiertes Head-Tail-Modell mit verlängertem Präfix (OPT-2)

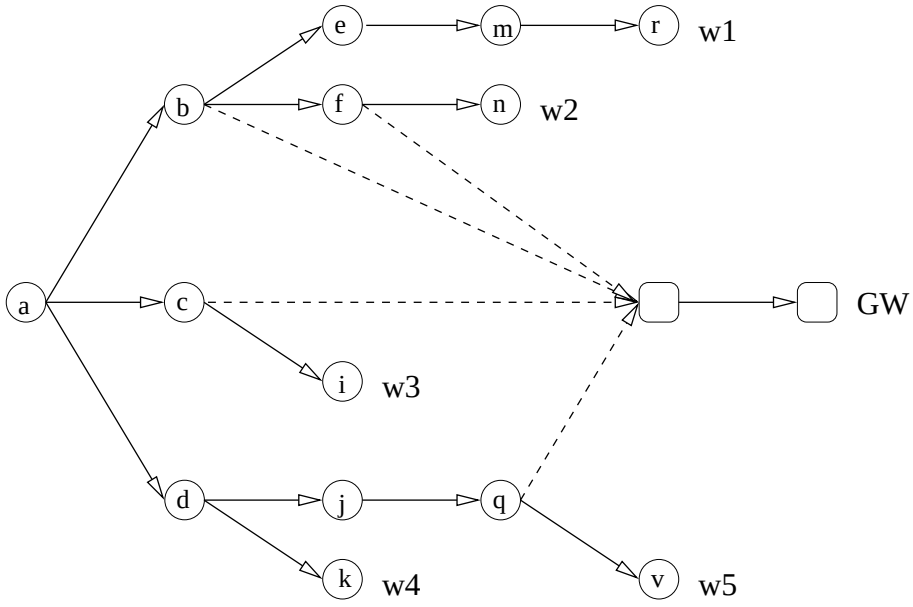

Abbildung 6.19: Optimiertes Head-Tail-Modell ohne verlängertem Präfix (OPT-1) und Vokabular

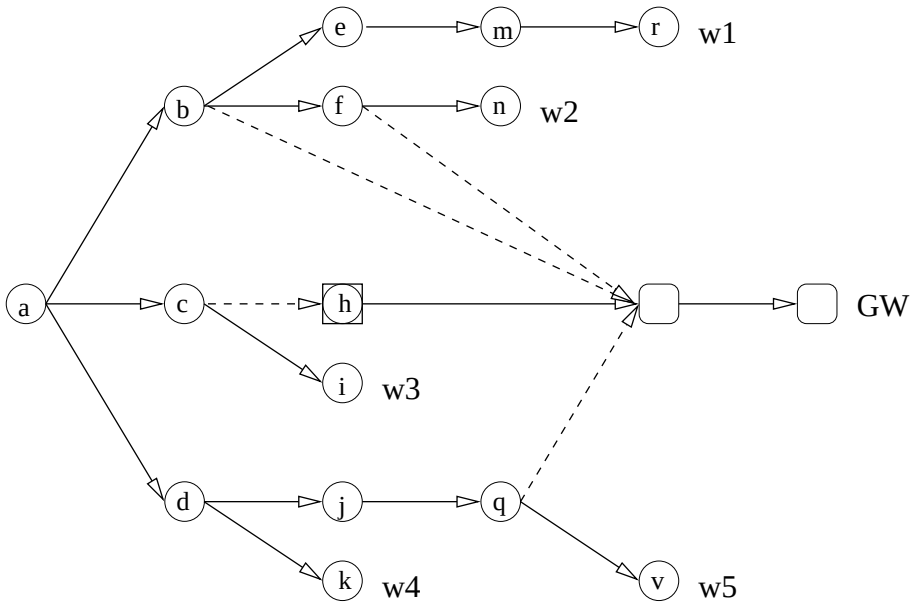

Abbildung 6.20: Optimiertes Head-Tail-Modell mit verlängertem Präfix (OPT-2) und Vokabular

Untersuchung der Präfixanzahl

Zunächst wird bei wachsender Vokabulargröße untersucht, wie viele verschiedene Präfixe sich jeweils ergeben, wenn das Vokabular V_{TopN} von V_{Top1} auf V_{Top40k} vergrößert wird. Die Präfixe werden dabei jeweils auf dem konstanten Vokabular $V_{\text{UNK}} = V_{\text{Top120k}} \backslash V_{\text{Top40k}}$ bestimmt. Die Anzahl der Präfixe entspricht dabei der Anzahl verschiedener Head-Modelle mit den Parametern $oov_c = 0$ und $min_p = 0$. Das Wörterbuch für diese Untersuchung wurde mit dem Graphem-zu-Phonem-Übersetzer von Fisher [26] generiert.

Wie man in Abbildung 6.21 sehen kann, wächst die Zahl der Präfixe mit dem Vokabularumfang. Für ein kleines Vokabular ist die Anzahl der verschiedenen Präfixe ebenfalls klein, was aufgrund der Konstruktion der Präfixe nicht anders zu erwarten ist.

Das Zahlenverhältnis von Präfixen und Vokabularumfang ist dabei bei kleinem Vokabular ungünstiger als bei einem umfangreichen Vokabular. Aufgrund der Beziehung $\text{Pref}_{\text{UNK-TopN}} \subset \text{Pref}_{TopN}$ ist die Anzahl der möglichen Präfixe und damit der Head-Teile durch das verwendete Vokabular nach oben hin beschränkt. Die maximale Zahl möglicher Präfixe ist dabei die Anzahl der Knoten des Aussprache-Präfix-Baum des verwendeten Vokabulars.

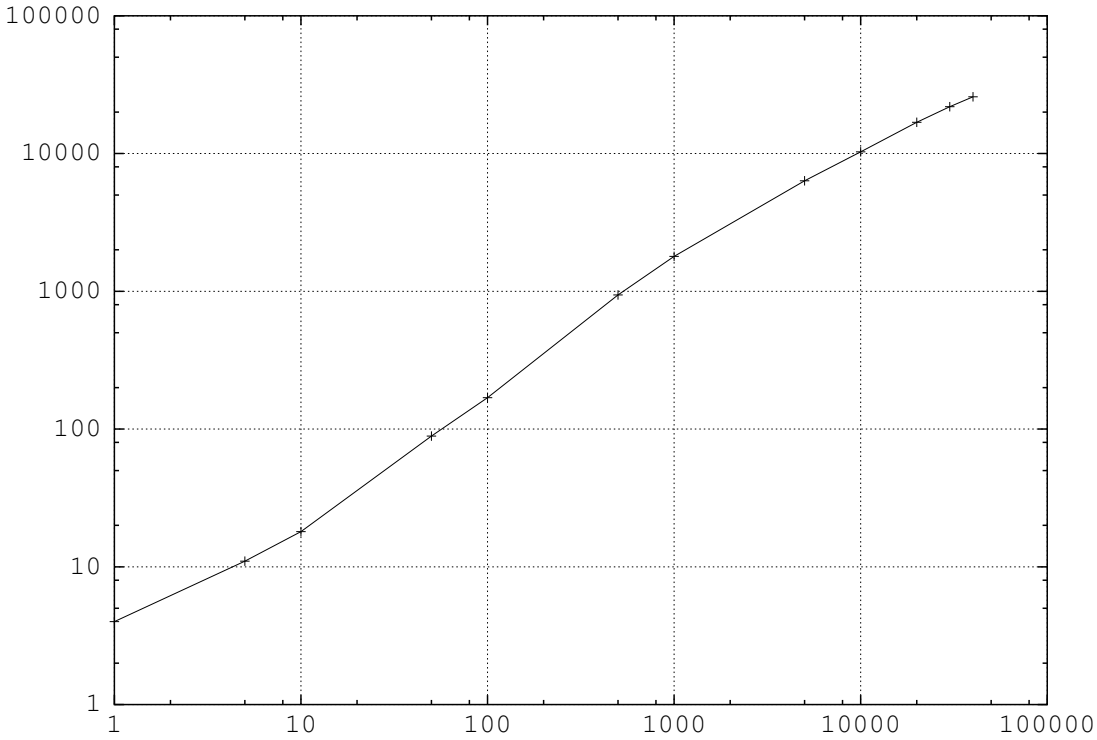

Abbildung 6.21: Anzahl Präfixe über der Vokabulargröße

Untersuchung der Längenverteilung der Präfixe

Für die Präfixe, die zum Vokabular V_{Top40k} gehören, wurde die Längenverteilung der Präfixe und abhängig davon die Zahl der OOV-Wörter bestimmt, aus denen diese Präfixe hervorgegangen sind. Wie man in Tabelle 6.4 sehen kann, gibt es im Vokabular V_{UNK} beispielsweise 5277 Wörter, die mit einem Wort aus V_{Top40k} in 3 Phonemen übereinstimmen, die sich also erst mit dem vierten Phonem vom Vokabular unterscheiden.

Wie man deutlich sehen kann, machen die Präfixe der Länge vier den größten Anteil an der Menge aller Präfixe aus. Die meisten Wörter aus V_{UNK} werden jedoch auf Präfixe der Länge drei abgebildet, unterscheiden sich also erst nach dem dritten Phonem von jedem Wort aus V_{Top40k}. Das bedeutet, dass es erst nach drei Phonemen möglich ist, festzustellen, dass es sich um ein unbekanntes Wort handelt.

Als nächstes werfen wir einen Blick auf die Längenverteilung der Präfixe.

Hier sei nochmals darauf hingewiesen, dass ein Präfix der Länge N bedeutet, dass die betreffenden Wörter mit dem $N+1$ten Phonem als unbekanntes Wort identifiziert wurden.

Betrachten wir die Ergebnisse in Tabelle 6.4 fällt zunächst auf, dass die Anzahl der verschiedenen Präfixe ansteigt, ein Maximum bei der Länge vier erreicht und dann wieder fällt. Es ist naheliegend, dass es mit steigender Länge der Präfixe auch mehr Möglichkei-

Gleiche Phoneme	Anzahl Präfixe	Unbekannte Wörter
1	30	306
2	602	5277
3	4046	**19804**
4	7206	18896
5	5735	10330
6	3900	5906
7	2455	3257
8	1642	2065
9	1076	1319
10	641	781
11	324	378
12	170	201
13	74	83
14	22	26
15	3	3
16	4	4
17	2	2

Tabelle 6.4: Präfixverteilung

ten gibt, Präfixe zu bilden. Um genau zu sein, wächst die Anzahl der möglichen Präfixe exponentiell schnell mit 39^N. Das Abflachen der Kurve ist eine Folge der Längenverteilung des Vokabulars.

Bezeichnung der Head-Tail-Modelle

Tabelle 6.5 gibt einen Überblick über die in Kapitel 7.1.4 verwendeten verschiedenen Head-Modelle.

Methode	Name	min_p	max_p	oov_p	min_c	min_l	min_f	max_f
VC	VC-2	2	2	-	-	5	-	-
	VC-3	3	3	-	-	5	-	-
	VC-4	4	4	-	-	6	-	-
IVP	IVP-3	3	3	-	1	5	-	-
	IVP-4	4	4	-	1	6	-	-
D	D-120-3	3	3	-	1	5	2	30
	D-120-4	4	4	-	1	6	2	30
OPT	OPT-1	3	6	0	2	5	2	30
	OPT-2	3	6	1	2	5	2	30

Tabelle 6.5: Übersicht der erzeugten Head-Modelle für den BN-Task

6.3 Zusammenfassung

In diesem Kapitel wurden verschiedene Sprachmodelle beschrieben, die bei den Untersuchungen in Kapitel 7 verwendet werden.

Dabei wurde eine neu entwickelte Methoden zur Modellierung unbekannter Wörter vorgestellt. Es handelt sich um ein Gruppierungsverfahren für unbekannte Wörter (GrOOV), das unüberwacht zu einem gegebenen Sprachmodellkorpus ein Klassensystem bestimmt, das für die Modellierung unbekannter Wörter in einem statistischen Sprachmodell geeignet ist. Dazu wird eine kleine Menge an Repräsentanten von potentiell unbekannten Wörtern (pUNK-Wörter) beobachtet, die durch ein datengetriebenes Clustern zusammen mit Wörtern des Vokabulars in Gruppen geballt werden.

Um die verschiedenen Eigenschaften der OOV-Wörter bei der Modellierung zu berücksichtigen, erzeugt das Verfahren einen speziell selektierten Korpus. Durch die Korpusselektion wird vermieden, dass fast alle pUNK-Wörter in einer einzigen Klasse konzentriert werden.

Es wurden Kriterien zur Bestimmung geeigneter Klassen zur Modellierung unbekannter Wörter untersucht, mit denen sichergestellt werden kann, dass genügend unbekannte Wörter in einer Klasse enthalten sind, damit diese sich nicht überspezialisiert und trainiert werden kann.

Da das Verfahren automatisch arbeitet, eignet es sich insbesondere für komplexe Domänen mit umfangreichem Datenmaterial, die sich manuell nicht mehr bearbeiten lassen, wie beispielsweise die englische BN-Domäne.

Das GrOOV-Verfahren wurde auf dem BN-Korpus durchgeführt. Für die Modellierung unbekannter Wörter wurden alle Klassen ausgewählt, die mindestens 50 der pUNK-Wörter enthalten. Das ergab 72 Klassen, bei denen sich zeigte, dass die OOV-Wörter sowohl nach syntaktischen als auch nach semantischen "Kategorien" in Klassen eingeteilt wurden. Es lassen sich dabei Klassen identifizieren, die, wie beim NE-Task, Namen von Personen, Organisationen oder Lokationen modellieren.

Besonders interessant aber sind dabei domänenspezifische Klassen, wie beispielsweise eine Klasse, in der sich Namen für Krankheiten und Begriffe aus dem medizinischen Fachbereich gesammelt haben. Eine solche Klasse ist beispielsweise beim NE-Task nicht vorgesehen, aber in der BN-Domäne sind unbekannte Wörter aus diesem Bereich so häufig aufgetreten, dass mit dem GrOOV-Verfahren eigens eine Klasse dafür entstand.

Der zweite Teil des Kapitels befasst sich mit neuen Verfahren zur akustischen Modellierung von unbekannten Wörtern, bei denen die Eigenschaften der exakten Phonemschleife und der flachen akustischen Modellen kombiniert werden. Die Head-Modelle der Head-Tail-Modelle werden von Vokal-Konsonanten-Folgen weiterentwickelt, um die Präfixe von unbekannten Wörtern kompakt repräsentieren zu können.

Es wird eine Methode vorgestellt, die aus einem großen Hintergrundwörterbuch zu einem gegebenen Vokabular eine für das Vokabular optimierte Menge an Head-Modellen erzeugt, die sich kompakt in den durch das Vokabular definierten Suchraum einfügen lässt (OPT-1).

Bei Methode OPT-2 wird der Abstand zum bekannten Vokabular durch das Verlängern der Head-Modelle um Phoneme eingeführt, die den Abstand zum bekannten Vokabular vergrößern.

Eine besondere Eigenschaft der optimieren Head-Modelle ist, dass im Gegensatz zu den Vokal-Konsonanten-Folgen oder zur Phonemschleife, die Zahl der hinzugefügten generischen Wörter sich an die Größe des verwendeten Vokabulars anpasst. So entstehen bei einem kleinen Vokabular wenige Head-Modelle und bei großem Vokabular entsprechend mehr. Die vorgestellten Head-Tail-Modelle besitzen verschiedene Eigenschaften in Bezug auf das erforderliche Wissen. So wird für die optimierten Head-Modelle ein Hintergrundwörterbuch und das Vokabular des Spracherkenners benötigt, bei den Vokal-Konsonanten-Head-Tail-Modellen nur eine Einteilung des Phonemsatzes in Vokale und Konsonanten.

Kapitel 7

Experimente

In diesem Kapitel wird untersucht, wie sich der Einsatz der im Rahmen dieser Arbeit entwickelten Techniken, wie Head-Tail-Modelle und die unüberwacht gelernten Klassen auf die Qualität der Erkennungsleistung auswirkt. Dabei wird das Verhalten dieser Modelle anhand der spontan-sprachlichen GSST-Domäne und der englischen BN-Domäne genauer betrachtet.

Bei der GSST-Domäne, für die ein aufwendig erzeugtes Sprachmodell mit semantischen Klassen eingesetzt wird, zeigt sich, dass die Head-Tail-Modellierung gute Ergebnisse bei der Erkennungsleistung liefert. Die dort eingesetzten Head-Modelle sind als Vokal-Konsonanten-Folgen realisiert, und es werden verschiedene Möglichkeiten der akustischen Modellierung dieser Head-Modelle in einer Untersuchung vorgestellt.

Für die BN-Domäne werden die in Kapitel 6.2 weiterentwickelten Head-Tail-Modelle untersucht und wie sich die verbesserten Head-Tail-Modelle verhalten, wenn sie mit den Klassen aus dem GrOOV-Verfahren kombiniert werden.

Um abzuschätzen, wie viele unbekannte Wörter überhaupt gefunden werden können und wo die Leistungsgrenzen der Detektion und dem Lernen unbekannter Wörter liegen, werden auf dem englischen BN-Task ORAKEL-Experimente[1] durchgeführt, bei denen das Wissen über fehlende Wörter auf der Testmenge "orakelhaft" eingesetzt wird. Die unbekannten Wörter werden für diese Untersuchungen in das Vokabular des Spracherkenners aufgenommen.

7.1 Akustische Modellierung von Head-Tail-Modellen

In diesem Abschnitt werden verschiedene Möglichkeiten untersucht, Head-Tail-Modelle mit den akustischen Modellen des Spracherkenners zu kombinieren. Die Experimente wurden mit einem deutschen Spracherkenner für die GSST-Domäne auf den Top30HH-Daten durchgeführt (vgl. Kapitel 3.2).

[1] Cheating-Experimente

Um die Auswirkung der Head-Tail-Modelle auch mit spontanen Daten zu untersuchen, wurden dazu die GSST-Daten Eval99.1 benutzt. Das statistische Sprachmodell des Spracherkenners besitzt eine semantische Klasse für Nachnamen, um die Head-Tail-Modelle zu integrieren.

Folgende Untersuchungen wurden durchgeführt:

- Grenzen mit einem ORAKEL-Experiment bestimmen
- Auswirkung der Länge des verwendeten Head-Modells
- Auswirkung der Länge des verwendeten Tail-Modells
- Verschieden exakte akustische Modelle im Head-Modell
- Auswirkungen auf das mittlere Realzeit-Verhalten

Eine inkrementelle Adaption des GSST-Spracherkenners wurde für die Experimente auf den Top30HH-Daten nicht benutzt.

7.1.1 Untersuchung der Top30HH-Daten mit einem ORAKEL-Experiment

Zunächst wurde auf den Top30HH-Daten in einem Experiment (BASELINE-Experiment) untersucht, wie groß die Auswirkung der unbekannten Namen auf die Fehlerrate ist und wo die Grenzen liegen. Hierzu wurde zunächst der Spracherkenner ohne Erweiterung der Head-Tail-Modelle und ohne das Aufnehmen der fehlenden Wörter in das Vokabular eingesetzt. Aufgrund der großen Anzahl unbekannter Wörter kam es zu einer hohen Wortfehlerrate von 38.9%, die auch der Klassenfehlerrate (CER) entspricht.

Um abzuschätzen, wie viele der unbekannten Wörter überhaupt gefunden werden können, wurden in einem weiteren Versuch (ORAKEL-Experiment) die fehlenden Wörter in das Vokabular des Spracherkenners aufgenommen und mit der gleichen klassenbedingten Wahrscheinlichkeit versehen, wie sie in den späteren Experimenten bei den Head-Tail-Modellen verwendet werden. Head-Tail-Modelle selbst waren aber nicht im Vokabular des Spracherkenners enthalten.

Die resultierende Klassenfehlerrate von 3.4%[2] bei einem Recall (REC) von 96.6% zeigt, dass auf diesen Daten mit den geeigneten akustischen Umschriften fast alle der gesuchten Wörter gefunden werden können (vgl. Tabelle 7.1).

Vergleicht man beide Ergebnisse, stellt man fest, dass bei dem BASELINE-Experiment 2240 Fehler und im ORAKEL-Experiment nur 195 Fehler gemacht, dafür aber 1159 der Namen korrekt gefunden wurden. Rechnet man dies nun um auf Fehler je unbekanntem Wort,

[2] in [65] ist aufgrund eines Tippfehlers 0.4% angegeben

System	CER	REC	PRC
BASELINE	38.9%	-	-
ORAKEL	3.4%	96.6%	100.0%

Tabelle 7.1: Ergebnis auf Top30HH-Daten (Auszug)

ergibt sich, dass im Mittel 1.8 Fehler pro unbekanntem Wort erzeugt wurden, dabei entfällt ein Fehler auf das unbekannte Wort; 0.8 Fehler sind Folgefehler. Das Ergebnis liegt damit in der gleichen Größenordnung wie die Ergebnisse der Untersuchungen von Hetherington [42].

7.1.2 Verwendete Head-Tail-Modelle

Für die akustische Modellierung des Tail-Modells wird ein generisches akustisches Modell verwendet, das mit Sprachdaten von allen Phonemen trainiert wurde und über drei HMM-Zustände verfügt (generisches Phonem). Für die Experimente mit verschieden langen Head-Modellen wurde zunächst eine feste Länge für die Tail-Modelle von zwei generischen Phonemen festgelegt.

Es wurden Kombinationen aus abwechselnd Vokalen und Konsonanten, beziehungsweise Konsonanten und Vokalen, für die Modellierung der Head-Modelle eingesetzt (Kapitel 6.2). Für die in den Head-Modellen verwendeten Phoneme wurden akustische Modelle benutzt, die diese Phoneme mit unterschiedlicher Genauigkeit modellieren. Zum einen wurden für die Phoneme kontextunabhängige akustische Modelle (CI) benutzt, die extra für diesen Einsatz trainiert wurden und zum anderen kontextabhängige akustische Modelle (CD), wie sie auch in der Modellierung der zum Vokabular gehörenden Wörter des Spracherkenners eingesetzt werden. Bei der kontextabhängigen Modellierung an den Wortgrenzen handelt es sich um ein Cross-Word-Triphone-Modell (XWT).

Abhängig von der Länge der Head-Modelle ergeben sich unterschiedliche Mengen an Head-Tail-Modellen. Bei einer Länge von null entsteht nur ein einziges Head-Tail-Wort, was der Modellierung mit einem flachen Modell für unbekannte Wörter entspricht. Bei einer Länge von einem Phonem erhält man 39 verschiedene Modelle der generischen Wörter. Bei einer Länge von zwei Phonemen ergeben sich aufgrund der Einschränkung auf Vokal-Konsonanten-Kombinationen nur **780** verschiedene generische Wörter. Wären alle Kombinationen erlaubt, hätten sich $39 * 39 = 1521$ Modelle ergeben, von denen natürlich viele aufgrund der Phonotaktik nicht plausibel sind.

Die untersuchten Kombinationen der Längen der Head-Modelle mit den verwendeten akustischen Modellen sind mit der Festlegung ihres Systemnamens in Tabelle 7.2 aufgeführt.

Experiment	Länge Head-Modell	erstes Phonem	zweites Phonem
GW1	0	-	-
GW38-CI	1	CI	-
GW38-XWT	1	XWT	-
GW**780**-CI-CI	2	CI	CI
GW780-XWT-CI	2	XWT	CI
GW780-XWT-CD	2	XWT	CD

Tabelle 7.2: In den Head-Modellen verwendete akustische Modellierung

Akustische Modellierung der Head-Modelle

In Tabelle 7.3 sind die Ergebnisse der oben beschriebenen Experimente zusammengefasst. Wie erwartet werden mit längeren Head-Modellen auch mehr unbekannte Wörter gefunden. Insbesondere führte die flache Modellierung (Länge 0) zum schlechtesten Ergebnis. Außerdem ist zu sehen, dass es wichtig ist, die gleichen kontextabhängigen akustischen Modelle für die Head-Modelle zu verwenden, wie sie auch für die bekannten Wörter im Vokabular des Spracherkenners benutzt werden. Vergleicht man den Recall (REC) von Experiment GW38-XWT, bei dem die Länge des Head-Modells nur eins beträgt, dafür aber ein kontextabhängiges akustisches Modell für das Phonem benutzt wird, mit dem Experiment GW780-CI-CI, bei dem das Head-Modell zwar eine Länge von zwei Phonemen hat, aber dafür beide Phoneme mit kontextunabhängigen akustischen Modellen realisiert sind, wird dies besonders deutlich. So verdoppelt sich der Recall von 19,2% auf 38,8% durch die Verlängerung um ein CI-modelliertes Phonem, aber zwei CI-modellierte Phoneme führen zu deutlich schlechteren Ergebnissen als nur ein einziges XWT-modelliertes Phonem.

System	Länge Head-Modell	CER	REC	PRC
BASELINE	-	38,9%	-	-
GW1	0	38,7%	2,2%	100,0%
GW38-CI	1	36,9%	8,7%	100,0%
GW38-XWT	1	34,2%	**19,2%**	100,0%
GW780-CI-CI	2	35,2%	**12,8%**	99,4%
GW780-XWT-CI	2	27,0%	**38,8%**	99,8%
GW780-XWT-CD	2	21,1%	58,9%	100,0%
ORAKEL	-	3,4%	96,6%	100,0%

Tabelle 7.3: Ergebnis auf Top30HH-Daten

Bei Systemen, die den Suchraum stark beschneiden müssen, um ein Ergebnis schnell zu erreichen, ist für den Fall, dass bereits beim ersten Phonem ein Kandidat für ein unbe-

kanntes Wort aus dem Suchraum entfernt wird, dieses später auch nicht mehr zu finden. Aufgrund der weniger genauen Modellierung durch kontextunabhängige akustische Modelle kann dies aber sehr viel früher geschehen als bei der Verwendung von gleich guten akustischen Modellen in den Head-Modellen. Da der Spracherkenner Cross-Word-Triphone-Modelle für seine Wörter benutzt, ist es somit besonders wichtig, hier ebenfalls die gleichen Modelle zu verwenden wie für das restliche Vokabular.

Außerdem fällt die mittlere Bewertung eines passenden Head-Tail-Wortes, das kontextabhängig modellierte Phoneme benutzt, über den Bereich eines unbekannten Wortes besser aus als bei Verwendung kontextunabhängiger akustischer Modelle, was natürlich diesem Wort insgesamt eine höhere Wahrscheinlichkeit zuordnet.

Dies erklärt ebenfalls, warum die Verwendung von zwei kontextabhängigen Phonemen im Head-Teil (GW780-XWT-CD) hier die besten Ergebnisse liefert. Verglichen mit dem ORAKEL-Experiment wurde auf diesen Daten mehr als 50% der Reduktion der Klassenfehlerrate (CER) erreicht. Insgesamt machte das System 1215 Fehler und konnte 707 der unbekannten Wörter detektieren. Daraus ergibt sich, dass die Zahl der Folgefehler um 37% (0.3 absolut) je gefundenem Wort reduziert werden konnte. Diese Reduktion der Folgefehler ist vergleichbar mit den Ergebnissen der Untersuchungen von Fetter [22].

Akustische Modellierung der Tail-Modelle

Nachdem die beste Modellierung (GW780-XWT-CD) für das Head-Modell gefunden ist, wird der Einfluss untersucht, den die Länge des Tail-Modells auf das Erkennungsergebnis hat. Hierzu wurde die Länge der Tail-Modelle von null bis vier variiert und auf den Top30HH-Daten die Klassenfehlerrate (CER) und die Wortfehlerrate (WER) bestimmt. Die Ergebnisse dazu sind in Tabelle 7.4 zusammengefasst.

Länge Tail-Modell	CER	WER
0	26,3%	42,6%
1	21,0%	35,5%
2	20,7%	34,2%
3	21,1%	33,4%
4	22,5%	33,3%

Tabelle 7.4: Verschiedene Längen des Tail-Modells (Head-Modell GW780-XWT-CD)

Wie man sehen kann, hat die Länge des Tail-Modells einen deutlichen Einfluss auf die Fehlerraten. Insbesondere führt ein Tail-Modell der Länge null zum schlechtesten Ergebnis, weil es noch zu vielen Folgefehlern kommt. Das beste Ergebnis bezüglich der Klassenfehlerrate wird mit einer Länge des Tail-Modells von zwei erreicht; das beste Ergebnis bezüglich

der Wortfehlerrate bei einer Länge von drei. Bei einer minimalen Wortfehlerrate entstehen die wenigsten Folgefehler, es wird aber nicht notwendigerweise auch das Optimum an unbekannten Wörtern detektiert. Die beste Verbindung von Detektion und Reduktion der Folgefehler wird mit dem Minimum der Klassenfehlerrate bei einer Länge des Tail-Modells von zwei erreicht. Insgesamt ist der Einfluss der Länge des Tail-Modells auf die Erkennungsleistung aber nicht so groß wie der Einfluss der Länge des Head-Modells.

7.1.3 Head-Tail-Modelle und spontane Daten

Die Erfahrungen und Ergebnisse der vorherigen Untersuchungen werden nun in einem weiteren Experiment auf deutsche spontane Daten übertragen.

