Trainable Text-to-Speech Synthesis for European Portuguese

Inaugural-Dissertation
zur Erlangung der Doktorwürde
der Philosophischen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität
zu Bonn

vorgelegt von

Maria João Almeida de Sa Barros Weiss

aus Luanda

Bonn, 2010

Cuvillier Verlag Göttingen
Internationaler wissenschaftlicher Fachverlag

Trainable Text-to-Speech Synthesis for European Portuguese

Inaugural-Dissertation
zur Erlangung der Doktorwürde
der Philosophischen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität
zu Bonn

vorgelegt von

Maria João Almeida de Sá Barros Weiss

aus Luanda

Bonn, 2010

Gedruckt mit der Genehmigung der Philosophischen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

Zusammensetzung der Prüfungskommission:

Prof. Dr. Caja Thimm

(Vorsitzende)

Prof. Dr. Bernd Möbius

(Betreuer und Gutachter)

Prof. Dr. Wolfgang Hess

(Gutachter)

Prof. Dr. Ulrich Schade

(weiteres prüfungsberechtigtes Mitglied)

Tag der mündlichen Prüfung: 02. August 2010

Diese Dissertation ist auf dem Hochschulschriftenserver der ULB Bonn unter

http://hss.ulb.uni-bonn.de/diss online elektronisch publiziert.

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.
1. Aufl. - Göttingen: Cuvillier, 2011
Zugl.: Bonn, Univ., Diss., 2010

© CUVILLIER VERLAG, Göttingen 2011
Nonnenstieg 8, 37075 Göttingen
Telefon: 0551-54724-0
Telefax: 0551-54724-21
www.cuvillier.de

1. Auflage, 2011
Gedruckt auf säurefreiem Papier

978-3-86955-773-1

I dedicate this dissertation to Christian, Jan Rodrigo and Tim Miguel,
For all their Love,
With all my Love.

Whenever a new discovery is reported to the scientific world, they say first,

‘‘It is probably not true’’.

Thereafter, when the truth of the new proposition has been demonstrated beyond question, they say,

‘‘Yes, it may be true, but it is not important’’.

Finally, when sufficient time has elapsed fully to evidence its importance, they say,

‘‘Yes, surely it is important, but it is no longer new’’.

Michel Eyquem Montaigne (1533-1592)

Acknowledgements

I thank Prof. Bernd Möbius and Prof. Wolfgang Hess for supervising my work and for the interesting discussions and valuable comments.

I thank Prof. Diamantino Freitas for the years I worked with him and the knowledge he helped me to acquire.

I thank Prof. Keiichi Tokuda for the months I worked at his lab and Ranniery Maia for all the support during this period. I thank Prof. Gil Resende for making the contact with Prof. Tokuda's lab and his support in Japan.

I thank ISEL for the sabbatical year to work on the thesis.

And I thank David Kennedy for permitting me to finish the dissertation.

I thank Christian for the motivation, long discussions and all kinds of support. And I thank Jan and Tim for the inspiration.

I thank my family for believing in my work, in particular my parents for their support and encouragement and tia Fernanda, tio Plácido and Margarida Varela for the motivation.

I thank Ellen for all her help, without which would not be possible to finish the thesis, extended to Hans for his help during many weekends.

Finally, I want to thank the many other persons whose work I used, whose work inspired me and whose strength motivated me.

Contents

List of Figures

List of Tables

List of Acronyms

AP Angolan Portuguese

ASR Automatic Speech Recognition

BP Brazilian Portuguese

CART Classification And Regression Tree

DAM Diagnostic Acceptability Measure

DCT Discrete Cosine Transform

DIXI+ TTS Synthesizer for European Portuguese

DRT Diagnostic Rhyme Test

EM Expectation Maximization

EP European Portuguese

F0 Fundamental Frequency

FFT Fast Fourier Transform

FST Finite-State Transducer

G2P Grapheme-To-Phoneme

HMM Hidden Markov Model

HTK HMM ToolKit

HTS HMM-based Speech Synthesis System

IBM International Business Machines Corporation

IIR Infinite Impulse Response

IPA International Phonetic Alphabet

logF0 logarithm of Fundamental Frequency

LP Linear Prediction

LPC Linear Prediction Coding

LSF Linear Spectral Frequency

MFCC Mel-frequency Cepstral Coefficient

ME Maximum Entropy

MLSA Mel Log Spectrum Approximation

MOS Mean Opinion Score

MRT Modified Rhyme Test

MSD Multi-space Probability Distribution

MSD-HMM Multi-Space Probability Distribution Hidden Markov Model

Multivox G2P conversion system for Portuguese language

NLP Natural Language Processing

OLA OverLap and Add

OVE Orator Verbis Electris

PAT Parametric Artificial Talker

PCM Pulse-Code Modulation

PDF Probability Density Function

POS Part-Of-Speech

PSOLA Pitch Synchronous OverLap and Add

RN Rio Grande do Norte

RJ Rio de Janeiro

SAMPA Computer Readable Phonetic Alphabet

SPTK Speech Signal Processing Toolkit

SUS Semantically Unpredictable Sentences

TBL Transformation-Based Learning

TCL Program development tool

TD-PSOLA Time Domain Pitch Synchronous OverLap and Add

TTS Text-to-Speech

WER Word Error Rate

WFST Weighted Finite-State Transducer

WPB Words Phonetically Balanced

Chapter 1

Introduction

1.1 Zusammenfassung

Die vorliegende Arbeit trägt zum aktuellen Forschungsstand der Sprachsynthese für europäisches Portugiesisch bei, indem ein Sprachsynthese- System entwickelt wurde, welches mit wenig Rechenleistung auskommt. Ebenso wurde ein spezielles Korpus hinzu entwickelt, mit dem das Sprachsythese-System trainiert werden kann. Weiterhin werden neue Ansätze zur Sprachsynthese eingeführt, die natürlich klingende und sehr verständliche Sprache aus Text generieren. Die Arbeit stellt innovative Lösungen vor, um Sprachsynthese auf Geräten mit wenig Rechenleistung, wie mobile Computer, oder Mobiltelefone auszuführen. Das entwickelte Sprachsynthese System ist das erste für europäisches Portugiesisch, das die Sprachparameter direkt aus Hidden-Markov Modellen erzeugt. Das Sprachsynthese System ist vollständig implementiert und läuft unter dem Linux Betriebsystem sowie unter dem Windows Betriebssystem. Die vorliegende Arbeit trägt auch zur Forschung bezüglich Sprachdatenkorpora für europäisches Portugiesisch bei. Es wurde ein kontext-basiertes Sprachdatenkorpus manuell ausgearbeitet und erstellt. Das Sprachdatenkorpus besteht aus je einem Satz, in dem jedes Diphon abhängig von der Koartikulation und zugehöriger Vokalreduktion in der gesprochen Sprache abgebildet mit zugehöriger phonetischer Transkription wird. Weiterhin wurde ein Text-Vorverarbeitungsmodul für europäisch portugiesische Sprachsynthese entwickelt, bestehend aus einem automatischen Graphem-Phonem Umsetzungsmodul, einer automatischen Silbengrenzen-Erkennung, sowie einer automatischen Silbenakzent Vorhersage. Die automatischen Vorhersagen werden mittels statistisch motivierten Modellen erzeugt. Für die Berechnung der Modelle wird der Maximum Entropie Algorithmus eingesetzt, der erfolgreich für natürlichsprachliche Textverarbeitung eingesetzt an anderer Stelle verwendet wird.

Es wird eine ausführliche Übersicht gegeben über Algorithmen und Methoden zum Einsatz des Quelle-Filter Modells, sowie Algorithmen, die die Vokaltrakt Funktion simulieren und manipulieren. Des Weiteren findet sich eine grundlegende Übersicht wieder, über konkatenative Sprachsynthese Ansätze, bei denen Sprachbausteine aus einem Sprachdatenkorpus ausgeschnitten werden und zu einem neuen Sprachsignal wieder zusammengesetzt werden. Es wird das Hidden-Markov Modell (HMM) basierte Sprachsynthese Verfahren eingeführt und die Verwendung von statistischem HMM Lernverfahren zur Generierung von Sprachsignalparametern erläutert, die dann automatisch als statistisches Modell trainiert werden können. Dieser Ansatz zur Generierung der Sprachsignalparameter mittels HMMs verwendet Spektralkoeffizienten, wie z.B. Linare Prädiktive Koeffizienten (LPC) oder Mel-Frequenz-Cepstrum-Koeffizienten (MFCC). Am häufigsten werden, wie auch in dieser Arbeit, MFCCs als Datenbasis für das Training des statistischen Modells herangezogen. Der Grund für die Auswahl der Parameter liegt im Ansatz, ob eine einfache Impulsfolge mit einem Gaussschen Rauschen als Anregung verwendet wird, wie in dieser Arbeit in Kapitel 5 beschrieben, oder aber LPCs, wenn zum Beispiel die Nutzung des verbleibenden Signals, das Residual-Signal, als Filter für die Anregung verwendet wird, wie es in Kapitel 6 als System-Erweiterung vorgeschlagen ist.

In dieser Arbeit liegt der Fokus auf statistischen Modellen, da diese Vorteile hinsichtlich der verwendeten Sprachressourcen haben. Es wird der Einsatz von HMMs in der Sprachsynthese, im Speziellen für Portugiesisch, untersucht und ein Sprachsynthesesystem entwickelt. HMMs zählen zu den prominentesten statistischen Sprachsynthese Ansätzen. Um die Vorteile der HMM basierten Sprachsynthese besser verstehen zu können, werden die unterschiedlichen Herangehensweisen von Unit-Selection basierten Synthesesystemen und der HMM basierten Sprachsynthese herausgearbeitet und dargestellt. Es wird die Entwicklung der HMM basierten Sprachsynthese aufgezeigt, bei denen zunächst ein Modell für alle Einheiten verwendet wurde, bis zu aktuellen Entwicklungen, bei denen pro Sprachsegment ein Modell generiert und verwendet werden. Ein wichtiger Schritt für den Erfolg der HMMs in der Sprachsynthese ist die Verwendung von dynamischen Eigenschaften von Sprache, welche durch Einschliessen der Vorgängerinformationen der jeweiligen Sprachsegmente verbessert wird. Es werden für den herausgearbeiteten HMM Ansatz die wichtigsten Techniken für den Einsatz von HMMs in der Sprachsynthese beschrieben und im entwickelten System eingesetzt. Die Mel-Cepstrum Analyse-Technik zur Extraktion von Sprachsignalparametern, wird verwendet um Entscheidungsbäume zu trainieren, die den sprachlichen Kontext abbilden und diesen in die einzelnen Zustände der HMMs mit einbeziehen. Weiterhin wird die mehrdimensionale Wahrscheinlichkeitsverteilung für die Unterscheidung von stimmhaften und stimmlosen Lauten zur Integration in die HMMs angewendet, sowie ein spezieller Algorithmus für die Sprachsignalparameter Generierung, welche dann zu dem Sprachsignal syntheisiert werden.

Eine wichtige Methode der Sprachsynthese, der Unit-Selection basierte Ansatz, wurde untersucht und die Verfahren und Algorithmen dargestellt. Dieser Ansatz nutzt als Datenbasis groe Sprachdatenkorpora aus denen Sprachbausteine als Segmente durch Ausschneiden aus Trägersätzen verwendet werden, um dann die für das Ziel-Sprachsignal geeigneten besten Sprachbausteineinheiten zu verketten. Dieser Ansatz hat eine zweidimensionale Kostenfunktionen als Basis-Algorithmus zur Grundlage, die über manuelle Gewichtung oder durch eine statistische Übergangswahrschein- lichkeit die besten Übergänge der Sprachsegmente errechnet.

In der aktuellen Forschung und Anwendung beruhen Sprachsynthese Systeme meist auf sehr groen Sprachdaten-Korpora. Das hat den Vorteil, dass eine Vielzahl von kontextbasierten Sprachbausteinen repräsentiert werden. Soll nun eine neue Äuerung synthetisiert werden, so nutzt der Unit-Selection basierte Algorithmus eine bestimmte Zielvorgabe, die das Sprachsegment erfüllen muss, und sucht das geeignete Sprachsegment aus den Sprachdaten zur Verkettung mit den anderen Sprachsegmenten heraus. Diese Zielvorgaben werden über eine zweidimensionale Kosten-Funktion abgebildet, um eben die geeigneten Einheiten für die Verkettung zu identifizieren. Die Kostenfunktion selbst ist ein metrischer Algorithmus, der eine quasi Entfernung des phonetischen Zusammenhangs der aufeinanderfolgenden Sprachsegmente wie auch zusätzlich phonologische und prosodische Eigenschaften mit einbezieht. Weiterhin werden die Sprachsignaleigenschaften, die im Spektralbereich identifiziert wurden, in die metrische spektrale Abstandsberechnung der aufeinanderfolgenden Einheiten einbezogen. Der Vorteil eines solchen Systems ist, dass ein sehr natürlich klingendes Sprachsignal erzeugt werden kann. Der Nachteil liegt vor allem in den Expertenkosten bei der Gestaltung und Aufzeichnung der Sprachdatenkorpora, die in der Regel mehrere Stunden Sprache repräsentieren, und den Kosten für die Experten, die das Korpus nachbearbeiten. Darüber hinaus beeinflusst die Qualität der aufgenommen Sprachdaten die Qualität des erzeugten Sprachsignals. So kann es durchaus vorkommen, dass kein geeignetes Sprachsegment im Sprachdatenkorpus abgebildet ist und daher ein unpassendes Sprachsegment ausgewählt wurde. Dieser Umstand beeinträchtigt die Gesamtqualität des erzeugten Sprachsignals erheblich.

Ein Ansatz, der die Nachteile der Unit-Selection basierten Synthesen überwindet, besteht in der Verwendung eines Quelle-Filter-Modells. Die Quelle des Filters kann durch eine einfache Pulsfolge mit akustischen Parametern, wie der Grundfrequenz (F0), angeregt und das Filter mittels der spektralen Parameter realisiert werden, das dann die gesamte Eingabe der Signalparameter zu einer sprachlichen Äuerung resynthetisiert. Um nun die richtigen Parameter für die Quelle und das Filter auszuwählen, wird ein HMM trainiert. Dieser Ansatz ist für die Generierung der Sprachsignale des entwickelten Systems umgesetzt. Die extrahierten Parameter, die als Trainingsdaten für die HMMs verwendet werden, sind die Mel-Cepstrum Parameter sowie F0 und Dauer in Millisekunden. Mit Hilfe von statistisch motivierten Entscheidungsbäumen werden Kontextabhängigkeiten, wie sie in gesprochener Sprache vorkommen, miteinbezogen und die Sprachsegmente in Cluster zusammengefasst. Der Vorteil eines HMM basierten Sprachsynthese-Systems ist, dass es verständliche Sprache mit einer kleinen Menge von Sprachdaten erzeugen kann. Ebenso ist die HMM basierte Sprachsignalgenerierung weniger anfällig für Inkonsistenzen in der Sprachdatenbank. Aufgrund der geringen Sprachdaten sind auch die Expertenkosten zur Erstellung der Sprachdaten- Trainingskorpora geringer. Ein weiterer Vorteil ist die Adaption an neue Sprachen bzw. Stimmen. Der Nachteil eines solchen Systems ist immer noch der Vocoder-Klang der synthetisierten Äusserung, der sich aus dem Quelle-Filter-Modell bzw. aus der Transferfunktion ergibt.

Das entwickelte Sprachsynthese System wurde für in Europa gesprochenes Portugiesisch entwickelt und implementiert. Der Bedarf für ein Sprachsynthese System für Portugiesisch resultiert aus der Anforderung, dass Portugiesisch die siebthäufigste Sprache der Welt in Bezug auf die Zahl der Muttersprachler ist, und mit rund 178 Millionen Muttersprachlern, und es ist die zweithäufigste gesprochene Sprache in Lateinamerika. Portugiesisch wird unter anderem in Angola, Brasilien, auf den Kap Verdischen Inseln, China (Macau) und in Guinea-Bissau gesprochen. Weiterhin in der Indische Union (Daman, Diu und Goa), Indonesien (Flores Island), Ara, Malaysia (Mallaca), Mosambik, Portugal, Sao Tome & Principe, Timor Lorosa'e und Uruguay. Wobei es Amtssprache in acht Ländern ist: Angola, Brasilien, Kap Verde, Guinea-Bissau, Mosambik, Portugal, Sao Tome & Principe, Timor Lorosa'e.

Das portugiesische Alphabet besteht aus dem ursprünglichen lateinischen Alphabet mit 23 Buchstaben. Das Europäische Portugiesisch besitzt ein phonetisches Inventar mit achtunddreiig Phonemen. Besonderheiten des Portugiesischen in der gesprochenen Sprache sind vor allem die Koartikulation zwischen Wrtern und Vokalreduktionen, die sehr häufig auftreten. Beide Effekte werden in dieser Arbeit behandelt und bezüglich der Korporaerstellung für das Portugiesische Sprachsynthesesystem mit einbezogen. Die Besonderheit der Koartikulation zwischen Wrtern wirkt sich zum Beispiel so aus, dass die phonetischen Transkriptionen von aufeinanderfolgenden Worten beeinflusst werden gegenüber den phonetischen Standard Transkriptionen. Die Auswirkungen der Vokalreduktion sind bezüglich gesprochener Sprache und synthetisierter Sprache in der Weise unterschiedlich, dass die synthetisierte Sprache flschlich Vokale hörbar macht, die in der gesprochenen Sprache nicht hörbar wären.

In den Abschnitten 3 bis 5 wird die Implementierung eines vollständigen Sprachsynthesesystems beschrieben, beginnend von der Eingabe des Textes bis zur Ausgabe einer sprachlichen Äuerung. Zu Beginn kommt ein Text Vorverarbeitungs- modul zum Einsatz, welches den eingegebenen Text mit dem Text Vorverarbeitungs- modul Textnormalisierung bearbeitet. Textnormalisierung heisst, dass Abkürzungen, Datum, Telefonnummern, Zahlen, Akronyme und andere Symbole, in lesbaren Text graphemisch umgesetzt werden muss. Wurde der Text aufbereitet und alle nicht graphemischen Textstellen transformiert, wird eine linguistische Analyse der Texteingabe erstellt. Das linguistische Analyse Modul führt zum Beispiel eine morphosyntaktische Analyse durch, die sehr hilfreich ist, um Homographen auszulösen. Auch für die prosodische Vorverarbeitung und die Extraktion prosodischer Merkmale ist das linguistische Analysemodul notwendig. Während die Textnormalisierung und das linguistische Analysemodul auf Graphembasis arbeiten, kommt für die eigentliche Eingabe in das Synthesesystem ein phonetisch transkribierter Text zum Einsatz. Die Umwandlung der Eingabe von Text in ihre korrespondierende phonetische Transkription übernimmt ein Verarbeitungsmodul für natürliche Sprache, kurz NLP-Modul, welches einen Graphem-Phonem Umsetzungsalgorithmus enthält. Das NLP-Modul erfüllt noch weitere wichtige Aufgaben, um Merkmale aus der Texteingabe zu erhalten. So werden weitere nützliche Informationen wie Wörter- und Silbengrenzen erkannt, sowie Wort- und Silbenprominenz bzw. Akzent markiert. Damit die Prosodie auch in der gesprochenen Sprache in der Sprachsynthese wiedergegeben werden kann, müssen auch die prosodischen Muster bestimmt werden. Diese werden vor allem durch die Grundfrequenz und die Dauer bestimmt. Hierzu kommt ein Prosodie-Modul zum Einsatz, dass in das Sprachsynthese System integriert wird. Information, wie F0 und segmentale Dauer werden hier über statistische Verfahren geschätzt. Das letztes Modul, welches dann die extrahierten Merkmale und Sprachsignalparameter verarbeitet und den Text in eine sprachliche Äuerung überträgt, ist das Signalgenerierungsmodul.

Von groer Wichtigkeit für die Qualität von Sprachsynthese Systemen ist die Gestaltung der Sprachdatenkorpora. Kapitel 5 befasst sich ausführlich mit der Gestaltung und Entwicklung eines geeigneten Sprachdatenkorpus für die HMM basierten Synthese-Systeme, wie in dieser Arbeit umgesetzt. Statistische Systems wie das HMM System berechnen die Übergangswahrscheinlichkeiten zwischen den einzelnen Zuständen. Von daher ist das Einbeziehen von sprachlichem Kontext in der Korpuserstellung notwendig, um alle Ereignisse, die auftreten können, abzudecken. Gewöhnlich ist es nicht möglich alle auftretenden Ereignisse in der gesprochenen Sprache mit einem kleinen Sprachdatenkorpus zu berücksichtigen und diese aufzunehmen. Bei der Erstellung eines geeigneten Sprachdatenkorpus wurde speziell auf diese Einschränkungen eingegangen und ein Korpus entwickelt, dass mit wenigen

Daten möglichst alle sprachlichen Ereignisse abdeckt. Dieses Korpus wurde manuell zusammengestellt. Das neue Sprachdatenkorpus wurde auch unter Berücksichtigung der speziellen Anforderungen und den bisherigen wissenschaftlichen Erkenntnissen zur Erstellung von Sprachsynthesekorpora entwickelt. Hier wurde vor allem auf das Korpus der Ingenieursfakultät der Universität Porto, Portugal, FEUP-IPB-Datenbank zurückgegriffen. Es wurde ein professioneller männlicher Sprecher aufgenommen, der im Vorlesestil die aufzunehmenden Sätze gesprochen hat. Das neue Sprachdatenkorpus ist folgendermaen repräsentiert: Graphem-Sätze, phonetische Transkription, typische Phänomen mit Markierungen der in Europäischem Portugiesisch typischen Effekte der Koartikulation und Vokalreduktion. Das Korpus wird für den späteren Einsatz durch die Verteilung der Phoneme im Gesamten, sowie die Verteilung von Phonemen innerhalb der Sätze und Wörter repräsentiert. Es wurde eine Analyse zwischen der Anzahl der Einheiten in der phonetischen Transkription mit und ohne Berücksichtigung der Vokalreduzierung vorgelegt, um die Bedeutung dieses Effektes im Europäischen Portugiesisch nachvollziehen zu können, sowie eine Analyse der verwendeten Diphone. Zur Darstellung wurde eine Verwechslungsmatrix generiert. Für die Graphem-Phonem Umsetzung wurde ein statistisches Werkzeug auf Basis des Maximum Entropie Frameworks entwickelt und eingesetzt. Maximum Entropie ist ein statistisch motiviertes Modell, das u.a. erfolgreich für das Tagging von sequentiellen Daten wie Part-of-Speech (POS) Tagging oder für das syntaktische Parsing angewandt wurde. Die Entscheidung für statistisch motivierte Werkzeuge basiert auf einer Kostenabschätzung für die einzelnen Teilschritte. Regel-basierte Systeme erfordern Expertenwissen, wohingegen statistisch motivierte Systeme auch ohne linguistisches Fachwissen umgesetzt und betrieben werden können. Vor allem bei der Umsetzung von Graphemen in entsprechende Phoneme ist Expertenwissen und Erfahrung des Experten entscheidend. Statistische Systeme zeigen sich hier flexibler, sofern die statistischen Modelle mit ausreichenden und sinnvollen Daten trainiert werden, wie zuvor angedeutet mit Berücksichtigung von Effekten der gesprochenen Sprache, Koartikulation und Vokalreduktion. In europäischem Portugiesisch existieren achtunddreiig Phoneme, da einige Grapheme in den Transkriptionen durch eine Kombination von mehr als einem Phonem repräsentiert werden, entstehen vierundvierzig Phon-Klassen statt achtunddreiig.