Für das Experiment auf den GSST-Daten (Eval99.1) wird beim Tail-Modell die Länge zwei gewählt, da die Folgefehler, die sich aus den unbekannten Wörtern ergeben reduziert und möglichst viele unbekannten Wörter (optimales CER) gefunden werden sollen.

Das Head-Modell wird ebenfalls mit einer Länge von zwei und mit kontextabhängigen akustischen Modellen konstruiert (GW780-XWT-CD), da mit dieser Kombination die bisher besten Ergebnisse erzielt wurden.

In einem weiteren Experiment wurden die in den Testdaten fehlenden neun Namen, jedoch keines der Head-Tail-Modelle (ORAKEL2), in das Vokabular des Spracherkenners aufgenommen.

Ferner wurde das BASELINE-System, das keine Head-Tail-Modelle besitzt, auf den spontanen Daten getestet.

Bei allen durchgeführten Experimenten auf den spontanen Daten wurde auch die mittlere Laufzeit als Realtime-Faktor (RTF) auf einer 300MHz Sparc ermittelt, um die Auswirkung der Head-Tail-Modelle auf das Laufzeitverhalten zu untersuchen.

In Tabelle 7.5 sind die Ergebnisse der verschiedenen Untersuchungen gegenübergestellt. Bei der Verwendung der Head-Tail-Modelle konnte über 50% der mit dem ORAKEL2-Experiment erreichten Fehlerreduktion erzielt werden. Dabei wurden von den 46 Auftreten unbekannten Namen 26 korrekt gefunden und 13 an falscher Stelle hypothetisiert. Die Reduktion der Folgefehler je gefundenem unbekannten Wort liegt im gleichen Bereich wie auf den Top30HH-Daten.

Die Verwendung von Head-Tail-Modellen führt zu keinen Nachteilen bei den Fehlerraten und die Head-Tail-Modelle beeinträchtigen die mittlere Laufzeit des Spracherkenners nicht, wie ebenfalls aus Tabelle 7.5 zu ersehen ist.

Wegen der geringen Anzahl an unbekannten Namen auf den spontanen Daten liegt die Reduktion der Fehlerrate zwar im Bereich dessen, was bei einem Spracherkenner

System	CER	REC	PRC	WER	RTF
BASELINE	22,6%	-	-	22,6%	2,0
GW780-XWT-CD	22,2%	57%	77%	22,5%	2,0
ORAKEL2	21,9%	74%	100%	21,9%	2,0

Tabelle 7.5: Ergebnis auf Eval99.1 Daten

normalerweise nur als Rauschen angesehen werden kann, aber der Vergleich mit den ORAKEL2-Experimenten zeigt, dass durch das Hinzufügen der unbekannten Namen ebenfalls keine große Reduktion in der Fehlerrate zu erwarten ist. Die Ergebnisse bezüglich des Recall liegen bei ähnlichen Werten wie die bei den Top30HH-Daten, auf denen wegen der künstlich hohen Rate unbekannter Namen keine Aussage über die Präzision möglich ist. Dagegen sind die spontanen Daten umfangreich genug, um einen Rückschluss über die Anzahl fehlerhaft hypothetisierter unbekannter Namen zu erlauben. Daraus lässt sich schließen, dass es mit dieser Modellierung möglich ist, 57% - 59% der unbekannten Namen zu finden und dies mit einer Präzision von annähernd 77% erreicht werden kann, ohne dabei den Spracherkenner in der Laufzeit zu beeinträchtigen. Im Mittel kann hier je gefundenem unbekannten Wort die Zahl der Folgefehler um 37% reduziert werden.

Die bisher in diesem Kapitel beschriebenen Untersuchungen führen zu dem Schluss, dass für die akustische Modellierung der Head-Modelle die gleichen akustischen Modelle eingesetzt werden sollen, wie für die regulären Wörter des Spracherkenners. Das bietet zusätzlich den Vorteil, dass die Head-Modelle gut in eine Aussprache-Präfix-Baum-Struktur des Suchraums eingebettet werden können und somit die gleichen Zustände der regulären Wörter des Vokabulars teilen, wodurch sich auch ein kompakterer Suchraum ergibt.

7.1.4 Übertragung der Head-Tail-Modelle auf die BN-Domäne

Die Untersuchung der Head-Tail-Modelle wird nun auf der komplexen BN-Domäne weitergeführt, da es sich dabei um real gefundene Sprache handelt, die ein großes Vokabular voraussetzt und gegenwärtig einen wichtigen Referenztask der Spracherkennung darstellt. Die verwendeten BN-OOV-Daten, die einen erhöhten Anteil an OOV-Wörtern besitzen (1,86%), wurden in Kapitel 3.3 beschrieben.

Alle nun folgenden Ergebnisse sind mit einem festen Wert für das Gewicht des Sprachmodells und dem Wortübergangsstrafterm berechnet, für die das BASELINE-System das beste Ergebnis auf den DARPA-Evaluationsdaten Testset-1 1998 erzielte.

Wie aus Tabelle 7.6 zu ersehen ist, wächst die Zahl der Head-Modelle konstruktionsbedingt bei Vokal-Konsonanten-Folgen exponentiell. Bei einer Länge der Vokal-Konsonanten-Folgen von vier (VC-4) ergeben sich dann 430592 Head-Tail-Modelle. Somit würden bei einem Erkennervokabular von ungefähr 40000 Wörtern bei Verwendung

einer Wortmodellierung zehnmal so viele Head-Tail-Wörter für die Detektion unbekannter Wörter wie für die Erkennung bekannter Wörter benutzt werden. Bei einer Länge der Vokal-Konsonanten-Folgen von drei sind dies immerhin noch 20880 Head-Tail-Modelle.

System	Länge Head-Modell	# Heads
VC-2	2	982
VC-3	3	20880
VC-4	4	430592

Tabelle 7.6: Anzahl Head-Modelle in Abhängigkeit von der Länge

Für die Experimente auf dem BN-Task wurden deshalb Head-Modelle der Länge zwei und drei untersucht. Als Sprachmodell wurde das in Kapitel 6.1.3 beschriebene Einklassensprachmodell (LM-1) für unbekannte Wörter benutzt. Alle Head-Tail-Modelle wurden als Wörter in das Erkennervokabular aufgenommen und die für die Modellierung unbekannter Wörter vorgesehene Wahrscheinlichkeitsmasse wurde gleichmäßig auf alle Head-Tail-Wörter w_{HTM} verteilt. Die entsprechenden LM-Faktoren $p(w_{\mathrm{HTM}}|<\mathrm{OOV}>)$ sind in Tabelle 7.7 angegeben. w_{HTM} ist dabei eines der Head-Tail-Modelle.

System	# Heads	LM-Faktor	REC	PRC	CER
BASELINE	-	-	-	-	20,0%
VC-2	982	1/982	2,4%	60,0%	20,0%
VC-3	20880	1/20880	3,8%	63,6%	20,0%

Tabelle 7.7: Ergebnis auf BN-OOV-Daten

Wie man in Tabelle 7.7 sieht, verbessert sich sowohl die Präzision als auch der Recall, wenn längere Head-Modelle verwendet werden. Aufgrund des umfangreicheren Vokabulars wird im Weiteren eine Modellierung der Head-Modelle mit mindestens drei Phonemen durchgeführt. Dennoch erscheint ein Recall von 3.8% im Vergleich zu den Ergebnissen auf der GSST-Domäne als gering. Als Ursache können die verwendeten Head-Modelle und die Verwendung einer einzigen Klasse für die Modellierung unbekannter Wörter angesehen werden, weshalb im Folgenden eine Verbesserung der Head-Tail-Modelle und der Modellierung unbekannter Wörter im statistischen Sprachmodell untersucht wird.

Untersuchung der weiterentwickelten Head-Tail-Modelle

In diesem Abschnitt werden weitere Experimente mit Head-Tail-Modellen durchgeführt, zu deren Erzeugung in Kapitel 6 verschiedene Verfahren vorgestellt wurden. Die untersuchten

Head-Modelle sind in Tabelle 7.8 aufgeführt. Wie im vorangegangenen Abschnitt wird ein Tail-Modell der Länge zwei verwendet.

Head-Modelle	# Heads
VC-3	20880
IVP-3	4662
IVP-4	9696
D-1-120-3	7482
OPT-1	8912
OPT-2	9819

Tabelle 7.8: Verwendete Head-Modelle

Da die verschiedenen Head-Modelle ausführlich in Kapitel 6.2 Tabelle 6.5 beschrieben sind, folgt hier nur eine kurze Übersicht.

Head-Modell VC-3 bezeichnet die bereits verwendeten Vokal-Konsonanten-Folgen der Länge drei.

Die Head-Modelle IVP-3 und IVP-4 sind allein vom bekannten Vokabular des Spracherkenners abgeleitet, indem alle Präfixe der Länge drei bzw. der Länge vier der Wörter mit mindestens fünf (bzw. sechs) Phonemen als Head-Modelle benutzt werden.

Bei den Head-Modellen D-1-120-3, OPT-1 und OPT-2 wird ein großes Hintergrundwörterbuch mit Aussprachen für die OOV-Wörter des Sprachmodellkorpus benutzt. Die Head-Modelle für D-1-120-3 besitzen dabei die Länge drei und sind aus den Wörtern des Sprachmodellkorpus erzeugt, die mehr als einmal und weniger als 120-mal auftreten und mindestens aus fünf Phonemen bestehen. Wörter, die zwischen dreißig und 120-mal aufgetreten sind, gehören zum Vokabular des Spracherkenners (ca. 20000 Wörter).

Für OPT-1 wurden initiale Präfix-Phonem-Folgen aus dem gleichen Hintergrundwörterbuch mit OOV-Wörtern so bestimmt, dass sie Präfix eines Wortes aus dem Vokabular des Spracherkenners sind und diese Präfixe maximale Länge besitzen. Wird die minimale Länge von drei Phonemen unterschritten, wird der Phonem-Präfix entsprechend verlängert. Wird die maximale Länge von sechs Phonemen überschritten, wird der Präfix auf diese Länge gekürzt. Ist der Präfix nicht mindestens um zwei Phoneme kürzer als die ursprüngliche Phonemfolge, so wird die Phonemfolge entsprechend gekürzt. Die aus dem Hintergrundwörterbuch verwendeten OOV-Wörter sind im Sprachmodellkorpus mehr als einmal aufgetreten und besitzen mindestens sechs Phoneme. In die Menge der schließlich verwendeten Head-Modelle werden nur die Präfix-Phonemfolgen aufgenommen, die mindestens zweimal aufgetreten sind.

Die Head-Modelle mit der Bezeichnung OPT-2 sind ebenfalls für das Erkennervokabular optimiert, aber die Phonem-Folgen-Präfixe sind um ein Phonem verlängert. Das

initiale Präfix unterscheidet sich somit im letzten Phonem von allen Präfixen gleicher Länge, die aus dem Vokabular des Spracherkenners abgeleitet werden können. Es wird aber gegebenenfalls noch wie bei OPT-1 modifiziert, um die Mindest- oder Maximal-Längenbedingungen zu erfüllen.

Für die Untersuchung wurde wie bei den Experimenten im vorhergehenden Abschnitt das Sprachmodell LM-1 verwendet, und die, für die Modellierung unbekannter Wörter reservierte Wahrscheinlichkeitsmasse, gleichmäßig auf die jeweils verwendeten Head-Tail-Modelle verteilt.

Die Ergebnisse aus den Experimenten mit den verschiedenen Head-Tail-Modellen sind in Tabelle 7.9 zusammengefasst.

System	LM-Faktor	REC	PRC	CER
BASELINE	-	0,0%	-	20,0%
VC-3	1/20880	3,8%	63,6%	19,9%
IVP-3	1/4662	3,8%	40,0%	19,9%
IVP-4	1/9696	5,6%	52,5%	19,9%
D-1-120-3	1/7482	4,3%	47,1%	19,9%
OPT-1	1/8912	7,5%	54,9%	19,8%
OPT-2	1/9819	8,0%	61,2%	19,8%

Tabelle 7.9: Vergleich Head-Modelle mit LM-1 Sprachmodell (Tail-Länge 2)

Zunächst ist zu bemerken, dass alle Systeme bessere Ergebnisse liefern als die Baseline. Vergleicht man System VC-3 mit System IVP-3 sieht man, dass bei den IVP-3-Modellen öfter an falscher Stelle unbekannte Wörter vorhergesagt werden. Durch die große Anzahl an Head-Modellen in VC-3 wurde eigentlich erwartet, dass die Gefahr für Verwechslungen mit bekannten Wörtern größer ist. Die geringe Präzision bei IVP-3 resultiert aus der großen Ähnlichkeit der Modelle mit den häufigen Wörtern aus dem Erkennervokabular und einem vier mal so hohen LM-Faktor, wodurch das Hypothetisieren von generischen Wörtern erleichtert wird.

Durch die Verlängerung der Head-Länge auf vier (IVP-4) wird sowohl die Präzision als auch der Recall gesteigert. Es ist hervorzuheben, dass keines der bisher diskutierten Modelle phonetische Informationen über OOV-Wörter benutzt hat, abgesehen vom Wissen, dass diese in der Regel länger sind als bekannte Wörter, insbesondere Funktionswörter. Die Head-Modelle von D-1-120-3 besitzen ebenfalls eine Länge von drei, benutzen aber auch OOV-Wörter und seltene bekannte Wörter für die Generierung der Head-Modelle. Im Vergleich zu den gleich langen Modellen von IVP-3 finden sie mehr der unbekannten Wörter mit einer höheren Präzision. Die Verwendung eines großen Hintergrundwörterbuches kann die Ergebnisse also verbessern, schneidet aber gegenüber den längeren Modellen von IVP-4 sowohl bei der Präzision als auch beim Recall schlechter ab.

Die besten Ergebnisse liefern aber erwartungsgemäß die für das Erkennervokabular optimierten Head-Modelle, die über eine variable Länge verfügen. Die Anzahl der verschiedenen Head-Modelle von den Systemen OPT-1 und OPT-2 liegt in der gleichen Größenordnung wie von IVP-4. Die Systeme OPT-1 und OPT-2 sind aber sowohl beim Recall als auch bei der Präzision System IVP-4 überlegen. Durch das zusätzliche Phonem bei OPT-2 wird sowohl die Präzision als auch der Recall im Vergleich zu OPT-1 nochmals gesteigert.

7.1.5 Zusammenfassung

Durch die Verwendung von ungefähr 10000 Head-Tail-Modellen können, ohne die Verwendung eines Hintergrundwörterbuches, *allein* aus dem Erkennervokabular (IVP-4) Head-Tail-Modelle gewonnen werden, die 5.6% der unbekannten Wörter mit einer Präzision von 52.5% detektieren. Durch die Optimierung der Head-Modelle an das Erkennervokabular (OPT-2) kann das Ergebnis nochmals deutlich auf 8.0% Recall und 61.2% Präzision gesteigert werden. Im Vergleich zum ursprünglichen Vokal-Konsonanten-System VC-3 konnte der Recall bei vergleichbarer Präzision mehr als verdoppelt werden und das bei Verwendung von weniger als der Hälfte an Head-Tail-Modellen.

7.2 Untersuchung der unüberwacht gelernten Klassensysteme

Im Rahmen dieser Arbeit wurden verschiedene Experimente durchgeführt, um festzustellen, ob die Verwendung von Sprachmodellklassen zur Modellierung unbekannter Wörter *überhaupt* einen Einfluss auf die Erkennungsleistung hat, und in *welchem* Umfang sich gegebenenfalls dieser Einfluss auswirkt. Für die folgenden Experimente wurden hauptsächlich die Head-Modelle D-1-120-4 verwendet, da diese Experimente *vor* der Entwicklung der vokabular-optimierten-Head-Bestimmung durchgeführt wurden. Daraus resultiert aber keinerlei Einschränkung, da durch Verwendung dieser Modelle die Unterschiede stärker zum Vorschein kommen, die durch die Sprachmodule verursacht werden.

7.2.1 Experiment über den Einfluss der Klassenanzahl

Um die Auswirkung auf die Erkennungsleistung von mehr als einer Klasse zur Modellierung unbekannter Wörter zu untersuchen, wird in diesem Abschnitt zunächst ein Vergleich des in Kapitel 6 beschriebenen statistischen Sprachmodells mit unüberwacht gefundenen Klassen (LM-72) und dem Einklassensprachmodell (LM-1) sowie dem Named-Entity-Sprachmodell (LM-NE) durchgeführt. LM-0 ist ein Sprachmodell, bei dem die Modellierung nur über die Backoff-Wahrscheinlichkeiten stattfindet. Es wird ein Bias eingeführt, der den LM-Faktor verändert und es somit ermöglicht, das Verhalten an

unterschiedlichen Arbeitspunkten zu bestimmen und zu vergleichen.

In Abbildung 7.1 ist die Präzision über dem Recall der verschiedenen Sprachmodelle für unterschiedliche Bias-Werte aufgetragen (ROC-Kurve vgl. Kapitel 2.9.2). Wie man gut erkennt, bricht die Präzision bei Verwendung nur einer Klasse deutlich stärker ein als bei der Verwendung von 72 Klassen. Erwartungsgemäß liefert das 0-Klassen-Sprachmodell, bei dem die Head-Tail-Modelle alleine über die Backoff-Wahrscheinlichkeiten modelliert werden, das mit Abstand schlechteste Ergebnis bei der Detektion.

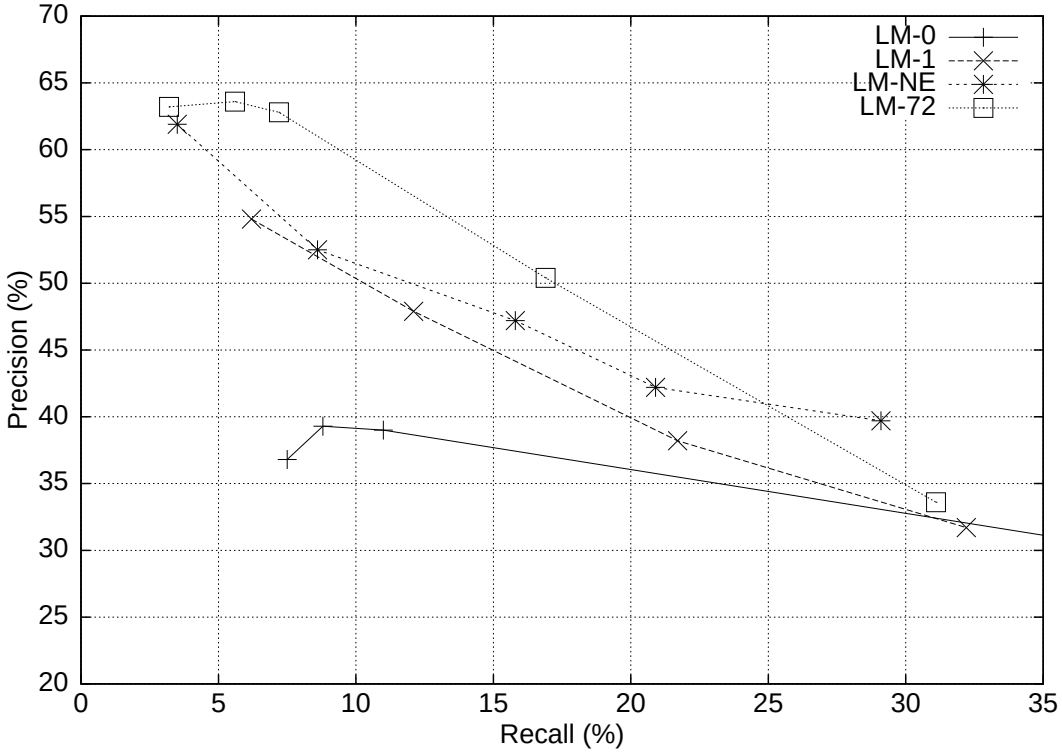

Abbildung 7.1: Einfluss des Sprachmodells auf die Präzision (D-1-120-4)

Die Verwendung der NE-Klassen führt ebenfalls zu schlechteren Ergebnissen als das Sprachmodell mit den 72 unüberwacht gelernten Klassen. Gründe hierfür sind, dass diese Klassen nicht unbedingt für die Modellierung in einem statistischen Sprachmodell entworfen wurden und dass bei automatischer Einteilung der Wörter in NE-Klassen gerade OOV-Wörter verstärkt unkorrekt zugeordnet werden. Insgesamt liefert das Sprachmodell mit den NE-Klassen aber bessere Ergebnisse als die alleinige Verwendung einer einzigen Klasse und belegt damit, dass die Verwendung von Klassen einen nützlichen Beitrag zur Detektion unbekannter Wörter leistet.

Mit dem optimierten Head-Tail-Modell OPT-2 wurde auch der Vergleich des 72-Klassen-Sprachmodells mit dem Einklassensprachmodell durchgeführt. In Abbildung 7.2

ist die Präzision über dem Recall für verschiedene Bias-Werte aufgetragen. In diesem Graphen ist zu erkennen, dass das Sprachmodell LM-1 in der Präzision früher einbricht als Sprachmodell LM-72. Es ist aber auch zu erkennen, dass durch den Bias der Unterschied zwischen den beiden Sprachmodellen verschwindet und beide ab einem Recall um die 8% gleich gute Ergebnisse liefern. Links von dieser Marke wird aber durch die Klassen bei vergleichbarem Recall eine deutlich höhere Präzision erreicht. So werden bei einer Präzision von ungefähr 79% durch das LM-72 6,2% Recall erreicht, womit 50% mehr unbekannte Wörter gefunden wurden als durch das LM-1.

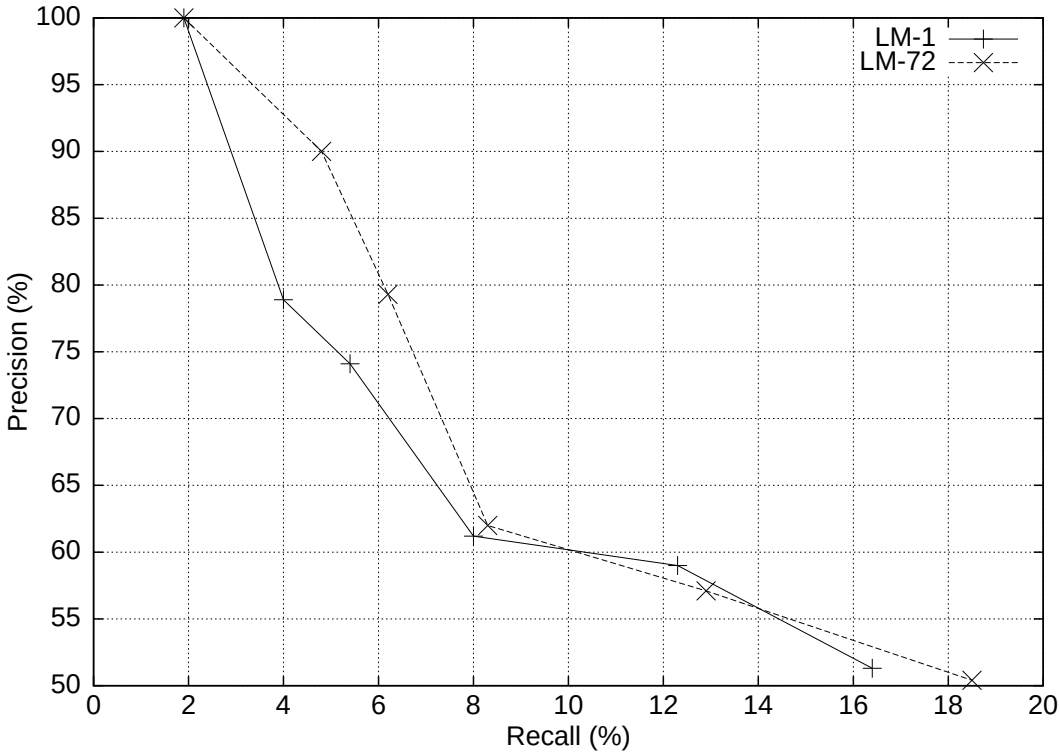

Abbildung 7.2: Vergleich der Sprachmodelle LM-1 und LM-72 mit OPT-2-Modellen

Da das 72-Klassen-Sprachmodell konstant zu besseren Erkennungsergebnissen führte als die anderen Sprachmodelle, soll es im nächsten Abschnitt eingehender untersucht werden.

7.2.2 Untersuchung der Beiträge der Klassen zur Detektion

In folgendem Experiment wurde der Einfluss der einzelnen Klassen des LM-72 auf die Präzision und den Recall untersucht.

Dazu wurden jeweils Teilmengen der 72 Klassen zur Detektion von unbekannten Wörtern gebildet. Als Kriterium wurde die Wahrscheinlichkeit für das Auftreten eines

unbekannten Wortes innerhalb der Klasse ausgewählt. Die Idee, die hierbei verfolgt wurde ist, dass durch die unüberwachte Partitionierung der unbekannten Wörter in Klassen, jede Klasse einen Vektor für typische Kontexte modelliert, in dem unbekannte Wörter auftreten können. Die Verwendung mehrerer Klassen führt dann zu einer Überlagerung dieser Kontextvektoren und zu einer genaueren Modellierung des Raumes für unbekannte Wörter. Es sollten also mit zunehmender Anzahl Klassen immer mehr Kontexte unbekannter Wörter modelliert werden was somit zu einem Ansteigen des Recall beziehungsweise der Präzision führt.

Für dieses Experiment werden die Head-Tail-Modelle D-1-120-4 jeweils mit einer Teilmenge der 72 Klassen versehen, die einen OOV-Anteil besitzen, der eine bestimmte Schwelle s überschreitet. Die Klassen werden nach dem Kriterium

$$p(\text{<OOV:Class-x>}|\text{<Class-x>}) > s$$

mit verschiedenen Schwellen ausgewählt. Die Menge der ausgewählten Klassen kann in Tabelle 7.10 abgelesen werden. Um welche Klassen es sich dabei genau handelt, kann aus Anhang A entnommen werden.

Schwelle s	Anzahl Klassen
40,0%	1
20,0%	12
10,0%	29
5,0%	49
2,5%	62
-	72

Tabelle 7.10: Klassen bei verschiedenen Schwellen

Das Ergebnis der Untersuchung ist in Tabelle 7.11 dargestellt. Es zeigt sich, dass durch das Hinzunehmen der Klassen sich der Recall erhöht und dass bis zu einem bestimmten Punkt die Präzision dabei unverändert bleibt.

Bereits ab der Verwendung von 29 Klassen sind diese dem Einklassensprachmodell LM-1 überlegen. Ab der Verwendung von 49 Klassen ist das Ergebnis auch besser als bei Benutzung der NE-Klassen. Erst beim Hinzufügen der letzten 23 Klassen sinkt die Präzision bei gesteigertem Recall, insgesamt ist das Ergebnis aber immer noch besser als bei der Verwendung der NE-Klassen oder nur einer einzigen Klasse für alle unbekannten Wörter.

Anzahl Klassen	CER	REC	PRC
1	19,9	1,6	60,0
12	19,8	3,5	59,1
29	19,8	7,0	59,1
49	19,8	8,6	59,3
62	19,8	9,1	55,7
72	19,8	9,9	56,9
LM-1	19,8	6,7	56,8
LM-NE	20,0	8,6	52,5

Tabelle 7.11: Einfluss der Anzahl der Klassen bei der Detektion

7.2.3 Analyse einer Zufalls-Baseline

In diesem Abschnitt wird beschrieben, wie das Ergebnis von Recall und Präzision bei einer Zufallsauswahl ausfallen würden, um zu zeigen, dass die hier erreichten Ergebnisse kein Zufallsprodukt sind. Hierzu werden aus den Hypothesen des Spracherkenners zufällig Wörter ausgewählt, die durch das Symbol <OOV> ersetzt werden. Der Anteil von Wörtern, die auf das Symbol <OOV> abgebildet werden, ist durch die Zufallsrate ZuR bestimmt. Werden 5% der Hypothesenwörter auf das Symbol <OOV> abgebildet, so beträgt die Zufallsrate ZuR 5%. Würde dieser Prozess auf den Referenzen durchgeführt, wären im Mittel 5% der zufällig ausgewählten Wörter OOV-Wörter, was einen Recall von 5% ergeben würde. Durch die zufällige Auswahl der Wörter ist der Anteil der OOV-Wörter in dieser Menge im Mittel gerade der Anteil der OOV-Wörter auf dem gesamten Text, was genau der OOV-Rate entspricht. Da der Anteil der korrekt gefundenen OOV-Wörter die Präzision ist, ist diese gleich der OOV-Rate. Bei den BN-OOV-Daten sollte also bei einer Zufallsrate von 5% die Präzision 1,86% betragen.

Wenn bei der Abbildung die Referenz benutzt wird, ist dies auch richtig, da der Align-Algorithmus kaum Freiheiten besitzt, wie die zufälligen OOV-Hypothesen den OOV-Referenzen zugeordnet werden können, da alle umgebenden Wörter ja korrekt zugeordnet werden. Wird dieser Zufallsprozess auf den Hypothesen eines Spracherkenners durchgeführt, so kommt es durch den Align-Algorithmus dazu, dass "Beinahe-Treffer" von OOV-Hypothesen auf OOV-Wörter ausgerichtet werden, da insbesondere in der Umgebung eines OOV-Wortes mehr Erkennungsfehler auftreten. Dies führt dazu, dass bei einer Zufallsrate ZuR von 5% ein Recall von ungefähr 15.5% entstehen kann, was zu einer ungefähr drei bis vier mal höheren Präzision als erwartet führt. Diese Präzision ist aber weitestgehend konstant.