Bei der Vorhersage des Wort- und Silbenakzents kommt eine binäre Entscheidung zum Tragen, indem eine Klasse die betonten Silben widerspiegelt und eine Klasse für keinen gesetzten Akzent steht. Die gleiche binäre Klassifikation kommt bei der Entscheidung Silbengrenze oder nicht zum Einsatz. Zur Einschätzung der Qualität der Vorhersagen der statistischen Werkzeuge wird das Qualitätsma Word Error Rate (WER) und Phoneme Error Rate (PER) eingeführt, wobei sich ersteres auf die Fehlerrate der Worte bezieht und das zweite Ma auf die Substitutionsfehler der

Graphem-Phonem Umsetzung. Letzteres kommt auch zum Tragen, wenn in dem System eine Einfügung stattfindet, obwohl in der manuellen Transkription ein leeres Resultat vorhanden ist, und ohne dass ein klangliches Ereignis damit verbunden ist. Die Ergebnisse sind für Vokale und Konsonanten getrennt aufbereitet und durch Verwendung einer Konfusionsmatrix angeordnet.

Wie zuvor erwähnt kommt für die Generierung der sprachlichen Äuerung eine HMM basierte Synthese zum Einsatz, die auf dem HTS Framework der Technischen Universität Nagoya aufbaut. Die HMM Topologie für das eingeführte System ist auf einem sieben dimensionalen Links-Rechts Kontext bei dem die Zustände nicht übersprungen werden. Es gibt zwei nicht emittierende Zustände, die durch den Start- und Endzustand repräsentiert sind. Die anderen fünf Zustände sind mit Ausgaben verbunden und besitzen jeweils eine Übergangswahrscheinlichkeit. Die Erstellung des Hidden-Markov Modells teilt sich in drei Aufgabengebiete auf: die Extraktion der Sprachsignalparameter, die Aufbereitung der kontextuellen Merkmale sowie das Trainieren des Hidden-Markov Modells selbst.

Die Sprachsignalparameter werden durch die Extraktion der Grundfrequenz und der Mel-Cepstrum Koeffizienten aus dem Trainingskorpus gewonnen. Die phonetischen, phonologischen und metrischen Merkmale werden zu einem Merkmalsvektor zusammengefasst. Dieser Merkmalsvektor muss für das Training wie auch später zur Laufzeit des Systems generiert werden. Die Merkmale werden mit den zuvor genannten Werkzeugen extrahiert. Für die Erstellung eines Hidden-Markov Modells werden als akustische Parameter die logarithmierte Grundfrequenz, die Mel-Cepstrum Parameter und die logarithmierte Dauer verwendet. Der Eingabevektor des HMM Trainings umfasst achtundsiebzig Dimensionen, die in vier Ströme aufgeteilt in das Hidden-Markov Modell einflieen. Der erste Strom ist eine gemischte Komponente und spiegelt fünfundsiebzig Dimensionen wider und beinhaltet ebenso Mittelwerte und Varianzen. Aufgeteilt entspricht dies: fünfundzwanzig Parameter für die Mel-Cepstrum Parameter, fünfundzwanzig auf die Delta-Werte und die letzten fünfundzwanzig für die Delta-Delta Koeffizienten. Die anderen Ströme sind zusammengesetzt aus ihrem Gewicht und dem Mittelwert sowie der Varianz für die Gausssche Wahrscheinlichkeitsdichtefunktion. Der dritte Strom entspricht dem logF0 und der vierte dem Delta-F0.

Die kontextuellen Faktoren speziell für das Sprachsynthese System für Europäisch- es Portugiesisch sind die Kontexte der Phoneme und die Positionen des aktuellen Phonems in der aktuellen Silbe und dem Wort, jeweils gezählt von rechts und links. Ein weiteres Merkmal ist der Akzent der aktuellen Silbe, der vorhergehenden und der nachfolgenden Silbe, die Anzahl der Phoneme in der vorangehenden Silbe und der nachfolgenden Silbe, jeweils von Silbenanfang und Silbenende gezählt.

Das Merkmal der Position der Silbe im aktuellen Wort und in der aktuellen Phrase; die Anzahl der betonten Silben vor und nach der aktuellen Silbe in der aktuellen Phrase und wie viele Silben zwischen der vorangegangenen betonten Silben und der aktuellen Silbe liegen. Weiterhin fliet der Vokal der Silbe ein. Auf Wortebene wird die Anzahl der Silben im aktuellen, vorhergehenden und nachfolgendem Wort gezählt. Weiterhin die Positionen des aktuellen Wortes gezählt von Phrasenanfang und Phrasenende der aktuellen Phrase. In den Merkmalsvektor flieen auch noch Angaben ber die Anzahl der Silben und die Anzahl der Wörter im aktuellen, vorangehenden und nachfolgenden Satz, und die Positionen der aktuellen Phrase in der Äuerung ein.

Zur Auswahl der kontextuellen Hidden-Markov Modelle für die Verkettung werden binäre Entscheidungsbäume verwendet, die die einzelnen HMMs bezüglich der Phoneme auswählen. Diese HMMs werden zusammengesetzt, und die Generierung der Mel-Cepstrum Parameter und der logarithmierten Grundsequenz und Dauer wird gestartet. Die Generierung des akustischen Parameters Grundfrequenz ist durch eine mehrdimensionale Wahrscheinlichkeitsverteilung (MSE) modelliert und die Mel-Cepstrum Parameter mittels einer multivariaten Gauss-Verteilung. Beide werden dann durch den Signalparametergenerierungsalgorithmus erzeugt und als kontinuierliche HMMs abgespeichert. Die Dauer wird durch eine mehrdimensionale Gauss-Verteilung bestimmt. Jedem Zustand im Hidden-Markov Modell wird eine Dauerverteilung zugewiesen. Ein wichtiger Baustein in einem HMM-basierten Sprachsynthese System ist das Anregungsmodul, das hier für das verwendete Quelle-Filter-Modell durch das Mel-Log-Spektrum-Approximations-Filter umgesetzt ist. Das Anregungssignal wird am Eingang des MLSA Filter entweder durch eine Impulsfolge für stimmhafte Laute oder durch Gausssches weies Rauschen für stimmlose Laute nachgebildet. Die Übertragungsfunktion des Filters basiert auf den Mel-Cepstrum Eingabeparametern. Die synthetische Sprache wird dann durch die Wellenform des MLSA Filters erzeugt. Der Vokaltrakt wird durch die erzeugte Mel-Cepstrum Sequenz und die Übertragungsfunktion des Filters modelliert. Die Quelle des Filters ist die logarithmierte Grundfrequenzabfolge.

Um die Qualität der Ergebnisse zu bewerten, wurde das System im Vergleich zu zwei anderen Sprachsynthese Systemen evaluiert. Zum einen kommt als Vergleichssystem das kommerzielle System RealSpeak von ScanSoft zum Einsatz, und zum Anderen das an der Universität Porto entwickelte System basierend auf dem konkatenativen PSOLA-Ansatz. Dieser Vergleich ermöglicht die Bewertung der Ergebnisse des Systems. Für die Evaluationen der Sprachsynthese wurden subjektive Bewertungen von zwei Benutzergruppen durchgeführt. Folgende zwei Tests wurden entwickelt: Tests zur Akzeptanz, die sich durch die subjektiven Meinungen der Testhörer über die Sprachqualität ausdrücken, und ein Test zur Verständlichkeit, der mit Hörverstehen auf offenen Fragen oder Multiple-Choice-Antworten durchgeführt wurde.

Ein Überblick über die Evaluation für die Sprachsynthese wird im Verlauf vorgestellt. Das ausgewählte Bewertungsschema ist der Mean Opinion Score (MOS)-Test, der mit einem Testset für Europäisches Portugiesisch entwickelt wurde. Der MOS-Test verwendet Sätze, die dem Hörer für die Bewertung mittels einer Skala von 1 bis 5 vorgespielt werden, wobei fünf hier den besten Wert widerspiegelt. Die Testergebnisse zeigen, dass das HMM basierte Sprachsynthese System eine gute Akzeptanz hat.

Im Anschluss an die Kapitel zur Entwicklung eines HMM basierten Sprachsynthese Systems werden zwei weitere Verbesserungen für ein solches System vorgeschlagen. Zum Einen wird die Verwendung des Residualsignals als Eingabeparameter der Anregung für das Filter erarbeitet. Diese Signale haben einen geringen Amplitudenausschlag und sind daher bei niedrigen Frequenzen gut geeignet. Die Auswirkungen auf das Synthesemodul wären, dass mehr Speicherplatz notwendig ist, um die Residualsignalparameter zu speichern und dass bei der Übertragung eine höhere Bandbreite als bei der ursprünglichen Version mit Impulsfolgen und Gaussschem Rauschen als Eingang nötig ist. Die übrigen Ressourcen würden gleich bleiben. Dieser Ansatz ist in Betracht zu ziehen, da das entwickelte und voll funktionsfähige Sprachsynthese System mit 21 Minuten Sprache auskommt und der höhere Speicherbedarf nicht wesentlich zum Tragen kommt. Zum anderen wird ein Modul konzeptionell erarbeitet, um in der Sprachsynthese Wörter zu synthetisieren, die nicht der Ursprungssprache entsprechen. Dies ist ein zunehmend wichtiger Aspekt bei der Entwicklung von Sprachsynthese Systemen, vor allem wenn Sprachen gemischt werden, wie in Emails, Webseiten, Kino- und Fernsehprogrammen, bei denen z.B. Deutsch und Englisch regelmäig in einem Satz vorkommen. Der vorgeschlagene Ansatz beruht auf einer Phonem Zuordnung und wird in Kapitel 6 beschrieben.

1.2 Summary

The scope of this thesis is to contribute to European Portuguese (EP) state of the art in speech synthesis for Text-to-Speech (TTS), with a small footprint TTS system, using new approaches to synthesize natural sounding and highly intelligible speech. The work introduces innovative solutions to the process of TTS synthesis for low resource devices, such as mobile devices. The developed TTS system is the first TTS for EP in which speech parameters are generated from the Hidden Markov Models (HMMs) themselves [Barros et al., 2005]. It is fully implemented and runs on Linux and Windows. The Natural Language Processing (NLP) module consists of Grapheme-To-Phoneme (G2P) conversion, Syllable Boundary Detection and Syllable Stress Prediction tasks and it is a statistically motivated model based on the Maximum Entropy (ME) method, which was successfully applied to NLP tasks for EP [Barros & Weiss, 2006]. The work also contributes to the EP language resources for language context-based systems with a manually designed speech corpus comprising one sentence targeting each diphone of the language, 1436 sentences, which is completely phonetically transcribed considering the word coarticulation and vocalic reduction effects [Barros & Moebius, 2009].

The dissertation starts with a technical and detailed survey about speech synthesis methods and approaches. The survey considers the vocal tract approaches, either with explicit source-filter models based on rules and data, or speech production models simulating the human organs. The concatenative approaches, in which waveform concatenation is used, are also overviewed. The HMM-based synthesis approaches, implemented by applying the HMM statistical learning algorithms to generate speech parameters that can be automatically trained are considered as well. These approaches use spectral coefficients, like Linear Prediction Coding (LPC) or Mel-frequency Cepstral Coefficients (MFCCs), to implement a source-filter model. The most common are the MFCCs, for when it is intended to have a simple pulse train/Gaussian noise filter excitation, as is used in this thesis, but it is also usual to use LPCs, for instance when it is to make use of the residual signal as the filter excitation, an approach that is proposed in this thesis as a system enhancement, although using the MFCCs instead of the LPCs. The unit-selection based approaches are also considered, using large speech corpora to choose the best units to use, or via the classical approach with the two-dimensional cost functions or via statistical selection.

The statistical models in speech synthesis can be used to choose the best speech units for a specific speech output target or specification, or to generate the best speech parameters to represent the speech output target. Statistical models are chosen in this thesis because of the advantages of statistical systems in giving the possibility of training a new model on any particular database.

A survey about the use of HMMs in speech synthesis is presented, as they are used in this thesis. HMMs can be applied to the most prominent speech synthesis approaches nowadays. The thesis highlights the unit-selection based synthesis and the HMM-based synthesis as the most prominent speech synthesis approaches today. The survey includes the history of HMMs in speech synthesis, which were first used as a model for all the units simultaneously, until today's approaches, which use one model per unit. One important step in the success of HMMs for speech synthesis was the use of dynamic features of speech frames, besides the static ones, which means including information relating the current speech frame with the previous ones. The HMM method is explained and the main techniques for using HMMs in speech synthesis are described. The mel-cepstral analysis technique is the technique used for the speech feature analysis. Decision trees based in the language context are used to cluster the HMMs states. Multi-Space Probability Distribution Hidden Markov Models (MSD-HMMs) are used in order to deal with voiced and unvoiced sounds. And a specific algorithm for speech parameter generation is used.

Nowadays, TTS systems are based on large speech corpora, with several representations of each speech unit. To synthesize a new utterance, a selection algorithm finds the best speech units according to a certain target specification. This is the unit-selection based synthesis. The classical unit-selection approach uses a two-dimensional cost-function to identify the appropriate units for concatenation, through the calculation of the distance in the phonetic context with additional phonological and prosodic features, and the spectral distance regarding the succeeding units. The advantage of such a system is the highly natural sounding speech. The disadvantage is the expert costs in designing and recording large amounts of speech data, typically of several hours of speech. Furthermore, the naturalness of the speech is dependent on the units found in the database. If no appropriate unit is found the quality of the produced utterances decreases rapidly.

An approach which overcomes the disadvantages of the unit-selection based synthesis consists of using a source-filter model. The source of the filter can be a simple pulse train built with acoustical parameters, like Fundamental Frequency (F0), and the spectral parameters are the coefficients of the filter that re-synthesizes the utterances. Statistical methods are commonly used to select both the acoustical and the spectral parameter values. The HMM-based synthesis, which is the synthesis approach implemented in this thesis, is one of the approaches that use this method. In this work a source-filter model with HMM estimated speech parameters is used to generate speech. The extracted parameters are the MFCCs, F0 and durations. Decision trees considering the language contexts are used to cluster the HMMs. The advantage of a HMM-based speech synthesis system is that it can produce highly intelligible speech with a small amount of data, as the HMMs are less susceptible for inconsistencies in the database. The expert costs are much lower in designing such a speech synthesis system and the adaptation to new databases is straightforward. The disadvantage of such a system is still the vocoder like sound which results from the use of a parametric model of speech production, the source-filter model, during the synthesis process.

The TTS system presented in this thesis is implemented for the EP language. The EP language is introduced, to help the reader that is not familiar with it. Portuguese is an Iberian-Romance language derived from Latin. It is the seventh most common language in the world regarding the number of native speakers, with around 178 million native speakers and it is the second most spoken Latin language [Wikipedia, visited in 2010]. Portuguese language is spoken in Angola, Brazil, Cape Verde, China (Macau), Guinea-Bissau, Indian Union (Daman, Diu and Goa), Indonesia (Flores Island), Macaw, Malaysia (Mallaca), Mozambique, Portugal, Sao Tome & Principe, Timor Lorosa´e and Uruguay. It is the official language in eight countries: Angola, Brazil, Cape Verde, Guinea-Bissau, Mozambique, Portugal, Sao Tome & Principe and Timor Lorosa´e. It is largely used in many others. There are several different Portuguese dialects, which differ from each other not only in their grammar constructions and vocabulary, but also on their prosody and on some phonetic units. The Portuguese alphabet consists of the original Latin alphabet, with 23 letters and the EP phonetic inventory has thirty-eight phonemes. The EP language in continuous speech suffers effects of coarticulation between words and natural vocalic reduction. Both effects are covered in this thesis. The coarticulation between words means that the word transcriptions are influenced by the neighboring words. The natural vocalic reduction means that some phonemes are reduced or even suppressed in continuous speech.

A complete TTS system for EP is implemented for this thesis. TTS systems are systems that convert text into speech, meaning that these systems receive text as input and read it, producing voice as output. A TTS system involves several steps, beginning from the input of the text to the output of the sound. A general TTS system is composed by different modules. The text pre-processing module performs tasks that convert short-length text forms, like dates, phone numbers, numerals, abbreviations, acronyms and other symbols, into readable full-length text forms. The linguistic analysis module performs for instance morpho-syntactic analysis, which can be very helpful for homograph disambiguation and for an adequate prosodic manipulation. The NLP module performs tasks like converting the input text into its correspondent phonetic transcription, using a G2P converter, and add other useful information like word and syllable boundaries detection, and word stress prediction. The prosodic pattern determination is another module that can integrate a TTS system. Its function is to estimate prosodic information, like F0 and segmental durations, to produce the right prosody for the synthetic speech. Finally, the signal generation module processes the speech signal according to the used speech synthesis approach.

Of major importance to the quality of the TTS systems is the design of the speech corpus. HMM-based synthesis systems, like the one implemented in this thesis, are data-driven language context-based statistical systems. This means that they rely on the quality and richness of language contexts of the corpus to produce better or worse speech synthesis results. In this thesis a new speech corpus especially designed for language context-based systems with small footprint engines [Barros & Moebius, 2009] is developed for EP. It is fully manually designed, being composed by one sentence concerning each of the EP language diphones, to which certain language contexts are considered. The new speech corpus was designed after the experience of the TTS system implementation, using the FEUP/IPB-DB database [Teixeira et al., 2001], where a professional male speaker was recorded using a reading style in a recording booth.

The new speech corpus is presented in three forms: its orthographic sentences, its phonetic transcriptions considering the word coarticulation, typical phenomenon in the EP, and its phonetic transcriptions considering the word coarticulation and the vocalic reduction effects, another typical phenomenon of the language. The corpus is described and the results are presented in total number of phonemes and their distribution in the beginning/middle/end of a word and beginning/end of a sentence and some details of the language, which had to be considered in the corpus design, are described. A comparison analysis between the number of units in the phonetic transcriptions with and without considering the vocalic reduction is presented, in order to understand the importance of this effect in the EP, as well as an analysis of the diphones present in the orthographic sentences and in each of the phonetic transcriptions, for which the confusion matrix method is used.

The NLP tasks constitute another important factor in determining the quality of TTS system's results, because they provide significant information regarding the choice of the best units to use for the speech output. The NLP tasks used for the developed EP TTS system are G2P conversion, syllable boundary detection and syllable stress prediction.

The NLP tasks implemented for this thesis are based on the ME method [Barros & Weiss, 2006]. The ME is a statistically motivated model, which was successfully applied to the labeling of sequential data such as Part-Of-Speech (POS) tagging or shallow parsing. Statistical systems are not so cost intensive as rule-based systems and can be setup even without linguist knowledge, although the rule-based ones are more common by now. Statistical systems have proved to be more flexible according to natural sounding synthetic speech of continuous speech, as their statistical models can be trained with data as it was spoken, for instance considering coarticulation effects and vocalic reduction effects.

In the EP there are 38 phonemes, but because some grapheme transcriptions are a combination of more than one phoneme, the G2P conversion task has 44 classes instead of 38. Each phoneme or combination of phonemes from the phonetic set represents a class. For stress prediction a binary classification is considered, where the class is true for stressed syllables and false for non-stressed. The same binary classification task is solved for the syllable boundary detection, where a syllable boundary exists or not. The results for the three tasks are presented by giving the logarithmic likelihood and the performance of the system. Another test is performed with the G2P converter, as this module is more complex for classification than the other two, which are binary classification problems. All the EP inventory phonemes were covered in this test, with different numbers of occurrences. The test consists in comparing the system results of the test corpus with the entries from the training corpus, giving the average number of phoneme errors taking into account three types of errors, a measure known as Word Error Rate (WER). The first type relates to substitution errors and refers to the situations where the system replaces the correct phoneme by another. The second type relates to insertion errors and refers to the situations for which the system gives a result to a phoneme that in the manual transcription is an empty result, meaning that the grapheme being processed should not give a phonetic result. And the third type is related to deletion errors and refers to the situations where the system gives an empty result when the manual transcription gives a phoneme as result. The results are presented for vowels and consonants independently, using the confusion matrix method.

For the speech synthesis approach implemented in this thesis, the HMM-based synthesis, as it was already mentioned, the HMM-based Speech Synthesis System (HTS) framework is used. The HMM topology for the implemented system is a 7-state, left-right, no skip HMM, with two non-emitting states, the first and the last ones. The other five states, the middle ones, are emitting states and have output probability distributions associated.

The HMM training process consists of three main parts: the speech parameter extraction; the contextual labels definition; and the HMM modeling. The speech parameters extracted for the training are the F0 and the MFCCs. For the contextual label definition, utterance information from the whole speech database is converted into labels with the same format as those used during the synthesis phase for text knowledge. During the HMM modeling a unified framework that models logarithm of Fundamental Frequency (logF0), MFCCs and durations simultaneously is used. Each HMM includes state duration densities and its observation vectors have a length of seventy eight values, divided into four streams. The first stream has one mixture component and presents seventy five mean and variance values: twenty-five correspond to the MFCCs, twenty-five to their delta and the last twenty-five to their delta delta coefficients. The other streams have two mixture components, each represented by its weight and the mean and variance values for the Gaussian Probability Density Function (PDF). The second stream corresponds to the logF0, the third to its delta, and the fourth to its delta delta.

The main modules of a HMM-based synthesis system, in the way they were implemented to develop the EP system, are explained in the thesis. First there is the contextual label generation, where contextual labels that represent HMM units in the database are generated according to the utterance information of the input text to be synthesized. The contextual factors used for EP are the following: At phone level the current, previous and next phones; the phones before previous phone and after next phone; and the positions, forward and backward, of the current phone in current syllable are considered. At syllable level the stress condition of the current, previous and next syllables; the number of phones in the previous, current and next syllable; the positions, forward and backward, of the current syllable in the current word and current phrase; the number of stressed syllables before and after the current syllable in the current phrase; the syllable counts between the previous stressed syllable and the current syllable and between the current syllable and the next stressed syllable in the utterance; and the vowel of the syllable are considered. At word level the number of syllables in the current, previous and next words; and the positions, forward and backward, of the current word in the current phrase are considered. At phrase level the number of syllables and number of words in the current, previous and next phrases; and the positions, forward and backward, of the current phrase in the utterance are considered. At utterance level the number of syllables, words and phrases in the utterance are considered.

In the contextual HMM selection and concatenation module, binary decision trees are created from contextual information and used to decide which HMMs are going to be selected to represent each of the speech units. These HMMs are concatenated in a sequence of HMMs and used for generating the MFCCs, logF0 and state durations. The generation of these parameters is the next step. The logF0s are modeled according to the Multi-space Probability Distributions (MSDs) and the MFCCs are modeled according to the multivariate Gaussian distributions. Both are then determined through an algorithm for speech parameter generation from multi-mixture continuous HMMs. The state durations are determined according to the multi-dimensional Gaussian distributions. The distribution dimensionality is equal to the number of states of the corresponding HMM and each state duration density dimension corresponds to the respective HMM state.

An important module of a HMM-based synthesis system is the excitation generation module, for which a source-filter model implemented with a Mel Log Spectrum Approximation (MLSA) filter is used. The excitation signal, the input of the MLSA filter, is either a pulse train or Gaussian noise, for voiced and unvoiced segments respectively, based on the determined pitch sequence. The transfer function of the filter is based on the MFCCs. The synthesis filter constitutes the last module of a HMM-based synthesis system. The synthesized speech waveform is generated through the MLSA filter. The vocal tract is modeled by the generated MFCC sequence, which is the transfer function of the filter. The source of the filter is obtained by the generated logF0 sequence.

In order to evaluate the quality of the results, the system is compared with two other systems based on different approaches: the commercial system *RealSpeak*, from Scansoft [Realspeak, visited in 2010], based on unit-selection synthesis, and an academic concatenative synthesizer using Pitch Synchronous OverLap and Add (PSOLA) [Barros, 2002]. This comparison allows to evaluate the system's results and the advantages of HMM-based synthesis for low resources languages. In speech synthesis the evaluation is usually carried out through the use of subjective tests performed by groups of listeners, since there is not a metric or parameter that allows to classify the quality of speech according to human perception. Two kinds of evaluation schemes can be carried out: tests to acceptability, which deal with subjective opinions relating to the audible speech quality, and tests to intelligibility, which deal with listening comprehension based on open or multiple choice answers. An overview of evaluation tests for speech synthesis is presented in the thesis. The selected evaluation scheme is an acceptability test, the Mean Opinion Score (MOS) test, with a test-set designed for EP [Barros, 2002]. The MOS test scheme uses sentences, which are presented to the listeners for evaluation through a scale from one to five. The test results show that the HMM-based TTS system has a good acceptability, approaching the commercial system in score, even though the HMM-based system was trained with only twenty-one minutes of speech, what is less than five per cent of the nine hours of speech usually considered necessary for a unit-selection based system to present good results.

Besides the use of the new speech corpus, rich in language contexts, two other improvements to the system are proposed. The use of the residual signals from the acoustic units used for training is proposed as the excitation of the speech reconstruction filter. These signals are flat enough for most of the cases, at least at low frequencies. The synthesis engine would need more storage space than the original version which uses the pulse trains and Gaussian noise as input, as the residual signals have more or less the same size as the original sound waves signals, but this is not a problem for HMM-based speech systems because the amount of speech required is very small, e.g., the EP system that was implemented in this thesis showed good results with twenty-one minutes of speech.

The other proposal is to create a module to resolve foreign words. Foreign words are another concern when developing TTS systems, mainly because nowadays TTS systems are confronted more and more with the language mixing phenomenon, e.g., in e-mail readers, in speech enabled web sites, or even in automatic cinema program announcements. A method based on the mapping of phonemes from EP with phonemes of other languages or dialects is proposed to deal with foreign words when implementing a HMM-based TTS system. The approach allows synthesizing text in different languages using the same voice.

1.3 Outline

This chapter, chapter 1, introduces the summary of the thesis and gives an outline of the dissertation.

Chapter 2 describes the different speech synthesis approaches.

Chapter 3 overviews the use of HMMs in speech synthesis. The most prominent speech synthesis approaches, and the ones that present the best results in public evaluations, like the Blizzard Challenge [Blizzard, 2009], use HMMs applied to synthesis. The history of HMMs in speech synthesis is outlined, the main techniques used to apply HMMs to speech synthesis are explained and the speech parameter generation algorithm is described.

Chapter 4 presents the implemented TTS system for EP. An analysis of the EP language and phonetic inventory is provided. The NLP module is described, with a brief description of the use of ME in NLP and the results for EP. The EP language dependent module is described, referring to the language contextual factors and decision tree questions for phoneme clustering. A description of the speech synthesis module and of the training process of a synthesis system with the speech parameters generated from HMMs themselves is given. The results of the TTS system evaluation using a MOS test-set to test its acceptability and compare it with two other systems conclude the chapter.

Chapter 5 describes a speech corpus especially designed for context-based EP TTS systems. Appendix E presents a subset of the complete corpus, organized in orthographic sentences together with their phonetic transcription.

Chapter 6 proposes two improvements to the system. Section 6.2 presents a study proposing a hybrid system based on the residual signal, in substitution for the pulse train used in the source-filter model of the speech synthesizer for the voiced sounds. Finally, section 6.1 proposes a method for dealing with foreign words, based on the mapping of phonemes from EP with phonemes of other languages or dialects.

Chapter 7 summarizes the conclusions.

Chapter 2

Speech Synthesis

This chapter presents a survey about speech synthesis. There are different speech synthesis methods, following different approaches, some of which will be described in the next sections. The speech synthesis approaches can be classified in:

- **Vocal Tract approaches**
  Either explicit source-filter models and techniques based on rules, e.g. formant synthesis, or data, e.g. Linear Prediction (LP) synthesis, or speech production models that simulate the human organs, called the articulatory models.

- **Concatenative approaches**
  These approaches do not make use of the explicit source-filter models anymore, instead they simply use waveform concatenation, with or without the use of some kind of speech signal modification model, like OverLap and Add (OLA), for speech smoothing.

- **HMM-based synthesis approaches**
  These are data-driven approaches implemented by applying statistical learning algorithms, such as HMMs, to generate the speech parameters directly from the models. The models can be automatically trained.

- **Unit-selection based synthesis approaches**
  These are data-driven approaches, which are based on the concatenative approaches, but more sophisticated. These approaches use large corpora with several realizations of each speech unit. Search algorithms are used to choose the best units to use, based on two-dimensional cost functions or on statistical selection. Unit-selection based approaches rely less on signal processing and more on the variety of speech databases.

2.1 Vocal Tract Models

The vocal tract is a three-dimensional cavity with losses, composed by non-uniform transversal sections which are approximately $2cm$ large in average, which is smaller than a typical speech sound wave length. E.g., if considering a 4KHz frequency sound wave, common in speech sounds, its wave length is $\lambda = c/f = 34300/4000 = 8.6cm$, and c is the speed of sound.

The vocal tract models try to simulate the shape of the vocal tract, like the vocal tract acoustic model that determines its acoustic features, or the resonances tube model that represents the vocal tract through the use of several tubes.

The source-filter models do not simulate the vocal tract but its effect on speech, through the use of a filter whose transfer function is composed by some speech features, like formants or any spectral coefficients. Two of these types of synthesis systems, formants and LPC, are explained below.

It is common to classify as source-filter models those models that use some kind of filter to simulate the vocal tract and excite the filter with any signal source that allows to have the best possible synthetic speech in the output, not only for voiced but also for the unvoiced regions.

The articulatory models do not only physically simulate the vocal tract, but also all the speech articulators, as explained below.

One can say that the first vocal tract models for speech synthesis were Kratzenstein's resonators simulating the vocal tract for each vowel in 1779, Kempelen's talking machine with the lungs and articulators simulation in 1791 and Wheatstone's model, based on Kempelen's but with the possibility of deforming a tube in order to simulate the vowels' sounds, in 1835.

During the Electronic Era there were two different kinds of vocal tract transfer function models: the ones that used formant resonators and the ones that used electrical transmission line segments, known as the electrical vocal-tract analog models.

The vocal tract models are still used today, now mainly through the use of statistical data-driven models.

2.1.1 Source-Filter Models Synthesis

The first spectral information used to model the vocal tract were the formants, in the beginning of the Electronic Era, in 1922, with Stewart's circuit, a formant synthesizer. Another kind of source-filter model, LPC synthesis, became very successful during the 1990s.

With the statistical data-driven approaches, the use of spectral features in source-filter models became largely used.

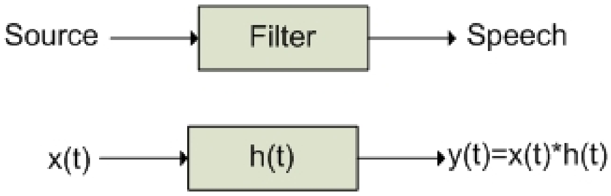


Figure 2.1: The source-filter speech model.

Figure 2.1 shows a simple source-filter speech model and its signal processing representation. This is a model in which speech is the output signal, $y(t)$, of a filter with impulse response, or transfer function, $h(t)$, representing the vocal tract, excited by a source component coming from the vocal cords, $x(t)$.

There are several models to simulate the vocal cords source signal, but the most widely used ones are the pulse train/Gaussian noise model and the use of the residual signal. The residual signal is the error between the original speech signal and the predicted (synthetic) signal. Figure 2.2 shows both schemes.

The pulse train/Gaussian noise model is a model that uses for the voiced signals a pulse train with a period corresponding to the fundamental period as the input of the filter representing the vocal tract and a Gaussian noise signal as the input for the unvoiced ones. For this kind of model a voicing decision is needed.

The model using the residual signal is more sophisticated and has more natural results, but it also consumes more resources as a way of extracting and/or storing the residual signals for each speech unit is needed.

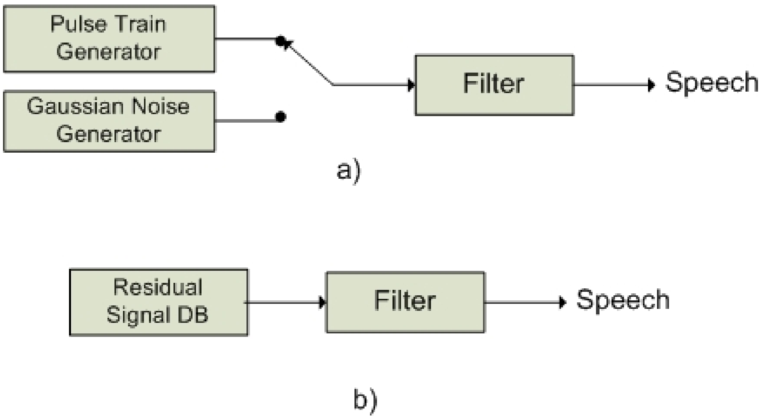


Figure 2.2: The source-filter speech model excitations: a) pulse train/Gaussian noise; b) residual signal.

2.1.2 Formant Synthesis

Formant synthesizers represent the beginning of speech synthesis Electronic Era, with Stewart's circuit, in 1922, and Dudley's Voder, in 1939. Later, in 1953, Lawrence's Parametric Artificial Talker (PAT) was the first parallel formant synthesizer and Fant's Orator Verbis Electris (OVE) was one of the most famous cascade formant synthesizer. Klatt's cascade/parallel synthesizer, in 1981, is one of the most famous digital rule based formant synthesizers. In 1999 Acero proposed a data-driven formant synthesizer using HMMs.

In the formant source-filter model, the filter smoothly varies the formant frequencies, and usually the excitation signal follows the pulse train/ Gaussian noise from the "a)" scheme in figure 2.2. To achieve a good quality at least four formants are needed, but there are systems that use up to six formants.

A formant resonance can be implemented with a second-order Infinite Impulse Response (IIR) filter [Huang, Acero & Hon, 2001]:

$$H_i(z) = \frac{1}{1 - 2e^{-\pi b_i}\cos(2\pi f_i)z^{-1} + e - 2\pi b_i z^{-2}} \tag{2.1}$$

in which $f_i = F_i/F_s$ and $b_i = B_i/F_s$, being F_i the formants' frequencies, B_i the formants' bandwidths and F_s the sampling frequency.

2.1.3 Linear Prediction Coding Synthesis

LPC synthesis, like formant synthesis, is based on the source-filter model of speech, using a recursive filter that predicts the current sample from a finite number of previous samples. The LPC coefficients are estimated using the minimal quadratic error between the current sample and its prediction.

Both the "a)" and "b)" schemes from figure 2.2 are possible excitations for a LPC vocal tract filter. The output signal of the filter is:

$$s\left[n\right] = -\sum_{k=1}^{p} a_k s_q\left[n-k\right] + e_q\left[n\right] \tag{2.2}$$

being S_q the samples, a_k the LPC coefficients and e_q the excitation signal of the filter.

The transfer function of the filter is an all-pole filter:

$$H(z) = \frac{G}{1 + \sum\limits_{k=1}^{p} a_k z^{-k}} \tag{2.3}$$

The LPC coefficients can be calculated using several methods: through the autocorrelation coefficients; the covariance coefficients; or the cepstral coefficients. More about these methods can be found in [Childers, 1999].

Figure 2.3 shows some LPC synthesis results, using different excitation signals for the filter:

- a) represents the original signal to be synthesized;
- b) represents the residual signal, this is the error signal that is obtained through the difference between the original and the predicted signal;
- c) output signal of the filter, using Gaussian noise as the filter excitation;
- d) output signal of the filter, using a fixed period as the filter excitation, correspondent to a 100Hz F0, pulse train;
- e) output signal of the filter, using as the filter excitation a F0 synchronous pulse train;
- f) output signal of the filter, using the residual signal as the filter excitation.

2.1.4 Articulatory Synthesis

The first vocal tract mechanical models can also be considered as the first articulatory models. Dunn's machine, in 1950, was the first electrical analog articulatory model.

Articulatory synthesis uses a model that represents the physical speech production system, including all the articulators and the vocal cords. Articulatory systems model the shape of the human speech production system and not its signal or acoustic features. Their implementation is based on a mathematical-physical model of the vocal tract in which the articulators are modeled by a set of functions corresponding to its small sections' areas.

An articulatory synthesizer [Mermelstein, 1973] is composed of two subsystems:

- The glottal model transforms the articulator positions in a function of the vocal tract section areas in order to their length;

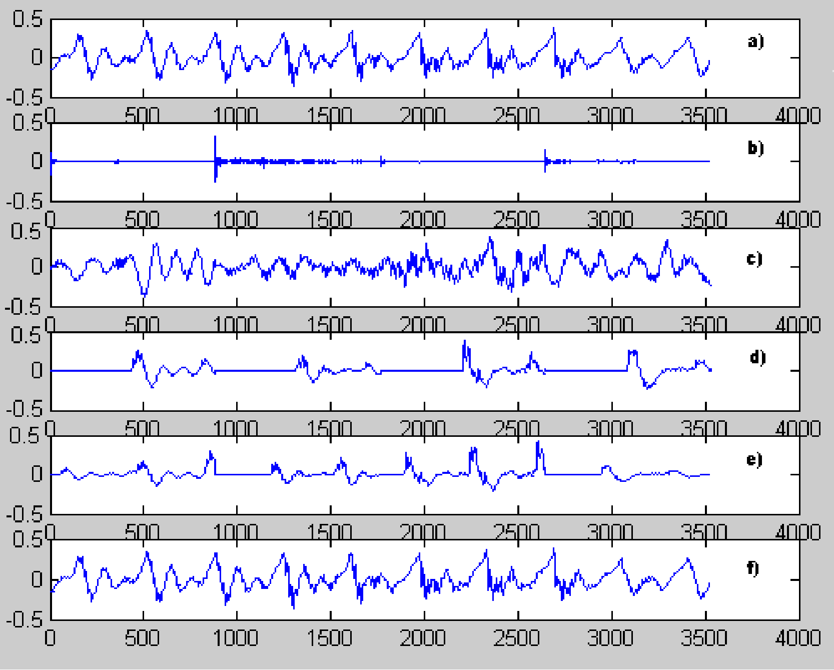


Figure 2.3: LPC synthesis results to different filter excitation signals.

- The articulatory model models the sound propagation in the glottal model structures.

The glottal model is important to the speech production process to model the oral and nasal cavities. The oral cavities are modeled with a shape variable in time and the nasal cavities with a constant shape.

2.2 Concatenative Models

Concatenative synthesis started being a theory in 1958 [Peterson et al., 1958], with Peterson, Wang and Sivertsen's proposal, on the use of diphones as speech units for synthesis.

Concatenative synthesis is an approach that concatenates speech unit waveforms in the time domain in order to obtain synthetic speech. This is what characterizes this approach, because almost all the approaches for speech synthesis use some kind of concatenation, e.g. the source-filter models, like those used for HMM-based or LPC synthesis, where vectors of features corresponding to a speech unit, or a speech frame, are concatenated.

Concatenative systems can simply concatenate the units or make use of some kind of speech signal modification model like OLA, to modify the pitch and improve the overall acceptability.

Concatenative synthesis can be:

- Uniform

  The system only uses a specific speech unit type, that can be phones, phonemes, diphones, triphones, syllables, words, etc...

- Non-uniform

  The system can choose the best output, using different types of units for selection, which can be phones, phonemes, diphones, triphones, syllables, words, etc...

With the statistical approaches and the development of the computational systems, the concatenative systems evolved to unit selection systems, for which a large speech corpus and several realizations per speech unit are used.

In systems where it is possible to choose between several units for the same output target, a discrimination technique is needed. The use of HMM is the most widely used method to select the best units for the synthesis output specification and it was first presented in 1995 [Donovan & Woodland, 1995a].

For generic TTS synthesis, with unlimited vocabulary, it is impossible to cover all possible words and therefore the speech units used by the system have to be a subword unit, like syllable, triphone, diphone or phone. Syllables and triphones are units that are still not practical, because the set of units would be too big to completely cover the vocabulary. Diphones, which are speech segments from the middle of a phoneme to the middle of the next one, are the most common units as they preserve the spectrally unstable parts of the phonemes and consider the neighbor context of the phones.

Usually, the speech segments used in the databases for concatenative systems were manually segmented, however this was very slow and time consuming. With the use of statistical methods in speech, statistical approaches were used for data segmentation systems.

2.3 Unit-Selection based Synthesis

Unit-selection synthesis is based on the concatenation of speech segments, in which the system is not limited to a reduced set of representations for each unit and, in some systems, can even choose among different kinds of units, like phonemes, triphones, syllables, words, etc...

There are two different approaches for the unit selection: the classical cost functions approach and a statistical approach trained with contextual factors and speech parameters.

Unit-selection based synthesis is the approach that provides the greatest naturalness, mainly because it applies only a small amount of signal processing to the recorded speech, with the use of speech signal modification models like overlap-and-add, to no processing at all and just simple unit concatenation. Large unit databases, segmented into individual phones, syllables, words and utterances, is the strength of a unit-selection based synthesis system.

For this synthesis approach there are two important factors regarding the choice of the best units to be used in the synthesis: the choice of each unit regarding its features, the target factor, and the choice of each unit regarding its neighbor units, the concatenation factor.

Usually a multi-dimensional cost function is associated to both of these factors and the process of selecting the best units to use in a unit-selection based synthesis system depends on the so called target cost and concatenation cost.

The notions of target and concatenation costs were first defined in 1996 by Hunt and Black [Hunt & Black, 1996], which became the classical search technique for unit-selection based synthesis.

According to their definition, the target cost is calculated as the weighted sum of the differences between the elements of the target and candidate feature vectors, these differences being the p target sub-costs, $C_j^t(t_i, u_i)(j = 1, ..., p)$:

$$C^t(t_i, u_i) = \sum_{j=1}^{p} w_j^t C_j^t(t_i, u_i) \tag{2.4}$$

and the concatenation cost is determined by the weighted sum of q concatenation sub-costs, $C_j^c(u_{i-1}, u_i)(j = 1, ..., q)$:

$$C^c(u_{i-1}, u_i) = \sum_{j=1}^{q} w_j^c C_j^c(u_{i-1}, u_i) \tag{2.5}$$

The weights, w_j^t and w_j^c, must be found for each feature. This is usually achieved by statistical training and then manual tune.

The sub-costs can be acoustical features, e.g. F0 or duration, or spectral features, e.g. MFCCs or LPC coefficients. They are measured by a spectral distance measure, e.g. Euclidean or Mahalanobis.

The total cost for a sequence of n units is the sum of the target and concatenation costs:

$$\begin{aligned} C(t_1^n, u_1^n) &= \sum_{i=1}^{n} C^t(t_i, u_i) + \sum_{i=2}^{n} C^c(u_{i-1}, u_i) + \\ &\quad + C^c(S, u_1) + C^c(u_n, S) \end{aligned} \tag{2.6}$$

where S is silence and $C^c(S, u_1)$ and $C^c(u_n, S)$ are the start and the end conditions given by the concatenation of the first and last units to silence.