Tabelle 7.12 stellt die Auswirkungen für verschiedene Zufallsraten auf Recall, Präzision und Wortfehlerrate zusammen. Wie zu ersehen ist, werden bei einer Zufallsrate von $\frac{1}{3}$ fast alle OOV-Wörter "gefunden"; bei einer Zufallsrate von $\frac{1}{2}$ sind es bereits alle. Dabei sinkt

ZuR	REC	PRC	CER	WER
0.0	0.0	0.0	20.0	20.0
0.5	1.1	4.2	20.4	20.4
1.0	3.2	6.4	20.7	20.8
2.0	6.7	6.6	21.5	21.7
4.0	13.6	6.6	23.1	23.3
5.0	15.5	6.2	23.8	24.1
10.0	35.9	7.1	27.5	28.2
20.0	69.4	6.9	35.6	36.9
33.3	97.1	5.8	46.0	47.8
50.0	100.0	4.0	59.1	61.0

Tabelle 7.12: Ergebnis aus Zufallsauswahl auf BN-OOV-Daten

die Präzision auf den ungefähr doppelten Wert der OOV-Rate. Würden 100% aller Wörter abgebildet, würde die Präzision weiter sinken, bis sie ungefähr den Wert der OOV-Rate erreicht hat, da die Zahl der Wörter in den Hypothesen ungefähr der Zahl der Wörter in den Referenzen entspricht. Dabei würde die Wortfehlerrate ungefähr 100% − OOV-Rate betragen[3]. Die Wortfehlerrate steigt durch diesen Zufallsprozess im Mittel linear abhängig von der Zufallsrate an; die Steigung ist dabei durch den Anteil korrekter Wörter in den Hypothesen gegeben. Sind beispielsweise 80% der Wörter in der Hypothese korrekt und werden 5% der Wörter auf das Symbol <OOV> abgebildet, so steigt die Wortfehlerrate im Mittel um fast 4% an. Die zusätzliche Fehlerrate wird dabei durch die korrekt gefundenen OOV-Wörter reduziert (Recall ∗ OOV-Rate).

Mit dieser Untersuchung ließ sich zeigen, daß die Präzisionen, die in den Experimenten gefunden wurden, signifikant höher liegen, als bei einer zufälligen Auswahl der Wörter. In Abbildung 7.3 ist die Präzision über dem Recall für den Zufallsprozess und das beste System, das in dieser Arbeit vorgestellt wurde (OPT-2 mit 72 unüberwacht gefundenen Klassen), gemeinsam aufgetragen. Es ist deutlich zu sehen, dass die Präzision bei dem Verfahren OPT-2 immer weit über dem Zufall liegt, was auch für alle übrigen Experimente in dieser Arbeit gilt. Insbesondere steigt die CER nicht oder kaum an, wie aus der Abbildung rechts zu ersehen ist.

[3]Dabei wird davon ausgegangen, dass die Zahl der Symbole in den Hypothesen, die Anzahl der Wörter in den Referenzen nicht übersteigt.

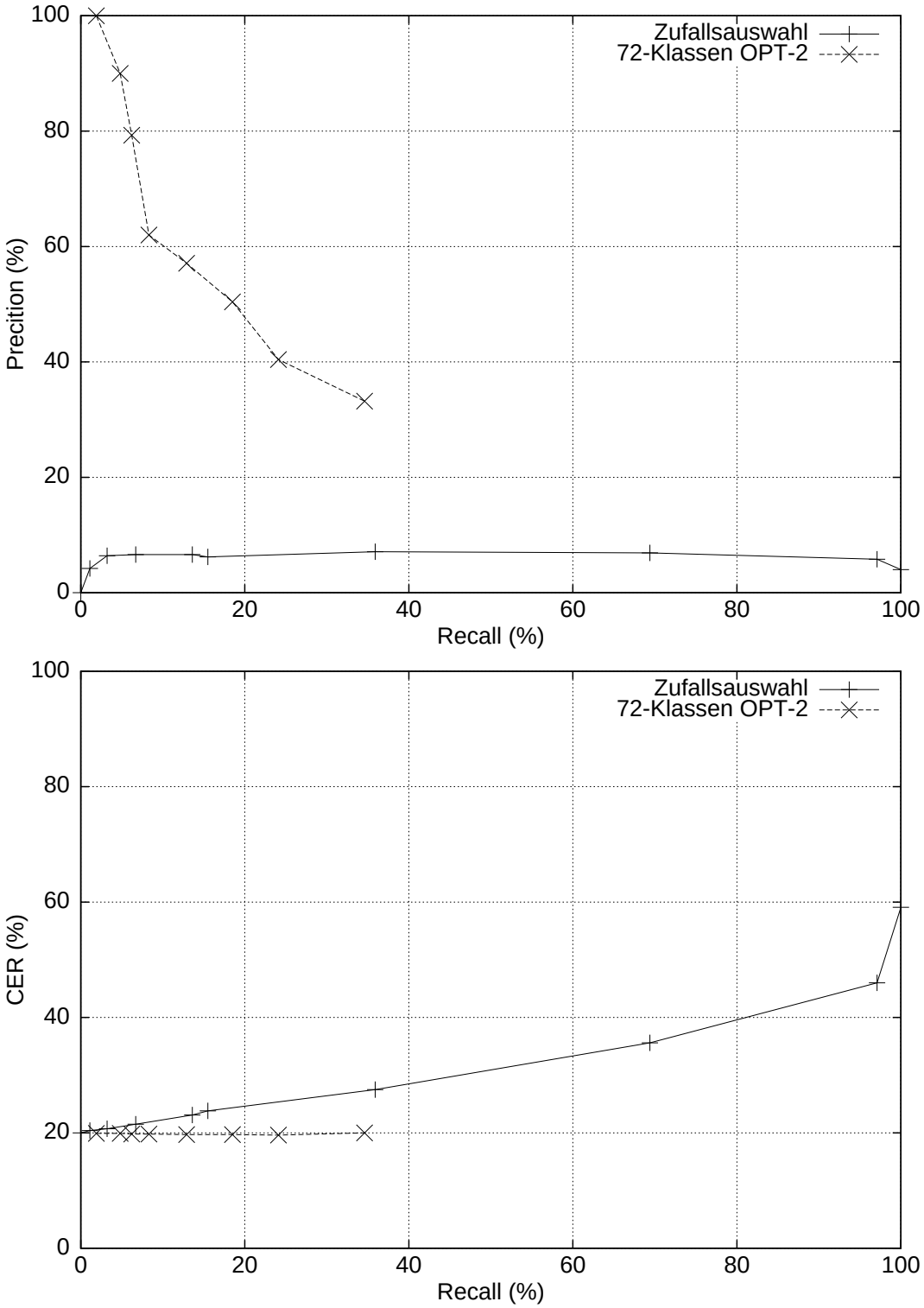

Abbildung 7.3: Vergleich des Zufallsexperiments mit den Head-Tail-Modellen OPT-2 und 72-Klassen-LM

7.2.4 Zusammenfassung

Die in diesem Abschnitt vorgestellten Experimente belegen, dass durch die Verwendung von Klassen bei der Modellierung unbekannter Wörter die Qualität der Detektion verbessert wird. Insbesondere zeigen Abbildungen 7.1 und 7.2, dass bei der Verwendung eines Bias zur Erhöhung des Recall die Präzision deutlich weniger stark einbricht, als bei Verwendung einer einzigen Klasse für alle unbekannten Wörter.

Die Untersuchungen der einzelnen Klassen ergab, dass in der Tat die Verwendung von mehreren Klassen den Recall bei vergleichbarer Präzision steigert.

7.3 Orakel-Experimente auf dem BN-Task

Auf dem englischen BN-Task wurden schließlich Experimente durchgeführt, bei denen Wissen über fehlende Wörter auf der Testmenge verwendet wurde. Ein solcher Test ist im Grunde unfair, da dieses Wissen zu deutlich besseren Ergebnissen führen kann als bei einer realen Anwendung. Diese Orakel-Experimente dienen jedoch eigens dazu, *Grenzen* bei der Detektion und dem Lernen unbekannter Wörter auszuloten.

In den verwendeten Testdaten sind 259 verschiedene unbekannte Wörter enthalten, die insgesamt 373 mal auftreten. Für diese unbekannten Wörter wurden mit dem Graphem-zu-Phonem-Übersetzer von Fisher [26] Phonemumschriften für das Wörterbuch erzeugt. Die so gewonnenen Phonemumschriften dienen als Grundlage für die hier beschriebenen Experimente, in denen verschiedene Möglichkeiten untersucht werden, die Wörter in die jeweils verwendeten statistischen Sprachmodelle aufzunehmen.

Folgende Integrationsmethoden wurden näher betrachtet:

- Integration als Monogramm-Wahrscheinlichkeit (kontextfreie Modellierung) (LM-0)
- Neuberechnung des Sprachmodells mit erweitertem Vokabular (LM-Ext)
- Verwendung von Sprachmodellen mit Klassen für unbekannte Wörter
  - Modellierung aller unbekannten Wörter mit einer Klasse (LM-1)
  - Modellierung der unbekannten Wörter mit den Klassen aus dem NE-Task (LM-NE)
  - Modellierung der unbekannten Wörter mit unüberwacht gefundenen Klassen (LM-72)

Die Experimente, die nun folgen, wurden alle nach einem ähnlichen Schema durchgeführt. Die Wörter wurden mit ihrer *Wortidentität* in das Vokabular des Spracherkenners

aufgenommen, weshalb es möglich ist, eine Wortfehlerrate anzugeben. Je nach Sprachmodell wird diesen Wörtern zunächst eine klassenbedingte Wahrscheinlichkeit zugeordnet, die der mittleren Wahrscheinlichkeit eines OOV-Wortes in dieser Klasse entspricht. Wären beispielsweise 500 OOV-Wörter während des Trainings einer OOV-Klasse OOV_{g} zugeordnet worden, der nun das Wort w_{neu} zugeordnet wird, so wird als klassenbedingte Wahrscheinlichkeit für w_{neu}

$$p(w_{\mathrm{neu}}|OOV_{\mathrm{g}}) = \frac{1}{500} \tag{7.1}$$

verwendet.

Da bekannt ist, dass die hinzugefügten Wörter alle OOV-Wörter auf den Testdaten repräsentieren, werden anschließend Untersuchungen durchgeführt, bei denen die gesamte Wahrscheinlichkeit für unbekannte Wörter auf die in der Klasse hinzugefügten Wörter verteilt wird. Abschließende Experimente versuchen durch die Anwendung eines Bias das Ergebnis noch weiter zu verbessern.

7.3.1 Orakel-Experiment mit Monogramm-Wahrscheinlichkeit

Für dieses Experiment wurden die 259 unbekannten Wörter als Monogramm in das Sprachmodell LM-0 aufgenommen. Das heißt, dass alle benötigten Wahrscheinlichkeiten für einen konkreten Kontext über die Backoff-Wahrscheinlichkeiten des Sprachmodells berechnet werden.

Mit diesem Sprachmodell wurden auf den Testdaten 38 der als unbekannt angesehenen Wörter korrekt erkannt (10.2% REC), keines davon wurde an falscher Stelle hypothetisiert (100% PRC). Insgesamt ergab sich eine Wortfehlerrate von 19.6%.

Um dieses Ergebnis noch zu verbessern wurde zusätzlich ein Bias angewendet, der die Monogramm-Wahrscheinlichkeit erhöht. Es wurde mit verschiedenen Gewichtungsfaktoren experimentiert und mit einem Faktor von 100000 für die Monogramm-Wahrscheinlichkeit konnte dabei das beste Ergebnis bezüglich der Wortfehlerrate von 18.5% erreicht werden. 185 der aufgenommenen Wörtern wurden korrekt gefunden (49.6% REC), 93 mal wurde eines der Wörter an falscher Stelle hypothetisiert (66.5% PRC).

7.3.2 Orakel-Experiment mit erweitertem Sprachmodellvokabular

Für dieses Experiment (LM-Ext) wurde ein neues 3-Gramm-Sprachmodell berechnet indem das Vokabular, über dem das Sprachmodell des BASELINE-Systems berechnet wurde, um die 259 unbekannten Wörter erweitert wurde. Dabei wurden die gleichen Parameter wie für das BASELINE-Sprachmodell verwendet.

Mit diesem neu berechneten Sprachmodell wurden auf den Testdaten 65 der als unbekannt angesehen Wörter korrekt erkannt (17.4% REC), kein einziges davon wurde fälschlicherweise hypothetisiert (100% PRC). Insgesamt ergab sich eine Wortfehlerrate von 19.3%.

Da nun aber bekannt ist, dass die 259 unbekannten Wörter auch *tatsächlich* in der Testmenge enthalten sind, lässt sich dieses Erkennungsergebnis noch verbessern, indem man die Wahrscheinlichkeit des Auftretens dieser Wörter mit einem Bias erhöht.

Auch hier wurde mit verschiedenen Gewichtungsfaktoren experimentiert um das Ergebnis zu verbessern. Mit einem Faktor von 10000 konnte eine Wortfehlerrate von 18.5% erzielt werden. Dabei wurden 189 der Wörter korrekt gefunden (50.7% REC), aber 94 mal eines an falscher Stelle hypothetisiert (66.8% PRC).

7.3.3 Orakel-Experiment mit einer einzigen Klasse

Bei diesem Experiment wurde das Sprachmodell LM-1 benutzt, bei dem alle Wörter, die nicht zum Vokabular V_{Top40k} gehören, in einer *einzigen* Klasse zusammengefasst wurden (vgl. Kapitel 6.1.3). Während des Sprachmodelltrainings wurden 196646 (= Klassenumfang) verschiedene Wörter dieser Klasse zugeordnet.

Die mittlere klassenbedingte Wahrscheinlichkeit für *ein konkretes* Wort aus dieser Klasse wird, wie bereits erwähnt, mit Formel 7.2 geschätzt.

$$p(w_{\text{neu}}|\text{<OOV>}) = \frac{1}{196646} \tag{7.2}$$

Bei diesem Experiment wurden 66 der hinzugefügten Wörter korrekt gefunden (17.7% REC), keines an falscher Stelle hypothetisiert (100% PRC) und eine Wortfehlerrate von 19.3% erreicht.

Da die 259 unbekannten Wörter die *einzigen unbekannten* Wörter sind, kann der Klassenumfang in Formel 7.2 mit 259 angenommen und somit die *gesamte* Wahrscheinlichkeitsmasse der unbekannten Worte auf die neu hinzugefügten Wörter verteilt werden:

$$p(w_{\text{neu}}|\text{<OOV>}) = \frac{1}{259} \tag{7.3}$$

Das entspricht einem Faktor von 1237, mit dem die Wahrscheinlichkeiten für die neu hinzugefügten Wörter im Vergleich zu obigem Versuch erhöht werden.

Mit dieser klassenbedingten Wahrscheinlichkeit ergibt sich eine Wortfehlerrate von 18.5%, wobei 149 der Wörter korrekt gefunden (39.9% REC) und 12 an falscher Stelle hypothetisiert wurden (92.5% PRC).

In einem weiteren Experiment wurde mit einem Bias die Wahrscheinlichkeiten für das Auftreten dieser Wörter noch einmal angehoben, da man ja *ganz sicher* weiß, dass diese 259 hinzugefügten Wörter auftreten müssen. Bei einem zusätzlichen Faktor von 25 wurde das beste Ergebnis bezüglich der Wortfehlerrate von 18.3% erreicht. Dabei wurden 185 der gesuchten Wörter gefunden (49.6% REC). Allerdings wird nun von den hinzugefügten Wörtern 93 mal eines an falscher Stelle hypothetisiert (66.5% PRC).

7.3.4 Orakel-Experiment mit NE-Klassen

Für dieses Experiment wurde das LM-NE verwendet, dass mit Hilfe des Named-Entity-Tagger (vgl. Kapitel 6.1.4) erstellt wurde. Das Auftreten der Wörter, die nicht im Vokabular V_{Top40k} enthalten sind, werden dabei in die vier Klassen "PERSON", "ORGANISATION", "LOCATION" und "REST" eingeteilt und durch Symbole ersetzt, die die Klassen repräsentieren. Anschließend wird ein Sprachmodell mit diesen Klassen berechnet.

Die klassenbedingte Wahrscheinlichkeit für ein Wort aus dieser Klasse wird durch Formel 7.4 geschätzt:

$$p(w_{\text{neu}}|\text{<OOV-ClassX>}) = \frac{1}{Klassenumfang_{\text{ClassX}}} \tag{7.4}$$

$classX \in \{PERSON, ORGANISATION, LOCATION, REST\}$.

Der Klassenumfang ist dabei jeweils die Anzahl verschiedener Wörter, die das Klassensymbol repräsentiert. In Tabelle 7.13 ist der Klassenumfang jeder Klasse angegeben. Die Summe aller Klassenelemente übersteigt dabei die Gesamtzahl der verschiedenen Wörter im Sprachmodellkorpus, da ein Wort anhand des Kontextes nach seiner Benutzung auf ein Klassensymbol abgebildet wird und nicht nur alleine nach dessen Wortidentität. So ist es

möglich, dass ein konkretes Wort in mehreren Klassen auftreten kann, beispielsweise der Name "Washington" könnte sowohl einen Ort als auch eine Person bezeichnen.

Klasse	Anzahl verschiedener Wörter (Klassenumfang)
PERSON	61672
LOCATION	8409
ORGANISATION	9246
REST	155054

Tabelle 7.13: Klassenumfang auf dem Trainingskorpus

Bei diesem Experiment wurden die 259 unbekannten Wörter in *jede* der vier Klassen aufgenommen. Mit dieser Zuordnung wurden 67 der hinzugefügten Wörter korrekt erkannt (18.0% REC), wobei keines davon an falscher Stelle hypothetisiert wurde (100% PRC). Dabei ergab sich eine Wortfehlerrate von 19.4%.

Da die 259 Wörter die einzigen unbekannten Wörter sind, wird auch hier die gesamte Wahrscheinlichkeitsmasse der OOV-Wörter jeder Klasse gleichmäßig auf sie verteilt.

$$p(w_{\text{neu}}|\text{<OOV-ClassX>}) = \frac{1}{259} \tag{7.5}$$

$classX \in \{PERSON, ORGANISATION, LOCATION, REST\}$.

Danach wurden 148 Wörter korrekt gefunden (39.7% REC), und 10 an falscher Stelle hypothetisiert (93.7% PRC). Dabei ergab sich eine Wortfehlerrate von 18.7%.

7.3.5 Orakel-Experiment mit unüberwachten Klassen

Für das Experiment mit dem Sprachmodell LM-72 wurden folgende Untersuchungen durchgeführt:

- Einfügen aller unbekannten Wörter in genau eine Klasse (Top-1)
- Einfügen aller unbekannten Wörter in alle 72 Klassen sowie der Restklasse (TopAlle)
- Einfügen aller unbekannten Wörter in die Top-N Klassen (Top-N)
- Einfügen aller unbekannten Wörter in Top-N plus Restklasse (Top-N + REST)

Aufgrund der Art und Weise, wie die Klassen gebildet wurden, kann für die Berechnung der Wahrscheinlichkeit, dass ein konkretes unbekanntes Wort w_n aufgetreten ist, folgende Formel benutzt werden ($classX \in \{1 \ldots 72, REST\}$):

$$p(w_n|w_1^{n-1}) = p(w_n|\text{<OOV}-\text{ClassX>})*p(\text{<OOV}-\text{ClassX>}|\text{<ClassX>})*p(\text{<ClassX>}|w_1^{n-1}) \tag{7.6}$$

Dabei wurde $p(\text{<OOV}-\text{ClassX>}|\text{<ClassX>})$ beim Erstellen des Sprachmodells bestimmt, als der Anteil der Wörter der Klasse <ClassX>, die nicht im zugrundeliegenden Vokabular V_{Top40k} enthalten sind.

Die Wahrscheinlichkeit für ein konkretes unbekanntes Wort w_{neu}, das einer Klasse zugeordnet ist, kann durch

$$p(w_{\text{neu}}|\text{<OOV}-\text{ClassX>}) = \frac{1}{Klassenumfang_{\text{OOV}-\text{ClassX}}} \tag{7.7}$$

angenähert werden. Dabei wird als Größe der Klasse die Anzahl der verschiedenen unbekannten Wörter benutzt, die während des Erstellens des Sprachmodells der betreffenden Klasse zugeordnet wurden. Damit wird die mittlere Wahrscheinlichkeit eines unbekannten Wortes aus dem Trainingskorpus verwendet, das in dieser Klasse abgebildet wurde.

Andererseits kann unter Verwendung des Wissens, *welche* 259 Wörter auf den Testdaten unbekannt sind, die gesamte Wahrscheinlichkeitsmasse auf diese Wörter verteilt werden, indem man sie den Klassen zuordnet und danach bestimmt, welche Klasse wie viele unbekannte Wörter zugeteilt bekommen hat, und diese Zahl als Klassengröße benutzt. Wenn also beispielsweise der Klasse Class-1 zehn Wörter zugeordnet würden, so wäre $p(w_{\text{neu}}|\text{<OOV} - \text{Class1>} = \frac{1}{10})$.

Für die Wörter, die mindestens einmal im Sprachmodellkorpus aufgetreten sind, und somit über einen Kontext verfügen, wurde eine Zuordnung in eine der 72 Klassen bestimmt. Das waren 2/3 (173) der 259 Wörter. Die restlichen Wörter wurden der Restklasse zugeordnet. Daraus ergibt sich also $p(w_{\text{neu}}|\text{<OOV} - \text{REST>} = \frac{1}{86})$.

Bei Verwendung dieser Zuordnung und unter Annahme der Klassengrößen, wie sie auf der Trainingsmenge vorlagen, wurde ein Wortfehlerrate von 19.1% erreicht, wobei 80 Wörter korrekt gefunden (21.4% REC) und keines der hinzugefügten Wörter an falscher Stelle hypothetisiert wurde (100% PRC).

Wird stattdessen als Klassengröße die Anzahl der Wörter verwendet, die aufgrund ihres Kontextes der Klasse zugeordnet wurden, ergibt sich eine Wortfehlerrate von 18.8%, bei der 122 Wörter korrekt gefunden (32.7% REC) und 11 der hinzugefügten Wörter an

falscher Stelle hypothetisiert wurden (91.7% PRC).

Nimmt man alle der 259 hinzugefügten Wörter nicht nur in eine einzige sondern in alle Klassen auf, und nimmt man als Klassengröße dementsprechend 259 an, ergibt sich eine Wortfehlerrate von 18.6%, bei der 144 der hinzugefügten Wörter gefunden (38.6% REC) und 12 an falscher Stelle hypothetisiert wurden (92.3% PRC).

Es wurde eine Verbesserung des Erkennungsergebnisses erwartet, wenn man die hinzugefügten Wörter in eine kleine Anzahl Klassen aufnimmt, wie dies beispielsweise in [65] beobachtet wurde. Überraschend kam es aber zu keinen nennenswerten Unterschieden zwischen den Ergebnissen, wenn die Wörter nicht nur der Top-1-Klasse sondern den Top-N-Klassen zugeordnet wurden und dabei gleichzeitig die klassenbedingte Wahrscheinlichkeit entsprechend angepasst wurde (vgl. dazu Tabelle 7.14). Ein Grund ist möglicherweise, dass die Zuordnung hier auf zur Domäne passenden Kontexten stattfand und somit die erste Klasse bereits gut passt.
Ein weiterer Grund ist, dass die Zuordnung in Klassen keinen bedeutenden Unterschied macht, wenn die Wahrscheinlichkeiten der betreffenden Wörter entsprechend hoch gewichtet werden.

Top-N	WER	REC	PRC
1	18,8%	32,7	91,7
2	18,8%	32,2	91,4
4	18,8%	32,2	90,2
8	18,8%	32,7	91,0
16	18,8%	32,2	90,2
32	18,8%	32,4	91,0

Tabelle 7.14: Einteilung in Top-N-Klassen

Da es aber immer noch einen deutlichen Unterschied zur Verwendung von allen Wörtern in allen Klassen gab, wurde untersucht, welchen Einfluss die Restklasse besitzt.

Hierzu wurde das Experiment mit den Top-N-Einteilungen wiederholt, jedes Wort wurde aber *zusätzlich* in die Restklasse eingeteilt (vgl. Tabelle 7.15). Das heißt, bei Top-1 wird ein Wort in die wahrscheinlichste Klasse und die Restklasse eingeteilt. Bei diesem Experiment wurde das beste Ergebnis mit einer Wortfehlerrate von 18.5% erreicht, wobei 156 der hinzugefügten Wörter korrekt gefunden (41.8% REC) und 17 an falscher Stelle hypothetisiert wurden (90.2% PRC).

Durch die Restklasse erhöht sich nochmals die Anzahl der gefundenen Wörter. Eine Erklärung hierfür ist, dass diese Klasse die 173 Wörter der im Trainingskorpus aufgetretenen Wörter kennt und eine große Zahl unbekannter Wörter repräsentiert.

Top-N	WER	REC	PRC
1	18,5	41,8	90,2
2	18,5	41,6	90,1
4	18,5	41,6	90,1
8	18,5	41,6	89,6
16	18,5	41,6	89,6
32	18,5	41,3	89,5

Tabelle 7.15: Verwendung von Top-N-Klassen mit zusätzlicher Restklasse

Auch bei diesem Experiment lässt sich die Anzahl der gefundenen Wörter durch die Verwendung eines Bias noch erhöhen. Mit einem weiteren Faktor von 15 konnte mit dem System Top1+Rest ein Ergebnis von 18.3% bei der Wortfehlerrate erreicht werden, wobei 196 der hinzugefügten Wörter korrekt gefunden wurden (52.5% REC) und 78 mal an falscher Stelle hypothetisiert wurde (71.5% PRC).

7.3.6 Zusammenfassung - Orakel-Experimente

Betrachtet man die Versuchsergebnisse aus den verschiedenen Orakel-Experimenten, bei denen die Wahrscheinlichkeit für ein unbekanntes Wort im Allgemeinen mit einer mittleren Wahrscheinlichkeit für ein beliebiges unbekanntes Wort aus der gesamten Trainingsmenge geschätzt wird, fällt auf, dass die Wortfehlerraten alle sehr nahe beieinander liegen (Tabelle 7.16). Die Integration ohne Unterstützung durch das Sprachmodell (LM-0) ist bezüglich des Auffindens der hinzugefügten Wörter erwartungsgemäß am schlechtesten. Der Effekt durch die Verwendung von mehr Klassen für unbekannte Wörter im Sprachmodell fällt gering aus, aber dennoch wird das beste Ergebnis bei Verwendung des Sprachmodells mit den unüberwacht gefundenen Klassen für unbekannte Wörter erzielt (LM-72).

Besonders interessant ist, dass das System LM-Ext, bei dem das Sprachmodell mit erweitertem Vokabular neu berechnet wurde, nicht am besten abgeschnitten hat. Das entspräche dem Ergebnis der Wortfehlerrate, wenn die fehlenden Wörter dem Spracherkenner schon immer bekannt gewesen wären. Ferner wird dadurch auch gezeigt, dass die verwendeten Integrationsmethoden plausible Ergebnisse liefern, wenn nicht bekannt ist, dass die fehlenden Wörter auch in den Testdaten vorkommen müssen.

Unter Ausnutzung des Wissens, dass die hinzugefügten Wörter alle unbekannten Wörter auf den Testdaten darstellen, kann die Wahrscheinlichkeitsmasse, die für unbekannte Wörter reserviert ist, komplett auf die hinzugefügten Wörter verteilt werden. In Tabelle 7.17 werden die Ergebnisse der Versuche dargestellt, bei denen die gesamte Wahrscheinlichkeitsmasse der Klassen für das Auftreten eines unbekannten Wortes auf die 259 hinzugefügten Wörter verteilt und jedes Wort in alle verfügbaren Klassen aufgenommen wurde.

Wie man sieht, liegen alle Ergebnisse nun sehr nahe beieinander. Das zeigt, dass es besonders wichtig ist, die Wahrscheinlichkeit für die Wörter entsprechend hoch zu schätzen und dass in diesem Fall Klassen dann keine so bedeutende Rolle mehr spielen.

System	WER	REC	PRC
BASELINE	20,0	-	-
LM-0	19,6	10,2	100,0
LM-1	19,3	17,7	100,0
LM-NE	19,4	18,0	100,0
LM-72 Top1	19,1	21,4	100,0
LM-72 TopAlle	18,9	28,2	94,6
LM-Ext	19,3	17,4	100,0

Tabelle 7.16: Klassengröße wie beim Training des Sprachmodells

Vergleicht man dies mit den Detektionsergebnissen der Head-Tail-Modelle, zeigt sich, dass die korrekten Aussprachen wichtiger sind, als die Verwendung von Klassen.