The unit-selection procedure is the task of determining the set of units $\overline{u}_1^n$ that minimizes the total cost, defined by equation (2.7):

$$\overline{u}_1^n = \arg min_{u_1,\ldots u_n} C(t_1^n, u_1^n) \tag{2.7}$$

With the increase of the available computational resources, many systems started using statistical models to evaluate these costs, clustering the units with similar phonetic contexts and using decision trees, in order to handle the large number of units and minimize the problems with the missing ones. The target cost will be automatically given by the choice of the cluster through the decision tree and the concatenation cost will be given by the parameters used for the training of the model.

2.4 Hidden Markov Models based Synthesis

HMM-based speech synthesis is a data-driven approach in which the speech is generated from a concatenation of mathematical models that represent the statistics of the acoustic features of the phonetic units, usually monophones or triphones. The speech parameters, MFCCs, F0 and duration, are generated from HMMs themselves, as explained in section 3.2.1.

The HMM-based speech synthesis uses a source-filter model with the HMM estimated speech parameters to generate the speech. The filter can be a MLSA filter [Fukada et al., 1992], in case of using MFCCs as spectral parameters, as explained in section 4.3.3. The spectral parameters generated from the HMMs are the filter coefficients. The voiced/unvoiced information, together with the F0 values, produce the filter excitation, as it can be seen in scheme “a)” from figure 2.2. Decision trees considering the phonetic, phonological and prosodic contexts are used to cluster the HMMs. An important aspect of the approach is that besides the static features, related to the actual speech segment, it also considers the dynamic ones, from the previous speech segments: the velocity, or the delta coefficients, and the acceleration, or the delta delta coefficients [Tokuda, Kobayashi, & Imai, 1995], as explained in section 3.2.1.

HMM-based systems commonly use the HTS framework [HTS, visited in 2010]. This framework builds context-dependent HMMs of monophones and uses binary decision trees for the MFCCs, F0 and duration densities. A detailed description of this approach is covered in chapter 3 and chapter 4, as it was the approach selected for the work of this thesis.

Chapter 3

Hidden Markov Models Applied to Speech Synthesis

3.1 Motivation for using Hidden Markov Models in Speech Synthesis

This chapter constitutes a survey about the use of HMMs in speech synthesis, as they can be applied to the most prominent speech synthesis approaches: the unit-selection based and the HMM-based synthesis.

In the HMM-based synthesis, during system training, context-dependent monophone or triphone HMMs, usually with 5 states, model the spectral and excitation parameters. During the synthesis process, these parameters are generated for the utterances to be synthesized, from the HMMs themselves. The excitation parameters will be the input of a vocal tract model filter, whose transfer function is given by the spectral parameters.

In unit-selection based synthesis, during system training, context-dependent monophone or triphone HMMs, usually with 5 states, model the acoustic units through their spectral and excitation parameters, which are then indexed, during the synthesis process, by the sequence of HMM states that most probably produced the sequence of observations. The acoustic signals are concatenated with or without some signal modification techniques.

Both techniques use decision trees to choose which HMMs best represent a target speech specification. They use one model per unit, which is usually monophone or triphone. The models representing all the units of the target speech specification are concatenated in a sequence of HMMs.

Besides speech synthesis itself, HMMs can be applied to other tasks from speech synthesis processing, e.g. transcription and segmentation of speech databases; construction of speech segment inventories; and run-time selection of speech segment units.

The use of HMMs in speech synthesis involves two main problems:

- In the training part: Which are the best HMMs to describe the training observations?
- In the synthesis part: Which are the models and the state sequences of those models that could most probably have generated that observation sequence?

The first problem is part of the system training, in which information is extracted from the data and modeled. The second problem is related to the synthesis process, in which the models are used to choose the best units, parametric ones or waveforms, to synthesize a particular utterance.

When using HMMs in speech synthesis, instead of memorizing the data, the model tries to learn the general properties of the data. The main advantage is synthesis results with good prosody, using very small amounts of speech. Another advantage is to have automatic training based on statistics, with models that do not require many parameters. This considerably reduces the memory requirements.

The main disadvantage is the sound quality of the synthesis results, which is still not comparable to the high quality of speech achieved with the “classical” unit-selection based synthesis approach.

3.1.1 Introduction to Hidden Markov Models

The HMM is a statistical method for classifying observed data that constitutes one of the most important statistical methods for modeling speech signals [Huang, Acero & Hon, 2001]. The assumption of the HMM is that the data can be classified as a parametric random process and the parameters of the stochastic process can be estimated in a well defined framework. Being a stochastic process means that all state transitions are probabilistic.

The Markov Chain

A Markov chain models a class of random processes, in which the future states depend only on the present state and a finite number of past states.

A first order discrete-time Markov chain is defined by:

$$P(o_1, o_2, ..., o_n) = P(o_1) \prod_{i=2}^{n} P(o_i|o_{i-1}) \tag{3.1}$$

where $O = o_1, o_2, ..., o_n$ is a sequence of random variables.

Equation 3.1 is based on the Bayes' rule:

$$P(o_1, o_2, ..., o_n) = P(o_1) \prod_{i=2}^{n} P(o_i|o_1, o_2, ..., o_{i-1}) \tag{3.2}$$

having in consideration the Markov assumption that the probability of having an observation at a given time depends only on the observation at the preceding time:

$$P(o_i|o_1, o_2, ..., o_{i-1}) = P(o_i|o_{i-1}) \tag{3.3}$$

If we associate the observation, o_i, to a state, q, the Markov chain can be represented by a finite state process with transition between states specified by the probability function:

$$P(o_i = q_t|o_{i-1} = q_{t-1}) = P(q_t|q_{t-1}) \tag{3.4}$$

and the Markov assumption, represented in equation 3.3, is translated to the probability that the Markov chain is in a particular state at a given time, t, depends only on the state of the Markov chain at the previous time, $t-1$:

$$P(q_t|q_1, q_2, ..., q_{t-1}) = P(q_t|q_{t-1}) \tag{3.5}$$

The parameters of a Markov chain with N states, being q_t the state at time t, are described as:

$$a_{ij} = P(q_t = S_j|q_{t-1} = S_i), 1 \leq i, j \leq N \tag{3.6}$$

$$\pi_i = P(q_0 = S_i), 1 \leq i \leq N \tag{3.7}$$

where a_{ij} is the transition probability from state i to state j;

π_i is the initial probability that the Markov chain will start in state i;

and

$$\sum_{j=1}^{N} a_{ij} = 1, 1 \leq i \leq N; \tag{3.8}$$

$$\sum_{i=}^{N} \pi_i = 1 \tag{3.9}$$

This Markov chain is also called Observable Markov Model, because each state corresponds to an observable event.

Definition of Hidden Markov Model

The Markov chain is a deterministic observable event, because the output of any given state is not random. The HMM is an extension of the Markov chain to a non-deterministic process that generates random output observations at any given time. This means that there are two stochastic processes, as the observation is generated according to a probabilistic function associated with each state. This means that there is no longer a one-to-one correspondence between the observation sequence and the state sequence, so it is not possible to determine the state sequence for a given observation sequence, because the state sequence is hidden.

The specification of a first order HMM is represented by:

$$\lambda =< N, M, \{\pi_i\}, \{a_{ij}\}, \{b_i(k)\} > \tag{3.10}$$

with:

N representing the total number of states;

M representing the size of the observation set;

$\pi = \{\pi_i\}$, representing the initial state distribution, in which π_i is as described in equation (3.7);

$A = \{a_{ij}\}$, representing the transition probability matrix, where a_{ij}, described in equation (3.6), is the probability of having a transition from state i to state j;

$B = \{b_i(k)\}$, representing the output probability matrix, where $b_i(k)$, described in equation (3.11) below, is the probability of emitting k when in state i.

$$b_i(k) = P(o_t = k | q_t = S_i) \tag{3.11}$$

The first order HMM, besides the Markov assumption presented in equation (3.5), follows the assumption of output independency, which states that the probability of a particular output being emitted at time t depends only on the state S_i:

$$P(o_t|o_1, o_2, ..., o_{t-1}, q_1, q_2, ..., q_t) = P(o_t|q_t) \tag{3.12}$$

Basic Hidden Markov Model Operations

When using HMMs to work with a sequence of observations $O =< o_1, o_2, ..., o_T >$, three basic problems may be taken in consideration [Moore, consulted in 2010] [Yamagishi, 2006]:

- Learning HMM: Given a sequence of observations O, what is the maximum likelihood HMM that could have produced this observation sequence?

  The answer to this problem is $\lambda^* = \arg\max_\lambda P(O|\lambda)$.

  The problem is solved by Dynamic Programming, using Baum-Welch Expectation Maximization (EM) algorithm.

- State estimation: Which is the state at time t, $q_t \in \{S_1, S_2, ..., S_N\}$, giving a sequence of observations O?

  The answer to this problem is $P(q_t = S_i|o_1 o_2 ... o_t)$.

  The problem is solved by Dynamic Programming, using the Forward-Backward Algorithm.

- Most Probable Path: Given an observation sequence O, what is the most probable path or sequence of states, $Q =< q_1 q_2 ... q_S >$, S being the number of states, and the probability of that path being taken?

  The answer to this problem is $Q^* = \arg\max_Q P(Q|O)$.

  The problem is solved by Dynamic Programming, using the Viterbi Decoding Algorithm.

So, for an observation sequence $O =< o_1, o_2, ..., o_T >$, there are 3 basic HMM problem operations:

- Learning, solved using Baum-Welch EM Algorithm, to compute $\lambda^* = \arg\max_\lambda P(O|\lambda)$;

- Evaluation, solved using Forward-Backward Algorithm, to calculate $P(q_t = S_i|o_1 o_2 ... o_S)$;
- Inference, solved using Viterbi Decoding Algorithm, to compute $Q^* = \arg\max_Q P(Q|O)$.

In the training stage of speech synthesis we deal with the learning problem. In the synthesis stage we deal with the evaluation and the inference problems to find the best sequence of states for each of the HMMs chosen to represent the target speech specification (the observation sequence).

Speech Synthesis Hidden Markov Model Structures

There are different HMM structures, but the most common ones in speech synthesis are the ergodic model, used in the first approaches with HMMs, as explained in section 3.1.2, and the left-right model, used in today's approaches. Figure 3.1 gives an example of each structure [Yamagishi, 2006].

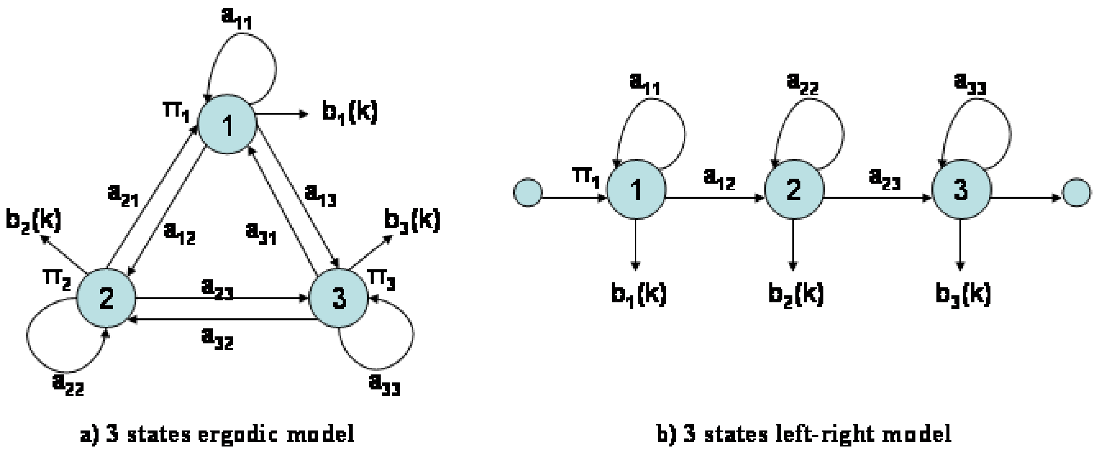


Figure 3.1: Example of common HMM structures.

The ergodic model has state transitions that make it possible to reach any state from any other, in a finite number of transitions.

The left-right model is a model in which the transitions are only allowed from states with lower indices to states with higher ones and self-loops. This model is used for systems with variable properties over time.

For more details on the use of HMMs the tutorial by Rabiner [Rabiner, 1989] is suggested.

3.1.2 History of Hidden Markov Models in Speech Synthesis

Since the late 1980s there have been several approaches trying to use the HMM-based techniques of speech recognition to build TTS systems. The HMMs were applied to different modules of TTS and in particular to the speech synthesis part.

The first attempts to apply HMM-based techniques to speech synthesis evolved from the need of low bit-rate coding for communication purposes and ended up in modern times in a new generation of speech synthesis techniques of choice, with a very reasonable, although vocoder like, quality of synthetic speech. The first approaches were based on a single 64 to 256 states (6-8 bits coding) ergodic HMM, having evolved, almost one decade later, into multiple monophone or triphone left-to-right no-skip HMMs, still used today.

The first reference regarding the use of HMMs in speech synthesis dates back to 1988 [Farges & Clements, 1988]. It is related to Eric Farges' PhD work, presented in 1987, and it represents the first attempt to the concept of speech synthesis from a state sequence, resulting from attempts of continuous speech analysis with a single global ergodic HMM for very-low-bit-rate speech coding systems [Farges & Clements, 1986]. It uses an optimization of the forward-backward algorithm that could accommodate large training databases, with a scaling procedure on its top to eliminate underflow problems and maximum likelihood estimation.

Another HMM approach to speech synthesis, but to synthesize isolated words, appeared in 1989 [Falaschi, Giustiniani, & Verola, 1989], using continuous autoregressive Gaussian output distributions, instead of discrete ones as was done in the previous approach. The model was Viterbi aligned to each word to be synthesized and the state sequence obtained was used to build a smaller left-to-right HMM whose observation vectors were composed by autoregressive Gaussians and speech parameters like energy, voicing and F0. The HMM was then trained on the multiple occurrences of the word to be synthesized and the synthesis of the word was achieved from a sequence of feature vectors calculated from the new HMM mean vectors, using weight functions based on mean state durations to determine the contribution of each state to the feature vector at each point in time [Donovan, 1996].

Using the same HMM method, in 1991, a new system [Giustiniani & Pierucci, 1991] was presented, this time synthesizing speech from a phoneme string specification. The authors submitted a patent for the first statistical approach to speech synthesis, which was patented in 1993 as a phonetic HMM synthesizer [Patent 5230037].

The patented system was based on the interaction of two HMMs, both ergodic and with the same number of states. One had continuous and the other had discrete observation probability functions.

The first HMM models the spectral features, giving the most probable sequence of spectral features corresponding to the input phonetic symbol string. The observation probability functions are continuous Gaussian distributions, giving for each state the probability of the parametric vector being observed in that state. The observed process is the sequence of features extracted from speech and the hidden process is the sequence of states that most probably generated the observed speech sequence. This means that the HMM associates the features of each speech frame to the state or set of states, and therefore the corresponding signal sources that most probably have emitted that signal frame feature. The distance measure used to build the model was the likelihood ratio distortion.

The second HMM models the phoneme sequence, aligning a string of synthetic observations with the phoneme sequence and therefore giving for each state the probability of a phoneme being generated for that state. Giving a string of phonetic symbols, the HMM computes the most probable sequence of labels that constitutes the hidden state sequence. The two HMMs output the most probable sequence of spectral features corresponding to the phonetic symbol string.

In 1994, a system using a global n-gram HMM [Sharman, 1994] was presented, in which each state was associated with an individual phoneme, with discrete output distributions modeling the probabilities of sub-phoneme units being generated by the states.

An n-gram model is a statistical model for predicting the next item in a sequence. The idea of incorporating the n-gram into the HMM structure was to enable the model to be constrained to long state sequences. Each state of the HMM is associated with a sub-phoneme unit and the output distributions model the probabilities of every phoneme being generated by each state [Donovan, 1996].

There are also publications of the use of HMMs in pitch contour generation for speech synthesis, in 1986 [Ljolje & Fallside, 1986] and in 1994 [Fukada et al., 1994]. In 1995 there was a significant change in HMM-based approaches, evolving from a single HMM to multiple language context-based HMMs. Two implementations were presented.

One was introduced by Donovan and Woodland [Donovan & Woodland, 1995b], with some improvements presented in the same year [Donovan & Woodland, 1995a]. This system automated the construction of an acoustic inventory using decision-tree state-clustered triphone HMMs using single Gaussian distributions, in which synthesis was achieved by concatenating representations of the clustered states representing sub-words units. The states were clustered using a set of automatically generated decision trees that enabled HMMs to be constructed for all possible triphones, even those not present in the training database. Each state probability was based on the average energy per sample, the average zero crossing rate of all the speech occurrences in its cluster and the LP coefficients. To obtain the LP coefficients several models were proposed. The voiced/unvoiced decision to determine the excitation signal to be used in the synthesis was based on the average zero crossing rate for each state.

The other approach was introduced by Keiichi Tokuda and colleagues [Tokuda, Kobayashi, & Imai, 1995], also using context-based multiple HMMs. The highlight that distinguished it from the others was that both static and dynamic features were taken into account for speech parameter generation from HMMs. Including the dynamic features proved to be essential for the smoothness of synthetic continuous speech, resulting in the search for the optimum state sequence and in solving a set of linear equations for each possible state sequence, without which the generated parameter sequence becomes a sequence of the mean vectors independently of contexts. The models were continuous Gaussian single mixture HMMs, with 3-states, left-to-right and no-skips. The spectral coefficients used were the MFCCs and the dynamic features were given by their movements, delta MFCC, as it was used in speech recognition before by Furui [Furui, 1986], from where the idea of this approach arose. Still in the same year, the accelerations, delta delta MFCCs, were also included as dynamic features [Tokuda et al., 1995]. This approach has been the basis to a speech synthesis technique that is today considered as a new speech synthesis generation.

Still in 1996, an HMM-based speech synthesis system with various voice characteristics was presented [Masuko, 1996]. In fact, the flexibility of HMM-based synthesis in voice characteristics modification has been one of the major reasons for the research interest in this type of systems, one decade later.

Also in 1996, Huang and colleagues from Microsoft Research presented Whistler [Huang et al., 1996], another trainable HMM-based TTS system that models speech parameters in order to automatically generate speech. The prosody model only used pitch as a parameter, but due to the rich context-dependent units the default amplitudes and durations resulted in fairly natural speech. The acoustic speech representations were the pitch-synchronous LPC parameters and their residual waveforms. Voicing and epoch detection was achieved without manual intervention, using a laryngograph signal that measured the movement of the vocal cords directly. The system used decision tree based senones as the synthesis units. A senone is described to be a context-dependent sub phonetic unit which is equivalent to a HMM state in a triphone. The senone decision trees are generated automatically from the database according to the minimum entropy (or within-unit distortion). The use of decision trees resolves the contexts not seen in the training data, based on phonetic categories of neighboring contexts, while providing detailed models for contexts that are represented in the database, as it was proposed in the previous systems [Donovan & Woodland, 1995b] [Tokuda, Kobayashi, & Imai, 1995].

Acero, in 1999, also applied HMMs to formant synthesis [Acero, 1999]. In 2002, Tokuda and his working group released the HTS [HTS, visited in 2010], a free open source engine for speech synthesis based on HMMs. The highlights of the system are [Yoshimura, 1999]:

- Spectrum, pitch and state durations are modeled simultaneously in a unified framework of HMMs;
- Pitch is modeled by MSD-HMMs;
- State duration is modeled by multidimensional Gaussian distributions;
- The distributions for spectral parameters, pitch parameter and the state duration are clustered independently by using a decision-tree based context clustering technique;
- Synthetic speech is generated by using a speech parameter generation algorithm from HMM and a mel-cepstral based vocoding technique.

The HTS was the framework chosen to implement the TTS synthesis engine presented in chapter 4.

3.2 Main Techniques for using Hidden Markov Models in Speech Synthesis

There are some basic techniques that are responsible for the success of the use of HMMs in speech synthesis and will be explained in this section. The speech parameter generation using dynamic features, which revealed to be essential for the use of HMMs in speech synthesis when using the HMM-based synthesis approach, is one example of these techniques. The decision tree based context clustering of HMM states is also relevant for the success of HMM-based synthesis results. Another important technique is the use of MSD HMMs for F0 modeling.

There is another technique that is important for the success of the results when using MFCCs, the mel-cepstral analysis, which is used as a speech feature analysis technique.

3.2.1 Speech Parameter Generation Algorithm

The use of dynamic features was the most important factor for improving the results of the speech parameters generated by HMMs. Tokuda et al. brought this method to speech synthesis [Tokuda, Kobayashi, & Imai, 1995], from Furui's work in speech recognition [Furui, 1986], in which dynamic spectral features were used for the recognition of isolated words with good improvements over the previous results.

The speech parameter sequence is generated from HMMs whose observation vector consists of a spectral parameter vector and its dynamic feature vectors. It is assumed that the state sequence, or part of the state sequence, is unobservable. The algorithm iterates the forward-backward algorithm and the parameter generation algorithm for the case where a state sequence is given [Tokuda et al., 2000a]. The phoneme durations are given from the phoneme duration densities.

The quality of the synthetic speech is considerably improved by increasing mixtures, because multi-mixture HMMs give a clear formant structure. With the use of single-mixture HMMs, the formant structure of the spectrum corresponding to each mean vector $\mu_{q,i}$ might be vague since $\mu_{q,i}$ is the average of different speech spectra.

The procedure used to generate the speech parameters from HMMs is described [Tokuda, consulted in 2009] in the following:

For each given HMM λ the speech parameter vector sequence, or the observations vector, $O =< o_1, o_2, ..., o_T >$, has to be determined, which maximizes:

$$\begin{aligned} P(O|\lambda, T) &= \sum_Q P(O|Q, \lambda, T) P(Q|\lambda, T) \\ &\simeq \max_Q P(O|Q, \lambda, T) P(Q|\lambda, T) \end{aligned} \tag{3.13}$$

where:

$$Q_{\max} = \arg\max_Q P(Q|\lambda, T)$$

$$O_{\max} = \arg\max_O P(O|Q_{\max}, \lambda, T)$$

In order to obtain the best Q it is needed to maximize $P(Q|\lambda, T)$:

$$P(Q|\lambda, T) = \prod_{i=1}^{K} p_i(d_i) \tag{3.14}$$

where:

$p_i(d_i)$ has exponential distribution and it is a Gaussian defined by its mean m_i and variance σ_i^2

and being:

$$(d_i) = m_i + \rho \cdot \sigma_i^2$$

$$\rho = (T - \sum_{k=1}^{K} m_i)$$

Then, in order to obtain the best O it is required to maximize $P(O|Q_{\max}, \lambda, T)$.

The first systems did not use dynamic features, but O would become a sequence of mean vectors, producing flat values with discontinuities at the state boundaries, because the parameters were being determined without considering the parameters from the preceding and succeeding frames. The integration of dynamic features solved this problem [Tokuda, Kobayashi, & Imai, 1995].

Without the use of dynamic features, the observation sequence is composed by a vector of parameters, usually MFCC: $o_t = [c_t^\top]^\top$, being $O = [o_1^\top, o_2^\top, ..., o_T^\top]^\top$.

In order to obtain O it is needed to maximize:

$$P(O|Q_{\max}, \lambda, T) = \prod_{t=1}^{T} b_{q_t}(O_t) \tag{3.15}$$

$$O_t = \arg\max_{O} b_{q_t}(O), t = 1, 2, ..., T \tag{3.16}$$

With the integration of dynamic features, each observation sequence is composed by a vector of vectors of parameters, usually called a set of streams.

$O = [o_1^\top, o_2^\top, ..., o_T^\top]^\top$, where $o_t = [c_t^\top, \Delta c_t^\top, \Delta^2 c_t^\top]^\top$.

The delta and delta delta coefficients are respectively:

$$\begin{aligned}\Delta c_t &= \sum_{\tau=-L_-^{(1)}}^{L_+^{(1)}} \omega^{(1)}(\tau)c_{t+\tau} \\ &= \frac{1}{2}(c_{t+1} - c_{t-1})\end{aligned} \tag{3.17}$$

$$\begin{aligned}\Delta^2 c_t &= \sum_{\tau=-L_-^{(2)}}^{L_+^{(2)}} \omega^{(2)}(\tau)c_{t+\tau} \\ &= \frac{1}{2}(\Delta c_{t+1} - \Delta c_{t-1}) \\ &= \frac{1}{4}(\Delta c_{t+2} - 2c_t + \Delta c_{t-2})\end{aligned} \tag{3.18}$$

where $L_-^{(0)} = L_+^{(0)} = 0$ and $\omega_0^{(0)} = 1$.

Considering that $O = WC$, where:

$C = [c_1^\top, c_2^\top, ..., c_T^\top]^\top$

$W = [\omega_1, \omega_2, ..., \omega_T]^\top$ with $\omega_t = [\omega_t^{(0)}, \omega_t^{(1)}, \omega_t^{(2)}]$

it is possible to obtain the best O by obtaining the best C:

$$P(O|Q_{\max}, \lambda, T) = P(WC|Q_{\max}, \lambda, T) \tag{3.19}$$

This achieved by setting

$$\frac{\partial}{\partial C} \log P(O|Q_{\max}, \lambda, T) = 0 \tag{3.20}$$

to obtain:

$$W^\top U^{-1} W C = W^\top U^{-1} M \tag{3.21}$$

being:

$$U^{-1} = diag[U_{q_1}^{-1}, U_{q_2}^{-1}, ..., U_{q_T}^{-1}] \tag{3.22}$$

$$M = [\mu_{q_1}^\top, \mu_{q_2}^\top, ..., \mu_{q_T}^\top]^\top \tag{3.23}$$

the covariance matrix and the mean vector of the mixture component of state q_t, respectively.

The equation 3.21 gives the speech parameters vector C that maximizes $P(O|Q_{\max}, \lambda, T)$.

3.2.2 Decision Tree Based Context Clustering

There are many contextual factors that cause acoustic variation in phonemes by affecting the spectrum, F0 and duration. A context-clustering technique must be used to overcome the impossibility of designing a corpus that covers all possible unit contexts and all the variations in acoustic parameters like F0, duration, or energy.

Some techniques were proposed to cluster HMM states and share model parameters among states in each cluster, two of the most widely used ones being data-driven clustering and the decision tree based-context clustering [Masuko, 2002].

Data-Driven Clustering

There are several data-driven approaches to cluster HMM states, but the one that went furthest is the neighbor hierarchical clustering algorithm.

The neighbor hierarchical clustering algorithm consists of the training of a set of context-dependent HMMs with single Gaussian output distributions, in which all states at the same position of HMMs are gathered and placed in their own individual clusters. Distances between two clusters are then calculated for all combinations of two clusters.

The distance between two clusters is defined as the maximum distance between any distributions in the two clusters, and the distance $D(i, j)$ between distributions i and j is calculated using:

$$D(i,j) = [\frac{1}{d}\sum_{k=1}^{d}\frac{(\mu_{ik}-\mu_{jk})^2}{\sigma_{ik}\sigma_{jk}}]\frac{1}{2} \tag{3.24}$$

where d is the dimensionality of the observation and μ_{sk} and σ_{sk} are the mean and variance of the k^{th} dimension of the Gaussian distribution s.

The clusters which have minimum distance are merged and all states at the same position of HMMs are re-gathered and replaced in their own individual clusters again, with the distances between two of all the new clusters being re-calculated for all combinations of two clusters again, until the minimum distance exceeds a threshold [Masuko, 2002].

Decision Tree Clustering

A decision tree is a binary tree, meaning that there are always two different branches to choose, depending on the decision of a statement being true or false, or being yes or no, until reaching the leaf nodes that have output distributions for the corresponding cluster.

The statements are context related questions, e.g., "Is the previous phoneme a vowel or the next phoneme a consonant?"; "Is the next consonant a fricative?".

Using decision tree based context clustering, it is possible to model parameters for unseen contexts, because any context reaches one of the leaf nodes.

To construct a decision tree, a set of context-dependent HMMs with single Gaussian output distributions has to be trained and all states that are to be clustered have to be gathered and placed in the root node of the tree. For each leaf node, a statement is chosen that gives maximum increase in log-likelihood when splitting the leaf node in two. The choice of the statements is language dependent and very important for the quality of the models. Among all leaf nodes, a node which give maximum increase in log-likelihood is selected and split into two using this statement. These steps are repeated until this increase falls below a threshold [Masuko, 2002].

Some examples of parts of decision trees from the implemented system are shown in section C.3, from appendix C, for duration, F0 and MFCCs, in figures C.1, C.2 and C.3, respectively.

3.2.3 Multi-space Probability Distribution Hidden Markov Model

The MSD-HMM is a new type of HMM [Tokuda et al., 2002] based on MSD in order to overcome the limitations of discrete and continuous HMMs when modeling observation sequences like F0 patterns that consist of continuous values and discrete symbols, simultaneously.

The F0 patterns are composed of one-dimensional continuous values for voiced regions and discrete symbols to represent the unvoiced regions of the phones. The MSD-HMM is able to model a sequence of observation vectors with variable dimensionality, including zero-dimensional observations for the discrete symbols. This new method allows us to model and generate the spectral features and F0 patterns in a unified HMM framework.

The output probability in each state of an MSD-HMM is given by the MSD. An N-state MSD-HMM λ is specified by the initial state probability distribution

$\pi = \{\pi_j\}_{j=1}^{N}$, the state transition probability distribution $A = \{a_{ij}\}_{i,j=1}^{N}$, and the state output probability distribution $B = \{b_i(\cdot)\}_{i=1}^{N}$, where

$$b_i(O) = \sum_{g \in S(O)} \omega_{ig}(V(O)) \tag{3.25}$$

Each state i has G PDFs $\mathcal{N}_{i1}(\cdot), \mathcal{N}_{i2}(\cdot), \cdots, \mathcal{N}_{iG}(\cdot)$ and their weights $\omega_{i1}, \omega_{i2}, \cdots, \omega_{iG}$, where $\sum_{g=1}^{G} \omega_{ig} = 1$. The observation probability of $O = \{o_1, o_2, \cdots, o_T\}$ is:

$$\begin{aligned} P(O|\lambda) &= \sum_{allq} \prod_{t=1}^{T} A_{q_{t-1}q_t} b_{q_t}(O_t) \qquad (3.26) \\ &= \sum_{allq,l} \prod_{t=1}^{T} a_{q_{t-1}q_t} \omega_{q_t l_t} \mathcal{N}_{q_t l_t}(V(O_t) \end{aligned}$$

where $q = \{q_1, q_2, \cdots, q_T\}$ is a possible state sequence, $l = \{l_1, l_2, \cdots, l_T\} \in \{S(O_1) \times S(O_2) \times \cdots \times S(O_T)\}$ is a sequence of space indices which is possible for the observation sequence O, and $a_{q_0 j}$ denotes π_j [Masuko, 2002].

Equation 3.26 can be calculated efficiently through the forward-backward procedure, using:

$$\alpha_t(i) = P(O_1, O_2, ..., O_t, q_t = i|\lambda) \tag{3.27}$$

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, ..., O_T, q_t = i|\lambda) \tag{3.28}$$

3.2.4 Mel-Cepstral Analysis Technique

The spectrum that is represented by the MFCCs has a frequency resolution similar to that of the human ear which has high resolution at low frequencies (from G.Fant, “Speech sound and features”) [Fukada et al., 1992].

In the mel cepstral analysis, the spectrum of the speech signal is modeled by the M^{th} order MFCC, $\tilde{c}(m)$, as in equation 3.29:

$$H(z) = \exp \sum_{m=0}^{M} \tilde{c}(m)\tilde{z}^{-m} \tag{3.29}$$

where $\tilde{z}^{-1}$ is a first order all-pass transfer function:

$$\tilde{z}^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}, \mid \alpha \mid< 1 \tag{3.30}$$

Its phase response gives the frequency scale, $\tilde{\omega}$:

$$\tilde{\omega} = \tan^{-1} \frac{(1-\alpha^2)\sin\omega}{(1+\alpha^2)\cos\omega - 2\alpha} \tag{3.31}$$

The phase response $\tilde{\omega}$ will be a good approximation to the auditory frequency scale if α is well chosen. For instance, when using mel scale with a sampling frequency of 16KHz, the ideal α is 0,42. Table 4.10, from section 4.3.2, presents some values that are suggested to better approximate the mel-scale and other scales to the human auditory frequency scales.

The mel-cepstral analysis method is efficient for the estimation of spectra which have resonances and anti-resonances at low frequencies.

In order to obtain an unbiased estimate of log spectrum, the following criterion is minimized:

$$E = \frac{1}{2\pi}\int_{-\pi}^{\pi} \{\exp R(\omega) - R(\omega) - 1\} d\omega \tag{3.32}$$

where:

$$R(\omega) = \log I_N(\omega) - \log |H(e^{jw})|^2 \tag{3.33}$$

and $I_N(\omega)$ is the modified periodogram of a weakly stationary process $x(n)$ with a time window $\omega(n)$ of length N:

$$I_N(\omega) = \frac{\mid \sum_{n=0}^{N-1} \omega(n)x(n) \exp^{-j\omega n} \mid^2}{\sum_{n=0}^{N-1} \omega^2(n)} \tag{3.34}$$

Equation (3.29) is rewritten as

$$H(z) = \exp \sum_{m=0}^{M} b(m)\Phi_m(z) = K \cdot D(z) \tag{3.35}$$

where $K = \exp b(0)$ is the gain factor;

$$D(z) = \exp \sum_{m=1}^{M} b(m)\Phi_m(z) \tag{3.36}$$

$$\Phi_m(z) = \begin{cases} 1 & \text{m} = 0 \\ \frac{(1-\alpha^2)z^{-1}}{1-\alpha z^{-1}} \tilde{z}^{-(m-1)} & \text{m} \geq 1 \end{cases} \tag{3.37}$$

and the relationship between the coefficients $c(m)$ and $b(m)$ is:

$$\tilde{c}(m) = \begin{cases} b(m) & \text{m = M} \\ b(m) + \alpha b(m+1) & 0 \leq \text{m ¡ M} \end{cases} \tag{3.38}$$

Because $H(z)$ is considered to be a synthesis filter of speech, $D(z)$ must be stable and so assumed to be a minimum phase system yielding the relationship:

$$E = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(e^{j\omega})|^2 d\omega = \log k^2 \tag{3.39}$$

Using this assumption on the minimization of the spectral criterium E, equation 3.32, becomes:

$$E = \frac{\varepsilon}{k^2} - \frac{1}{2\pi} \int_{-\pi}^{\pi} \log I_N(\omega) d\omega + \log k^2 - 1 \tag{3.40}$$

where:

$$\varepsilon = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{I_N(\omega)}{|D(e^{j\omega})|^2} d\omega \tag{3.41}$$

The minimization of E with respect to c leads to the minimization of ε with respect to c_1 and the minimization of E with respect to k [Masuko, 2002].

The gain factor K that minimizes E is obtained by setting $\partial E / \partial K = 0$:

$$K = \sqrt{\varepsilon_{\min}} \tag{3.42}$$

where $\varepsilon_{\min}$ is the minimized value of ε.

In this chapter it is explained how to apply HMM to speech synthesis and its motivation. The techniques described in the chapter: speech parameter generation algorithm, decision tree based context clustering, multi-space probability distribution hidden Markov model, and mel-cepstral analysis, are included in the HTS framework [HTS, visited in 2010] selected to implement the speech synthesis system of the thesis.

Chapter 4

European Portuguese Text-to-Speech System

TTS systems are systems that convert text into speech, meaning that these systems receive text as input and read it, producing voice as output. A general TTS system is composed of different modules, from the pre-processing of the text to be synthesized, where the text that is in the form of acronyms, abbreviations, numerals, dates, etc. is converted in its extended, plain text form, to its audio signal generation, where the text is transformed into voice and spoken out. A TTS system does not have to contain all the modules from the general system. Many systems do not use linguistic analysis, others do not need text pre-processing because the text is already formatted and others do not use prosodic modulation. Figure 4.1 presents a general TTS system and each of its modules is described in the following.

Linguistic Text Analysis

The TTS system presented in figure 4.1 receives the text to be synthesized and labels it with several text characteristics and parsing information, from the utterance level to the phone level. There are different types of text pre-processing tasks that can be used, like for instance, short-length text forms conversion into a readable full-length text form, like dates, phone numbers, numerals, abbreviations, acronyms and other symbols. There are also several levels of linguistic text analysis that can be used in a TTS system. For instance a morpho-syntactic analysis can be very helpful for homograph disambiguation and for an adequate prosodic manipulation. The linguistic text analysis also involves the NLP tasks, which convert the input text into its correspondent phonetic transcription, using a G2P converter, and add other useful information like word and syllable boundaries detection, and word stress prediction.

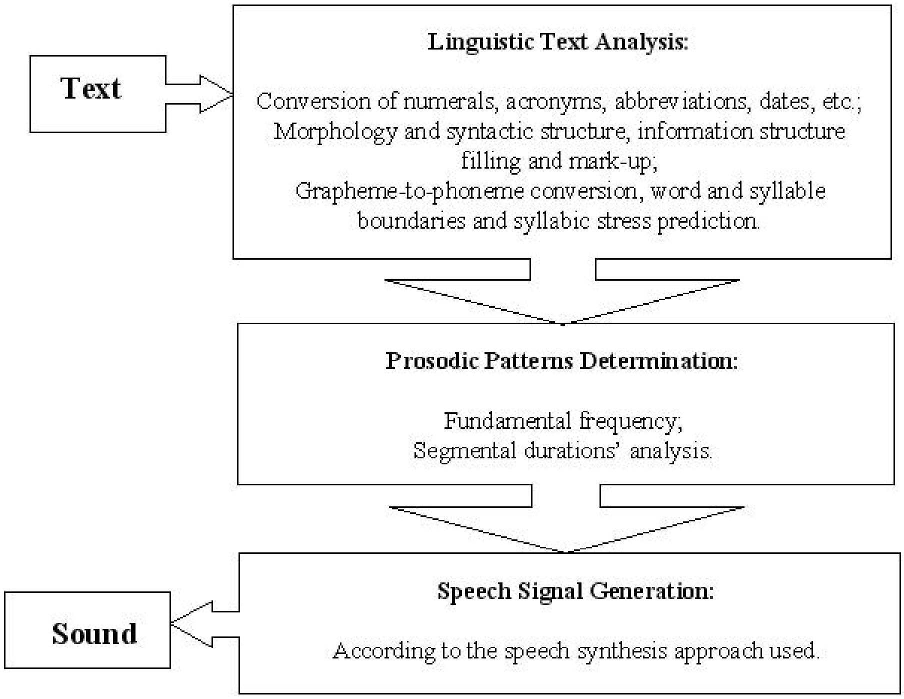

Text
Linguistic Text Analysis:
Conversion of numerals, acronyms, abbreviations, dates, etc.;
Morphology and syntactic structure, information structure filling and mark-up;
Grapheme-to-phoneme conversion, word and syllable boundaries and syllabic stress prediction.
Prosodic Patterns Determination:
Fundamental frequency;
Segmental durations' analysis.
Speech Signal Generation:
According to the speech synthesis approach used.
Sound

Figure 4.1: General TTS system architecture.

Prosodic Patterns Determination

This module estimates prosodic information, like F0 and segmental duration, to produce the right prosody for the synthetic speech.

Speech Signal Generation

This module processes the speech signal according to one of the speech synthesis approaches presented in chapter 2. The produced speech signal is then buffered and delivered to the operating system sound sub-system.

The TTS implemented in this thesis is based on statistical methods. The synthesis system is HMM-based and the language dependent model is implemented for EP. The HMM-based synthesis was originally tested in the speech synthesis area for the Japanese language [Yoshimura, 1999] and meanwhile implemented for other languages [Black, Zen & Tokuda, 2007] like English [Tokuda, Zen & Black, 2002], Brazilian Portuguese (BP) [Maia et al., 2003], German [Weiss et al., 2005], Mandarin, Korean, Swedish, Finnish, Slovenian, Croatian, Arabic, Farsi and now EP [Barros et al., 2005]. Figure 4.2 presents the main modules of a HMM-based synthesis system, which are then briefly explained.

Contextual Label Generation

According to the utterance information of the input text to be synthesized, contextual labels that represent HMM units in the database are generated. These contextual labels are generated in a phone by phone basis.

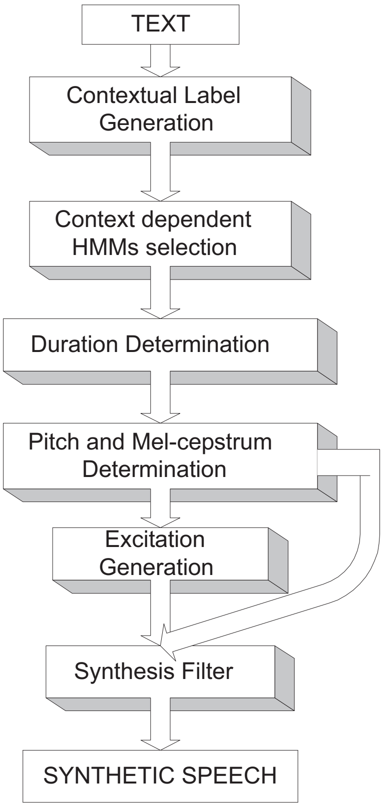

TEXT
Contextual Label Generation
Context dependent HMMs selection
Duration Determination
Pitch and Mel-cepstrum Determination
Excitation Generation
Synthesis Filter
SYNTHETIC SPEECH

Figure 4.2: HMM Based TTS System

Contextual HMM selection and concatenation model

Decision trees, created from contextual information, are used to decide which HMMs are going to be selected to represent each of the speech units. These HMMs are then concatenated in a sentence of HMMs and used for generating the MFCCs, logF0 and state durations.

Duration Determination

The state durations are determined according to multi-dimensional Gaussian distributions [Yoshimura, 1999]. The distribution's dimensionality is equal to the number of states of the corresponding HMM, where the n^{th} state duration density dimension corresponds to the n^{th} respective HMM state.

LogF0 and Mel-Cepstral Coefficients Determination

The logF0s and MFCCs are modeled in accordance with MSD [Tokuda et al., 2002] and multivariate Gaussian distribution, respectively. These parameters are then determined through an algorithm for speech parameter generation from multi-mixture continuous HMMs [Tokuda et al., 2000a].

Excitation Generation

A source-filter model is used, implemented with a MLSA filter. The excitation signal, the input of the MLSA filter, is either a pulse train or random noise, for voiced and unvoiced segments respectively, based on the determined pitch sequence. The transfer function of the filter is based on the MFCCs.

Synthesis Filter

The synthesized speech waveform is generated through the MLSA filter. The vocal tract is modeled by the generated MFCC sequence that will be the transfer function of the filter. The source of the filter is obtained by the generated logF0 sequence, as explained in section 2.1.1.

4.1 Language Dependent Module

4.1.1 European Portuguese Language

Portuguese is an Iberian-Romance language derived from Latin. It is the seventh most used language in the world regarding the number of native speakers, with around 178 million native speakers and it is the second most spoken Latin language [Wikipedia, visited in 2010].

Portuguese language is spoken in Angola, Brazil, Cape Verde, China (Macau), Guinea-Bissau, Indian Union (Daman, Diu and Goa), Indonesia (Flores Island), Macaw, Malaysia (Mallaca), Mozambique, Portugal, Sao Tome & Principe, Timor Lorosa´e and Uruguay. It is the official language in eight countries: Angola, Brazil, Cape Verde, Guinea-Bissau, Mozambique, Portugal, Sao Tome & Principe and Timor Lorosa´e. And it is largely used in many others.

There are several different Portuguese dialects which differ from each other, not only in their grammar constructions and vocabulary, but also in their prosody and even in some phonetic units. For instance, between EP and BP the dialectal variations do not have academic or literary importance, nor do they compromise the intelligibility, as is the case between EP and African dialects.

The Portuguese alphabet consists of the original Latin alphabet, with 23 letters. The letters <k>, <y> and <w> do not belong to it, although they can appear in some foreign words imported to the Portuguese vocabulary.

The EP language in continuous speech suffers effects of coarticulation between words and natural vocalic reduction. Coarticulation between words means that the words' pronunciation is influenced by the neighboring words. The natural vocalic reduction, also called Sandhi effect [Braga, Freitas & Ferrreira, 2003] [Amaral et al., 1999], means that some phonemes are reduced or even suppressed in continuous speech.

The EP phonetic inventory has thirty-eight phonemes, as is summarized in Table 4.1, both in Computer Readable Phonetic Alphabet (SAMPA) and International Phonetic Alphabet (IPA) alphabets.