System	WER	REC	PRC
LM-1	18,5	39,9	92,5
LM-NE	18,7	39,7	93,7
LM-72 TopAlle	18,6	38,6	92,3

Tabelle 7.17: Klassengröße 259

Ein Grund, warum das Ergebnis bei Verwendung der unüberwacht gefundenen Klassen in der Wortfehlerrate geringfügig schlechter ausfällt, ist, dass alle Wörter völlig unspezifisch in alle Klassen aufgenommen wurden. Wird dagegen eine Klasse ausgewählt und das Wort in diese und in die Restklasse aufgenommen, fällt die Wortfehlerrate vergleichbar gut aus (Tabelle 7.18).

System	WER	REC	PRC
LM-1	18,5	39,9	92,5
LM-72 Top1+REST	18,5	41,8	90,2

Tabelle 7.18: Vergleich “nur Restklasse” versus ”LM-72 Top1 + Restklasse”

Erstaunlich ist, dass trotz des unfairen Ausschöpfen der Information, welche Wörter auf dem Testset unbekannt sind, nur zirka 42% der hinzugefügten Wörter gefunden wurden. Durch Verwendung eines Bias für die hinzugefügten Wörter, konnten noch einmal bessere Ergebnisse erzielt werden (Tabelle 7.19).

Dabei wurden mit dem Sprachmodell, das nur eine einzige Klasse für unbekannte Wörter besitzt, gleich gute Ergebnisse erzielt, wie mit dem Sprachmodell, das die unüberwacht gefundenen Klassen besitzt. Die Verbesserung der Wortfehlerrate im Vergleich zu

System	WER	REC	PRC
LM-1	18,3	52,3	72,8
LM-72 Top1+REST	18,3	52,5	71,5

Tabelle 7.19: Beste Ergebnisse mit einem Bias

Tabelle 7.18 wurde hierbei auf Kosten der Präzision erreicht (Tabelle 7.19). Bei weiterem Erhöhen des Bias kommt es zu einem Ansteigen der Wortfehlerrate. Daraus lässt sich schließen, dass eine Unterstützung durch mehr Klassen anscheinend keine bedeutende Auswirkung mehr hat, wenn man über das Wissen verfügt, dass bestimmte Wörter vorkommen müssen und man die Wahrscheinlichkeiten für die hinzugefügten Wörter nur entsprechend hoch schätzt.

Diese Ergebnisse bedeuten aber keineswegs, dass die Einteilung von unbekannten Wörtern in Klassen *keine* bedeutende Rolle spielen. Zum einem wäre es ungerechtfertigt anzunehmen, dass sich unbekannte Wörter nicht in Gruppen einteilen lassen, wie beispielsweise verschiedenen Arten von Namen. Diese Einteilung erlaubt eine präzisere Möglichkeit der Modellierung von unbekannten Wörtern. Dies hätte sich eigentlich auch in einer höheren Präzision bei vergleichbarem Recall bemerkbar machen sollen, was hier aber bei Verwendung des Bias anscheinend nicht der Fall ist. Zum anderen führt im Allgemeinen eine bessere Modellierung auch zu besseren Ergebnissen in der Spracherkennung. Insbesondere wenn ein Spracherkennungssystem den Suchraum stark beschneiden muss, um einen geringen Real-Time-Faktor zu erreichen, ist es wichtig, gute Vorhersagen durch das Sprachmodell zu erhalten.

Sind die ursprünglich unbekannten Wörter alle im Vokabular des Spracherkenners enthalten, bedeutet es nicht automatisch, dass diese auch gefunden werden. Wenn bestimmte Wörter nicht gefunden werden, kann das beispielsweise daran liegen, dass die phonetischen Umschriften automatisch erzeugt wurden und sie daher nicht immer die tatsächliche Aussprache repräsentieren. Außerdem sind die akustischen Modelle seltener Wörter, zu denen die unbekannten Wörter in der Regel gehören, nur mit wenig oder wenig geeignetem Datenmaterial trainiert worden.

Durch das ungenierte Benutzen von unfairem Wissen kann man zwar über 52% der unbekannten Wörter finden. Wären es aber reguläre Wörter des Erkenners, hätte dieser lediglich 18% bis 22% der Wörter gefunden. Nimmt man an, dass jedes der verschiedenen unbekannten Wörter mindestens einmal detektiert wurde, und aufgrund dessen die Wortidentitäten der unbekannten Wörter zur Verfügung stehen würden, könnten bei einem zweiten Erkennungslauf 42% der Auftreten der detektierten Wörter gefunden werden (ohne Bias).

7.4 Zusammenfassung

Es wurde gezeigt, dass eine Modellierung generischer Wörter durch Head-Tail-Modelle effektiv möglich ist und diese selbst dann gute Ergebnisse erzielt, wenn der Spracherkenner den Suchraum (stark) beschneiden muss, um ein vorgegebenes Zeitverhalten zu erreichen. Durch die generischen Wörter konnte 50% der Reduktion der Fehlerrate des Spracherkenners gemessen am ORAKEL-Experiment auf den GSST-Daten erreicht werden.

Ferner haben die Untersuchungen gezeigt, dass für die akustische Modellierung der Head-Modelle die gleichen kontextabhängigen Modelle zu verwenden sind, wie für die regulären Wörtern des Spracherkenners, damit die generischen Wörter nicht zu früh eine schlechte akustische Bewertung erhalten und aus dem Suchraum entfernt werden. Im durchgeführten Experiment bedeutet dies, dass für die akustische Modellierung des ersten Phonems (Wortgrenze) im Head-Modell Cross-Word-Triphone-Modelle eingesetzt wurden. Für alle weiteren Untersuchungen wurde daher stets diese akustische Modellierung verwendet. Ein positiver Seiteneffekt ist, dass für diese Art der Modellierung generischer Wörter keine neuen akustischen Modelle zu trainieren sind, bis auf die generischen Modelle, die im Tail-Modell benutzt werden. Der Einfluss des Tail-Modells auf die Erkennungsleistung ist erwartungsgemäß nicht so groß wie der Einfluss der Länge des Head-Modells.

Mit diesen Erkenntnissen wurde der Ansatz der Head-Tail-Modelle von der deutschen GSST-Domäne auf die sehr viel umfangreichere englische BN-Domäne übertragen. Die auf der GSST-Domäne eingesetzten Head-Modelle, die aus Vokal-Konsonanten-Folgen bestanden, wurden mit weiter entwickelten Verfahren zur Bestimmung geeigneter Head-Modelle verglichen.

Die besten Resultate erzielte das Verfahren, das unter Verwendung des Erkennervokabulars und eines Hintergrundwörterbuches die in den Head-Modellen eingesetzten Phonemfolgen bestimmt (OPT-2). Das Verfahren (IVP), das nur das Vokabular des Spracherkenners benutzt, war aber ebenfalls besser in der Erkennungsleistung als die Vokal-Konsonanten-Folgen.

Durch Optimierung der Head-Modelle an das Erkennervokabular (OPT-2) konnte unter Verwendung eines 1-Klassen-Systems die Verdopplung des Recall im Vergleich zu Vokal-Konsonanten-Folgen (VC-3) bei gleichbleibender Präzision von 3.8% auf 8% erreicht werden. Wenn allein das Erkennervokabular zur Gewinnung von Head-Tail-Modellen (IVP-4) benutzt wurde, stieg der Recall von 3.8% auf 5.6%.

Für die englische BN-Domäne steht umfangreiches Datenmaterial zur Verfügung und die Domäne selbst ist komplex. Eine geeignete Modellierung von unbekannten Wörtern im Sprachmodell erfordert im allgemeinen ein genaues Wissen über die Domäne. Eine manuelle oder halbautomatische Einteilung der unbekannten Wörter in ein für die Domäne zu entwickelndes Klassensystem, das auch für die Modellierung in einem Spracherkenner geeignet ist, ist sehr aufwendig.

Die Ergebnisse der Untersuchungen mit den in dieser Arbeit entwickelten Systemen belegen, dass durch Verwendungen von überwacht gelernten Klassen (GrOOV-Verfahren) mehr unbekannte Wörter mit einer höheren Präzision gefunden werden können, als bei Verwendung einer einzigen Klasse für unbekannte Wörter. Dabei wurde mit dem unüberwacht gefundenen Klassensystem mit 72-Klassen bessere oder gleich gute Ergebnisse erzielt, wie bei Verwendung von Named-Entitiy-Klassen und einer Restklasse. Dafür wird aber kein Named-Entity-Tagger benötigt, der entweder auf manuell annotierten Daten trainiert werden muss oder mit Regeln von Experten arbeitet. Das neu entwickelte Verfahren kann sich somit besser an die Domänen anpassen und liefert geeignetere Klassen für eine Modellierung in einem Sprachmodell. Im Vergleich mit der Verwendung einer einzigen (unüberwachten) Klasse konnte der Recall um über 50.0% bei gleicher Präzision vom 79% gesteigert werden, wenn die Modelle OPT-2 verwendet wurden.

Mit den Orakel-Experimenten auf der BN-Domäne wurde schließlich gezeigt, dass man zwar durch das unfaire Ausschöpfen der Information, welche Wörter auf den Testdaten im Vokabular fehlen, über 52% der unbekannten Wörter detektieren kann, aber dass trotz dieses Wissens nicht *alle* hinzugefügten Wörter gefunden werden konnten.

Das System LM-Ext, bei dem das Sprachmodell mit erweitertem Vokabular neu berechnet wurde, demonstriert das Verhalten des Erkenners, falls die Wörter keine OOV-Wörter gewesen wären und findet bescheidene 17.4% der Wörter. Durch die Verwendung eines Bias kann das Ergebnis zwar deutlich gesteigert werden, dies führt aber nicht zum besten Ergebnis.

Es zeigte sich, dass die Verwendung von Klassen keinen Unterschied mehr macht, wenn die Wahrscheinlichkeiten für unbekannte Wörter nur hoch genug angesetzt werden. Das beste Ergebnis durch das Verwenden von normalerweise nur einem Orakel bekannten Informationen wurde mit einer Wortfehlerrate von 18.3% erreicht, was ein 8.5% relativ besseres Ergebnis als die Baseline ist. Es wurde aber hierbei in Kauf genommen, dass die hinzugefügten Wörter bis zu 78 mal falsch aufgetreten sind, was auf anderen als den hier verwendeten Daten sicher zu einer Verschlechterung des Ergebnisses führen würde.

Die Ergebnisse zur Detektion unbekannter Wörter mit Head-Tail-Modellen und dem Sprachmodell mit dem GrOOV-Verfahren sind damit am besten mit den Ergebnissen aus Tabelle 7.16 zu vergleichen, die die Ergebnisse zusammenfasst, wenn es sich bei den gesuchten Wörtern nicht um unbekannte Wörter gehandelt hätte. Danach wären 17,4% bis 28,2% der Wörter gefunden worden.

In Relation dazu konnte mit den Head-Tail-Modellen OPT-2 und dem Sprachmodell LM-72 bei einer Präzision von 79,3% immerhin 36% der Leistung des um die fehlenden Wörter erweiterten Systems LM-Ext erreicht werden. Wird in Kauf genommen, dass jede zweite Hypothese fehlerhaft ist, könnten sogar 18,5% der gesuchten Wörter gefunden werden, was mehr als beim Orakel-Experiment LM-Ext ist und 66% der Leistung des Experiments LM-72 entspricht.

Kapitel 8

Integration eines neuen Wortes in den Spracherkenner

Bei einem System, bei dem sowohl die Detektion als auch das Erlernen von unbekannten Wörtern möglich ist, bedarf es sowohl einer Wahrscheinlichkeit für das erstmalige Auftreten eines unbekannten Wortes, als auch einer Wahrscheinlichkeit, dieses während des Lernens und danach wiederzufinden.

In diesem Kapitel wird untersucht, wie man diese Aufteilung der Wahrscheinlichkeitsmasse durchführen und wie die Wahrscheinlichkeit für ein neu gelerntes Wort geschätzt werden kann.

8.1 Aufteilung der Wahrscheinlichkeitsmasse auf Detektion und Wiederfinden

Sollen unbekannte Wörter nicht nur detektiert sondern für eine spätere Verwendung auch noch gelernt werden, ergeben sich neben der Frage nach einer geeigneten Aussprache und der Zugehörigkeit zu einer oder mehreren Klassen noch weitere Fragen. Es muss auch geklärt werden, mit welcher Wahrscheinlichkeit ein gelerntes Wort in das statistische Sprachmodell aufgenommen werden soll, und woher die dafür verwendete Wahrscheinlichkeitsmasse stammt.

Es wäre im Prinzip möglich, für ein zu lernendes Wort w_{neu} das Sprachmodell noch einmal komplett neu zu berechnen. Doch ist dieser Ansatz einerseits recht aufwendig, andererseits kann nicht sichergestellt werden, dass auch alle zu lernenden Wörter in einem für die Domäne geeigneten Trainingskorpus vorkommen. Ist dies doch der Fall, handelt es sich in der Regel um seltene Wörter, zu denen nur eine geringe Menge an Beispielen verfügbar ist und dadurch die Gefahr besteht, über nicht genügend geeigneten Kontext zu verfügen. Durch ein einfaches Neuberechnen des Sprachmodells über den gesamten Sprachmodellkorpus fällt ferner die Wahrscheinlichkeit für das Auftreten des zu lernenden Wortes insgesamt sehr gering aus.

Dies ist beispielsweise bei den Orakel-Experimenten in Kapitel 7.3 zu sehen, bei denen auch eine völlige Neuberechnung des Sprachmodells durchgeführt wurde (LM-Ext), und in denen die hinzugefügten Wörter ohne ein Anheben der Wahrscheinlichkeit nur zu 17.4% gefunden wurden. Das Ergebnis ist zwar deutlich besser als die 10.2% wiedergefundener Wörter, die sich ergeben, wenn diese ohne Kontext in das Sprachmodell aufgenommen werden (LM-0). Aber bereits bei Verwendung von nur einer *einzigen* Klasse kann schon ein vergleichbar gutes Resultat (17.7%) erzielt werden, ohne dass hierzu ein neues statistisches Sprachmodell berechnet werden muss.

Klassenbasierte Sprachmodelle bieten für die Modellierung von zu lernenden Wörtern nicht nur den Vorteil, dass sie das Hinzufügen und Entfernen eines zu lernenden Wortes ohne aufwendige Neuberechnung des Sprachmodells erlauben, sie führen vor allem zu besseren Ergebnissen.

Die in Kapitel 7.3 durchgeführten Orakel-Experimente lassen sich auch so interpretieren als wären alle unbekannten Wörter nach einem *gedachten* ersten orakelhaften Erkennungslauf in das Sprachmodell aufgenommen, bei dem jedes unbekannte Wort mindestens einmal gefunden worden wäre. Mit dem erweiterten Vokabular wurde dann ein weiterer *realer* Erkennungslauf, eben die Orakel-Experimente aus Kapitel 7.3, durchgeführt. Das beste Ergebnis wurde dann erreicht, wenn die gesamte Wahrscheinlichkeitsmasse, die in einer Klasse für die Modellierung unbekannter Wörter reserviert ist, auf die neu hinzugefügten Wörter verteilt wurde. Dies war möglich, weil alle generischen Wörter (Head-Tail-Modelle) für die Detektion unbekannter Wörter im zweiten Erkennungsschritt nicht mehr im Vokabular enthalten waren. Im Prinzip kann diese Kombination als eine Batch-Verarbeitung betrachtet werden, bei der zunächst eine Detektion durchgeführt wurde, um anschließend alle korrekt gefundenen unbekannten Wörter in den Spracherkenner aufzunehmen und danach alle Äußerungen nochmals zu verarbeiten.

Der im folgenden Kapitel beschriebene Prototyp verfügt über die Fähigkeit, neue Wörter zu lernen, *sobald* sie detektiert wurden (online-lernen). Daraus ergibt sich aber die Notwendigkeit, die für die Modellierung unbekannter Wörter reservierte Wahrscheinlichkeit auf die für die Detektion verwendeten generischen Wörter *und* die während der Benutzung zusätzlich gelernten Wörter $w_{\text{neu}} \in g$ aufzuteilen, da die Summe der Wahrscheinlichkeit einer Klasse eins ergeben soll.

Im folgenden wird mit "$w, P1$" das erste Auftreten eines Wortes w bezeichnet und mit "$w, P2+$" das wiederholte Auftreten von w. Mit $p(w, P1|g)$ wird demnach die Wahrscheinlichkeit bezeichnet ein Wort w, das zur Klasse g gehört das erste Mal zu beobachten und mit $p(w, P2 + |g)$ die Wahrscheinlichkeit, dass es sich dabei um ein wiederholtes Auftreten nach einem erstmaligen Auftreten $P1$ handelt. Gesucht wird nun eine Schätzung dieser Wahrscheinlichkeiten für das zu detektierende Auftreten $P1$ und das gelernte wiederholte Auftreten $P2+$.

Für die folgenden Analysen und Berechnungen wird vereinfachend angenommen, dass

alle unbekannten Wörter detektiert werden und dies jedesmal bereits beim ersten Auftreten gelingt, wenn das Wort unbekannt ist. Um die Normierung der Wahrscheinlichkeiten nicht zu verletzen, muss für die Wörter w, die einer Klasse g zugeordnet sind und für die auf einem Trainingskorpus die klassenbedingten Wahrscheinlichkeiten $p(w|g)$ geschätzt wurden, folgende Beziehung gelten:

$$1 = \sum_{w \in g} p(w|g) = \sum_{w \in g} (p(w, P1|g) + p(w, P2 + |g)) = p(P1|g) + p(P2 + |g) \tag{8.1}$$

Dabei ist $p(P1|g) = \sum_{w \in g} p(w, P1|g)$ die Wahrscheinlichkeitsmasse, die für die Detektion unbekannter Wörter in der Klasse g zu reservieren ist und $p(P2 + |g) = \sum_{w \in g} p(w, P2 + |g)$ die Wahrscheinlichkeitsmasse, die für ein wiederholtes Auftreten eines Wortes aus der Klasse benutzt wird. Normalerweise ist in Sprachmodellen $p(w, P1|g) = 0$, da keine Detektion unbekannter Wörter vorgesehen ist, weil bereits alle Wörter während des Trainings gelernt wurden, die er Spracherkenner beherrschen soll.

8.1.1 Aufteilung aufgrund der Häufigkeit der Wörter in den Klassen

Aus der Annahme, dass das erste Auftreten eines Wortes durch eine Detektion für unbekannte Wörter gefunden und damit gelernt wird, lässt sich die Wahrscheinlichkeit $p(w, P1|g)$ aus der Verteilung der potentiell unbekannten Wörter einer Klasse schätzen. Dabei wird die Summe N_1 über das erste Auftreten der Wörter $w \in g$ berechnet durch

$$N_1 = \sum_{w \in g} 1 = |g| \tag{8.2}$$

Da das Lernen im Prototypen aus Kapitel 9 einen Schritt beinhaltet, bei dem das erfolgreiche Lernen eines neuen Wortes überprüft wird, wird der Wert beim wiederholten Auftreten (P2+) um eins erhöht. Damit ist auch sichergestellt, dass für das Wiederfinden mindestens die gleiche Wahrscheinlichkeitsmasse zu Verfügung steht, wie für die Detektion. Würde dies nicht gemacht, könnte es vorkommen, dass *keine* Wahrscheinlichkeit für das Wiederfinden reserviert würde. Dieser Grenzfall tritt aber nur dann auf, wenn eine Klasse nur Wörter enthält, die im Training nur ein einziges Mal aufgetreten sind. Die Zahl des Wiederauftretens berechnet sich dann durch

$$N_{2+} = \sum_{w \in g} freq(w) \tag{8.3}$$

Dabei bezeichnet $freq(w)$ die Häufigkeit des Wortes w auf dem Trainingskorpus.
Damit ist die Gesamtsumme N_+ für die Normierung gegeben durch

$$N_+ = \sum_{w \in g} (freq(w) + 1) = N_1 + N_{2+} \tag{8.4}$$

Es wurde vereinfachend angenommen, dass die Klasse g nur Wörter enthält, die zu den pUNK-Wörtern zählen (vgl. Kapitel 6). Eine Erweiterung auf gemischte Klassen aus pUNK-Wörtern und bekannten Wörtern ist einfach möglich.

Die geschätzte Wahrscheinlichkeit ein beliebiges unbekanntes Wort zum ersten Mal zu beobachten, das zur Klasse g gehören könnte, ergibt sich dann durch

$$p(P1|g) \approx \frac{N_1}{N_+} = \frac{|g|}{N_+} \tag{8.5}$$

Es folgt aus der Voraussetzung, dass g nur potentiell unbekannte Wörter enthält, dass die verbleibende Wahrscheinlichkeit $p(P2|g) \approx 1 - p(P1|g)$ für das Wiederfinden von gelernten Wörtern benutzt werden kann.

Am Beispiel des Sprachmodells mit einer einzigen Klasse für unbekannte Wörter (vgl. Kapitel 6.1.3), soll nun berechnet werden, welcher Anteil der Wahrscheinlichkeitsmasse auf das Detektieren von unbekannten Wörtern entfällt.

Häufigkeit	Wörter	Häufigkeit	Wörter	Häufigkeit	Wörter
1	92189	11	2173	21	806
2	31890	12	1941	22	860
3	15709	13	1714	23	756
4	10386	14	1478	24	714
5	7031	15	1355	25	705
6	5385	16	1283	26	674
7	4198	17	1135	27	635
8	3455	18	1085	28	569
9	2885	19	969	29	549
10	2451	20	974	30	506

Tabelle 8.1: Häufigkeiten und Anzahl von OOV-Wörtern auf dem BN-Korpus

Tabelle 8.1 gibt dabei für diese Klasse die Häufigkeit der Wörter und deren Anzahl an. Nimmt man diese Häufigkeiten, so ergibt sich für $N_1 = 196460$, $N_+ = 993501$ und $N_{2+} = 797041$. Aus Gleichung 8.5 folgt, dass 19.8% der Wahrscheinlichkeitsmasse für die Detektion von unbekannten Wörtern verwendet werden würde. Die Wahrscheinlichkeit, die dann beispielsweise einem *konkreten* generischen Wort zur Detektion zugeordnet wird, kann entweder durch $p(w_1|g) = \frac{1}{N_1}\frac{N_1}{N_+} = \frac{1}{N_+}$ bestimmt werden oder indem die Wahrscheinlichkeitsmasse gleichmäßig auf alle generischen Wörter verteilt wird. In Kapitel 7 wurde die gesamte Wahrscheinlichkeitsmasse gleichmäßig auf die generischen Wörter verteilt um die Detektionsexperimente durchzuführen.

8.1.2 Anwendung der Zipf'schen Verteilung

Ein weiterer Ansatz geht davon aus, dass sich die Zahl der seltenen Wörter in Abhängigkeit der Häufigkeit wie von Zipf [84] (siehe Kapitel 2.10) beobachtet, durch $y = a * \frac{1}{i^b}$ beschreiben lässt. Die Werte für die Parameter a und b sind beispielsweise für die 72 unüberwacht gefundenen Klassen für unbekannte Wörter im Anhang A angegeben. Die Werte für a und b in dem hier bereits benutzten Beispiel mit nur einer einzigen Klasse, sind in Tabelle 8.2 angegeben.

Parameter	Wert
a	86923
b	1.451

Tabelle 8.2: Parameter für Zipf'sche Verteilung für Einklassensprachmodelle

Geht man nun von einem Vokabular aus, das alle Wörter eines Textkorpus enthält, die eine Mindesthäufigkeit M besitzen, so können alle Wörter, die seltener sind als potentiell unbekannte Wörter angesehen werden.

Verwendet man den gleichen Ansatz wie eben, erhält man die Gleichungen

$$N_1 = \sum_{i=1}^{M-1} a\frac{1}{i^b} = a\sum_{i=1}^{M-1}\frac{1}{i^b} \tag{8.6}$$

$$N_+ = \sum_{i=1}^{M-1} a\frac{1}{i^b}(i+1) = a\sum_{i=1}^{M-1}\frac{i+1}{i^b} \tag{8.7}$$

$$N_{2+} = \sum_{i=1}^{M-1} a\frac{1}{i^b}i = a\sum_{i=1}^{M-1}\frac{1}{i^b}i \tag{8.8}$$

Daraus ergibt sich, dass sich die Wahrscheinlichkeit schätzen lässt durch

$$p(\mathrm{P1}|g) \approx \frac{N_1}{N_+} = \frac{\sum_i^{M-1}\frac{1}{i^b}}{\sum_i^{M-1}\frac{i+1}{i^b}} \tag{8.9}$$

Dabei fällt auf, dass der Wert für a keinerlei Einfluss auf die Aufteilung der Wahrscheinlichkeit hat. Man benötigt nur den Wert M und eine geeignete Schätzung für die Steigung b der betrachteten Klasse. Die Steigung[1] b charakterisiert dabei das Verhältnis, wie häufig sich Wörter innerhalb eine Klasse wiederholen. Findet eine verhältnismäßig häufige Wiederholung statt, ist der Wert für b klein, ist die Wiederholung innerhalb der Klasse verhältnismäßig selten, ist der Wert für b groß.

[1] eigentlich das Gefälle

Im Fall unseres Beispiels ist $M = 31$ und es würden 18.1% der Wahrscheinlichkeitsmasse für die Detektion unbekannter Wörter reserviert werden. Das entspricht ungefähr dem Wert, wie er auch beim zuvor beschriebenen Ansatz gefunden wurde (19.8%).

8.2 Welche Wahrscheinlichkeit bekommt ein gelerntes Wort zugeordnet?

Nachdem nun Möglichkeiten beschrieben wurden, wie die Wahrscheinlichkeitsmasse zwischen Detektion und gelerntem Wort aufgeteilt werden kann, stellt sich als nächstes die Frage, welche Wahrscheinlichkeit nun einem *konkreten* Wort w_{neu} zugeordnet wird, das gelernt werden soll.

Da im Voraus nicht bekannt ist, wie oft eines der gelernten Wörter in Zukunft auftreten wird, besteht hier nicht die Möglichkeit für ein neu gelerntes Wort eine initiale klassenbedingte Wahrscheinlichkeit in Abhängigkeit dieses Wertes zuzuordnen. Für ein neu gelerntes Wort wird deshalb die Wahrscheinlichkeit nur in Abhängigkeit der Klasse g geschätzt. Hierdurch wird implizit eine Modellierung des erwarteten Wiederauftretens durchgeführt. Für bereits gelernte Wörter gibt es durchaus die Möglichkeit, die einmal zugeordnete Wahrscheinlichkeit in Abhängigkeit des Auftretens in den Hypothesen anzupassen, beispielsweise wird dies bei Trigger-Sprachmodellen (vgl. dazu Kapitel 2.7.3) gemacht. Im Folgenden wird nun beschrieben, wie die *initiale* Wahrscheinlichkeit für ein gelerntes Wort bestimmt werden kann.

Klassengröße wie beim Training

Werden der Klasse g während des Trainings N_g der potentiell unbekannten Wörter zugeordnet, so kann beispielsweise, wie in einigen Orakel-Experimenten (vgl. Kapitel 7.3), einem neuen Wort w_{neu} eine mittlere Wahrscheinlichkeit $p(w_{\text{neu}}|g) = \frac{1}{N_g} p(P1|g)$ zugeordnet werden.

Dies erlaubt die Aufnahme von N_g Wörtern in diese Klasse. Überschreitet die Zahl der Wörter, die in die Klasse g gelernt werden den Wert N_g , müssen die Wahrscheinlichkeiten der bereits gelernten Wörter angepasst werden, um die Normierungsbedingungen nicht zu verletzten.

Bei der Schätzung der Wahrscheinlichkeit eines gerade gelernten Wortes dieser Klasse wird davon ausgegangen, dass es die gleiche mittlere Wahrscheinlichkeit eines seltenen Wortes besitzt, die auf einem sehr viel größeren Textkorpus bestimmt wurde. Da in der Praxis ein zu erkennender Text sehr viel kürzer ist und es daher auch sehr viel weniger unbekannte Wörter zu lernen gibt, treten diese doch relativ gesehen sehr viel häufiger auf. Die so gewonnene Schätzung von $p(w_{\text{neu}}|g)$ tendiert also dazu, die Wahrscheinlichkeit für das Wort w_{neu} zu unterschätzen. Dies kann dazu führen, dass das gelernte Wort seltener gefunden wird.

Inkrementelle Aufteilung der verfügbaren Wahrscheinlichkeiten

Eine gleichmäßige Aufteilung der Wahrscheinlichkeit auf die zu erwartende Menge unbekannter Wörter erscheint sinnvoll. Daraus ergibt sich, dass eine Schätzung für die Anzahl der verschiedenen unbekannten Wörter benötigt wird, die in eine Klasse hinzugefügt werden.

Würde man stattdessen die gesamte Wahrscheinlichkeitsmasse unter den aktuell in der Klasse enthaltenen gelernten Wörtern aufteilen, so führt dies dazu, dass die Wahrscheinlichkeiten für die ersten, insbesondere *das erste* gelernte Wort, überschätzt wird, was zu häufigem fehlerhaften Auftreten dieser Wörter führen kann. Beobachtet werden kann das beispielsweise auch bei den Orakel-Experimenten in Kapitel 7.3, wenn diese mit einem großen Bias durchgeführt werden.