From the thirty-eight EP phonemes, thirty-two are voiced, meaning that during their production there is oscillation of the vocal cords, and six are unvoiced, meaning that there is no use of the vocal cords during their production. Table 4.2 presents the voiced and unvoiced EP phonemes.

Twenty-four EP phonemes are continuant, meaning that during their production there is a incomplete closure of the vocal tract, and fourteen are non-continuant, meaning that they are produced with a complete closure of the vocal tract. Table 4.3 presents the continuant and non-continuant EP phonemes.

The Vowels

The Portuguese language has five graphemes corresponding to vowel sounds: <a>, <e>, <i>, <o> and <u>. There are four types of accents that can be used with some of the vowels, but there is no accent used with consonants. The acute accent, < ´ >, used to open a vowel, can be used with <a>, <e>, <i>, <o> and <u>: <á>, <é>, <í>, <ó> and <ú>. The grave accent, < ` >, is an accent that is only used with <a>: <à>, and it is used for the contraction of the preposition "a" with the definite article "a" or a pronoun like "aquele" (meaning *"that one"* for the masculine case), "aquela" (meaning *"that one"* for the feminine case), or "aquilo" (meaning *"that one"* for the undetermined case). The circumflex accent, < ˆ >, used to close a vowel, can be used with <a>, <e> and <o>: <â>, <ê> and <ô>. The tilde, < ˜ >, used to nasalize a vowel, can be used with <a> and <o>: <ã> and <õ>.

From the fourteen EP phonetic vowels, /a/, /6/, /E/, /O/, /e/, /@/, /i/, /o/, /u/, /6∼/, /e∼/, /i∼/, /o∼/ and /u∼/, nine are oral vowels, meaning that during their production the air escapes only through the mouth: /a/, /6/, /E/, /O/, /e/, /@/, /i/, /o/ and /u/, and five are nasal vowels, meaning that during their production the air escapes both through the mouth and the nose: /6∼/, /e∼/, /i∼/, /o∼/ and /u∼/. There are four semivowels, also known as glides, which are vowel-like sounds with short duration and rapidly change from one position of articulation to another: /j/, /w/, /j∼/ and /w∼/, from which two are oral: /j/ and /w/, and two are nasal: /j∼/ and /w∼/.

Regarding the backness, referring to the position of the tongue relative to the back of the mouth, the EP vowels are distributed in five front, or anterior, vowels, which are those produced with the tongue positioned forward in the mouth: /E/,

/e/, /i/, /e∼/ and /i∼/, four central vowels, which are those produced with the tongue positioned in the center of the mouth: /@/, /a/, /6/ and /6∼/, and five back, or posterior, vowels, which are those produced with the tongue positioned back in the mouth: /O/, /o/, /u/, /o∼/ and /u∼/.

Regarding the height, referring to the vertical position of the tongue relative to either the mouth ceiling or the lower jaw, five are high vowels, which are those produced with the tongue positioned close to the mouth ceiling: /@/, /i/, /u/, /i∼/ and /u∼/, four are mid vowels, which are those produced with the tongue positioned neither too close to the mouth ceiling nor to the lower jaw, more in the middle of both: /e/, /o/, /e∼/ and /o∼/, and five are low vowels, which are those produced with the tongue positioned far from the mouth ceiling and closer to the lower jaw: /a/, /E/, /O/, /6/ and /6∼/.

Regarding the roundedness, referring to whether the lips are rounded or not, EP has five rounded vowels, which are all the back vowels: /O/, /o/, /u/, /o∼/ and /u∼/. All the others are non-rounded vowels.

The most complex vowels regarding their phonetic transcription are the ones represented by the graphemes <e> and <o>. There are about thirty grammatical rules for each of these graphemes when only considering the transcription of isolated words, and even more rules regarding the coarticulation effects between words.

The grapheme <e> has eleven possible transcriptions: /E/, /e/, /@/, /6/, /i/, /j/, /6j/, /e∼/, /6∼j∼/, /6∼j∼6∼j∼/ and “&”. The transcription as /6j/ appears in some situations of <e> followed by <x> preceding a consonant, like in a non-stressed syllable at the beginning of a word, or in a stressed syllable. The nasal vowels appear when the <e> is positioned in contexts that make it nasal. Some examples of these situations appear when <e> is followed by <m,n>. In these cases it is usually transcribed as /6∼j∼/, but there are situations where it is transcribed as /e∼/. Another example of context that makes the <e> nasal is when <e> has a circumflex accent and is followed by <m>. In this case it is transcribed as /6∼j∼6∼j∼/. The “&” represents situations of phonetic suppression, as explained in section 4.1.1.

The grapheme <o> has seven possible transcriptions: /o/, /O/, /u/, /w/, /o∼/, /o∼j∼/ and /w∼/. The nasal vowels appear when the <o> has a tilde accent or when it is positioned in contexts that make it nasal, like being followed by <m,n>.

The <a> is a grapheme that, although not so complex as the <o> and the <e>, still has three phonetic transcriptions: /a/, /6/ and /6∼/. The nasal vowel appears when the <a> has a tilde accent or else it is positioned in contexts that make it nasal, like being followed by <m,n>.

The <u> when phonetically transcribed corresponds to the vowel /u/ and semivowel /w/, and also to the phonetic nasal vowel /u∼/ and semivowel /w∼/ when it is positioned in contexts that make it nasal, like being followed by <m,n>. When between a <(q,g)> and a <(e,i)>, the <u> is mute.

The <i> corresponds to the vowels /i/ and /i∼/, the semivowels /j/ and /j∼/ and, in particular cases, to the vowel /@/. One of the cases where <i> is transcribed as a /@/ is when it appears in a sequence of two syllables with the vowel <i>, as in <feminino>, which is transcribed as /f@m@ninu/. The nasal vowel appears when the <i> is positioned in contexts that make it nasal, like being followed by <m,n>.

The Consonants

From the 18 graphemes existing for Portuguese corresponding to consonant sounds there are some particular cases that must be referred to.

There are no accents that can be used with consonants and the <c> is the only grapheme in Portuguese that can appear with a cedilla, < ¸ >. The <ç> is always transcribed as /s/ and never precedes a <e> or a <i>.

The <h> is a grapheme correspondent to a consonant that by itself does not have a transcription, as its aspiration was lost with the evolution of the language. When <h> is preceded by one of the graphemes <c>, <n> or <l> forms the combinations <ch>, <nh> and <lh>, which have the individual consonant phonetic transcriptions, /S/, /J/ and /L/, respectively.

The grapheme <q> is always transcribed as /k/ and only appears before the grapheme <u>, without exceptions. It can appear at the beginning or in the middle of a word.

The <m> and the <n> between a vowel and a consonant, or the <m> at the end of a word, do not have a phonetic transcription, but instead they nasalize the precedent vowel.

There are twenty EP phonetic consonants: /p/, /b/, /t/, /d/, /k/, /g/, /m/, /n/, /J/, /s/, /S/, /z/, /f/, /v/, /Z/, /l/, /l∼/, /L/, /R/ and /r/. From those, fourteen are voiced, meaning that during their production there is oscillation of the vocal cords: /b/, /d/, /g/, /m/, /n/, /J/, /z/, /v/, /Z/, /l/, /l∼/, /L/, /r/ and /R/, and six are unvoiced, meaning that there is no use of the vocal cords during their production: /p/, /t/, /k/, /s/, /S/ and /f/.

There are seventeen oral consonants, meaning that during their production the air escapes only through the mouth: /p/, /b/, /t/, /d/, /k/, /g/, /s/, /S/, /z/, /f/, /v/, /Z/, /l/, /l∼/, /L/, /R/ and /r/, and three nasal consonants, meaning that during their production the air escapes both through the mouth and the nose: /m/, /n/ and /J/.

Regarding the articulation manner, in the EP there are nine stops, meaning that during the production of these consonants there is an interruption of the airflow: /p/, /b/, /t/, /d/, /k/, /g/, /m/, /n/ and /J/. From those, six are plosives, or oral stops, meaning that during their production there is an interruption of the airflow in the vocal cord produced by complete interruption of the airflow and subsequent release that causes a burst of air: /p/, /b/, /t/, /d/, /k/ and /g/. The others are the nasal consonants, meaning that during their production the airflow through the mouth is blocked, but not through the nose: /m/, /n/ and /J/.

There are six fricatives, meaning that these consonants are produced by forcing the airflow through a narrow channel created by two articulators: /s/, /S/, /z/, /f/, /v/ and /Z/, and five liquids, meaning that during these consonants production there is a narrowing of the vocal tract, but still leaving enough space for a less audible airflow: /R/, /r/, /l/, /l∼/ and /L/. From the liquids, two are vibrant, meaning that there is a vibration caused by the tongue against the mouth ceiling: /R/ and /r/, and three are lateral, meaning that the airflow escapes from the sides of the tongue: /l/, /l∼/ and /L/.

Regarding the place of articulation the names are indicative of the articulators producing these consonants and in the EP there are three bilabial consonants: /p/, /b/ and /m/, two labiodental: /f/ and /v/, and three dental consonants: /t/, /d/ and /n/. There are five alveolar consonants: /l/, /l∼/, /r/, /s/ and /z/, four palatal: /J/, /S/, /Z/ and /L/, and three velar: /k/, /g/ and /R/.

The most difficult consonants regarding their phonetic transcriptions are the ones represented by the graphemes <x> and <s>.

The <x> has four possible phonetic transcriptions: /s/, /z/, /S/ or /ks/, according to its context. There is a list of complex rules, however some exceptional cases exist.

The <s> can be transcribed as /s/, /z/, /S/, or /Z/, not only depending on the context of the word, but also of the following word when in a position at the end of a word. It can be also a mute grapheme in situations of double <s>, as <ss> is transcribed as /s/.

The <r> can be transcribed as /r/ or /R/, depending on the context and can be also a mute grapheme in situations of double <r>, as <rr> is transcribed as /R/.

The <l> can be transcribed as /l/, if followed by a vowel, or as /l∼/, if followed by a consonant or placed at the end of a word.

The <g> can be transcribed as /Z/, if followed by <(e,i)>, or /g/, in all the other cases.

All the other non mentioned graphemes have only one possible transcription.

4.1.2 Contextual Information

The language dependent module will give the information needed to build the context-based decision trees used for clustering the phonemes. Two different types of information are used by the language dependent module: the language contextual factors to be considered and the language classification information. Both are described below.

The contextual factors used for EP can be classified according to the hierarchical structure: phone, syllable, word, phrase, and utterance.

At the phone level the contextual factors considered were the current, the previous and the next phones, and the phones before the previous phone and after the next phone. The positions, forward and backward, of the current phone in the current syllable were also considered as factors.

At the syllable level several contextual factors were considered. The stress condition of the current, the previous and the next syllables, together with the number of stressed syllables before and after the current syllable in the current phrase were considered. The number of phones in the previous, the current and the next syllable, together with the number of syllables between the previous stressed syllable and the current syllable and between the current syllable and the next stressed syllable in the utterance were also considered. The positions, forward and backward, of the current syllable in the current word and in the current phrase were other considered factors, and vowel of the syllable.

At the word level the contextual factors considered were the number of syllables in the current, the previous and the next words, and the positions, forward and backward, of the current word in the current phrase.

At the phrase level, the number of syllables and the number of words in the current, the previous and the next phrases were considered. The positions, forward and backward, of the current phrase in the utterance were also considered.

At the utterance level, the contextual factors considered were the number of syllables, the number of words and the number of phrases in the utterance.

The system uses particular labels for the contextual factors, which can be seen in section B.2, from appendix B.

The phonetic classification for performing the tree-based context-clustering was developed according to the following characteristics:

- Silence or breathing;
- Voiced or unvoiced, continuant or non-continuant;
- Vowel: oral or nasal; front (anterior), central, or back (posterior); high, middle, or low; rounded or non-rounded;
- Semivowel: oral or nasal;
- Consonant: voiced or unvoiced; oral or nasal; stop, plosive, fricative, liquid, vibrant-liquid, lateral-liquid; bilabial, labiodental, dental, alveolar, palatal, velar.

A representation of the list of features used for the language classification information can be seen in section B.4, from appendix B.

4.2 Natural Language Processing Module

Due to the need of expensive work by experts to implement rule-based NLP tasks and also due to the lack of flexibility of these methods, more recently the main focus has been on the use of data-driven methods or lexical lookup in large databases to implement NLP modules.

The main task of the NLP module is G2P conversion, which plays an important role in the quality of TTS systems. In fact, G2P together with stress prediction and syllabification are essential tasks for NLP when considering building TTS systems, because they provide significant information regarding the choice of the best units to use for the speech output.

Much work has been done on the speech processing domain like TTS or Automatic Speech Recognition (ASR) for EP. However, there are only few works done in statistically motivated G2P conversion, stress prediction and syllabification, and the use of rule-based systems is more common by now than the statistical approaches.

Rule-based systems are expensive, as they need a linguist expert to set up all rules and exceptions needed to produce the results. Statistical systems are not that cost intensive and can be set up even without linguist knowledge. They appear to be better according to natural sounding synthetic speech of continuous speech, as their statistical models can be trained with data in such a manner as they were spoken, and in a particular speaking rhythm that is intended to be used for each application, with corresponding allophones from coarticulation effects and phonetic reductions.

The advantages of the statistical systems regarding the rule-based systems were the motivation to develop a statistical NLP module based on the ME method, presented in section 4.2.2.

Grapheme-to-phoneme Conversion

The G2P conversion is the translation of any orthographic word into its phonetic representation. The most important aspect of G2P conversion is the choice of the symbol inventory used for the transcription system. Although IPA [IPA, visited in 2010] is the most complete and most widely used symbol inventory for transcription systems, SAMPA [SAMPA, visited in 2010] is usually adopted for computational systems. One of the reasons for this is its reduced phonetic set related to IPA, which is sufficient for G2P transcriptions for computational systems and reduces the system complexity when compared to IPA.

Many are the difficulties in G2P conversion that makes it such a complex process. The words spelled alike but different in pronunciation and meaning, called the homographs, are probably one of the most complex tasks to resolve. A POS tagger can be sufficient to disambiguate the homographs belonging to different POS, but to disambiguate homographs belonging to the same POS a semantic analyzer is needed, and the Portuguese language is rich in both types of homographs [Braga, Coelho & Resende, 2007].

The existence of word transcriptions dependent on the left and right word contexts is another difficulty when performing G2P conversion. The words that do not have a G2P correspondence according to the rules, like imported words from other languages that start belonging to the vocabulary, can also be a complex task.

The different accents of a language, with different pronunciations of the same word, and the tone effects in languages that use tone to distinguish lexical meanings are also important aspects to consider in G2P conversion. And, at a different level, there is the consideration of different rhythms of speech that can cause different transcription results, like faster rhythms of speech in the EP that suffer Sandhi effects such as vowel reduction [Braga, Freitas & Ferrreira, 2003] [Amaral et al., 1999].

A new complexity of G2P conversion is now the fashionable new style of writing that is completely informal and lacking grammar rules, like the styles used in casual emails, social networks, chat rooms, instant messaging applications, and short message systems.

As mentioned before, rule-based systems are more common by now than the statistical approaches but there are some statistical motivated techniques that already showed good results when applied to the G2P conversion, such as decision trees [Black, Lenzo & Pagel, 1998], Finite-State Transducer (FST) [Caseiro et al., 2002], or Transformation-Based Learning (TBL) [Brill, 1995] [Polyakova & Bonafonte, 2006]. Seeing G2P as a classification problem is another possible approach and different kinds of statistical modeling methods have been used for this purpose [Chen, 2003], like the ME method that was used in this thesis [Barros & Weiss, 2006], the HMMs [Taylor, 2005], and the TBL algorithm [Polyakova & Bonafonte, 2006], which was previously applied to POS tagging.

Combining the rule-based method for G2P with a data-driven technique is another possible approach. An overview and comparative evaluation of such approach is presented in [Damper et al., 1998], using pronunciation by analogy, a neural network, and a nearest neighbor algorithm.

There are several rule-based approaches to EP G2P conversion in the literature [Oliveira, Viana & Trancoso, 1992] [Barbosa et al., 2003] [Teixeira et al., 1998] [Teixeira, 1998] [Braga, 2008]. There are only a few statistical methods applied to EP G2P conversion, like a neural network that was introduced by [Trancoso et al., 1994] with fairly good results and which led to a Classification And Regression Tree (CART) based G2P converter developed within the TTS Synthesizer for European Portuguese (DIXI+) framework. A newer approach was meanwhile introduced by [Caseiro et al., 2002], where a Weighted Finite-State Transducer (WFST) was implemented using the rules of the DIXI+, and the same authors presented a hybrid solution using a combined rule-based and data-driven approach.

This dissertation contributes with another statistical method applied to EP G2P conversion, based in Maximum Entropy, which is explained in this section.

Syllable Boundary Detection

Syllable is what denominates the parts of the word which are pronounced in only one voice emission and can be classified in tonic, which is a syllable that carries a tone and also a type of stress, or non-tonic (non-stressed), which is the opposite of a tonic (stressed) syllable. Regarding the non-tonic syllables they can be classified in post-tonic or pre-tonic, both regarding their position relative to the tonic syllable.

A word that consists of a single syllable is called a monosyllable, or a monosyllabic word, while a word consisting of two syllables is called a disyllable, or a dissyllabic word, and a word consisting of three syllables is called a trisyllable, or a trisyllabic word. A word consisting of more than three syllables is called a polysyllable, or a polysyllabic word, but this term is usually also used to describe words of two syllables or more.

Statistical studies about the EP syllabic structure concluded that the "consonant-vowel" structure is the most frequent in the EP polysyllables as well as in the monosyllables [Oliveira, Moutinho & Teixeira, 2005a].

A syllable is a unit of organization for a sequence of speech sounds, typically made up of a syllable nucleus with optional initial and final margins [Wikipedia, visited in 2011].

The syllable is the phonological unit which organizes segmental melodies in terms of sonority [Blevins, 1995]. According to the "Onset-Rhyme" model, the general structure of a syllable consists of the following segments [Davies & Elder, 2005] [Wikipedia, visited in 2011]:

- Rhyme: is the part that is lengthened or stressed when a person elongates or stresses a word and it is usually the portion of a syllable from the first vowel to the end. It consists of a nucleus and an optional coda:
  - Nucleus: is the central part of the syllable, made up of a highly sonorous segment, usually a vowel;
  - Coda: is a less sonorous segment that follows the nucleus;
- Onset: is a less sonorous segment that precedes the nucleus.

In the EP the rhyme contains a maximum of three segments and the number of the elements in the onset is irrelevant for the maximum number of elements in the rhyme. The syllabic nucleus is always composed by vowels and can be filled by any vowel or by a diphthong. Only three consonants, /l/, /s/ and /r/, with its different realizations, can occupy the coda position [Oliveira, Moutinho & Teixeira, 2005a].

The concept of syllable based on the "Onset-Rhyme" model has proved to be productive and efficient in the EP description [Mateus & Andrade, 2000].

The Portuguese syllable boundary detection follows a set of grammatical rules, from which six are related to the vowels, six to the consonants and two are related to the combinations <gu> and <qu>, and to the prefixes <bis>, <cis>, <des>, <dis>, <trans> and <ex> [CRI, visited in 2010].

Unlike for other languages, for EP there is not much work reported about automatic syllable boundary detection. Rule-based systems are more common than statistical ones, and the rule-based approaches published in the literature show good results [Oliveira, Moutinho & Teixeira, 2005a] [Oliveira, 1996] [Gouveia, Teixeira & Freitas, 2000] [Teixeira, 2004]. There are only a few published statistical approaches. Besides the work presented here, there is one based on artificial neural networks [Meinedo, Neto & Almeida, 1999], and another based on FSTs [Oliveira, Moutinho & Teixeira, 2005b].

Stress Prediction

In Portuguese grammar, the words can be classified according to the stressed syllable position in the word. A word is oxytone if the tonic (stressed) syllable is the last syllable of the word, or paroxytone if the tonic syllable is the one before the last syllable of the word, or proparoxytone if the tonic syllable is the third syllable from the end of the word.

In the EP stress always falls into one of the last three syllables, but the second last is the most common. There are however some words without tonic syllable in the EP, called non-stressed words. These words can be monosyllabic function words such as definite and indefinite articles (<o, a, os, as, um, uns>), clitics (<me, te, se, o, a, os, as, lo, la, los, las, no, na, nos, nas, lhe, lhes, nos, vos>) and their contractions (<mo, ma, mos, mas, to, ta, tos, tas, lho, lha, lhos, lhas, no-lo, no-la, no-los, no-las, vo-lo, vo-la, vo-los, vo-las>), the relative pronoun <que>, monosyllabic prepositions (<a, com, de, em, por, sem, sob>) and their contractions (<do, da, dos, das, ao, à, aos, às, no, na, nos, nas, num, nuns>), and monosyllabic conjunctions (<e, mas, nem, ou, que, se>) [Braga & Coelho, 2008].

In Portuguese, if there is an accent in the word it indicates which syllable is stressed. The accent rules for Portuguese can be found in the Inter-institutional style guide from the European Union Publications Office [PT4100100, visited in 2010].

Although word stress for EP is widely studied in the literature, there is not much work published on automatic stress prediction. There are several rule-based approaches that were published [Braga & Coelho, 2008] [Oliveira, Viana & Trancoso, 1991] [Teixeira & Freitas, 1998], but with respect to using statistical approaches, besides this work, only one approach using neural networks [Teixeira, Trancoso & Serralheiro, 1996] was published.

4.2.1 Motivation for using Maximum Entropy in Natural Language Processing

Introduction to Maximum Entropy

ME is a statistical process, which relies on the assumption that there are n given feature functions, f_i, $i = 1...n$, important for modeling the process. That is, the probability p should lie on the subset $\mathcal{C}$ of $\mathcal{P}$, defined by:

$$\mathcal{C} \equiv \{P \in \mathcal{P} | p(f_i) = \tilde{p}(f_i), i \in \{1, 2, ..., n\}\} \tag{4.1}$$

Among the models $p \in \mathcal{C}$, the ME philosophy dictates that the distribution which is most uniform is selected.