Die Wahrscheinlichkeit für die bereits gelernten Wörter ist zusätzlich jedesmal anzupassen, wenn ein neues Wort gelernt wird. Beim Lernen des zweiten Wortes halbiert sich dann beispielsweise die Wahrscheinlichkeit für das erste Wort.

8.2.1 Schätzung der Zahl der aufzufindenden Wörter

Im Folgenden wird eine Schätzung für die Anzahl der verschiedenen unbekannten Wörter ($\#W$) hergeleitet, die gelernt werden sollen, um die *initiale* klassenbedingte Wahrscheinlichkeit $p(w_{\text{neu}}|g)$ zu erhalten, die einem neu gelernten Wort w_{neu} zugeteilt wird. Als Beispiel dient wie bisher dabei das Sprachmodell mit einer einzigen Klasse für unbekannte Wörter. Verglichen wird die Wahrscheinlichkeit in diesem Fall mit dem Ansatz, bei dem die mittlere Wahrscheinlichkeit eines Wortes aus dieser Klasse verwendet wird, also $p(w_{\text{neu}}|g_1) \approx \frac{1}{200000}$.

Die Zahl $N_1 = 196460$ (vgl. 8.1.1) wurde zur einfacheren Berechnung auf 200000 und die Wahrscheinlichkeit für ein Wiederfinden auf $\frac{4}{5}$ gerundet.

Konservative Schätzung

Eine sehr konservative Schätzung der Zahl zu lernender Wörter ist, die Zahl der erwarteten unbekannten Wörter insgesamt zu schätzen. Es wird eine erwartete OOV-Rate, eine geschätzte Gesamtlänge der Äußerungen und damit eine geschätzte Anzahl gesprochener Wörter vorausgesetzt. Bei englischsprachigen Aufnahmen enthält eine Stunde ungefähr 10000 Wörter. Die OOV-Rate liegt beim Vokabular V_{Top40k} üblicherweise zwischen 0.5% und 2.0%. Bei spontan gesprochenen Dialogen liegt sie dabei eher am unteren Ende. Je Stunde Sprachaufnahmen muss also mit etwa 50 - 200 unbekannten Wörtern gerechnet werden.

Wird also erwartet, dass man zwei Stunden Sprachaufnahmen automatisch transkribieren soll, entspräche dies ungefähr 100 bis 400 unbekannten Wörtern. Geht man sehr konservativ davon aus, dass jedes der Wörter unterschiedlich ist, also gelernt werden soll, kann die *initiale* Wahrscheinlichkeit für ein gelerntes Wort w_{neu} mit $p(w_{\text{neu}}|g_1) = \frac{1}{400} * \frac{4}{5} = \frac{1}{500}$

geschätzt werden. Verglichen mit dem Wert $\frac{1}{200000}$ fällt die so geschätzte Wahrscheinlichkeit um einen Faktor von ungefähr 400 höher aus.

Zipf'sche Verteilung

Hat man eine Schätzung darüber, wie viele OOV-Wörter *insgesamt* in einem Textabschnitt zu erwarten sind ($\tilde{N}$), kann man mit Hilfe der Zipf'schen Verteilung schätzen, wie viele *verschiedene* OOV-Wörter ($\#W$) dies sind. Tabelle 8.3 stellt für den Wert $b = 1.45$ eine Abbildung von $\tilde{N}$ auf $\#W$ dar. m gibt dabei an, wie häufig die am häufigsten zu erwartenden OOV-Wörter sind. Die Werte für $\tilde{a}$, $\tilde{N}$ und $\#W$ berechnen sich wie folgt: $\tilde{a} = m^b$ und $\tilde{N} = \tilde{a} \sum_{i=1}^{m} \frac{i}{i^b}$ und $\#W = \tilde{a} \sum_{i=1}^{m} \frac{1}{i^b}$.

m	$\tilde{a}$	$\sum_{i=1}^{m} \frac{i}{i^b}$	$\tilde{N}$	$\sum_{i=1}^{m} \frac{1}{i^b}$	$\#W$
1	1,0	1,00	1,0	1,00	1,0
2	2,7	1,73	4,7	1,37	3,7
3	4,9	2,34	11,5	1,57	7,7
4	7,5	2,88	21,5	1,70	12,7
5	10,3	3,36	34,7	1,80	18,6
6	13,4	3,81	51,2	1,87	25,2
7	16,8	4,23	71,0	1,93	32,5
8	20,4	4,62	94,2	1,98	40,4
9	24,2	4,99	120,7	2,02	49,0
10	28,2	5,34	150,6	2,06	58,1
11	32,4	5,68	184,0	2,09	67,7
12	36,7	6,01	220,7	2,12	77,8
13	41,2	6,33	260,9	2,14	88,3
14	45,9	6,63	304,5	2,16	99,4
15	50,7	6,93	351,5	2,18	110,8
16	**55,7**	**7,21**	**402,0**	**2,20**	**122,7**
17	60,8	7,49	455,9	2,22	134,9
18	66,1	7,77	513,3	2,23	147,6
19	71,5	8,03	574,2	2,25	160,6
20	77,0	8,29	638,5	2,26	174,1

Tabelle 8.3: Geschätzte Anzahl der verschiedenen Wörter ($b = 1.45$)

Im aktuellen Beispiel ist dann zu erwarten, dass ca. 120 statt der 400 verschiedenen unbekannten Wörter zu lernen sind, da nach Tabelle 8.3 für $m = 16$ ein Wert von $\tilde{N} = 402$ für die geschätzte Zahl unbekannter Wörter insgesamt angenommen wird. Die geschätzte Zahl der verschiedenen unbekannten Wörter wächst dabei überproportional zur erwarteten Zahl erwarteter unbekannter Wörter.

Würde, wie im bisherigen Beispiel, bei einer einzigen Klasse die mittlere Wahrscheinlichkeit eines unbekannten Wortes ($\frac{1}{200000}$) als initiale klassenbedingte Wahrscheinlichkeit

für ein neu gelernten Wortes verwendet, so würde diese Wahrscheinlichkeit mit dieser Schätzung $p(w_{\text{neu}}|g_1) = \frac{1}{120} * \frac{4}{5} = \frac{1}{150}$ um einem Faktor von über 1333 höher ausfallen. Im Gegensatz zu einem inkrementellen Aufteilen der gesamten verfügbaren Wahrscheinlichkeitsmasse fällt diese Wahrscheinlichkeit beispielsweise beim ersten Wort um einen Faktor von 120 geringer aus. Überschreitet die Zahl der gelernten Wörter die erwartete Zahl unbekannter Wörter, muss die Wahrscheinlichkeit der bereits gelernten Wörter angepasst werden.

Der Ansatz lässt sich einfach auf mehr als eine Klasse übertragen, indem eine Schätzung der OOV-Raten für die einzelnen Klassen verwendet wird.

Verhältnis zwischen dem häufigsten Wort in einem Korpus und dem häufigsten bzw. seltensten OOV-Wort

In diesem Abschnitt soll das Verhältnis näher betrachtet werden, in dem das häufigste OOV-Wort eines Textausschnittes zu dem Wort steht, das am häufigsten in einem Sprachmodellkorpus auftritt. Zu der Untersuchung wird das Wort "the" herangezogen, das im BN-Korpus am häufigsten auftritt.

Die zugrundeliegende Idee dabei ist, dass die Wahrscheinlichkeit für unbekannte Wörter um einen Faktor dieser Größenordnung angehoben werden könnte.

Das Wort "the" tritt im gesamten BN-Korpus 7001501 Mal auf, das häufigste OOV-Wort 30 Mal, das seltenste ein einziges Mal. Es ergibt sich also ein Verhältnis von $7001501 : 30 \approx 233400$. Betrachtet man nun kürzere Textabschnitte von beispielsweise 50000 Wörtern, so ergibt sich nun im Mittel ein Verhältnis vom häufigsten Wort zum Wort "the" von 288.7, womit diese seltenen Wörter im Mittel 808 Mal häufiger in den untersuchten Abschnitten sind als auf dem gesamten Text.

Ausschnittlänge	#OOV	#verschiedener Wörter	Verhältniszahl	Faktor
10000	55,3	40,5	128,8	1812
20000	110,5	80,1	177,4	1315
50000	276,3	197,1	288,7	808
100000	552,7	389,2	438,2	532
250000	1381,6	951,4	812,4	278
500000	2763,3	1864,8	1358,0	172
1000000	5526,5	3632,9	2291,0	102

Tabelle 8.4: Verhältnis des häufigsten OOV-Wortes zum Wort "the"

Betrachtet man Tabelle 8.4, in der für verschieden lange Textausschnitte die Mittelwerte der Zahlenverhältnisse und der daraus resultierende Faktor aufgelistet wurde, so sieht man, dass beispielsweise bei einer Textlänge von 20000 Wörtern die initiale Wahrscheinlichkeit für ein gelerntes Wort um den Faktor 1315 höher angesetzt werden kann, als beispielsweise mit $\frac{1}{200000}$ angenommen wird. Dieser Wert liegt dabei sehr nahe an dem durch die Zipf'sche Verteilung geschätzten Wert von 1333.

Die vorgestellten Methoden, mit der Zipf'schen Verteilung und dem Verhältnis, in dem das häufigste OOV-Wort zu dem am häufigsten im Sprachmodellkorpus auftretenden Wort "the" steht, kommen damit zu vergleichbaren Werten, um die die initiale Wahrscheinlichkeit eines neu gelernten Wortes höher geschätzt werden kann. Die Wahrscheinlichkeit eines gelernten Wortes kann bei einer erwarteten Textlänge von 20000 Wörtern um einen Faktor von 400 bis 1400 höher geschätzt werden, als es die Wahrscheinlichkeit eines durchschnittlichen OOV-Wortes auf dem Gesamtkorpus eigentlich wäre.

8.2.2 Simulation des Lernvorgangs

Die bisher beschriebenen Methoden gehen von einer zu erwartenden Länge des Sprachsignals aus und schätzen darüber die Anzahl zu erwartender unbekannter Wörter. Dabei werden implizit am Ende alle gelernten Wörter wieder aus dem Vokabular entfernt. Es ist aber auch denkbar, dass die Wörter für eine unbestimmt lange Zeitspanne gelernt werden.

Das Simulationsverfahren arbeitet mit einem Erweiterungsvokabular $V+$, in dem zusätzlich die Position eines Wortes gespeichert wird, an der es zuletzt aufgetreten ist. Es besitzt die Fähigkeit, Wörter wieder aus dem Vokabular zu entfernen, die einmal gelernt und danach für eine Zeitspanne nicht mehr aufgetreten sind. Treten die Wörter wieder auf nachdem sie verlernt wurden, müssen sie erneut detektiert werden. Daher ist zu erwarten, dass eine größere Wahrscheinlichkeitsmasse für die Detektion zu verwenden ist, sich dafür aber die Anzahl der gleichzeitig gelernten Wörter verringert, was die Wahrscheinlichkeit $p(w_{\text{neu}}|g)$ für ein aktuell gelerntes konkretes Wort w_{neu} in der Klasse g erhöht.

Beim Lernen eines unbekannten Wortes wird während der Simulation festgelegt, zu welchem Zeitpunkt es wieder vergessen werden soll, falls es in der Zwischenzeit nicht erneut aufgetreten ist. Tritt ein gelerntes Wort bis dahin wieder auf, wird der Zeitpunkt zu dem es vergessen werden soll, erneut bestimmt. Eine Beschreibung der Verfahren in Pseudo-Code ist im Anhang B gegeben.

Bei der Simulation SIM-1 ohne Langzeitgedächtnis ist die Gedächtnisdauer N für alle Wörter gleichlang, unabhängig davon, ob sie bereits einmal gelernt wurden oder nicht.

Simulation SIM-2 benutzt ein Langzeitgedächtnis, in dem für Wörter, die zuvor bereits gelernt waren, die Gedächtnisdauer so bestimmt wird, dass sie der Zeitspanne vom vorherigen bis zum wiederholten Auftreten entspricht. Damit wird die Zeitspanne so gewählt, dass das Wort *gerade nicht* verlernt worden wäre. Die Gedächtnisdauer passt sich somit dynamisch an die Länge der Pausen an, in denen das Wort nicht auftritt.

In Abbildung 8.1 kann man sehen, dass die Zahl der gleichzeitig gelernten Wörter zwar deutlich schwankt, aber dass sie über die Zeitdauer nicht anwächst. Bei einer Gedächtnisdauer von beispielsweise $N = 50000$ lässt sich mit diesem Simulationsverfahren die Zahl der gleichzeitig gelernten Wörter sehr gut durch eine Gauss-Verteilung $N(\mu, \sigma^2)$ mit $\mu = 209.9$ und $\sigma^2 = 3306$ modellieren, wie man in Abbildung 8.2 sehen kann.

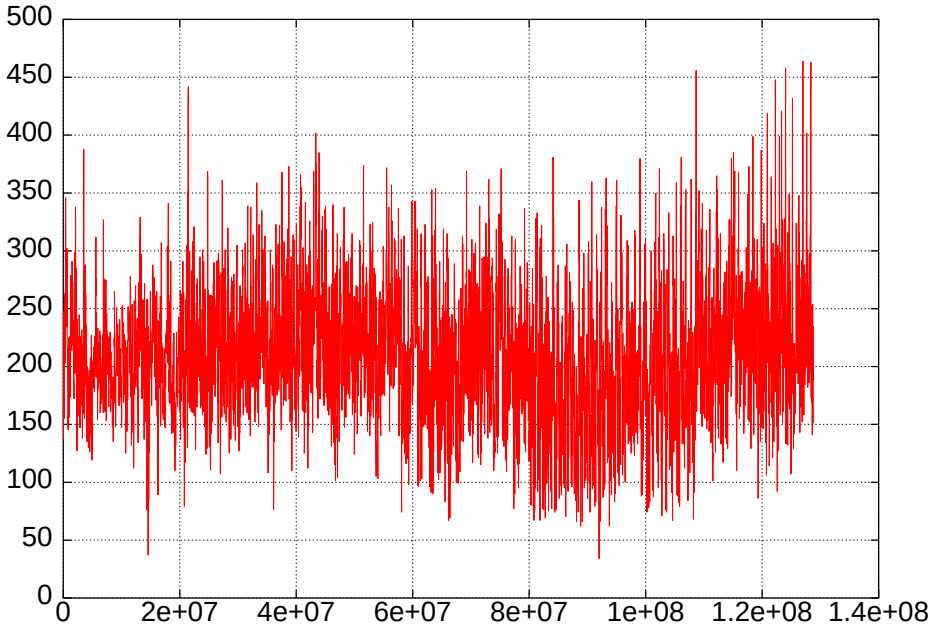

Abbildung 8.1: Anzahl aktiver Wörter in V_+ über dem BN-Korpus SIM-1 (N = 50000)

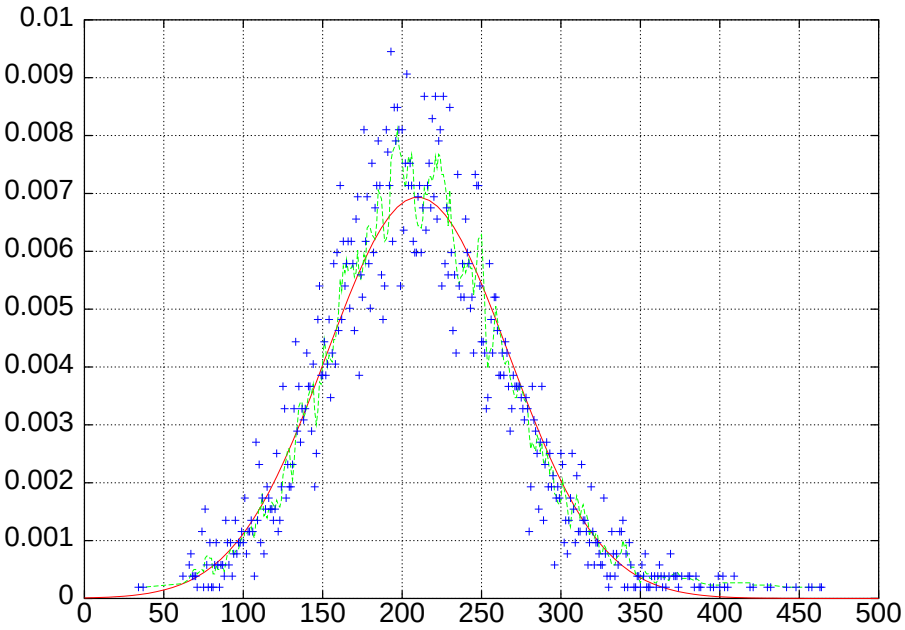

Abbildung 8.2: Häufigkeitsverteilung der Anzahl gelernter Wörter SIM-1 (N = 50000)

Methode	N	#P1	#P2+	μ_{V+}	σ_{V+}
SIM-1	20000	544189	219814	85,3	32,2
SIM-1	50000	534506	229497	209,9	57,5
SIM-2	50000	320447	443556	26201,9	6137,2

Tabelle 8.5: Statistik bei Simulation

In Tabelle 8.5 ist für die Verfahren SIM-1 und SIM-2 angegeben, wie häufig ein Wort gelernt werden muss (#P1), wie oft ein gelerntes Wort wiedergefunden wurde (#P2+) sowie die mittlere Anzahl gleichzeitig in das Vokabular $V+$ gelernter Wörter μ_{V+} mit Varianz σ_{V+}.

Die Untersuchungen wurden mit dem Basisvokabular V_{Top40k} der BN-Domäne durchgeführt. Nach jeweils $N/2$ Wörtern wurde der Umfang des gelernten Vokabulars auf dem Sprachmodellkorpus gemessen.

Vergleicht man das Ergebnis des Simulationsverfahrens SIM-1 für die Werte $N = 20000$ und $N = 50000$ fällt auf, dass die Zahl #P2+ um weniger als 5% ansteigt, obwohl die Gedächtnisdauer das 2.5fache beträgt. Dagegen wächst der Mittelwert μ_{V+} um ungefähr das 2.5fache an. Der lineare Zusammenhang zwischen dem Mittelwert μ_{V+} und N und der geringe Zuwachs an wiedergefundenen Wörtern deutet darauf hin, dass für die lokal auftretenden unbekannten Wörter eine Gedächtnisdauer von 20000 auf dem vorliegenden BN-Korpus ausreichend ist. Das Simulationsverfahren SIM-2 zeigt dagegen, dass zwischen dem Wiederauftreten einzelner Wörter auf dem BN-Korpus sehr große Pausen liegen können, was daran zu erkennen ist, dass in Abbildung 8.3 das Vokabular $V+$ stetig anwächst.

Gelernte Wörter wieder zu vergessen hat beispielsweise unter anderem den Vorteil, dass Wörter, die nur ein einziges Mal auftreten, rasch wieder aus dem Vokabular entfernt werden und dass Wörter, die nur über einen kurzen Abschnitt gebraucht werden, nur so lange in einem Erweiterungsvokabular gespeichert sind, wie sie tatsächlich auch benötigt werden.

Jedoch erscheint es wenig sinnvoll, ein bereits gelerntes Wort immer wieder zu vergessen, um es dann erneut zu lernen. Darum wurde für die Methode SIM-2, wie bereits beschrieben, das simulierte Langzeitgedächtnis realisiert. Dadurch müssen weniger Wörter detektiert werden, aber die Zahl der gleichzeitig gelernten Wörter fällt deutlich größer aus als im Vergleich zum Verfahren SIM-1.

Wie in Abbildung 8.3 zu sehen ist, wächst das Vokabular permanent über den gesamten Zeitraum und eine zu erwartende Sättigung ist noch nicht zu erkennen. Nach 120 Millionen Wörtern ist das Vokabular um zirka 45000 Wörter angewachsen. Ganz ohne Vergessen der gelernten Wörter, die längere Zeit nicht mehr aufgetreten sind, wären es zirka 200000 Wörter. Durch die Verwendung des Langzeitgedächtnisses verteilt sich zwar die Wahrscheinlichkeitsmasse zwischen Detektion $p(P1|g_1) = \frac{\#P1}{2\#P1+\#P2+}$ und Wiederfinden

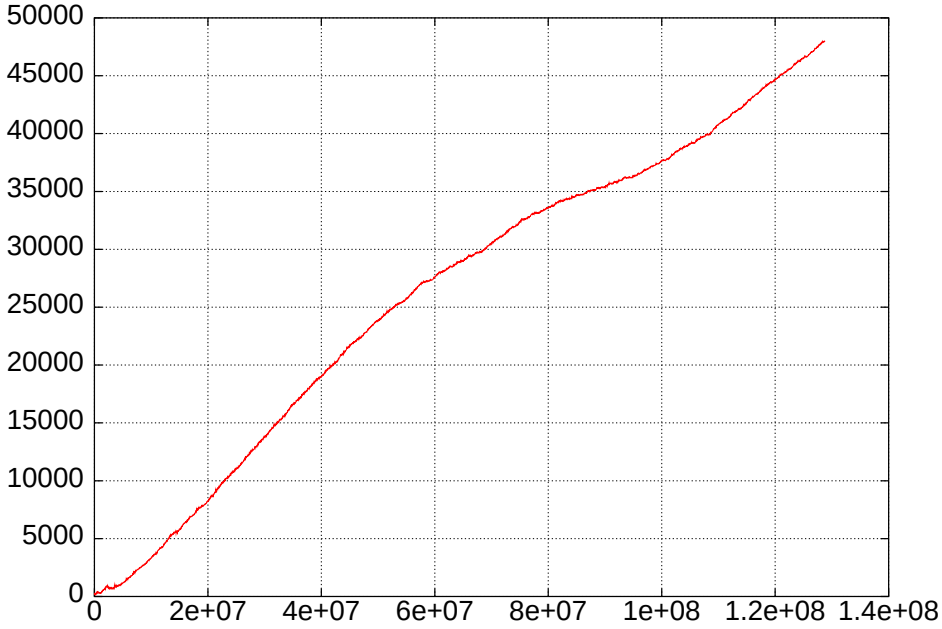

Abbildung 8.3: Anzahl aktiver Wörter in V_+ über dem BN-Korpus mit SIM-2 (N = 50000)

$p(P2+|g_1) = 1 - p(P1|g_1)$ so, dass nun eine größere Wahrscheinlichkeitsmasse auf das Wiederfinden entfällt, aber wegen der sehr viel größeren Zahl der gelernten Wörter fällt die Wahrscheinlichkeit für ein konkretes Wort geringer aus, wie in Tabelle 8.6 im Vergleich mit dem SIM-1 Verfahren zu erkennen ist.

Um mehr als 90% der zu erwartenden Fälle zu lernender Wörter abzudecken, wird für die erwartete Klassengröße $\mu + 1.3\sigma$ benutzt. Bei Verfahren SIM-2 wird ebenfalls diese Schätzung verwendet. Da sich die Klassengröße hier aber nicht mit einer Normalverteilung modellieren lässt, sind es keine 90% der Fälle, die dadurch abgedeckt werden, sondern nur ungefähr 64%.

Zur leichteren Vergleichbarkeit mit der angenäherten durchschnittlichen Wahrscheinlichkeit eines OOV-Wortes von $\frac{1}{200000}$ wird die resultierende Wahrscheinlichkeit

$$p(w_{\text{neu}}|g_1) = \frac{p(P2+|g_1)}{\mu + 1.3\sigma} \tag{8.10}$$

als Bruch angegeben. Zusätzlich gibt die Spalte *Faktor* an, um wie viel wahrscheinlicher ein neu gelerntes Wort im Vergleich dazu ist. Mit g_1 wird eine Klasse bezeichnet, um hervorzuheben, dass das Verfahren klassenabhängige Schätzung macht, auch wenn in diesem Abschnitt alle Untersuchungen stets mit einer einzigen Klasse durchgeführt wurden.

Welche Art der Simulation eingesetzt werden soll, hängt von der Anwendung ab. Eine Kombination beider Ergebnisse scheint für viele Anwendungen sinnvoll. Für ein gerade gelerntes Wort wird dabei eine Schätzung des Klassenumfangs nach SIM-1 verwendet und

Methode	N	$p(P1\|g_1)$	$p(P2+\|g_1)$	$\mu + 1.3\sigma$	$p(w_{\text{neu}}\|g_1)$	Faktor
SIM-1	20000	41,6%	58,4%	127,2	$\frac{1}{217,8}$	918
SIM-1	50000	41,2%	58,8%	284,7	$\frac{1}{484,2}$	413
SIM-2	50000	29,5%	70,5%	34180,3	$\frac{1}{48482,7}$	4,1

Tabelle 8.6: Vergleich der Wahrscheinlichkeiten

dann, falls das Wort nicht vergessen werden soll, nachdem das Wort über eine Zeitspanne von N Wörter nicht wieder aufgetreten ist, die Wahrscheinlichkeit auf einen mit SIM-2 geschätzten Wert gesetzt, bis das Wort vergessen oder erneut verwendet wird. Bei erneuter Verwendung wird es wieder mit einer Wahrscheinlichkeit versehen, als wäre es gerade gelernt worden. Dies entspricht der Verwendung eines einfachen Cache-Sprachmodells, bei dem lediglich eine klassenbedingte Wahrscheinlichkeit für ein Wort angepasst wird.

8.3 Zusammenfassung

In diesem Kapitel wurde dargelegt, warum es für das Online-Lernen notwendig ist, die Wahrscheinlichkeitsmasse, die für die Modellierung unbekannter Wörter reserviert ist, aufzuteilen in einen Teil, der für die Detektion unbekannter Wörter, und in einen Teil, der für neu gelernte Wörter benutzt werden kann, da sowohl gelernte Wörter wiedergefunden werden sollen als auch weitere Detektionen möglich sein sollen.

Es wurde gezeigt, welche Folgen eine zu geringe Wahrscheinlichkeit für ein gelerntes Wort haben kann, und dass die Wahrscheinlichkeit für gelernte Wörter höher geschätzt werden soll, als für ein allgemeines seltenes Wort. Ferner wurde erläutert, warum die initiale Wahrscheinlichkeit beim Lernen eines Wortes auf die *erwartete* Klassengröße angepasst und diese erst bei Überschreitung inkrementell an die *neue* Klassengröße angepasst werden soll.

Hierzu wurden die Orakel-Experimente aus Kapitel 7.3 weiter analysiert. Aus ihnen ist zu ersehen, dass eine zu große Wahrscheinlichkeit zu einem Ansteigen der fehlerhaften Hypothesen der gelernten Wörter führt. Bei einer zu niedrige Wahrscheinlichkeit werden aber deutlich weniger der "gelernten" Wörter gefunden.

Folgende Methoden zur Aufteilung der Wahrscheinlichkeitsmasse wurden entwickelt:

- Analyse der Verteilung von pUNK-Wörtern einer Klasse
- Parametrische Schätzung der Verteilung unbekannter Wörter durch Zipf'sche Verteilung
- Simulation des Lernvorgangs SIM-1 und SIM-2 mit Langzeitgedächtnis

Dabei ergab sich, dass auf dem BN-Korpus die Wahrscheinlichkeit, die für die Detektion unbekannter Wörter reserviert werden soll, bei der Simulation SIM-1 mit ungefähr $\frac{2}{5}$

doppelt so hoch ist, wie bei den anderen Schätzungen, da die gleichen Wörter mehrfach detektiert werden müssen. Bei Verwendung eines Langzeitgedächtnisses (SIM-2) liegt die Schätzung ungefähr in der Mitte zwischen $\frac{1}{5}$ und $\frac{2}{5}$ der Wahrscheinlichkeitsmasse, da weniger Wörter wiedergefunden werden müssen. In Tabelle 8.7 sind die Ergebnisse zusammengefasst.

Methode	$p(P1\|g_1)$	$p(P2+\|g_1)$
pUNK-Verteilung	18,1%	81,9%
Zipf'sche Verteilung	19,8%	80,2%
SIM-1	41,6%	58,4%
SIM-2	29,5%	71,5%

Tabelle 8.7: Vergleich Aufteilung der Wahrscheinlichkeitsmasse

Für die Schätzung der initialen Wahrscheinlichkeit, mit der ein Wort gelernt wird, wurden verschiedene Methoden vorgestellt, die versuchen, die zu erwartende Menge verschiedener unbekannter Wörter zu schätzen oder einen Faktor zu bestimmen, um den die Wahrscheilichkeit eines gelernten Wortes erhöht wird im Vergleich zu einem durchschnittlichen OOV-Wort.

Folgende Verfahren wurden betrachtet:

- Schätzung der Gesamtanzahl unbekannter Wörter (konservativ)
- Parametrische Schätzung mit der Zipf'schen Verteilung (Zipf)
- Schätzung des Verhältnisses zwischen häufigstem OOV-Wort und häufigstem Wort (The-Verhältnis)
- Simulation des Lernvorganges (SIM-1)

Dabei ergab sich, dass die Wahrscheinlichkeit für ein gelerntes Wort auf dem BN-Korpus um einen Faktor von ungefähr 400 bis ungefähr 2000 höher geschätzt werden kann, als ein durchschnittliches unbekanntes Wort, wenn eine Textlänge von 20000 Wörtern zugrunde gelegt wurde. In Tabelle 8.8 sind einige Ergebnisse zusammengestellt.