A mathematical measure of the uniformity of a conditional distribution $p(y|x)$ is provided by the conditional entropy:

$$H(p) \equiv -\sum_{x,y} \tilde{p}(x) p(y|x) \log p(y|x) \tag{4.2}$$

The entropy is bounded from below by 0 and above by log $\mid Y \mid$, being:

- 0, the entropy of a model with no uncertainty at all;
- log $\mid Y \mid$, the entropy of the uniform distribution over all possible values of y.

Being so, the principle of ME is to select a model from a set $\mathcal{C}$ of allowed probability distributions, choosing the model $p^* \in \mathcal{C}$ with ME $H(p)$:

$$p^* = \arg\max_{p \in \mathcal{C}} H(p) \tag{4.3}$$

It can be shown that p^* is always well-defined; that is, there is always a unique model p^* with ME in any constrained set $\mathcal{C}$.

The ME is a well known approach for ambiguity resolution, where many problems can be reformulated as classification problems. The task of such a reformulation is to include a context and to predict a correct class. The objective is to estimate a function $X \rightarrow Y$, which assigns an object $x \in X$ to its class $y \in Y$. Y represents the predefined classes for each task of this prediction problem.

In the case of G2P conversion, each phoneme from the phonetic set represents a class. In the EP there are 38 phonemes (see table 4.1), but because some grapheme transcriptions are a combination of more than one phoneme, as shown in section 4.2.2, there are 44 classes instead of 38.

In the field of stress prediction we are dealing with a binary classification, where the class is true for stressed syllables and false for non-stressed.

The same binary classification task has to be solved in the domain of the syllable boundary detection, where a syllable boundary exists or not after each grapheme/phoneme.

X consists of linguistic features where the context is included and the resulting input for the classification is a feature vector containing the object itself which has to be classified, as well as the context. The classifier $X \to Y$ can be seen as a conditional probability model in the sense of

$$C(x) = \arg\max_{y} P(y \mid x). \tag{4.4}$$

where x is the object to be classified and y is the class. Including the context, a more complex classifier is achieved:

$$C(x_1, x_2, ...x_n,) = \arg\max_{y_1...y_n} \prod_{i=1}^{n} p(y_i|x_1...x_n, y_1...y_{i-1}). \tag{4.5}$$

where $x_1...x_n, y_1...y_{i-1}$is the context at the ith decision and $y_i$is the outcome.

The objective is to construct a statistical model of the process which generates the training sample $\tilde{p}(x, y)$. The building blocks of this model will be a set of statistics of the training sample. The statistics can be:

- independent of the context;
- dependent on the conditioning information x.

A feature is a binary-valued function of (x, y), while a constraint is an equation between the expected value of the feature function in the model and its expected value in the training data.

Any probability of the sample can be expressed as the expected value of a feature function f:

$f(x, y) = 1$, if constraint is verified or $f(x, y) = 0$ otherwise.

The expected value of f, with respect to the empirical distribution $\tilde{p}(x, y)$, is the desired probability and it is denoted by:

$$\tilde{p}(f) \equiv \sum_{x,y} \tilde{p}(x, y) f(x, y) \tag{4.6}$$

When a probability that is useful is discovered, the model must consider it. This is done by constraining the expected value that the model assigns to the corresponding feature function f. The expected value of f, with respect to the model $p(y|x)$, is:

$$p(f) \equiv \sum_{x,y} \tilde{p}(x) p(y|x) f(x, y) \tag{4.7}$$

where:

- $p(f)$ is a constraint equation;
- $\tilde{p}(x)$ is the empirical distribution of x in the training sample, or the statistical phenomena inherent in a sample of data.

This expected value must be constrained to be the same as the expected value of f in the training sample. That is, it is required that the model of the process exhibit the phenomena $\tilde{p}(x)$, $p(f) = \tilde{p}(f)$, and so:

$$\sum_{x,y} \tilde{p}(x)p(y|x)f(x,y) = \sum_{x,y} \tilde{p}(x,y)f(x,y) \tag{4.8}$$

By restricting attention to the models $p(y|x)$, those models which do not agree with the training sample on how often the output of the process should exhibit the feature f are not considered.

The ME is a statistically motivated model, which was successfully applied to the labeling of sequential data such as POS tagging or shallow parsing. It was decided to apply this model to the NLP tasks and a framework for G2P conversion, stress detection and syllable boundary prediction for EP TTS systems is presented. This framework is based on the ME framework introduced by [Berger, Pietra & Pietra, 1996] and [Ratnarparkhi, 1998].

The ME principle is to model all that is known and assume nothing about that which is unknown. In other words, given a collection of facts, choose a model which is consistent with all the facts, but otherwise as uniform as possible. Only recently computers have become powerful enough to permit the application of this concept to the real world statistical problems [Berger, consulted in 2010]. Such a model is a method of estimating the conditional probability that given a context x the process will output y.

The Exponential form

The ME principle presents us with a problem in constrained optimization: find the $p^* \in \mathcal{C}$ which maximizes $H(p)$. In simple cases the solution can be found analytically, but the solution of the general ME cannot be written explicitly, and a more indirect approach is needed.

The general problem can be addressed applying the method of Lagrange multipliers from the theory of constrained optimization.

The constrained optimization problem at hand is to find:

$$p^* = \arg\max_{p \in \mathcal{C}}(-\sum_{x,y} \tilde{p}(x)p(y|x)\log p(y|x)) \tag{4.9}$$

from equation 4.2 in equation 4.3.

This is the primal problem and a way of saying that $H(p)$ must be maximized according to the following constraints:

- p must be a conditional probability distribution:

$$p(y|x) \geq 0, forallx, y \tag{4.10}$$

$$\sum_{y} p(y|x) = 1, forallx \tag{4.11}$$

- p must satisfy the active constraints $\mathcal{C}$, meaning $p \in \mathcal{C}$:

$$\sum_{x,y} \tilde{p}(x)p(y|x)f(x,y) = \sum_{x,y} \tilde{p}(x,y)f(x,y) fori \in \{1, 2, ..., n\} \tag{4.12}$$

To solve this optimization problem, the Lagrangian is used:

$$\begin{aligned} &\xi(p, \Lambda, \gamma) \equiv \\ &\quad -\sum_{x,y} \tilde{p}(x)p(y|x)\log p(y|x)) \\ &\quad +\sum_{i} \lambda_i(\sum_{x,y} \tilde{p}(x,y)f_i(x,y) - \tilde{p}(x)p(y|x)f_i(x,y)) \\ &\quad +\gamma \sum_{x} p(y|x) - 1 \end{aligned} \tag{4.13}$$

The real-valued parameters γ and $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_n\}$ correspond to the $n+1$ constraints imposed on the solution.

In order to get the optimal value of p, the p^*, the following steps must be performed:

- First hold γ and Λ constant and maximize equation 4.13 with respect to p, having this way an expression for p in terms of γ and Λ;

- Then substitute this expression back into equation 4.13, this time solving the optimal values of γ and Λ (γ^* and Λ^*, respectively).

This way, γ and Λ are held and the unconstrained maximum of the $\xi(p, \Lambda, \gamma)$ over all $p \in \mathcal{P}$ is computed:

$$\frac{\partial \xi}{\partial p(y|x)} = -\tilde{p}(x)(1 + \log p(y|x)) - \sum_i \lambda_i \tilde{p}(x) f_i(x, y) + \gamma \tag{4.14}$$

Equating this expression to zero and solving for $p(y|x)$, it is found that, at its optimum, p has the parametric form:

$$p^*(y|x) = \exp(\sum_{i=1}^{n} \lambda_i f_i(x, y)) \exp(-\frac{\gamma}{\tilde{p}(x)} - 1) \tag{4.15}$$

After the parametric form of p^* has been achieved, it is needed to solve it for optimal values γ^* and Λ^*.

Once the second factor of the equation 4.15 is the factor corresponding to the constraints γ^*, $p^*(y|x)$, it can be rewritten as:

$$p^*(y|x) = Z(x) \exp(\sum_i \lambda_i f_i(x, y)) \tag{4.16}$$

where $Z(x)$ is the normalizing factor and is given by:

$$Z(x) = \sum_y \exp(\sum_i \lambda_i f_i(x, y)) \tag{4.17}$$

γ^* was found, but not Λ^*. For that, the dual function $\Psi(\Lambda)$ must be defined as:

$$\Psi(\Lambda) \equiv \xi(p^*, \Lambda, \gamma^*) \tag{4.18}$$

and the dual optimization problem has to find:

$$\Lambda^* = \arg\max_{\Lambda} \Psi(\Lambda) \tag{4.19}$$

Since p^* and γ^* are fixed, there are only the free variables $\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_n\}$.

So, the solution to the constrained optimization problem was found with p^*, from equation 4.16, and Λ^*, from equation 4.19. This is due to a fundamental principle in the theory of Lagrange multipliers, the Kuhn-Tucker theorem, which asserts that the primal and dual problems are closely related when under suitable assumptions, which are satisfied here. As a result, the ME model subject to the constraints $\mathcal{C}$ has the parametric form p^* of equation 4.16, where Λ^* can be determined by maximizing the dual function $\Psi(\Lambda)$.

Maximum Entropy for Natural Language Processing

The implemented approach is fast, flexible and gives good results in each of the tasks, optimal for fast application development in the TTS domain.

It begins by seeking the conditional distribution $p(y|x)$ which had maximal entropy $H(p)$ subject to a set of linear constraints given in equation 4.9. Following the traditional procedure, in constrained optimization the Lagrangian $\xi(p, \Lambda, \gamma)$ is used, where Λ and γ are a set of Lagrange multipliers for the constraints imposed on $p(y|x)$.

To find the solution to the optimization problem, the Kuhn-Tucker theorem is followed, which states that $\xi(p, \Lambda, \gamma)$ should first be solved, for p to get a parametric form for p^* in terms of Λ and γ, and then p^* should be plugged back into $\xi(p, \Lambda, \gamma)$, this time solving Λ^* and γ^*. The parametric form for p^* has the exponential form of equation 4.16 and the γ^* gives rise to the normalizing factor $Z(x)$, given in equation 4.17. The Λ^* is solved using the dual function 4.19. This function, $\Psi(\Lambda)$, is the log-likelihood for the exponential model p.

4.2.2 Natural Language Processing Tasks for European Portuguese

Natural Language Processing Training Data

The data used for training the NLP models are manually labeled from continuous speech with natural vocalic reduction and coarticulation effects between words, common in Portuguese continuous speech. The idea was to use the same data as the ones used to train the HTS synthesis module in order to be in accordance with the trained speech units. The data samples were formatted in order to use the framework based on the ME framework introduced by [Berger, Pietra & Pietra, 1996] and [Ratnarparkhi, 1998] for labeling sequential data, as mentioned in section 4.2.1.

The training sample can be summarized in terms of its empirical probability distribution $\tilde{p}$, defined by:

$$\tilde{p}(x, y) = \frac{1}{N} \times U \tag{4.20}$$

where N is the total number of samples in training data and U is the number of times (x, y) occurs in the sample.

The framework was implemented in order that the syllable boundary detection and stress prediction follow after the G2P conversion [Barros & Weiss, 2006]. So, the training data consists of three different sets of the same word samples: one with the pairs “graphemes’ word- phonetic transcription” and the other two with the pairs “phonetic transcription- phonetic transcription with syllables boundaries” and “phonetic transcription- phonetic transcription with stress marks”. For the syllable boundary detection task 4283 training samples were used, for the stress prediction 4219 training samples and for the G2P conversion 7352 training samples, respectively. Excerpts of each of the three speech corpora for NLP tasks are presented in appendix A.

Both the syllable boundary detection and the stress prediction training data have a binary classification: “in” or “not in” boundary and “stressed” or “non-stressed” syllable. These classifications were annotated as “0” for no boundary or no stress; and “1”, for boundary or stress.

The G2P conversion is more complex in the sense that instead of predicting binary classes, the system needs to classify 44 different classes, as can be seen in table 4.4. By observation of the table it can be seen that some grapheme transcriptions have more than one phoneme. Due to this fact, some classes are a combination of phonemes, which is the reason for having 44 classes instead of the 38 corresponding to the number of phonemes for EP. Examples of these situations are found for instance in: <tem> (*has*), where the <e> is transcribed as /6∼j∼/ while the <m> is mute: /t6∼j∼/; <têm> (*have*), where the <ê> is transcribed as /6∼j∼6∼j∼/ while the <m> is mute: /t6∼j∼6∼j∼/; <mandam> (*send; order; impose*), where the second <a> is transcribed as /6∼w∼/: /m6∼d6∼w∼/; and <excelente> (*excellent*), where the first <e> is transcribed as /6j/: /6jSs@le∼t@/.

The symbol “&” represents a dummy class for the situations of mute graphemes or suppressed phonemes. For instance, the word <que> can be transcribed as “k&@”, which means /k@/, as the <u> is mute, or as “k&&”, which means /k/, as the <u> is mute and the /@/ can be suppressed due to the rhythm of speech.

Natural Language Processing Results

The results for the three tasks are presented by giving the logarithmic likelihood and the performance of the system. The performance is calculated using the number of correctly classified elements divided by the number of overall elements, multiplied by one hundred. The G2P transcription’s model results, on the phoneme level, are shown in table 4.5, the syllable boundary detection’s model results in table 4.6 and the stress prediction’s model ones in table 4.7.

Besides these results, another test was performed to the G2P converter as this module is more complex for classification than the other two. The corpus for this test has 550 words comprising of 3430 phonemes. All the phonemes from the EP inventory were covered, with different number of occurrences.

The test consisted in comparing the system results of the test corpus with the entries from the training corpus, giving the average number of phoneme errors taking into account three types of errors. The first type is called substitution and refers to the situations where the system replaced the correct phoneme by another. The second type is called insertion and refers to the situations where the system gave a result to a phoneme that in the manual transcription is a dummy, meaning that it should not give a phonetic result. And the third type is called deletion and refers to the situations where the system gave a dummy as a result and the manual transcription gives a phoneme as a result. This kind of measure is known in the speech recognition domain as WER, being used here not at a word level but at a phoneme level.

The results, given by the number of correct classified phonemes divided by the number of overall phonemes, multiplied by one hundred, are shown in table 4.9 and table 4.8, for vowels and consonants respectively, using the confusion matrix method [IRB, last visit in 2010], in which the rows are the actual classes and the columns the predicted classes.

The vowels show to be more difficult than the consonants due to the fact that most of them have several possible transcriptions. But there are also complex consonants, as the <x> and the <s>.

As it can be seen in figure 4.8, the /ks/, /S/, /Z/, /s/ and /z/ are the most difficult cases. It is important to mention that although the /l/ and the /l∼/ were sometimes confused with each other, this is not perceptually relevant and there are systems that do not even distinguish between them.

Regarding the vowels, from the analysis of table 4.9 it can be seen that some of the phoneme substitutions are not errors because they have no perceptual significance or they are both acceptable transcriptions. For example, substituting an /i/ by an /j/ and vice versa, or an /u/ by an /w/ and vice versa is acceptable. It is even acceptable that these phonemes are substituted by /@/ or "&", because the system considers the vocalic reduction effects from the EP language.

Suppressing the /@/ means substituting the /@/ by "&", which is also reasonable and only affects the rhythm of the speech, not the meaning of the word. Attending to these considerations, it can be considered that the /u/ has 96% correct values, the /w/ has 94%, the /j/ has 88% and the /@/ has 88% correct values.

The system represents a first approach to the given tasks and can be improved by adding more samples to the training data. Attending to the considerations made about the results presented in the confusion matrices, the system proves to be a reliable solution and the statistical ME framework shows to be a good and simple approach.

4.3 Synthesis Module

The main module of a TTS system is the module where the speech is actually generated, the synthesis module. The synthesis module implemented based on the work of this thesis uses the HMM-based synthesis approach. The speech parameters needed to produce the speech results are generated by the HMMs directly. For the speech production a source-filter model is used, for which the generated MFCCs compose the transfer function of the filter, and the logF0 values, together with the unvoiced region information, are used to create the signal for the filter source, or the filter excitation. The HMM models used for the implemented module have the left-right no-skip topology and correspond to a phoneme-sized speech unit.

The HMMs have state-duration densities to model the temporal structure of speech and a vector consisting of:

- a spectrum part with:
  - the MFCC vector, including the zeroth coefficients;
  - their delta coefficients;
  - their delta delta coefficients.
- an excitation part with:
  - the logF0;
  - its delta coefficient;
  - its delta delta coefficient.

The HMMs structure is defined in a model prototype definition file. This type of file allows to define the topology and the overall characteristics required for each HMM and consists of the number of states of the model, including the non-emitting states, and the information for each state, followed by the parameters of the transition matrix and the model duration parameters. The HMM prototype definition file for the EP system is presented in section C.1, from appendix C.

The information needed to define a single HMM is [Young et al., 2001]:

- Type of observation vector;
- Number and width of each data stream;
- Optional model duration parameter vector;
- Number of states;
- For each emitting state and each stream:
  - Mixture component weights or discrete probabilities;
  - If continuous density, then means and variances;
  - Optional stream weight vector;
  - Optional duration parameter vector;
- Transition matrix.

With the exception of the transition probabilities, all the HMM parameters given in the prototype definition are default values that will be modified during the training phase.

The HMM definition gives the topology of the HMM and has its formal representation between the tags:

```
~h "hmmPT"
<BeginHMM>
    ...
<EndHMM>
```

The HMM topology for the EP system implemented in this thesis is a 7-state, left-right, no skip HMM, with two non-emitting states, the first and the last ones. The other five states, the middle ones, are emitting states and have output probability distributions associated with them.

The transition matrix for this model has seven rows and seven columns, corresponding to the number of states of the model. Once the model is left to right with no skip, the transition matrix only has non-zero values in the diagonal corresponding to the elements $A_{i,i+1}$. Each row sums to one, except for the final row which is all zero, since no transitions are allowed out of the final state. See the transition matrices from the HMM files in section C.2 as example.

Each vector has a length of seventy eight values, divided into four streams. The first stream has one mixture component and presents seventy five mean and variance values: twenty-five correspond to the MFCCs, twenty-five to their delta and the last twenty-five to their delta delta coefficients. The other streams have two mixture components, each represented by its weight and the mean and variance values for the Gaussian PDF. The second stream corresponds to the F0, the third to its delta, and the fourth to its delta delta.

A variance floor factor can be used to impose a lower bound on the variance parameters. The variance flooring technique helps prevent the risk of over-fitting that is a usual problem when implementing models with a small amount of training data. Variance parameters are especially susceptible to over-fitting, because a variance estimated from a small amount of data can be very small and not representative of the real distribution.

A "project folder" must be created for the system structure. Several types of information need to be extracted from the speech database, as explained in section 4.3.1. This information will be stored in a subfolder of the "project folder", called "data". Figure 4.3 shows in a) the structure of the "project folder", called "EP_HTS_demo", and in b) the structure of the "data" subfolder.

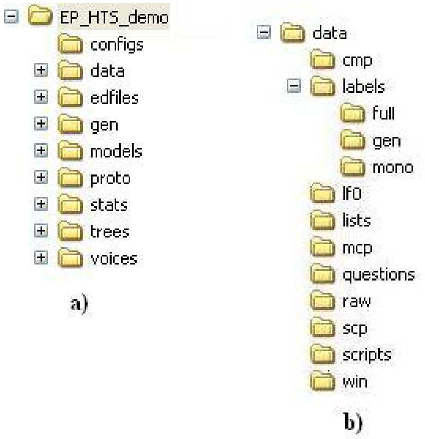


Figure 4.3: Structure of: a) the Project folder; b) its data subfolder

The different data collected from the speech database are organized in different folders inside the "data" folder, shown in figure 4.3:

- **Folder "raw"**

  Contains the sound files, which are in "raw" format. The convenience of using raw audio file, besides having no header, is that it is an uncompressed Pulse-Code Modulation (PCM) audio data file format. The sampling frequency used is 16 KHz.

- **Folder "lf0"**

  Contains the files with the logF0 values for each speech frame of each file from the database, the "lf0" files. The frames for the EP-HTS implementation are 25 ms Blackman window signals, with an overlap in the total signal of 20ms. An example of part of a "lf0" file can be seen in section B.6 of appendix B.

- **Folder "mcp"**

  Contains the files with the MFCC values for each speech frame in each of the files from the database, the "mcp" files. An example of these files can be seen in section B.7 of appendix B.

- **Folder "cmp"**

  Contains the files with the training data in the HMM ToolKit (HTK) data format. An example of these files, the "cmp" files, can be seen in section B.8 of appendix B.

- **Folder "scp"**

  Contains the "scp" files, which are files with the location of the data for the training, consisting of paths to all the "cmp" files from the database.

- **Folder "lists"**

  Contains the files that give the lists of the models to train. There are two different types of list files. One contains the list of all the phonemes present in the database. The other contains the list of the context-dependent feature label files of all the utterances in the database. Section B.3, from appendix B, shows an example of the context-dependent features label files.

- **Folder "labels"**

  Contains the master label files, "lab" files, which give the path to the two different types of files stored in the subfolders "mono" and "full". The files in the "mono" subfolder contain the phonetic transcriptions from the database utterances, with the initial and final time of each phoneme. An example of these files can be seen in section B.5 of appendix B. The files in the "full" subfolder contain the context-dependent features for all the utterances in the database, using the context-dependent features labels format shown in section B.3 of appendix B. Section B.3 of the same appendix presents the list of context-dependent feature labels used for EP in this thesis.

- **Folder "questions"**

  Contains the file with the questions about the contexts of the system's language, in this case EP. These questions will be used for the decision tree-based context clustering. A representation of the questions file for EP is shown in section B.4, from appendix B.

- **Folder "win"**

  Contains the files with the window coefficients to calculate the static and the dynamic features, the "win" files. These files can be seen in section B.9, from appendix B.

There are different stages in the HMM-based synthesis process, using different basic techniques associated with the success of the approach:

- During data preparation: A mel-cepstral analysis technique allows to obtain the spectral parameters in order to synthesize speech directly from the coefficients. This technique is described in section 3.2.4 [Fukada et al., 1992].
- During parameters modeling for training: Simultaneous modeling of spectrum, F0 and duration is achieved by a special type of HMMs, the MSD-HMM, which is described in section 3.2.3 [Yoshimura, 1999] [Tokuda et al., 2002].
- During parameter generation for synthesis: The speech parameters are generated directly from the HMMs using dynamic features. This approach is described in section 3.2.1 [Tokuda et al., 2000a].
- During synthesis: The speech waveform is reconstructed through the vocal tract source-filter model, using a MLSA filter as described in section 4.3.3 [Fukada et al., 1992].