Der geringste Wert von 4 ergab sich bei der Simulation mit Langzeitgedächtnis (SIM-2). Der Grund ist, dass durch das Langzeitgedächtnis das Vokabular konstant anwächst. Sollen Wörter, die ein zweites mal gelernt wurden, beibehalten werden oder erst zu einem sehr viel späteren Zeitpunkt möglicherweise wieder verlernt werden, kann die Wahrscheinlichkeit, nach Ablauf des Kurzzeitgedächtnisses auf diesen Wert angepasst werden (oder dorthin konvergieren).

Bei der Simulation ohne Langzeitgedächtnis (SIM-1) tritt auf dem untersuchten BN-Korpus für die untersuchten Gedächtnisdauern eine rasche Stabilisierung der Zahl

Methode	$p(w_{\text{neu}}\|g_1)$	Faktor
SIM-1	$\frac{3}{5} * \frac{1}{128} = \frac{1}{218}$	≈ 1000
Zipf	$\frac{4}{5} * \frac{1}{120} = \frac{1}{150}$	≈ 1333
Konservativ	$\frac{4}{5} * \frac{1}{400} = \frac{1}{500}$	≈ 400
The-Verhältnis	$\frac{1}{150} \approx \frac{4}{5} * \frac{1}{120}$	≈ 1315

Tabelle 8.8: Übersicht Faktoren für den BN-Task, Länge von 20000 Wörtern

im Mittel gelernter Wörter ein. Die Verteilung über die Zahl der in das Vokabular hinzugefügten Wörter, lässt sich dabei sehr gut mit einer Gauss-Verteilung modellieren und so eine initiale Anzahl zu lernender Wörter festsetzen, die je nach Bedarf eine vorgegebene Anzahl von Fällen abdeckt, beispielsweise 90%. Das heißt, in 10% der Fälle würde die erwartete Anzahl an verschiedenen OOV-Wörtern unterschätzt.

Die beschriebenen Verfahren lassen sich ohne weiteres auch für mehr als eine Klasse anwenden. In diesem Fall ist für jede Klasse eine separate Schätzung der Aufteilung und der initialen Wahrscheinlichkeit durchzuführen. Insbesondere bedeutet dies, dass bei den Simulationsverfahren ein klassenabhängiges Erweiterungsvokabular $V_{\text{g}+}$ zu benutzen ist. Da bei der Verwendung der Zipf'schen Verteilung, die Parameter für die Klassen unterschiedlich ausfallen, folgt daraus, dass sowohl die Aufteilung der Wahrscheinlichkeitsmassen als auch die initiale Wahrscheinlichkeit für ein neu gelerntes Wort klassenabhängig ist.

Kapitel 9

Online-Lernen unbekannter Wörter im Dialog

In diesem Kapitel wird einer der Prototypen zum interaktiven Online-Lernen von unbekannten Wörtern auf Initiative des sprachverarbeitenden Systems vorgestellt. Zunächst werden grob der Systemaufbau und anschließend die wesentlichen Komponenten und deren Zusammenspiel genauer beschrieben. Ein konkretes Beispiel dient zur Veranschaulichung des Dialogablaufs beim Erlernen eines unbekannten Wortes.

9.1 Grober Aufbau eines Prototypen

Der Prototyp besteht im Wesentlichen aus den drei Teilen Audioserver, Spracherkenner und Dialogmanager. Der Audioserver ist für die Sprachaufnahme und -ausgabe zuständig. Neben der eigentlichen Spracherkennung initiiert der Spracherkenner einen Dialog mit dem Benutzer, um die Informationen zu erhalten, die er zum Lernen von unbekannten Wörtern benötigt. Der Dialogmanager führt diesen Dialog durch und wertet die vom Benutzer gegebenen Informationen aus, anhand derer er den Spracherkenner dann in verschiedene Modi, wie beispielsweise den Buchstabiermodus versetzt.

In Abbildung 9.1 ist der prinzipielle Aufbau dieses Prototypen dargestellt. Der Audioserver nimmt dabei vom Benutzer ein Audiosignal (dicke Pfeile im Schaubild) über ein Mikrophon auf und leitet dieses weiter an den Spracherkenner. Audiosignale vom Dialogmanager werden über den Audioserver für den Benutzer auf einem Lautsprecher wiedergegeben. Der Audioserver ist als separater Prozess realisiert, der auch auf einem anderen, beispielsweise beim Benutzer stehenden, möglicherweise weniger leistungsfähigen Computer arbeiten kann, was aktuell für den Spracherkenner nicht gilt.

Die NLP-Komponente ist nur aus Gründen der Vollständigkeit in das Schaubild aufgenommen worden, um die prinzipielle Anbindung des Systems zu veranschaulichen. Dabei handelt es sich um eine Analysekomponente für natürliche Sprache (NLP = Natural

Language Processing), die versucht, die Bedeutung der Hypothese zur Weiterverarbeitung zu extrahieren und beispielsweise die Übersetzung in eine andere Sprache zu veranlassen.

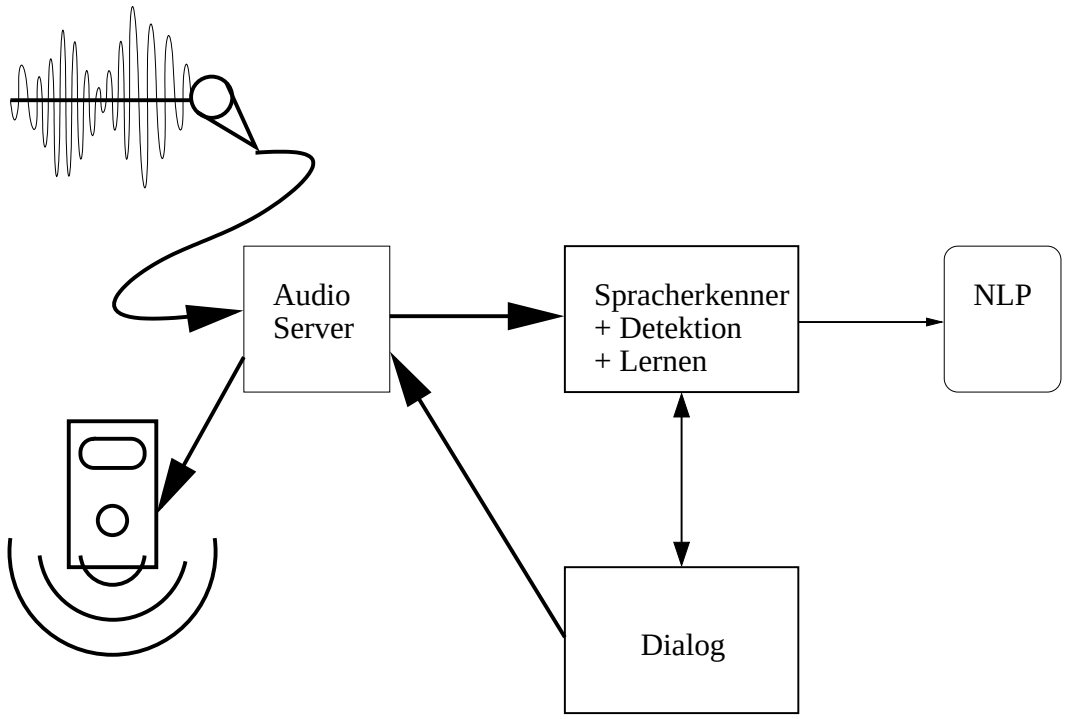

Abbildung 9.1: Allgemeiner Aufbau des Prototypen

9.2 Beschreibung der wesentlichen Komponenten

Der im Prototyp integrierte Spracherkenner unterstützt die Erkennung von Buchstaben-, Phonem- und Wortfolgen. Um das Vorhandensein eines unbekannten Wortes zu detektieren, wird die in Kapitel 6 und 7 beschriebene Kombination von Sprachmodellklassen für unbekannte Wörter und Head-Tail-Modellen als generische Wörter eingesetzt.

Als Auslöser für einen Dialog mit dem Benutzer dient das Auftreten eines generischen Wortes in der besten Hypothese, optional kann diese Entscheidung auch auf den N-Besten Hypothesen durchgeführt werden.

Wird ein unbekanntes Wort vermutet, so wird die Äußerung gespeichert, um in einem späteren Dialogschritt den Erfolg des Lernens überprüfen zu können. Außerdem wird nach Dialogende eine verbesserte Hypothese weitergegeben. Der initiierte Dialog wird vom Dialogmanager durchgeführt, der als endlicher Automat realisiert ist, und dessen Zustandsgraph in Abbildung 9.2 und den Tabellen 9.1, 9.2 und 9.3 beschrieben ist. Die dicken Pfeile

in Abbildung 9.2 entsprechen der Zustandsfolge, die den optimalen Dialogverlauf darstellt, wenn ein Wort zu lernen ist.

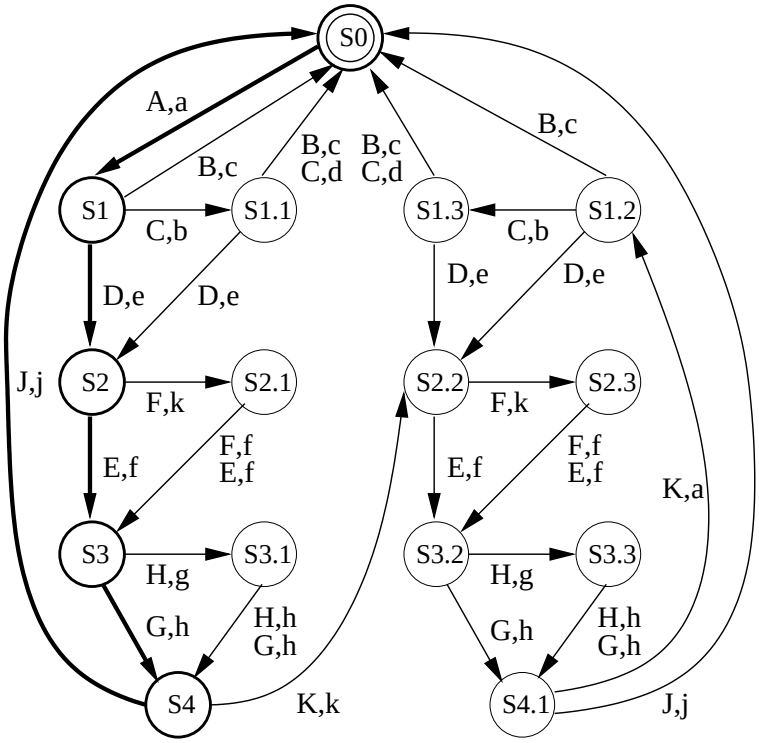

Abbildung 9.2: Zustandsautomat des Dialogs

Zustand	Beschreibung
S0	Initialer Zustand. Wartet auf Detektion eines zu lernenden Wortes
S1 S1.1 S1.2 S1.3	Wartet auf Antwort aufgrund einer Nachfrage beim Benutzer
S2 S2.1 S2.2 S2.3	Wartet auf Ergebnis der isoliert gesprochenen Wortwiederholung
S3 S3.1 S3.2 S3.3	Wartet auf Ergebnis der Buchstabierung des zu lernenden Wortes
S4 S4.1	Wartet auf Ergebnis der Überprüfung des Lernerfolges

Tabelle 9.1: Beschreibung der Zustände des Zustandsautomaten

In einem vollständigen System kann die verbesserte Ausgabe des Spracherkenners, also dessen Hypothese, auch an die NLP-Komponente weitergegeben werden. Für den (hier) realisierten Prototyp findet jedoch lediglich eine Bildschirmausgabe statt, die den Prozess beim Erlernen eines unbekannten Wortes veranschaulichen soll.

Für Verbmobil [66] wurde ein Mechanismus definiert, der es erlaubt Wörter, die der NLP-Komponente unbekannt sind, so zu kapseln, dass diese dennoch analysiert werden können. Dadurch ist es für die NLP-Komponente unerheblich, ob ein Wort bereits beim

Kürzel	Übergangsbedingung
A	Spracherkenner vermutet unbekanntes Wort
B	Es wurde eine Ablehnung erkannt
C	Es wurde keine eindeutige Ablehnung/Zustimmung erkannt
D	Es wurde eine Zustimmung erkannt
E	Es wurden mindestens N_p Phoneme erkannt
F	Es wurden weniger als N_p Phoneme erkannt
G	Es wurden mindestens N_b Buchstaben erkannt
H	Es wurden weniger als N_b Buchstaben erkannt
J	Das Wort wurde erfolgreich gelernt
K	Das Wort wurde nicht erfolgreich gelernt

Tabelle 9.2: Beschreibung der Übergangsbedingungen

Kürzel	Veranlasste Aktion
a	Frage an Benutzer, ob ein Name (der detektierten Klasse) benutzt wurde
b	Aufforderung mit "Ja" oder "Nein" zu antworten
c	Signalisierung des Dialogendes "Danke" mit der Vermutung, dass kein unbekanntes Wort vorliegt, und somit die Hypothese mit diesem Zusatzwissen erneut analysiert werden kann. Weitergabe einer verbesserten Hypothese an nachfolgende NLP-Komponente durch den Spracherkenner
d	Signalisierung des Dialogendes "Danke"; Ausgabe der bereits gefundenen Hypothese des Spracherkenners
e	Aufforderung, das betreffende Wort (isoliert) zu wiederholen
f	Aufforderung, das betreffende Wort zu buchstabieren
g	Erneute Aufforderung, das betreffende Wort zu buchstabieren
h	Information an Benutzer, dass versucht wird, das Wort zu lernen; Überprüfung des Lernerfolges beim Spracherkenner wird veranlasst
j	Signalisierung des Dialogendes "Danke" mit der Information, dass das Lernen erfolgreich war
k	Erneute Aufforderung, das betreffende Wort (isoliert) zu wiederholen
l	Frage an Benutzer, ob ein weiterer Lernversuch unternommen werden soll

Tabelle 9.3: Beschreibung der Aktionen

Starten des Spracherkenners bekannt war, oder nachträglich erst gelernt wurde. In dem entwickelten Format wird das Wort zusammen mit einer Phonemumschrift (Sampa) für die Sprachsynthese und der Klasseninformation übermittelt. Ein Beispiel hierfür ist das "Wort" <UNK:Surname.Sa:f.Schaaf>, das den Nachnamen "Schaaf" repräsentiert.

Durch diesen Mechanismus ist es möglich, das im Spracherkenner bekannte Vokabular einfach zu vergrößern, ohne dass die NLP-Komponenten davon betroffen sind. Eine sinnvolle Verarbeitung durch eine NLP-Komponente ist jedoch nur dann zu erwarten, wenn die hinzugefügten Wörter aus einer offenen Klasse stammen, also beispielsweise Orts- oder Familiennamen, oder aber die NLP-Komponente sich selbst das nötige Wissen für die Analyse des Wortes nachträglich aneignen kann.

Ein Vorteil dieses Mechanismus ist, dass für das Lernen eines (unbekannten) Wortes keine weiteren Änderungen mehr an der NLP-Komponente nötig sind, wenn man den Benutzerdialog mit einem Dialogmanager durchführt, der nahe beim Spracherkenner und vor einer NLP-Komponente angelegt ist (siehe Abbildung 9.2).

Durch die Verwendung eines Abfragedialoges und die enge Kooperation zwischen Dialogmanager und Spracherkenner kann die Komplexität der Dialogkomponente im Prototyp kleingehalten werden. Für flexiblere Dialoge, in denen kein so starrer Ablauf vorgegeben ist, kann auch ein komplexeres Dialogsysteme benutzt werden, wie beispielsweise Ariadne, das von Denecke in [19] beschrieben ist.

Der Automat in Abbildung 9.2 stellt die Grundstruktur zum Erlernen eines unbekannten Wortes dar, wenn kein zusätzliches Wissen über die Semantik dieses Wortes vorliegt.

Ist jedoch von vornherein ein semantisches Klassensystem im statistischen Sprachmodell vorgegeben, oder werden die unüberwacht gefundenen Klassen soweit möglich auf semantische Klassen abgebildet (vgl. Kapitel 6), lässt sich der Dialog auf einfache Weise erweitern, um diese Informationen in der Dialogführung und beim Erlernen eines unbekannten Wortes berücksichtigen zu können.

Falls beispielsweise der Benutzer die im Dialog vorgeschlagene Klasse ändert, wird die angepasste Klasse für die Ausgabe an die nachfolgende NLP-Komponente weitergegeben.

Dies muss aber nicht zwingend dazuführen, dass nun nur noch diese Klasse allein für die Spracherkennung verwendet wird. Sie wird mit in den Pool der zu überprüfenden Klassen im statistischen Sprachmodell aufgenommen und durch die Überprüfung des Lernerfolges wird dann entschieden, welche Klassen einem gelernten Wort zugeordnet werden.

Dazu werden die während des Benutzerdialogs gewonnenen Kandidaten für die Aussprache mit den bei der Detektion gefundenen Klassen des Sprachmodells kombiniert, und so in das Vokabular des Spracherkenners als Kandidaten für das zu lernende Wort aufgenommen. Bei der optionalen Verwendung einer N-Bestenliste bei der Detektion kann dies durchaus mehr als eine Klasse sein. Für den Fall, dass vom Benutzer eine andere Klasse ausgewählt wurde, wird diese ebenfalls mit aufgenommen. Dabei wird für jedes Kandidatenwort eine klassenbedingte Wahrscheinlichkeit $p(w_{\text{new}}|\text{<OOV-ClassX>})$, wie in Kapitel 8 beschrieben, bestimmt.

Das Lernen wird als erfolgreich angesehen, wenn beim erneuten Analysieren der ur-

sprünglichen Äußerung eines der Kandidatenwörter in der besten Hypothese an der entsprechenden Stelle auftritt. Insbesondere muss hierzu eines der Kandidatenwörter das generische Wort verdrängen, da diese im Spracherkenner für die Detektion weiterer unbekannter Wörter verbleiben. Falls dies nicht der Fall ist, wird dem Dialog signalisiert, dass das Lernen mit diesen Kandidaten nicht erfolgreich war und der Dialog wird fortgesetzt. Ist das Lernen erfolgreich, wird optional die N-Besten-Liste oder der geprunte Wortgraph auf weitere Kandidaten untersucht. Die dort zusätzlich gefundenen Kandidaten können sich von dem in der besten Hypothese gefundenen Kandidaten sowohl in der Aussprache/Phonemfolge als auch in der zugeordneten Klasse unterscheiden und werden zur Erhöhung der Robustheit gegen Erkennungsfehler während des Lernens ebenfalls fest in den Spracherkenner übernommen.

Bei der Übernahme der Kandidaten in das in Zukunft zu verwendende Vokabular wird festgelegt, welche Klasse, Phonem- und Buchstabenfolge für die Ausgabe des gelernten Wortes an eine nachfolgende NLP-Komponente benutzt wird. Diese Festlegung ist auch bei einer fehlerhaften Verschriftung sinnvoll, um bei einer späteren Verwendung des Wortes nicht annehmen zu müssen, dass es sich um verschiedene Wörter handelt. Dabei kann, abhängig von der Anwendung, beispielsweise die gleiche Buchstabenfolge für alle aufgenommenen Kandidaten festlegt werden. Für eine reine Textausgabe in einem Protokoll ist dies ausreichend. Falls für die Ausgabe eine Sprachsynthese verwendet wird, um das Wort bei einer Wiederholung immer wieder erkennen zu können, kann zusätzlich auch noch die Phonemumschrift, die an die NLP-Komponente weitergereicht wird, für alle Kandidaten einheitlich festgelegt werden. Dies ist unabhängig davon, welcher akustische Repräsentant im Spracherkennerwörterbuch verwendet wird. Ob auch die Klasse festgelegt wird, ist abhängig von der Fähigkeit der NLP-Komponente und davon, ob der Benutzer eine spezifische Klasse vorgegeben hat.

9.3 Beispiel für den Ablauf beim Erlernen eines unbekannten Wortes

Zur Veranschaulichung wird nun an einem konkreten Beispiel der Ablauf beim Erlernen eines unbekannten Wortes dargestellt. Dabei wird erläutert, welche Schritte im Spracherkenner durchgeführt werden. Das Beispiel folgt dabei dem Dialogverlauf in Abbildung 9.2, der durch die dick hervorgehobenen Pfeile markiert ist. Der Dialog ist ein reales Beispiel, das mit dem englischen Prototypen geführt wurde. Dabei wurde das NE-LM Sprachmodell (vgl. Kapitel 6) benutzt, das für unbekannte Wörter über die Klassen "PERSON", "LOCATION" und "ORGANISATION" verfügt.

Der Dialogmanager befindet sich im initialen Zustand S0.
Der Benutzer sagt: "I live in Maximiliansau."

Hypothese des Spracherkenners: "I LIVE IN <OOV:Location> MILLIONS OUT"

Das Auftreten des generischen Wortes "<OOV:Location>" in der Hypothese führt zu einem Zustandswechsel von S0 nach S1 im Dialogmanager und löst eine Rückfrage des Dialogsystems beim Benutzer aus.

Systemreaktion an den Benutzer: "Did you use the name of a location?"

Der Benutzer antwortet: "Yes."

Der Spracherkenner ist weiterhin im Modus für die Erkennung von Wortfolgen konfiguriert, wodurch der Benutzer nicht von vornherein auf eine "Yes" oder "No"- Antwort eingeschränkt ist.

Hypothese des Spracherkenners: "YES"

Aufgrund dieser Antwort nimmt das System an, dass dieses Wort gelernt werden soll und wechselt von Zustand S1 nach S2; dabei wird der Spracherkenner zur Phonemerkennung konfiguriert.

Systemreaktion an den Benutzer: "Please repeat the name."

Der Benutzer wiederholt: "Maximiliansau."

Die Hypothese des Spracherkenners ist eine N-Besten-Liste aus Phonemfolgen, in in Tabelle 9.4 angegeben ist.

Rang	Hypothese
1	/ n e k s @ v I l i: @ n z a U t /
2	/ m { k s @ v I l i: @ n z a U t /
3	/ n e k s @ v I l j @ n z a U t /
4	/ n e k s I v I l i: @ n z a U t /
5	/ n e k s I v I H l i: @ n z a U t /
6	/ n e k s @ v I l j @ n z a U t /
7	/ n e k s I v I l i: @ n z a U t /
8	/ n e k s I v I l i: @ n z a U t /
9	/ n e k s [v I l i: @ n z a U t /
10	/ n e k s @ v I l i: I n z a U t /

Tabelle 9.4: N-Bestenliste der Phonemerkennung

Da mindestens drei Phoneme ($N_p = 3$) in der besten Hypothese erkannt wurden, folgt keine Aufforderung zur Wiederholung, sondern es findet ein Zustandswechsel von S2 nach S3 statt und der Spracherkenner wird für die Erkennung von Buchstabenfolgen konfiguriert.

Systemreaktion an den Benutzer: "Please spell the name."

Der Benutzer buchstabiert fließend: "M A X I M I L I A N S A U"

Die Hypothese des Spracherkenners ist die N-Besten-Liste für die Buchstabenfolge aus Tabelle 9.5. Erkannt wurden dabei in der besten Hypothese mindestens drei Buchstaben ($N_b = 3$). Andernfalls hätte das Dialogsystem den Benutzer aufgefordert, die letzte Eingabe zu wiederholen. Nun findet ein Zustandswechsel von S3 nach S4 statt und es erfolgt die

Systeminformation an den Benutzer: "Please wait. Check success."

Rang	Hypothese
1	M A X I M I L I A N S A U
2	M A X I N I L I A N S A U
3	N A X I M I L I A N S A U
4	M A X I A N I L I A N S A U
5	M A X I A M I L I A N S A U

Tabelle 9.5: N-Bestenliste der Buchstabenerkennung

Nun wird aus der Phonem- und der Buchstabenerkennung eine Liste von Aussprachen erzeugt, die als Kandidaten für das zu lernende Wort dienen. Die Phonemfolgen, die aus Buchstaben gewonnen werden können, sind mit der ranghöchsten Buchstabenfolge assoziiert, von der die jeweilige Phonemfolge abgeleitet werden kann. Phonemfolgen, die durch die Phomenerkennung gewonnen wurden und nicht durch eine der Buchstabenfolgen generiert werden können, werden mit der Buchstabenfolge des Ranges eins assoziiert. Als Schreibweise für ein gelerntes Wort dient dann die assoziierte Buchstabenfolge der Aussprache, die beim Überprüfen des Lernerfolges in der besten Hypothese auftritt.

Diese Kandidatenliste von Aussprachen wird vom Spracherkenner benutzt, um zu überprüfen, ob das Wort erfolgreich gelernt werden kann, und welche der Kandidaten dann fest in den Spracherkenner übernommen werden, um das gelernte Wort zu repräsentieren. Dazu wird die Äußerung, die den Dialog ausgelöst hat, mit dem um die Kandidatenliste erweiterten Erkennervokabular erneut analysiert. An der verbesserten Hypothese des Spracherkenners wird festgestellt, dass das Lernen erfolgreich war, wenn ein Kandidat anstelle des generischen "<OOV:Location>" gefunden werden konnte. Der Spracherkenner signalisiert dem Dialogmanager, dass das Wort gelernt werden konnte und es wird das Wort

<NEW:Location.neks[vIli:@nzaUt.MAXIMILIANSAU>

für den zukünftigen Gebrauch in das Vokabular übernommen. Der Dialogmanager wechselt von Zustand S4 zurück in den initialen Zustand S0 und informiert den Benutzer

über das erfolgreiche Dialogende. Folgende verbesserte Hypothese könnte an eine (hier nicht existierende) nachfolgenden Komponente (NLP) weitergegeben werden:

Hypothese des Spracherkenners:
"I LIVE IN <NEW:Location.neks[vIli:@nzaUt.MAXIMILIANSAU>"

Interessant ist, dass bei diesem Beispiel vom Spracherkenner die Aussprache ausgewählt wurde, die von der Phonemerkennung auf Rang neun (Tabelle 9.4) gefunden wurde. Dabei ist erwähnenswert, dass die nach [20] korrekte englische Aussprache des Namens Maximilian /m ae k s i m i l e n/ ist und die korrekte deutsche Aussprache /m a k s i m i: l i a: n/ lautet. Wie man sehen kann, wurde hier vom Spracherkenner keine der Aussprachen ausgewählt, die mit einem "/m/" beginnt, was dadurch zu erklären ist, dass "/m/" und "/n/" -Laute vom Spracherkenner leicht verwechselt werden können, und sie somit bei der Auswahl kein so großes Gewicht haben. Bemerkenswerter ist, dass auf der "ungesehenen" ersten Äußerung diese Aussprachevariante wiedergefunden wird, weil dies ein Test auf die Generalisierungsfähigkeit der gefundenen Kandidaten ist.

9.4 Buchstabiererkenner und Phonemerkenner

Da für den Erfolg des Lernens eine gute Liste an Kandidaten für eine mögliche Aussprache notwendig ist, ist es wichtig, einen guten Phonemerkenner einzusetzen. Dieser hat aus den isolierten Wiederholungen des zu lernenden Wortes, die zu den später verwendeten akustischen Modellen passenden Phonemfolgen zu gewinnen. Deshalb werden in diesem Prototyp für die Erkennung der Phonemfolgen die gleichen kontextabhängigen akustischen Modelle eingesetzt wie für die Worterkennung. Dies bietet zusätzlich den Vorteil, dass es speicherschonender ist, da die Modelle nur einmal im Speicher gehalten werden müssen. Um die kontextabhängigen akustischen Modelle für die englische Phonemerkennung zu benutzen, wird ein Wörterbuch mit 6789 Multi-Phonem-Wörtern verwendet. Diese 6789 Wörterbucheinträge wurden dadurch gewonnen, indem die 4643 häufigsten (Maximum-Likelihood) Phonemkombinationen mit einer maximalen Länge von 5 Phonemen vereinigt wurden mit 5371 Phonemkombinationen, die durch ein iteratives Zusammenfügen gewonnen wurden, bei dem ein Maximum-Mutial-Information-Kriterium [9] optimiert wurde. Hier ist ebenfalls eine maximale Länge von 5 Phonemen benutzt worden. Die so gewonnenen Einträge erreichten eine Phonemfehlerrate von 32,8% auf dem BN-Testset. Dabei wurde ein 3-Gramm-Sprachmodell über dem verwendeten Phonemsatz benutzt.

Die Buchstabiererkennung sollte selbstverständlich ebenfalls beste Ergebnisse liefern, damit ein Nachschlagen in einem Hintergrundwörterbuch erfolgreich ist oder die Graphem-zu-Phonem-Umwandlung gute Ergebnisse liefert. Im Prototypen wird für die Buchstabiererkennung ebenfalls die gleiche Akustik wie für die Worterkennung benutzt. Für das deutsche System, das auf GSST-Daten trainiert ist, funktioniert dies gut, da die GSST-Daten künstliche Buchstabiersequenzen enthalten, bei denen versucht wurde, auch seltene Buch-

staben abzudecken. Dagegen enthalten die gefundenen Daten der BN-Domäne nur wenig Buchstabierungen, die im Allgemeinen Bestandteil von Namen von Personen oder Organisationen sind, wie beispielsweise "IBM" oder "CNN".

Anstelle eines HMM-basierten Buchstabiererkenners ist es beispielsweise auch möglich, einen nur auf Buchstabenfolgen diskriminativ trainierten MS-TDNN-Erkenner zu verwenden, wie er von Hild in [43] entwickelt wurde und wie er auch für den Verbmobil-Prototypen eingesetzt wird. Ein einzelner Buchstabe in einer kontinuierlich gesprochenen Buchstabensequenz wird mit einem MS-TDNN-Erkenner ohne Sprachmodell in etwa 90% der Fälle korrekt erkannt. Die Wahrscheinlichkeit alle Buchstaben, also beispielsweise einen ganzen Namen, richtig zu erkennen, ist geringer und liegt unter 60% (vgl. [43]).

Der in dieser Arbeit eingesetzte HMM-basierte Buchstabiererkenner führt zu gleich guten Erkennungsergebnissen und erlaubt zusätzlich das Arbeiten in einer homogenen Umgebung. So können beispielsweise auch die existierenden Adaptionsmethoden für die Buchstabier- oder Phonemerkennung eingesetzt werden.

Für die Umwandlung der Grapheme in Phoneme wird ein regelbasierter Ansatz benutzt, der auf eine Buchstabenfolge alle möglichen Regeln anwendet und daraus eine Menge an potentiellen Aussprachen erzeugt, die dann an der gegebenen Äußerung überprüft werden.

Es ist anzumerken, dass korrektes Buchstabieren eine für den Benutzer anstrengende Angelegenheit ist, insbesondere wenn die Zahl der Buchstaben zehn überschreitet. Nicht bei jeder Anwendung wird auch eine orthographische Form eines Wortes benötigt. Dies ist beispielsweise für ein Sprach-zu-Sprach-Übersetzungssystem nicht notwendig, es sei denn es soll beispielsweise auch ein Protokoll erzeugt werden. Wird also keine Repräsentation durch Buchstaben benötigt, kann bei einer solchen Anwendung im ersten Durchlauf auf die Buchstabierung verzichtet werden. Da aber im allgemeinen die Schreibweise eines Wortes nützliche Informationen enthält, sollte bei einem fehlgeschlagenen Lernversuch wieder auf eine zusätzliche Buchstabierung zurückgegriffen werden.

9.5 Lernen aus Fehlern

Für den Fall, dass die Überprüfung des Lernerfolges negativ ausgefallen ist, werden die Kandidaten, die es nicht in die N-Besten-Liste oder den geprunten Wortgraphen geschafft haben, in eine Negativliste für das Lernen dieses Wortes aufgenommen, so dass bei einem wiederholten Lernversuch diese von vornherein aus der Menge der Kandidaten entfernt werden können.

Im Falle des Fehlschlages lassen sich die zuvor gesammelten Daten mit den neu gewonnenen Informationen weiter sinnvoll kombinieren und abgleichen, indem beispielsweise die erneute Aufforderung zum Buchstabieren übersprungen werden kann, wenn das zweite Mal die gleiche Buchstabenfolge erkannt wurde. Bei mehr als einer isoliert gesprochenen Wiederholung des Wortes können die neuen Kandidatenphonemfolgen auch auf den vorher isolierten Äußerungen des Wortes bewertet werden, wodurch sich die Kandidatenliste noch weiter einschränken lässt und sich insbesondere Kandidaten beibehalten lassen, die im Mittel eine bessere Bewertung erhalten haben, was für das Erkennen bei einer Wie-

derholung positiv ist. Aus diesem Grund wird auch die Auswahl auf der "ungesehenen" ersten Äußerung durchgeführt, in der das zu lernende Wort auch noch in einem anderen akustischen Kontext vorkommt.

9.6 Zusammenfassung

In diesem Kapitel wurde ein Prototyp eines Spracherkenners vorgestellt, der von sich aus beim Auftreten eines generischen Wortes feststellen kann, dass ein neues Wort gelernt werden soll und dies mit Unterstützung des Benutzer durchführt. Dazu wird in einem Dialog mit dem Benutzer das benötigte Wissen erworben.

Der Aufbau des Prototypen mit seinen drei Komponenten Dialogmanager, Audioserver und Spracherkenner wurde erläutert. Der Zustandsgraph, die Beschreibung der Zustände des Zustandsautomaten, der Übergangsbedingungen und der ausgelösten Aktionen im Spracherkenner wurden detailliert beschrieben. Der für das Erlernen von Wörtern benutzte Dialogablauf zwischen Spracherkenner und Benutzer wurde in seiner Kernstruktur anhand eines konkreten Beispiels veranschaulicht.

Es wurde dargelegt, dass ein gelerntes Wort dabei nicht nur zur Verbesserung der aktuellen Hypothese führt, sondern auch bei zukünftigen Äußerungen zur Verfügung steht, ohne dabei nochmals gelernt werden zu müssen.

Im vorgestellten Prototypen wird auch versucht, möglichst viele der vom Benutzer zur Verfügung gestellten Informationen auszunutzen, insbesondere wenn das Lernen nicht auf Anhieb erfolgreich war. Dabei ist das Ziel, einerseits das Ergebnis des Lernens zu verbessern und andererseits, den Benutzer nicht mit wiederholten Eingabeaufforderungen unnötig zu strapazieren.

Es wurde erläutert, dass das Ergebnis des Lernens stark von der Qualität der Kandidaten und diese wiederum von den für die Phonemerkennung verwendeten Erkenner sowie der Kombination der Buchstabiererkennung und Graphem-zu-Phonem-Konvertierung, abhängt. Für das Lernen ist es notwendig, eine Phonemerkennung durchzuführen, die eine für die im Spracherkenner verwendeten akustischen Modelle passende Phonemfolge findet, weshalb für die Erzeugung der Phonemfolgen die gleichen kontextabhängigen akustischen Modelle eingesetzt werden, wie für die Worterkennung.

Auch das fehlerhafte Detektieren eines unbekannten Wortes kann zu einer Verbesserung der Hypothese führen, da durch die Rückmeldung des Benutzers die Detektion für unbekannte Wörter für die aktuelle Äußerung ausgeschaltet werden kann und dann eine neue Hypothese ohne generische Wörter erzeugt wird.

Es ist noch zu bemerken, dass die gelernten Wörter nicht immer dem Spracherkenner unbekannt sein müssen, so ist es beispielsweise auch möglich, dass ein bereits bekanntes Wort gelernt wird, weil es in einem für den Trainingskorpus ungewöhnlichen Kontext auftritt oder aber die Aussprache vom Benutzer nicht zu der im Spracherkenner modellierten Aussprache passt.

Kapitel 10

Zusammenfassung

Unbekannte Wörter stellen für Spracherkenner, die dem gegenwärtigen Stand der Technik entsprechen, nach wie vor ein schwieriges Problem dar. Zum einen führt ein unbekanntes Wort im Mittel zu 1.8 Fehlern bei der Erkennung, zum anderen handelt es sich häufig um Wörter, die wesentliche Informationen tragen, beispielsweise Eigennamen.

Da es aber unmöglich ist, beispielsweise alle aktuellen Personennamen der Welt zu kennen oder Wörter bzw. Namen vorauszusehen, die zum gegenwärtigen Zeitpunkt noch gar nicht existent sind, ist es nicht nur wünschenswert, sondern auch notwendig, dass ein Spracherkenner feststellen kann, ob ein unbekanntes Wort geäußert wurde und dann auch in der Lage ist, die notwendigen Schritte einzuleiten, um dieses Wort in sein Vokabular aufzunehmen.

Im Rahmen dieser Arbeit wurde ein Lösungsansatz entwickelt, der einem Spracherkennungssystems hilft, mögliche Erkennungsfehler durch unbekannte Wörter zu vermeiden und unbekannte Wörter zu detektieren. Erstmals wurde, ähnlich wie beim Menschen, ein kompletter Lösungsansatz bestehend aus der Detektion eines unbekannten Wortes, dem Erlernen des unbekannten Wortes durch Nachfragen sowie der Integration des gelernten Wortes in den Spracherkenner realisiert. Zu allen diesen Teilaufgaben wurden Untersuchungen durchgeführt und neue Methoden entwickelt, insbesondere:

- Detektion unbekannter Wörter:

  Entwicklung vokabular-optimierter Head-Tail-Modelle.

  Entwicklung einer neuen Methode zum Erstellen eines automatisch erzeugten Klassensystems zur Modellierung unbekannter Wörter.

- Erlernen unbekannter Wörter:

  Entwicklung und Realisierung eines Dialogs zum Erlernen von Kategorie, Aussprache und Schreibweise eines unbekannten Wortes.

- Integration unbekannter Wörter:

  Untersuchung einer neu entwickelten Methode zum Schätzen der Wahrscheinlichkeit eines neu gelernten Wortes für den wiederholten Gebrauch.

  Realisierung des ersten Dialogsystems, das auf eigene Initiative in Kooperation mit dem Benutzer ein unbekanntes Wort erlernen kann.

10.1 Die wichtigsten Ergebnisse und Beiträge

Die wichtigsten Ergebnisse und Beiträge sind hier noch einmal kurz skizziert. Eine detaillierte Zusammenfassung befindet sich jeweils am Ende der Kapitel 6, 7, 8 und 9.

Es wurden neuartige generischer Wörter (Head-Tail-Modelle) zur akustischen Modellierung unbekannter Wörter entwickelt und genauer untersucht. Die Head-Tail-Modelle besitzen den Vorteil, dass sie sich effizient und ohne Beeinträchtigung der mittleren Laufzeit in effiziente Suchstrukturen für LVCSR-Systeme einfügen lassen und eine präzisere Modellierung als flache akustische Modelle ermöglichen.

In Experimenten konnte mit den Head-Tail-Modelle auf der spontan-sprachlichen deutschen GSST-Domäne über die Hälfte (57%) der unbekannten Namen mit einer Präzision von 77% gefunden werden. Dabei wurden nur 780 verschiedene Head-Modelle benutzt und das Laufzeitverhalten wurde dabei im Mittel nicht beeinflusst.

Der Ansatz der Head-Tail-Modelle wurde zu den vokabular-optimierten Head-Modellen weiterentwickelt. Die optimierten Head-Tail-Modelle fügen sich noch effizienter als Vokal-Konsonanten-Folgen in bestehende Suchstrukturen ein und passen sich optimal an die Größe des verwendeten Vokabulars an.

Auf der sehr viel komplexeren englischen BN-Domäne konnten bei Verwendung der vokabular-optimierten Head-Modelle 18,5% der unbekannten Wörter mit einer Präzision von 50,4% detektiert werden. Ein Zufallsexperiment hätte eine Präzision von nur zirka 7% erreicht.

Ein neues Verfahren (GrOOV-Verfahren) wurde entwickelt, das automatisch ein geeignetes Klassensystem zur Modellierung unbekannter Wörter generiert. Das Verfahren ist insbesondere geeignet für sehr große und komplexe Domänen, wie beispielsweise die englischen BN-Domäne.

Mit dem GrOOV-Verfahren wird eine Einteilung der Menge von potentiell unbekannten Wörtern (pUNK-Wörter) in Klassen bestimmt, die sich zur Modellierung unbekannter Wörter eignet. Die so gefundenen Klassen führen bei vergleichbarem Recall zu einer höheren Präzision als bei Verwendung eines Einklassensystems.

Verglichen mit einem Sprachmodell mit nur einer einzigen Klasse für unbekannte Wörter konnten mit diesem neu entwickelten Verfahren 55% mehr unbekannte Wörter korrekt gefunden werden, bei einer vergleichbaren Präzision von ungefähr 80%.

Bei Verwendung eines hohen Bias (zur Erhöhung des Recalls) zeigt sich, dass der Einfluss des Sprachmodells gegenüber dem Bias keine allzu große Rolle mehr spielt und die in dieser Arbeit verglichenen Sprachmodelle ungefähr gleich gute Ergebnisse liefern.

Die Bedeutung eines geeigneten Sprachmodells für die Detektion unbekannter Wörter wird auch dadurch unterstrichen, dass das Sprachmodell bei dem die unbekannten Wörter nur durch die Back-off-Wahrscheinlichkeiten modelliert werden, erwartungsgemäß zu deutlich schlechteren Ergebnissen bei der Präzision führen.

Der Einfluss des Bias zeigte sich auch bei den durchgeführten Orakel-Experimenten. Ohne Anwendung des Bias sind Klassen für unbekannte Wörter bei der Erkennung nützlich, bei einem entsprechend hohen Bias macht die Verwendung von mehreren Klassen aber keinen Unterschied mehr. Die gelernten Wörter werden dann aber auch sehr häufig an unpassenden Stellen erkannt.

Dies zeigt, dass die Wahrscheinlichkeit für ein gelerntes Wort zwar höher geschätzt werden *kann*, aber nicht zu hoch geschätzt werden *darf*, da ansonsten das gelernte Wort zu häufig an falschen Stellen auftritt.

Um unbekannte Wörter zu erlernen, wurde ein Dialog zwischen Spracherkenner und Dialogpartner (Benutzer) entworfen und implementiert, der die für die Integration des Wortes benötigten Informationen wie Aussprache, Schreibweise und Klassenzugehörigkeit des unbekannten Wortes ermittelt. Die Integration des zu lernenden Wortes findet dabei während des Lernvorgangs statt, da der Erfolg der Integration sofort überprüft wird und somit den Lernvorgang mitbestimmt.

Wie auch aus den Orakel-Experimenten abgeleitet werden kann, ist ein wesentlicher Parameter, neben der Aussprache eines gelernten Wortes, die zu reservierende Wahrscheinlichkeit für sein wiederholtes Auftreten nach dem Lernen.

Es wurden verschiedene Verfahren untersucht und ein Simulationsverfahren entwickelt, das die Wahrscheinlichkeit für das Wiederauftreten eines gerade gelernten Wortes schätzt. Dabei zeigte sich, dass diese Wahrscheinlichkeit auf der untersuchten BN-Domäne im Vergleich zu einem beliebigen unbekannten Wort um einen Faktor von zirka 2000 höher zu schätzen ist.

Um unbekannte Wörter in einem Spracherkennungssystem detektieren *und* integrieren zu können, ist es notwendig, für beide Aufgaben einen Teil der Wahrscheinlichkeitsmasse des Sprachmodells zu verwenden. Es wurden verschiedene Verfahren beschrieben, die dies ermöglichen, indem die Verteilung der pUNK-Wörter in den Klassen untersucht wurde. Mit den Verfahren wurde bei Verwendung nur einer einzigen Klasse zirka 20% der Wahrscheinlichkeitsmasse für die Detektion verwendet und 80% für das Wiederfinden nach einem erfolgreichen Lernen.

Für die Sprachen Deutsch und Englisch wurden Prototypen realisiert, die mit den entwickelten Verfahren unbekannte Wörter nicht nur detektieren sondern auch erlernen.

Bei diesen Systemen handelt es sich meines Wissens nach, neben dem vom Autor im Rahmen des Verbmobil-Projektes realisierten Prototypen zum Erlernen unbekannter Wörter, um die ersten LVCSR-Spracherkennungssysteme, die, ähnlich wie beim Menschen, auf *eigene* Initiative in Zusammenarbeit mit dem Dialogpartner in der Lage sind, ihren Wortschatz zu vergrößern und somit von sich aus die Grenzen ihres Könnens zu erweitern.

10.2 Ausblick

Die in dieser Arbeit vorgestellten Methoden erlauben die Detektion unbekannter Wörter und das anschließende Lernen. Mit dem ebenfalls entwickelten Dialog kann das sprachverarbeitende System eigenständig seine Grenzen erweitern.

Um die Effektivität des Lernens insgesamt zu verbessern, ist eine Verbesserung aller beteiligten Komponenten sinnvoll. In Bezug auf ein zu lernendes Wort ist dies insbesondere der Phonemerkenner, der eine möglichst gute Liste an Erweiterungskandidaten erzeugen soll. Es ist zu erwarten, dass dies mit zukünftigen Verbesserungen der akustischen Modellierung zusammen mit einer allgemeinen Reduktion der Wortfehlerrate einhergeht. Dadurch ist auch zu erwarten, dass die Qualität der Detektion unbekannter Wörter verbessert wird, da weniger Fehler im allgemeinen zu weniger Stellen führt, an denen ein unbekanntes Wort fehlerhaft hypothetisiert wird. Je weniger Fehler das Baseline-System im allgemeinen macht, desto präziser werden die Hypothesen über das Auftreten unbekannter Wörter sein.

Aus den Untersuchungen geht hervor, dass erwartungsgemäß sowohl das Sprachmodell als auch die akustischen Modelle (Head-Tail-Modelle) einen Einfluss auf die Präzision des Ergebnisses haben. Dabei ist zu erwarten, je genauer ein Sprachmodell den möglichen Ort des Auftretens unbekannter Wörter vorhersagen kann, desto präziser das Ergebnis ausfallen wird. In Domänen, die sich beispielsweise durch Grammatiken beschreiben lassen, kann dieses Wissen explizit kodiert werden, wie beispielsweise von Asadi auf dem Ressource-Management-Task [1].

Bei der Verwendung der Head-Tail-Modelle in Grammatiken sind keine Schwierigkeiten zu erwarten. Eine interessante Frage ist aber, ob es möglich ist, automatisch aus der Grammatik weitere Optimierungsmöglichkeiten für die Head-Tail-Modelle zu gewinnen. Diese Frage stellt sich auch für die unüberwachten Klassen für unbekannte Wörter, bei der Klassen gefunden wurden, in der alle Wörter mit dem Suffix "-ing" oder "-s" enden. Anhand dieser Beobachtung kann man das Tail-Modell für diese Klassen speziell definieren, indem ein "-ing" oder "-s" bei den Phonemfolgen des Tail-Modells angehängt wird. Dies ermöglicht eine präzisiere Modellierung dieser speziellen Klassen.

Die in dieser Arbeit vorgestellten Head-Tail-Modelle führen zu sehr guten Ergebnissen;

trotzdem sind die Modelle zu verbessern. Dabei wird der akustischen Modellierung im Vergleich zur Sprachmodellierung ein größeres Potential zugeordnet. Eine logische Weiterentwicklung der Head-Tail-Modelle sind "Head-Middle-Tail-Modelle", bei denen sich die generischen Phoneme im Mittelteil befinden. Bei dieser Struktur wird Anfang und Ende des generischen Wortmodells präzise modelliert. Der nächste Schritt wäre dann die weitere Verallgemeinerung dieser Struktur.

Für die Komponenten, die dem Spracherkenner folgen, kann es nützlich sein, eine Relation für die unüberwacht gefundenen Sprachmodellklassen zu ihren semantischen Kategorien zu besitzen. Eine solche Relation kann mit genauer Analyse der einzelnen Klassen für einige der gefundenen Klassen manuell erstellt werden, aber um eine vollautomatische Verarbeitung für große Domänen zu erlauben, ist es von Vorteil, wenn es ein Verfahren gibt, dass diese Relation automatisch bestimmt.

Wenn die in einem sprachverarbeitenden System dem Spracherkenner folgenden Komponenten, wie beispielsweise der Dialogmanager, ebenfalls über Mechanismen verfügen, mit denen es möglich ist, dass sie die gelernten Wörter optimal für ihre Aufgaben verwenden können, so könnten gelernte Namen auch im Dialog eingesetzt werden. Eine vorstellbare Entwicklung wäre beispielsweise, dass einem Haushaltsroboter eine neuen Person vorgestellt wird: "Robbi, das ist Frau Brunner". Der Roboter stellt fest, dass er diesen Namen nicht kennt und lernt ihn. Das Dialogsystem kennt daraufhin ebenfalls diesen Namen und kann ihn nun verwenden. Beispielsweise könnte der Roboter auf die Aufforderung hin: "Wir haben Durst. Ich möchte ein Glas Wasser." nachfragen: "Und was möchte Frau Brunner trinken?" Eine Verwendung von gelernten Wörtern könnte somit die Natürlichkeit des Dialoges steigern.

Wünschenswert ist auch, dass der starre Dialogablauf, der bisher durch den endlichen Automaten realisiert ist, in einem zukünftigen System durch einen benutzerfreundlicheren, integrierten Dialog mit einem flexiblen Dialogmanager abgelöst wird, der dem Anwender mehr Freiheiten bei der Spracheingabe lässt. Daraus folgt, dass nun vor den Eingaben nicht mehr bekannt ist, ob eine Phonemerkennung oder Worterkennung benötigt wird. Bei freien Dialogen ist auch damit zu rechnen, dass der Anwender seine Antwort in eine Phrase einbettet. Beispielsweise bei Nennung des Namens "Brunner" nun antwortet "Brunner wie Brunnen nur mit einem R am Ende." Es ist in diesem Falle notwendig, den korrespondierenden Teil, der gelernt werden soll, zu identifizieren und daraus die benötigten Informationen für das Lernen zu extrahieren.

Anhang A

Statistische Eigenschaften der unüberwacht gefundenen Klassen für unbekannte Wörter

Klasse	OOV-Wörter	Wörter	OOV-Rate	a	b
0	600	1169	15.51	647.2	-1.134
1	115	390	5.90	27.6	-0.476
2	103	323	6.85	22.0	-0.433
3	57	173	7.09	25.1	-0.549
4	173	333	14.86	139.1	-0.964
5	169	329	13.51	75.6	-0.779
6	383	475	41.92	1422.2	-1.558
7	300	447	24.01	420.7	-1.234
8	62	128	0.22	13.9	-0.391
9	136	185	29.08	211.0	-1.208
10	90	117	36.88	50.9	-0.819
11	50	112	1.21	-	-

Tabelle A.1: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (1)

Klasse	OOV-Wörter	Wörter	OOV-Rate	a	b
12	103	172	13.80	49.6	-0.785
13	144	236	7.74	217.8	-1.218
14	244	322	24.09	367.2	-1.242
15	138	209	13.19	146.4	-1.081
16	50	72	1.46	-	-
17	143	250	12.02	83.1	-0.861
18	51	108	1.11	-	-
19	105	167	15.92	90.0	-0.965
20	50	103	1.74	-	-
21	130	183	22.59	220.5	-1.236
22	132	232	12.00	63.3	-0.776
23	54	90	1.94	19.6	-0.545
24	73	159	2.30	11.8	-0.372
25	58	133	2.40	7.0	-0.184
26	50	93	2.58	8.3	-0.319
27	67	150	9.68	19.6	-0.481
28	87	134	3.55	16.0	-0.412
29	62	126	8.36	9.0	-0.261
30	82	150	7.25	68.0	-0.910
31	70	101	10.10	35.9	-0.758
32	60	120	3.98	11.0	-0.383
33	52	104	9.30	19.8	-0.525
34	64	112	10.02	39.8	-0.770
35	316	429	21.86	413.1	-1.170
36	122	153	23.16	230.5	-1.257
37	73	99	14.55	11.2	-0.372
38	152	179	21.34	1157.8	-1.794
39	88	129	11.85	73.6	-0.976
40	91	178	4.69	12.2	-0.320
41	220	284	29.58	207.3	-1.029

Tabelle A.2: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (2)

Klasse	OOV-Wörter	Wörter	OOV-Rate	a	b
42	50	126	2.09	8.4	-0.269
43	97	199	4.69	29.8	-0.596
44	155	253	12.70	69.0	-0.801
45	50	92	2.52	5.0	-0.123
46	78	119	10.73	231.4	-1.383
47	90	142	4.11	17.9	-0.428
48	90	146	9.18	152.4	-1.189
49	50	137	2.52	-	-
50	137	150	33.07	85.2	-0.883
51	77	127	6.12	55.8	-0.842
52	58	89	3.75	22.9	-0.565
53	64	104	4.96	113.6	-1.197
54	109	186	6.15	185.2	-1.205
55	58	87	18.18	-	-
56	50	109	2.74	12.7	-0.429
57	122	176	9.00	37.7	-0.566
58	136	204	10.45	233.2	-1.214
59	50	72	2.12	-	-
60	50	87	8.27	6.9	-0.207
61	187	210	13.65	359.3	-1.301
62	65	97	8.39	34.2	-0.718
63	53	101	5.37	45.5	-0.843
64	60	94	3.80	26.8	-0.709
65	53	81	7.21	-	-
66	51	61	24.07	-	-
67	54	86	7.67	8.3	-0.284
68	50	83	8.14	-	-
69	58	89	7.12	-	-
70	65	93	3.51	31.2	-0.673
71	50	79	5.74	-	-

Tabelle A.3: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (3)

Klasse	Die fünf häufigsten Vokabular-Wörter der Klassen
0	ROSEANNE CATERPILLAR SEQUESTRATION SUNLIGHT DASH
1	SHELL DUCK OUTRIGHT DODGE SUBSTITUTING
2	SMILES MERCURY OFFSHORE SKIING RATIONING
3	BUCK WELLS YUP GAVEL DRILLING
4	NIELSEN DOYLE MALONE DEUTSCH ROSENTHAL
5	NIGGER BARBIE DEADLINES LO KHRUSHCHEV
6	STALIN CHERNOMYRDIN DURAN NORPLANT WHITEHURST
7	JORDAN'S CASTRO'S HARDING'S PAT'S ALLEN'S
8	PEOPLE HAITIANS CUBANS SOMALIS CALLERS
9	FORBES' KORESH'S MICHAEL'S NEWT'S HALL'S
10	MCVEIGH'S DAVID'S KIM'S ANOTHER'S NIGHTLINE'S
11	WOMEN LESBIANS MAYORS FEMALES COMMUTERS
12	HURTS TREATS VIOLATES REDUCES ELIMINATES
13	FRANKEN SIEGEL NOVAK SHAW FISKE
14	KRISTIN LAURIE MICHEL MARIE JOSH
15	LIAR MISTRIAL BACKLASH DIME BELIEVER
16	REPUBLICANS PLAINTIFFS IRANIANS PAKISTANIS SUBWAYS
17	BABBITT LINDSEY MCLARTY O'NEILL SANDERS
18	BOSNIA HAITI SOMALIA RWANDA CHECHNYA
19	MOHAMMED MOHAMED HERMAN MUHAMMAD AZIZ
20	TRUE RIDICULOUS UNTRUE LUDICROUS PREPOSTEROUS
21	KOBE NEWARK BRENTWOOD CAMBRIDGE HARTFORD
22	BURNS SAWYER OSWALD O'GRADY PATTERSON
23	DEMOCRATS CONSERVATIVES LIBERALS COWBOYS PROS
24	VIOLENCE TERROR CHAOS AGGRESSION PREJUDICE
25	REST PROSPECT MIDST AFTERMATH VERGE
26	GIVING LETTING DENYING INTRODUCING DIVIDING
27	DOWNS BALANCES ARROWS STRIPES CONS
28	PROSECUTION CLINTONS VATICAN PHILIPPINES SOVIETS
29	TUBERCULOSIS DIABETES ASTHMA CHOLERA ALCOHOLISM
30	BAKER MORET MCDOUGAL CLANCY CHILES
31	JESUS HONEY DADDY SHAKESPEARE MADONNA
32	BOOKS ARTICLES PILLS JEWELRY BUTTONS
33	JULIA ERIN JEANNETTE JEANETTE WHOOPI
34	WALTERS MONROE STREISAND BRYANT ELLIOT
35	MORTON EDWARDS WALSH LEVIN MATTHEWS

Tabelle A.4: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (1)

Klasse	Die fünf häufigsten Vokabular-Wörter der Klassen
36	PARTY'S ECONOMY'S BILL'S SENATOR'S PROGRAM'S
37	PORTLAND CHARLOTTE HOMESTEAD LOUISVILLE ROCHESTER
38	KEVORKIAN LAKSHMANAN GERDES BLAKE BADEN
39	PANETTA SESNO SPENCE UELMEN SINATRA
40	CAREFUL CAUTIOUS CYNICAL SEQUESTERED BORING
41	WILSON'S JOHNSON'S PACKWOOD'S NORTH'S SHAPIRO'S
42	SMITH DAVIS THOMPSON ROBERTS CHAPMAN
43	HARRIS GOLDBERG LEVINE COLLINS COHEN
44	MCCAIN RESNICK KANTOR KASICH MANTLE
45	FREEDOM DESTRUCTION SILENCE INNOCENCE SEPARATION
46	ROBB HUFFINGTON WILDER DUVALL THATCHER
47	SARAJEVO MOGADISHU BAGHDAD GORAZDE TUZLA
48	CUOMO FOLEY GARCETTI GIULIANI DASCHLE
49	BEAUTIFUL BRILLIANT NASTY ROMANTIC PRECIOUS
50	LIANE LAURIN HOWIE BEV BETTINA
51	KEMP BROOKS KINSLEY TYSON BENTSEN
52	CLARK MAZZOLA HOPKINS BRADLEY MCKINNY
53	NIXON GORBACHEV EISENHOWER LUGAR HOOVER
54	LEWIS SCHNEIDER BENNETT ASPIN GREENSPAN
55	WALDHOLTZ FAY PIPER FI CHANDLER
56	UNUSUAL UNKNOWN OUTRAGEOUS UGLY IMPRESSIVE
57	RENO KORESH STARR KERRIGAN ZHIRINOVSKY
58	MORRIS JENNINGS TSONGAS NEUFELD VERCAMMEN
59	FUHRMAN CASTRO VANNATTER LOPEZ LANGE
60	REYNOLDS HOFFMAN KOON GILLOOLY PACKARD
61	SHAPIRO DARDEN SIMS RUBIN MATHESON
62	MICROSOFT PARAMOUNT INTEL VIACOM BLOCKBUSTER
63	BAILEY FISHER BEGLEITER MYERS DONALDSON
64	WILLIAMS MILLER TAYLOR TURNER ANDERSON
65	GORE COWLINGS HINMAN PACINO O'DONNELL
66	POWELL'S SMITH'S GRAMM'S MANDELA'S GREENSPAN'S
67	BIERBAUER SHIPMAN FELDMAN BAIRD JACO
68	O'CONNOR GARSTEN ROLLINS HILLARD BERRY
69	DOUGHERTY MCDERMOTT WOODRUFF BROOKE MULLER
70	FORBES GRAMM QUAYLE D'AMATO ROSTENKOWSKI
71	MAN'S WOMAN'S PERSON'S CHILD'S GUY'S

Tabelle A.5: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (2)

Klasse	Die fünf häufigsten OOV-Wörter der Klassen
0	BLANCO HOBBES RHYMING TRISTAN DVORAK
1	AIMLESS REALPOLITIK WATERCOLORS CRIM FUNNELS
2	CROSSTALK MEDITATING KNOTTED LOBSTERING NAMECALLING
3	DOCKSIDE SER TRUJILLO LASALLE FELLER
4	GIRTH LAGER SOMERVILLE FONTANA BOSLEY
5	PCH REPARATION DEADBEATS GOOSEBUMPS INTERPOL
6	AVRAHAM JELKS CARRANZA ACCION AILICH
7	CARR'S DINKINS' ELVIS'S HOFFA'S LOZANO'S
8	TENNESSEANS BERLINERS ANGOLANS ENTERPRENEURS PROCRASTINATORS
9	DAWN'S LUCY'S GLORIA'S DEBORAH'S KATIE'S
10	DOUG'S HOWIE'S BOSKO'S TRACY'S EINSTEIN'S
11	COMMONERS FIREFLIES DIOXINS HOBOES DISCOS
12	UNDID SHORTENS ISOLATES NEGLECTS NEGATES
13	WIKOWSKY MAPPLETHORPE CRINGELY BRADEN REDFIELD
14	GINO NICKI HEINRICH MYRIAM NANA
15	CRYBABY CROSSWALK HERETIC WEREWOLF BUMMER
16	STEWARDESSES LITHUANIANS DILLEYS MULLAHS HOPIS
17	KRANTZ HOGUE NOVELLO HASENFUS SCHWARZENEGGER'S
18	GRADACAC TABA GARAZDE LUSAKA CINDERS
19	SAMIR XU AUGUSTO DAG MAJID
20	DISPOSITIVE EXPLAINABLE ANDORRA EXCUSABLE DECOMMISSIONED
21	VICKSBURG DANBURY HATTIESBURG MONTEZUMA NOTTINGHAM
22	HOUSER CATT HARWOOD BRUCKHEIMER SAENZ
23	FEDERALISTS MERCHANDISERS MOROCCANS EXPERIMENTERS RAINDROPS
24	RUTHLESSNESS BELLIGERENCY INVISIBILITY THUGGERY VITRIOL
25	RIGHTNESS VORACITY ALLEVIATION REEMERGENCE COARSENING
26	CLASSIFYING PREJUDGING IMPRISONING WHACKING OVERRULING
27	BUGGIES SCALLOPS OVERWORK PILLAGING CRANNIES
28	FONZ MALDIVES AZTECS MOSSAD GIFFORDS
29	GLAUCOMA PHARMACIA PERTUSSIS DCIS JAUNDICE
30	BARSHEFSKY BACHUS THORPE BILBRAY WYLER
31	GUNSMOKE SHEBA CONFUCIUS MOMMIE RAMTHA
32	MASCARA VASES AUTOBIOGRAPHIES FITTINGS CHEESEBURGERS
33	AMITAI GILBERTO FELICE HARLAND KRAVIS
34	ERIKSON MILNER PERL SAWYER'S CLACK
35	CARNES DELLINGER EHRLICH MCKERNAN SIEGAL

Tabelle A.6: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (1)

Klasse	Die fünf häufigsten OOV-Wörter der Klassen
36	CRIME'S INFORMATION'S PIANO'S RALLY'S COMPETITION'S
37	COMPTON'S ELGIN CALUMET KEITHSBURG RANTOUL
38	DESILVA TENNANT BRAZELTON HIMMELSTEIN TARNOWER
39	CARPIO CHALEFF LEDERMAN DUNKEL ELDAR
40	ANTICIPATORY BANKABLE DISBELIEVING FORGETFUL PRESENTABLE
41	BERRY'S GRIFFITH'S NICHOLSON'S RUBY'S SCHROEDER'S
42	LASSITER CANO NAVA PADILLA FREDRICK
43	BLUMBERG FABER CRUTCHFIELD FILIPOVIC KRAUSE
44	EHRLICHMAN ALSOP DILLINGER TRACHTENBERG CULLERTON
45	DISINFLATION STEADINESS PROBITY ZEALOTRY CORDIALITY
46	REIGLE ROLLO WANTZ MCKENNON ESLER
47	VOGOSCA BUTARE GONAIVES SKOPJE MATSUMOTO
48	HENN GARTEN HERSHISER GILLERS BERENGER
49	DOUR BLISSFUL TAUT BEEFY BOMBASTIC
50	CHAZ ELAINA VERNA EVERETTE FREDA
51	MASTROIANNI D'AMICO INGRAHAM STATMAN HEMMINGS
52	RASKIN UPSHAW TIPTON SAPIENS FALCO
53	HOSAKAWA SHIMOMURA GAVIRIA WODEHOUSE DICKER
54	GRANGER KOENIG MILLIKEN CASALS GLICKENHAUS
55	NICKERSON HITCHCOCK'S ALBEE ANTONINI BOUSQUET
56	UNGODLY INDETERMINATE UNSELFISH ANDROGYNOUS UNDEFENDED
57	BONIER KOSYREV DEMPSTER TROSCH ZUNIGA
58	EYAL GAMMONS BRAVERMAN COVERDALE KANJORSKI
59	NEUMANN GALANTER HULTGREEN MULDORFER KADISH
60	BEECHAM ACHESON LUCCA TILDEN KEMLER
61	ABADI BLAISER MABON JEWITT DELAND
62	CIGNA SAMSUNG MEDRAD BULLOCK'S GEMSTAR
63	BROWNBACK LANSKY ROCAR CALLACE KHALILI
64	HELVEY KISSLING BEGLEY HELLMAN OLDHAM
65	ELAN SARANT SHIMBUN DOGG ANON
66	DINGELL'S SANTORUM'S REINER'S HATCH'S DECONCINI'S
67	GIRARD PELLEGRINO HODSON ZOHAR KURALT
68	DABBA FOSLER CONVERTERS LABELLE LUPONE
69	BRUBECK HENNEN FORTON EDMONDSON TEICHOLZ
70	KALUI LAMBORN AKROYD GOODGAME GUTTENBERG
71	BUSINESSMAN'S MURDERER'S ELEPHANT'S COP'S GUEST'S

Tabelle A.7: Beschreibung der Klassen zur Modellierung von OOV-Wörtern (2)

Anhang B

Pseudo-Code für das simulierte Lernen

Der folgende Pseudo-Code beschreibt das Lernen, ohne Langzeit-Gedächtnis (SIM-1).

```
V+ := {} ;
pos = 0;

while not end of (text) {
  pos := pos + 1;
  w := get_next_word(text);

  if w in V {
    if w notin V+  {
       - Aufnamhme in das Kurzzeitgedächtnis
       c := bestimme_Klasse(w) ;
       V+ := V+ vereinigt mit { w } ;
       memDur[w] = N ;
    }
    checkPos[w] := pos + memDur[w] ;
  }

  foreach e aus V+ {
    if pos >= checkPos[e] {
       - Wort kann vergessen werden
       V+ := V+ minus { e } ;
    }
  }
}
```

Der folgende Pseudo-Code beschreibt das Lernen mit Langzeitgedächtnis (SIM-2).

```
V-  := {} ;
V+  := {} ;
pos := 0 ;

while not end of (text) {
  pos := pos + 1;
  w := get_next_word(text);

  if w in V {
    if w notin V+  {
       - Aufnamhme in das Kurzzeitgedächtnis
       c := bestimme_Klasse(w) ;
       V+ := V+ vereinigt mit { w } ;
       memDur[w] = N ;
    } else {
       if w in V- {
          - Aufnamhme in das Langzeitgedächtnis
          c := find_class($w) ;
          V+ := $V+ vereinigt mit { w } ;
          memDur[w] := pos - (checkPos[w] - memDur[w]) ;
       }
    }
    checkPos[w] := pos + memDur[w] ;
  }

  foreach e aus V+ {
    if pos >= checkPos[e] {
       - Wort kann vergessen werden
       V+ := V+ minus { e } ;
       V- := V- vereinigt mit { w } ;
    }
  }
}
```

Anhang C

Korrelationen von Konfidenzmerkmalen zu OOV-Fehlern

Tabelle C.1 fasst die gefundenen Korrelationen aus Arbeit [68] zusammen, in der alle Merkmale ausführlich beschrieben sind. Die Reihenfolge ist nach absteigender Korrelation zum Gesamtfehler geordnet. Die Untersuchung wurde auf Daten der ersten Phase von Verbmobil durchgeführt. Wie man sehen kann, sind nur wenige, redundante Merkmale mit OOV-Fehlern stärker korreliert als mit den Gesamtfehlern. Im Wesentlichen handelt es sich dabei um Abweichungen von der erwarteten Länge eines Wortes (WStreck3, WStreck2, WStreck1, WStauch1, WStauch2 und WStauch3) und darum, ob ein Wort, das hypothetisiert wurde, häufig, beziehungsweise selten, im Training aufgetreten ist. Diese Merkmale sind aber nicht sehr stark mit dem Auftreten von OOV-Fehlern korreliert. Die aufgeführten Merkmale gehören inzwischen zur Standardmenge an Merkmalen, die für die Bestimmung von Konfidenzen herangezogen wird.

	Gesamtfehler	OOV-Fehler
AStabil	-54,0 %	-18,1 %
H1	47,9 %	16,0 %
H2	46,9 %	16,5 %
H3	35,5 %	19,5 %
SM-NGRAM	-19,6 %	-14,3 %
AMWScore	13,5 %	5,9 %
PMWScore	13,1 %	3,9 %
MWScore	12,4 %	2,6 %
LogPhonAnz	-11,8 %	-1,0 %
EMWScore	10,7 %	11,8 %
LogAnzATrain	-10,7 %	-16,2 %
PhonAnz	-10,3 %	-0,5 %
WStreck3	6,6 %	11,5 %
WStreck2	5,0 %	10,1 %
WStreck1	3,5 %	9,4 %
WStauch1	-2,4 %	-7,9 %
WStauch2	-2,4 %	-8,7 %
WSchw	2,3 %	1,0 %
WStauch3	-1,3 %	-8,2 %
KSchw	-0,5 %	1,2 %
AnzATrain	-0,4 %	-0,8 %

Tabelle C.1: Zusammenfassung der Korrelationen

Literaturverzeichnis

[1] A. Asadi, R. Schwartz and J. Makhoul (1990). *Automatic detection of new words in a large-vocabulary continuous speech recognition system*. In: Proc. ICASSP '90, S. 125 ff., IEEE Signal Processing Society, Albuquerque, New Mexico, USA. April 1990.

[2] G. Bamberg and F. Baur (1989). *Statistik*. R. Oldenbourg Verlag, ISBN 3-486-21325-3. Munich/Vienna 1989.

[3] L. Baum, T. Petrie, G. Soules and N. Weiss (1970). *A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains*. In: The Annals of Mathematical Statistics, Vol 41, No 1, pp 164-171. 1970.

[4] L. Baum and T. Petrie (1966). *Statistical inference for probabilistic functions of finite state markov chains*. In: The Annals of Mathematical Statistics , Vol 37, No 6, pp 1554-1563. 1966.

[5] I. Bazzi and J. Glass (2001). *Learning units for domain-independent out-of-vocabulary word modelling*. In: Proc. EUROSPEECH, European Conference Speech Communication and Technology, pp. 61-64, Aalborg, Dänemark 2001.

[6] I. Bazzi and J. Glass (2000). *Modeling out-of-vocabulary words for robust speech recognition*. In: Proc. ICSLP '00, Beijing, China, Oktober 2000.

[7] D. M. Bikel, S. Miller, R. Schwartz and R. Weischedel (1997). *Nymble: a high-performance learning name-finder*. In: Proceedings of the Fifth Conference on Applied Natural Language Processing , Association for Computational Linguistics, pp. 194-201.31, Washington, D.C., USA, März 1997.

[8] G. Boulianne and P. Dumouchel (2001). *Out-of-vocabulary word modeling using multiple lexical fillers*. Proc. IEEE Workshop on Automatic Speech Recognition and Understanding, Madonna di Campiglio Trento, Italien, Dezember 2001.

[9] P. F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai and R.L. Mercer (1992). *Class-based n-gram models of natural language*. Computational Linguistics Vol. 18(4) S. 467 ff. 1992.

[10] L. Chase (1997). *Error-responsive feedback mechanisms for speech recognizers*. Ph. D. Thesis, Carnegie Mellon University, The Robotics Institute, Pittsburgh, Pennsylvania, April 1997.

[11] N. Chinchor, P. Robinson and E. Brown (1999). *The Hub-4 Named Entity Task Definition, Version 4.8*. In: Proceedings of DARPA: "Broadcast News Workshop", http://www.nist.gov/speech/tests/bnr/hub4_98/h4_iene_task_def.4.8.ps, 1999.

[12] P. Clarkson and T. Robinson (1998). *The applicability of adeptive language modelling for the Broadcast News Task*. Proc. ICSLP '98. Sydney, Australia, December 1998.

[13] G. Chung (2001). *Towards multi-domain speech understanding with flexible and dynamic vocabulary*. Ph. D. Thesis , Massachusetts Institute of Technology, June 2001.

[14] G. Chung (2000). *A three-stage solution for flexible vocabulary speech understanding*. In: Proc. ICSLP '00, Beijing, China, Oktober 2000.

[15] G. Chung (2000). *Automatically incorporating unknown words in Jupiter*. In: Proc. ICSLP '00, Beijing, China, Oktober 2000.

[16] T. M. Cover and J. A. Thomas (1938). *Elements of information theory*. John Wiley and Sons, ISBN 0-471-06259-6, New York, 1938.

[17] B. Decadt, J. Duchateau, W. Daelemans and P. Wambacq (2002). *Transcription of out-of-vocabulary words in large vocabulary speech recognition based on phonem-to-graphem conversion*. In Proc. ICASSP '02, 2002.

[18] B. Decadt, J. Duchateau, W. Daelemans and P. Wambacq (2001). *Phoneme-to-grapheme conversion for out-of-vocabulary words in large vocabulary speech recognition*. Proc. IEEE Workshop on Automatic Speech Recognition and Understanding. Madonna di Campiglio, Trento, Italy, Dezember 9-13 2001.

[19] M. Denecke (2002). *Generische Interaktionsmuster für aufgabenorientierte Dialogsysteme*. Ph. D. Thesis , Universität Karlsruhe, 2002.

[20] *Duden Aussprachewörterbuch*, 4. neu bearbeitete und aktualisierte Auflage, ISBN 3-411-04064-5. Mannheim 2000.

[21] *Duden Die deutsche Rechtschreibung*, 22. völlig neu bearbeitete und erweiterte Auflage, ISBN 3-411-04012-2. Mannheim 2000.

[22] P. Fetter (1998). *Detection and transcription of OOV words*. Dissertation, Verbmobil-Report 231, ISSN 0949-6048, 1998.

[23] P. Fetter, F. Class, U. Haiber, A. Kaltenmeier, U. Kilian and P. Regel-Brietzmann (1995). *Detection of unknown words in spontaneous speech*. In: Proc. EUROSPEECH '95, vol. 3, S. 1637 ff, Madrid, September 1995.

[24] M. Finke, P. Geutner, H. Hild, T. Kemp, K. Ries and M. Westphal (1997). *The Karlsruhe-Verbmobil Speech Recognition Engine*. In: Proc. ICASSP '97, Vol 1, S. 83 ff, München, April 1997.

[25] M. Finke and I. Rogina (1997). *Wide Context Acoustic Modelling in Read vs. Spontaneous Speech*. In: Proc. ICASSP '97, Vol 1, S. 83 ff, München, April 1997.

[26] W. M. Fisher (1999). *A statistical Text-To-Phone function using Ngrams and rules*. In: Proc. IEEE International conference on acoustics, speech and signal processing (ICASSP '99), Phoenix, AZ, USA, 15.-19. März 1999.

[27] J. Fritsch and I. Rogina (1996). *The bucket box intersection (BBI) algorithm for a fast aproximative evaluation of diagonal mixture gaussians*. In: Proc. ICASSP '96. Atlanta, Georgia, Mai 1996.

[28] G. D. Forney (1973). *The Viterbi algorithm*. In: Proc. IEEE, vol 61, S. 268-278. März 1973.

[29] K. Fukunaga (1990). *Introduction to statistical pattern recognition*. Academic Press Inc., San Diego, CA 92101, ISBN 0-12-269851-7, San Diego, 1990.

[30] M. J. F. Gales (1997). *Semi-tied full-covariance matrices for hidden markov models*. Cambridge University Technical Report TR 287, CUED/F-INFENG/TR 287. Cambridge, England, April 1997.

[31] F. Gallwitz (2001). *Integrated stochastic models for spontaneous speech recognition*. Dissertation, Studien zur Mustererkennung , Bd. 6 , Logos Verlag Berlin, ISBN 3-89722-907-2. Erscheinungsjahr 2002.

[32] F. Gallwitz, E. Nöth and H. Niemann (1997). *Recognition of out-of-vocabulary words and their semantic category*. In: Proceedings of the 2nd SQEL Workshop on Multi-Lingual Information Retrieval Dialogs, S. 114 ff. Pilsen, Tschechien, April 1997.

[33] F. Gallwitz, E. Nöth and H. Niemann (1996). *A category based approach for recognition of out-of-vocabulary words*. In: Proc. ICSLP '96, Vol 1, S. 228 ff, Philadelphia, 1996.

[34] J. Garofolo, J. Fiscus and W. Fisher (1997). *Design and preparation of the 1996 Hub-4 broadcast news benchmark corpora*. In: Proceedings of the DARPA Speech Recognition Workshop held in the Westfields Conference Center. Chantilly, VA, Februar 1997.

[35] P. Geutner (1998). *Adaptive vocabularies in large vocabulary speech recognition*. Ph. D. Thesis, University of Karlsruhe, Germany, Februar 1999.

[36] D. Gibbon and H. Lüngen (2000). *Speech lexica and consistent multilingual vocabularies*. In: Verbmobil: Foundations of Speech-to-Speech Translation, Springer Verlag, ISBN 3-540-67783-6, Berlin, 2000.

[37] I. J. Good (1953). *The population frequencies of species and the estimation of population parameters*. In: Biometrika, vol. 40, S. 237-264. Dezember 1953.

[38] Y. Gotoh, S. Renals and G. Williams (1999). *Named entity tagged language models*. In: Proceedings of the IEEE conference for acoustics, speech and signal processing (ICASSP '99), Phoenix, AZ, USA, 15.-19. März 1999.

[39] D. Graff (2002). *An overview of Broadcast News Corpora*. In: Speech Communication, Volume 37, May 2002, Pages 15-26.

[40] R. Haeb-Umbach, P. Beyerlein and E. Thelen (1995). *Automatic transcription of unknown words in a speech recognition system*. In: Proc. ICASSP '95, S. 840 ff., Madrid, Spanien, September 1995.

[41] T. J. Hazen and I. Bazzi (2001). *A comparison and combination of methodes for OOV word detection and word confidence scoring*. In: Proceedings of International Conference on Acoustics, Speech and Signal (ICASSP '01), Salt Lake City, UT, USA, Mai 2001.

[42] I. L. Hetherington (1995). *A Characterization of the problem of new, out-of-vocabulary words in continuous speech recognition and understanding*. Ph. D. Thesis, MIT, 1995.

[43] H. Hild (1997). *Buchstabiererkennung mit neuronalen Netzen in Auskunftssystemen*. Dissertation, Universität Karlsruhe, Fakultät für Informatik, Shaker Verlag, Aachen, ISBN 3-8265-3155-8, Karlsruhe, 1997.

[44] Fei Huang and A. Waibel (2001). *An adaptive approach to named entity extraction für meeting applications*. 2001.

[45] F. Jelinek (1998). *Statistical methods for speech recognition*. MIT Press, Cambridge, MA, ISBN 0-262-10066-5, 1998.

[46] F. Jelinek (1990). *Self-organized language modeling for speech recognition*. In: A. Waibel and K.-F. Lee: "Readings in Speech Recognition", Morgan Kaufmann Publishers, Inc., San Mateo, USA, 1990.

[47] F. Jelinek, R. Mercer and S. Roukos (1990). *Classifying words for improved statistical language models*. Proc. ICASSP '90, S. 621 ff., IEEE Signal Processing Society, Albuquerque, New Mexico, USA, April 1990.

[48] A. Jusek, G. A. Fink, F. Kummert and G. Sagerer (1996). *Automatically generated models for unknown words*. In: Proceedings of the Sixth Australian International Conference on Speech Science and Technology, S. 301–306, Adelaide, South Australia, 1996.

[49] A. Jusek, G. A. Fink, F. Kummert, H. Rautenstrauch and G. Sagerer (1995). *Detection of unknown words and its evaluation*. In: Proc. EUROSPEECH '95, vol. 3, S. 2107 ff, Madrid, Spanien, September 1995.

[50] A. Jusek, H. Rautenstrauch, G. A. Fink, F. Kummert, G. Sagerer, J. Carson-Berndsen and D. Gibbon (1994). *Detektion unbekannter Wörter mit Hilfe phonotaktischer Modelle*. In: W.G. Kropatsch and H. Bischof, editors, Mustererkennung 94, 16. DAGM-Symposium und 18. Workshop der ÖAGM Wien, S. 238-245. Springer-Verlag, 1994.

[51] T.Kemp and A.Jusek (1996). *Modelling unknown words in spontaneous speech*. In: Proc. ICASSP '96, pp 530 ff. ISBN 0-7803-3192-3. Atlanta, USA, Mai 1996.

[52] D. Klakow, G. Rose and X. Aubert (1999). *OOV-Detection in large vocabulary systems using automatically defined word-fragments as fillers*. In: Proc. EUROSPEECH '99, Budapest, Ungarn, September 1999.

[53] R. Kneser and H. Ney (1993). *Improved clustering techniques for class-based statistical language modelling*. In: Proc. EUROSPEECH '93, S. 973 ff., ESCA, Berlin, September 1993.

[54] R. Kneser and H. Ney (1995). *Improved backing-off for m-gram language modeling*. In: Proc. ICASSP '95, vol 1, pp 181 ff, Detroit, USA, Mai 1995.

[55] F. Kubala, R. Schwartz, R. Stone and R. Weischedel (1998). *Named entitiy extraction from speech*. In: Proceedings of DARPA Broadcast News Transcription and Understanding Workshop. Lansdowne, VA, Februar 1998.

[56] D. Miller, R. Schwartz, R. Weischedel and R. Stone (1999). *Named entity extraction from broadcast news*. In: Proceedings of the DARPA Broadcast News Workshop, pages 37–40, Herndon, Virginia, USA, 1999.

[57] Summary of the NIST Workshop on confidence measures, NIST, Januar 1996.

[58] L. R. Rabiner (1989). *A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*. In: Proceedings of the IEEE, 77(2):257-285, Februar 1989.

[59] L. R. Rabiner and S. E. Levison (1981). *Isolated and connected word recognition-theory and selected applications*. In: A. Waibel, K.-F. Lee: "Readings in Speech Recognition". Morgan Kaufmann Publishers, Inc., San Mateo, USA, 1990

[60] L. R. Rabiner and R. W. Schafer (1978). *Digital representations of speech signals*. In: A. Waibel and K.-F. Lee: "Readings in Speech Recognition". Morgan Kaufmann Publishers, Inc., San Mateo, USA, 1990.

[61] L. R. Rabiner and R. W. Schafer (1978). *Digital processing of speech signals*. Prentice-Hall, Inc., New Jersey, USA, 1978.

[62] M. Ravishankar (1996). *Efficient Algorithms for Speech Recognition*. Ph.D Thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania, May 1996, Tech Report. CMU-CS-96-143

[63] I. Rogina and T. Schaaf (2002). *Lecture and presentation tracking in an intelligent meeting room*. ICMI 2002, Pittsburgh, USA, 2002.

[64] Ch. Samuelsson and W. Reichel (1999). *A class-based language model for large-vocabulary speech recognition extracted from part-of-speech statistics*. In: Proceedings of the IEEE conference for acoustics, speech and signal processing (ICASSP), Phoenix, AZ, USA, 15.-19. März 1999.

[65] T. Schaaf (2001). *Detection of OOV words using generalized word models and a semantic class language model*. In: Proc. Eurospeech 2001, Aalborg, 2001.

[66] T. Schaaf and M. Dorna (1998). *Behandlung unbekannter Worte im Verbmobil-System*. Verbmobil Memo, April 1998.

[67] T. Schaaf and T. Kemp (1997). *Confidence measures for spontaneous speech*. In: Proc. ICASSP '97, Vol 2, pp 875 ff, München, April 1997.

[68] T. Schaaf (1996). *Vertrauensmaße für die maschinelle Spracherkennung*. Diplomarbeit an der Universität Karlsruhe, Oktober 1996.

[69] T. Schultz, A. Waibel, M. Bett, F. Metze, Y. Pan, K. Ries, T. Schaaf, H. Soltau, M. Westphal, H. Yu and K. Zechner (2001). *The ISL meeting room system*. In: Proceedings of the Workshop on Hands-Free Speech Communication (HSC-2001), Kyoto, Japan, April 2001.

[70] E. G. Schukat-Talamazzini (1995). *Automatische Spracherkennung*. Vieweg Verlag, Braunschweig, 1995.

[71] S. Sheremetyeva, J. Cowie, S. Nirenburg and R. Zajac (1998). *A multilingual onomasticon as a multipurpose NLP resource*. In: Proceedings of ELAR 1998, First International Conference on Language Resource & Evaluation, Granada, 28-30 Mai 1998.

[72] H. Soltau, F. Metze, C. Fügen and A. Waibel (2001). *A One pass decoder based on polymorphic linguistic context assignment*. In: Proceedings of the automatic speech recognition and understanding workshop, ASRU-2001, Madonna di Campiglio, Trento, Italien, Dezember 2001.

[73] H. Soltau, T. Schaaf, F. Metze and A. Waibel (2001). *The ISL Evaluation System for Verbmobil - II*. In: Proc. ICASSP '01, Salt Lake City, USA, 2001.

[74] V. Steinbiss, Bach-Hiep Tran and H. Ney (1994). *Improvements in beam search*. In: Proc. ICSLP '94 pp. 2143 ff, Yokohama, Japan, 1994.

[75] B. Suhm, M. Woszczyna and A. Waibel (1993). *Detection and transcription of new words*. In: Proc. EUROSPEECH '93, S. 2179 ff, Berlin, September 1993.

[76] A. J. Viterbi (1967). *Error bounds for convolutional codes and an asymptotically optimal decoding algorithm*. In: IEEE Transactions on Information Theory, vol. IT-13, S. 260-269, April 1967.

[77] A. Waibel, M. Bett, F. Metze, K. Ries, T. Schaaf, T. Schultz, H. Soltau, H. Yu, and K. Zechner. *Advances in automatic meeting record creation and access*, In: Proc. ICASSP '01. Salt Lake City, UT, USA, Mai 2001.

[78] A. Waibel, H. Soltau, T. Schultz, T. Schaaf and F. Metze *Multilingual Speech Recognition*. In: Verbmobil: Foundations of Speech-to-Speech Translation. Springer Verlag, Berlin, 2000, ISBN 3-540-67783-6.

[79] A. Waibel and K. F. Lee (1990). *Readings in Speech Recognition*. Morgan Kaufmann Publishers, Inc., ISBN 1-558-60124-4, San Mateo, CA, USA, 1990.

[80] J. C. Wells (1989). *Computer-coded phonemic notation of individual languages of the European Community*, Journal of the International Phonetic Association, Vol. 19, 1989.

[81] P. C. Woodland (2002). *The development fo the HTK Broadcast News transciption system: An overview*. In: Speech Communication, Volume 37, Issues 1-2, May 2002, Pages 47-67.

[82] H. Yu, M. Finke and A. Waibel (1999). *Progress in automatic meeting transcription*. EUROSPEECH '99, Budapest, 1999.

[83] H. Yu and A. Waibel (2000). *Streamlining the front-end of a speech recognizer*. ICSLP '00, Beijing, China, Oktober 2000.

[84] G. K. Zipf (1965). *Human behavior and the principle of least effort*. Nachdruck der Ausgabe 1949, Haffner. New York, 1965.