4.3.1 Hidden Markov Model Training Data

The synthesis training data is part of the published FEUP/IPB-DB database [Teixeira et al., 2001]. This database consists of 16 texts, with a total amount of 60 minutes of speech. The database was manually phonetically segmented and labeled by more than one person. The reason for not using the whole database is because when there is more than one person labeling a database this can cause some issues relating to the transcription results:

- Inconsistencies between labeling decisions;
- Inconsistencies between used labels;
- Inconsistency in the number of semivowels considered for EP;

- Inconsistency relating to the nasal diphthongs, as these diphthongs can be considered to be composed by a nasal and a non-nasal sound or by two nasal sounds as they are neighbors and one would always nasalize the other.

The files used are composed by utterances extracted from the original files, considering that in the nasal diphthongs both the vowel and the semivowel are nasal and considering the four semivowels presented in the table 4.1: /j/, /j∼/, /w/ and /w∼/. The system's training database has 104 utterances, giving the total amount of 21 minutes of speech.

Two types of text files are needed for each utterance used in the training process. One has the segmentation and labeling data, which is the initial and final time values of each phoneme of the utterance. An example of this type of file can be seen in appendix B, section B.5. The other one contains the utterance NLP information, which is the G2P conversion results and the syllable, word and phrase boundaries and stress information. This information will be used for the HMMs and for deriving the contextual factor information described in section 4.1.2. An example of this type of file can be seen in section B.1, from appendix B.

The system uses the SPTK toolkit [SPTK, visited in 2010], version 3.0, for signal feature extraction: the logF0 and the MFCCs. For more details, the Speech Signal Processing Toolkit (SPTK) site [SPTK, visited in 2010] or the SPTK reference manual [Tokuda et al., 2000b] should be consulted. The HTS [HTS, visited in 2010] toolkit, version 2.0, is used for building the HMMs and the decision trees. For more details, the HTS site [HTS, visited in 2010] or the HTK book [Young et al., 2001] should be consulted.

In speech signal processing it is common to consider the speech signals in short-time signals, due to speech signal "quasi-stationary" properties. For that, it is common to use frames of a particular period and length in order to use just some part of the signal and make the rest of the signal to be zero-valued. The period of the frame determines the size of the frame steps through the signal and the overlap between frames.

The short-time speech signal frames are usually multiplied by a window, in order to reduce leakage in the determination of the frequency spectra of sampled signals and in the design of finite impulse response digital filters. Windowing functions are usually symmetrical about their center and have a maximum value of one, going towards zero on both sides, although some window functions, such as the Hamming window, do not quite reach zero at either end. Some common windows are presented in figure 4.4.

The length used for the short-time signal frames for both the MFCCs and the logF0 extraction is 25 ms. In points, this is equivalent to $16000 \times 0.025 = 400$. The jump between frames is 5ms, corresponding to 20% of the frame length, in order to have a signal overlap of 20ms. In samples the jump is equivalent to $16000 \times 0.005 = 80$.

The frame is multiplied by a Blackman window of the same size. This window has slightly wider central lobes and less sideband leakage than the equivalent length Hamming and Hann windows, which could also be used.

The Blackman window is characterized by the function:

$$w(n) = 0.42 - 0.50 \cos[2\pi \frac{n}{N}] + 0.08 \cos[4\pi \frac{n}{N}] \tag{4.21}$$

$N + 1$ being the window length.

The Hamming window function would be:

$$w(n) = 0.54 - 0.46 \cos[2\pi \frac{n}{N}] \tag{4.22}$$

and the Hann window function would be:

$$w(n) = 0.5(1 - \cos[2\pi \frac{n}{N}]) \tag{4.23}$$

Figure 4.4 shows these three windows.

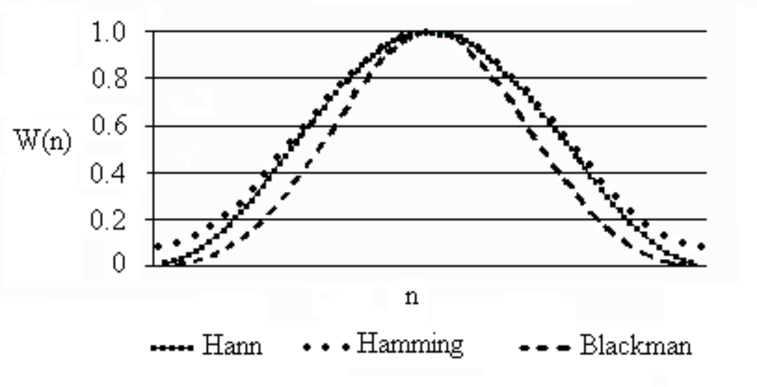


Figure 4.4: Window Functions

A window normalization by power is performed:

$$\sum_{n=0}^{L-1} w^2(n) = 1 \tag{4.24}$$

L being the frame length.

The algorithm to extract F0 returns the values in their logarithmic form and limits the values of F0 extraction to 60Hz as the lower limit and 400Hz as the upper one. This algorithm is a Program development tool (TCL) [ActiveTcl, visited in 2010] script that uses the Snack [Snack, visited in 2010] tool.

The algorithm used to extract the MFCCs uses a Fast Fourier Transform (FFT) length of 4096 points, a mel-cepstral analysis order of 24 coefficients and a frequency warping factor of 0.42. The frequency warping factor is a factor for the phase response of the vocal tract transfer function during mel-cepstral analysis. The appropriate choice of the frequency warping factor causes the phase response to give a good approximation to the human auditory frequency scale and is related to the used sampling frequency, 16000Hz, and scale, which for the cepstral analysis is the mel scale.

Table 4.10 shows some examples of frequency warping factor values for better approximating human auditory frequency scales, simulating the human sensitivity to the frequencies of speech signals.

The training data files are files with the data in the HTK data format, composed by the MFCCs and logF0 files together. The data are organized in frames and there are 78 values per frame: 25 MFCCs, 25 delta MFCCs, 25 delta delta MFCCs, logF0, delta logF0 and delta delta logF0. These files have the format extension "cmp", from which an example is shown in section B.8, from appendix B. The delta and delta delta features are extracted from the window files shown in section B.9, from appendix B.

The window files are the files with the window coefficients to calculate the static and dynamic features. Independent window files are used for the logF0 and the MFCCs.

Each of these files has the coefficients to calculate the windows for one type of feature:

- "win1", refers to the static feature window coefficients;
- "win2", refers to the velocity, delta, dynamic feature window coefficients;
- "win3", refers to the acceleration, delta delta, dynamic feature window coefficients.

4.3.2 The Training Process

Figure 4.5 shows a scheme of the HTS training process, which consists of three main parts:

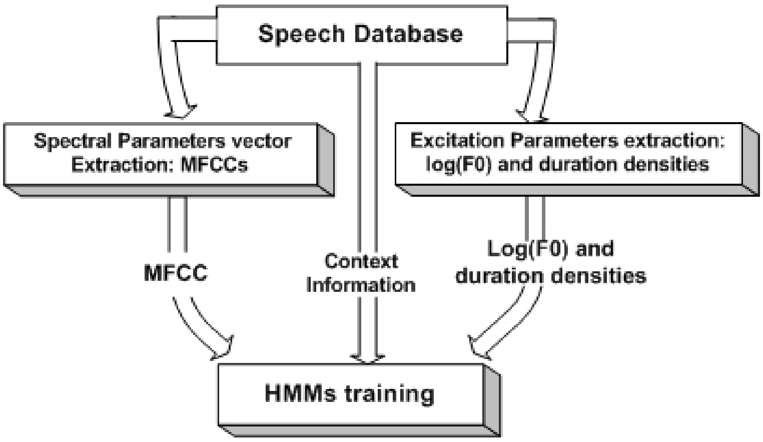

Speech Database
Spectral Parameters vector Extraction: MFCCs
Excitation Parameters extraction: log(F0) and duration densities
MFCC
Context Information
Log(F0) and duration densities
HMMs training

Figure 4.5: HMM-based training stages.

1. **Speech Parameter Extraction**
   The speech parameters extracted for the training are the F0 and the MFCCs. These parameters are stored in files that use the HTK data format [HTK, visited in 2010], the "cmp" files, presented in section B.8, from appendix B.

2. **Contextual Label Definition**
   Utterance information from the whole speech database that was converted into labels with the same format as those used during the synthesis phase. This information is stored in the context-dependent feature label files, presented in section B.3, from appendix B.

3. **HMM Modeling**
   A unified framework is used for the HMMs modeling that models logF0, MFCCs and durations simultaneously. Each HMM includes state duration densities and its observation vectors are composed of four streams: one with the MFCCs and their related delta and delta-delta parameters; and the other three with the logF0 and its related delta and delta-delta.

There are many contextual factors that cause acoustic variation in phonemes by affecting spectrum, F0 and duration, as described in section 3.2.2.

The language-dependent context factors are extracted from the utterance files received from the NLP module. An example of these files is presented in section B.1

of appendix B. For the EP system, the contexts chosen and the list of their corresponding label tags are presented in section B.2, from appendix B, and an example of a context feature label file is presented in section B.3 of the same appendix.

Context-dependent HMMs capture the contextual factors using the decision-tree-based context-clustering technique. The implemented TTS system uses a unified framework of HMMs, where F0, MFCCs and durations are modeled simultaneously, as it was explained in chapter 3. This allows us to apply the decision-tree-based context-clustering independently from the MFCCs, logF0 and state durations, to better capture the factors that interfere more with each of these features.

From the system training a set of automatically generated binary decision trees is produced, which are used for the generation of the MFCCs, logF0 and state durations during the synthesis process. The decision trees are created from the language contextual information questions presented in section B.4, from appendix B.

One decision tree is created for the durations, five for the logF0 distribution, and five for the MFCC distribution, one for each HMM emitting state. Section C.3, from appendix C, presents examples of part of the decision trees for the durations, figure C.1, for the fourth state of the logF0 distribution, figure C.2, and for the fourth state of the MFCC distribution, figure C.3.

The number of leaves of each tree from the implemented system are:

- Decision tree for durations: 250
- Decision tree for second state of the logF0 distribution: 158
- Decision tree for third state of the logF0 distribution: 136
- Decision tree for fourth state of the logF0 distribution: 161
- Decision tree for fifth state of the logF0 distribution: 98
- Decision tree for sixth state of the logF0 distribution: 123
- Decision tree for second state of the MFCC distribution: 123
- Decision tree for third state of the MFCC distribution: 116
- Decision tree for fourth state of the MFCC distribution: 115
- Decision tree for fifth state of the MFCC distribution: 103
- Decision tree for sixth state of the MFCC distribution: 135

To generate and train the HMMs, the first step is to define their structure. After that it is required to estimate their parameters from the database data, which are the sequences that they are intended to model. For the parameter estimation in the training process HTK [HTK, visited in 2010] is used.

An HMM will be created for each phoneme from the "mono.list", the list with the phonemes to model. This list is composed of the EP phonemes presented in table 4.1, from section 4.1.1, plus the silence and the breathing.

During system implementation, some technical issues related to the EP SAMPA set had to be resolved. Due to computational formatting problems, the symbol "∼" was substituted by "c", which is not a EP SAMPA symbol. Due to the HTS file format, the SAMPA character "6" was substituted by "A", otherwise it would have been confused with numerical values. Due to Windows problems in not distinguishing uppercase names from the corresponding lowercase ones, all the phonemes whose SAMPA symbols use uppercase letters were substituted by the same letters followed by "q", which is not a EP SAMPA symbol.

Each HMM is generated individually. The training process starts by reading the prototype HMM definition, which defines the required HMM topology, and the training data from the database, in order to output a new HMM definition. The prototype HMM definition file for the EP system is presented in section C.1, from appendix C. For the new HMM definition the means and variances of the Gaussian components are calculated to be equal to the global mean and variance of the speech training data, respectively.

Having the same parameters initially given to all models is used as a strategy, known as "flat start training", to make all models initially equal so the first iteration of the embedded training relies on a uniform segmentation of the data [Young et al., 2001].

The process consists in first initializing for each of the phoneme HMMs their parameters with default values and then re-estimating their parameters. Both models, before and after re-estimation, are stored in the folder "models" that can be seen on the tree structure in figure 4.3. Section C.2, of appendix C, presents the example of a HMM before and after the re-estimation, for better understanding the process.

The essential problem is to estimate the means and variances of the HMM in which each state output distribution is a single component Gaussian, that is:

$$b_j(O_t) = \frac{1}{\sqrt{(2\pi)^n |\sum_j|}} \exp^{-\frac{1}{2}(O_t-\mu_j)' \sum_j^{-1}(O_t-\mu^j)} \tag{4.25}$$

where j are the states, and:

$$\hat{\mu}_j = \frac{1}{T}\sum_{t=1}^{T} O_t \tag{4.26}$$

$$\hat{\sum}_j = \frac{1}{T}\sum_{t=1}^{T}(O_t - \mu_j)(O_t - \mu_j)' \tag{4.27}$$

All the data corresponding to each of the required phonemes are read and uniformly segmented. Each successive state is associated with each successive data segment. Equations 4.26 and 4.27 are used to give initial values for the mean and variance of each state. The maximum likelihood state sequence is found with the Viterbi estimation algorithm and the observation vectors are reassigned to the states. Equations 4.26 and 4.27 are used again, to improve the initial values. The procedure is repeated until the estimates do not change any more.

The transition matrix of the prototype specifies both the allowed transitions and their initial probabilities. Transitions which are assigned zero probability will remain zero, denoting non-allowed transitions. The transition probabilities are estimated by counting the number of times that each state is visited during the alignment process.

The parameters of the created HMMs are refined using Baum-Welch re-estimation. Since the full likelihood of each observation sequence is based on the summation of all possible state sequences, each observation vector O_t contributes to the computation of the maximum likelihood parameter values for each state j. This means that instead of assigning each observation vector to a specific state, as in the above approximation, each observation is assigned to every state in proportion to the probability of the model being in that state when the vector was observed.

The equations 4.26 and 4.27 become the weighted averages, corresponding to the Baum-Welch re-estimation functions for the means and variances of the HMM:

$$\hat{\mu}_j = \frac{\sum_{t=1}^{T} L_j(t) O_t}{\sum_{t=1}^{T} L_j(t)} \tag{4.28}$$

$$\hat{\sum}_j = \frac{\sum_{t=1}^{T} L_j(t)(O_t - \mu_j)(O_t - \mu_j)'}{\sum_{t=1}^{T} L_j(t)} \tag{4.29}$$

Where $Lj(t)$ is the probability of being in state j, at time t.

$Lj(t)$ can be calculated using the Forward-Backward algorithm, as explained in [Young et al., 2001].

The refinement process consists in first performing a basic Baum-Welch re-estimation of the parameters from each single HMM, using a set of observation sequences. It operates on HMMs with initial parameter values estimated by the previous Viterbi estimation. On the output of the basic Baum-Welch re-estimation an embedded training version of the Baum-Welch algorithm is then performed, in order to perform another re-estimation of the HMM parameters. The training samples are seen as the output of the HMMs whose parameters are to be estimated. For each training utterance, a composite model is synthesized by concatenating the phoneme models given by the transcription.

It was already mentioned before that, when training large model sets from limited data, setting a floor is often necessary to prevent variances from being badly underestimated because of data sparsity. All the processes used for the HMM training use a variance floor, which is a floor set on each individual variance in order to prevent any variance of getting too small. For that, a variance floor macro is defined, with values equal to a specified fraction of the global variance.

At this stage, the models are ready for the tree-based context clustering, and the embedded re-estimation is again performed for the clustered models.

Several folders are created during the training process, which can be seen in the structure presented in figure 4.3:

- **Folder "models"**

  Contains the model files for the durations and for the MFCCs and logF0 training data. It has two subfolders: one, named "dur", for the duration model files and the other, named "cmp", for the training data in the HTK format.

- **Folder "stats"**

  Contains files with statistics.

- **Folder "edfiles"**

  Contains the folder structure for the model edit files. The model edit files contain a set of commands needed for the HMM modeling.

- **Folder "trees"**

  Contains the folder structure for the binary decision trees, created from language context information to decide MFCC, logF0 and state durations. Examples of these trees are presented in section C.3, from appendix C.

- **Folder "voices"**

  Contains the folder structure for the model and tree files converted to the HTS engine format.

- **Folder "gen"**

  Contains the synthesis results. Besides the wave files, other files can be created, like the "trace" file, which is a file with the parameters generated by the models for the utterance to be synthesized. An example of a "trace" file is presented in section C.5, from appendix C.

- **Folder "configs"**

  Contains the variable configuration files, which are files with the data format settings that are needed for the data computation or to customize the working environment.

- **Folder "proto"**

  Contains the HMM topology definition file. The HMM definition file for the EP system is presented in section C.1, from appendix C.

4.3.3 The Synthesis Process

The synthesis process consists of several stages for which the input is the text sequence to be synthesized. The sequence of stages for HTS synthesis is shown in figure 4.6.

The text sequence to be synthesized is first converted into a context-based label sequence and then, according to the label sequence, a HMM sequence is constructed by concatenating context-dependent HMMs.

The context-dependent HMMs are then used to generate the speech parameters for synthesis. During the parameter generation state durations of the HMM sequence are determined to maximize the output probability of state durations, and a sequence of MFCC and logF0 values, including voiced/unvoiced decisions, is generated in a way that the sequence output probability for the HMM is maximized using the speech parameter generation algorithm described in section 3.2.1 [Tokuda et al., 2000a].

Using the information from the models and the decision trees as input, the speech parameter trajectories will be generated in order to maximize their output probabilities for a given HMM sentence, under the constraints between static and dynamic features.

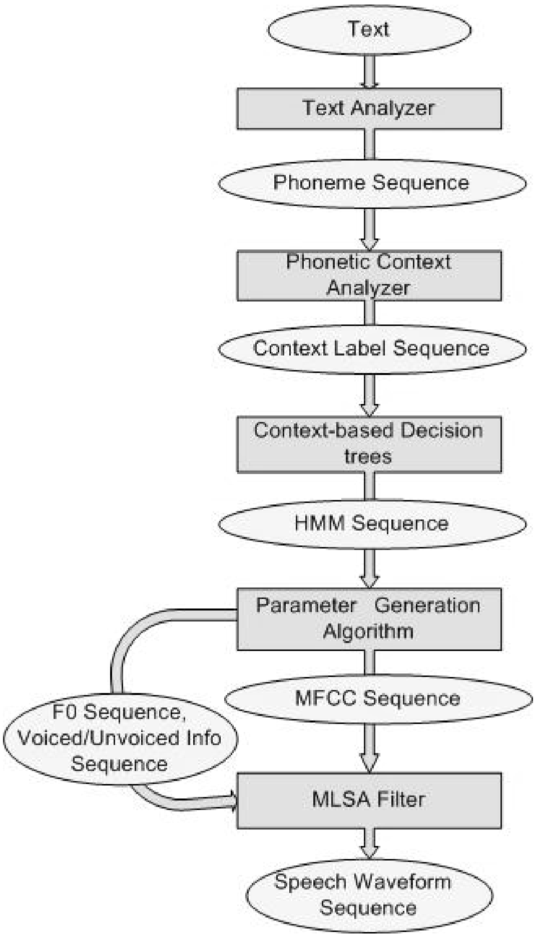

Text
Text Analyzer
Phoneme Sequence
Phonetic Context Analyzer
Context Label Sequence
Context-based Decision trees
HMM Sequence
Parameter Generation Algorithm
MFCC Sequence
F0 Sequence, Voiced/Unvoiced Info Sequence
MLSA Filter
Speech Waveform Sequence

Figure 4.6: HMM-based synthesis stages.

The speech waveform is synthesized directly from the generated MFCC and logF0 values by using the MLSA filter.

The logF0 values are transformed in F0 values, which together with the information of the voiced/unvoiced regions are converted into a pitch sequence to be used as the excitation of the MLSA filter, composed of:

- a pulse train for voiced speech regions;
- a Gaussian noise sequence for unvoiced speech regions.

The autocorrelation coefficients are obtained from the MFCCs through the inverse Fourier transform of the power spectrum. The power spectrum is calculated from the logarithmic spectrum, which is obtained from the Fourier transform of the M^{th} order cepstral analysis.

The following frequency transformation is performed on the M^{th} order MFCCs, $c_{\alpha_1}(0), c_{\alpha_1}(1), ..., c_{\alpha_1}(M)$:

$$c^{(i)}_{\alpha_2}(m) = \begin{cases} c_{\alpha_1}(-i) + \alpha c^{(i-1)}_{\alpha_2}(0) & m = 0 \\ (1-\alpha^2)c^{(i-1)}_{\alpha_2}(0) + \alpha c^{(i-1)}_{\alpha_2}(1) & m = 1 \\ c^{(i-1)}_{\alpha_2}(m-1) + \alpha(c^{(i-1)}_{\alpha_2}(m) - c^{(i)}_{\alpha_2}(m-1)) & m = 2, ..., M \end{cases} \tag{4.30}$$

$\alpha = (\alpha_1 - \alpha_2)/(1 - \alpha_1\alpha_2)$ and $i = -M, ..., -1, 0$

where α_1 and α_2 are frequency warping parameters, as explained in section 4.3.1.

In order to use the MFCCs in the MLSA filter, the following transformation is performed:

$$b(m) = \begin{cases} c_\alpha(M) & m = M \\ c_\alpha(m) - \alpha b(m+1) & 0 \leq m < M \end{cases} \tag{4.31}$$

Being $c_\alpha(m)$ the MFCCs and $b(m)$ the coefficient values used for the MLSA filter.

The exponential transfer function of the MLSA filter, $H(z)$, is obtained by the M^{th} order MFCC $c_\alpha(m)$, as follows:

$$H(z) = \exp \sum_{m=0}^{M} c_\alpha(m)\tilde{z}^{-m} \tag{4.32}$$

where z^{-1} is a first order all-pass transfer function:

$$\tilde{z}^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}, \mid \alpha \mid < 1 \tag{4.33}$$

The HTS engine allows to create a file, the “trace” file, with the parameters generated by the models for the utterance to be synthesized. An example of this type of file is presented in section C.5, from appendix C, showing the results of the TTS synthesis process for the input utterance <Olá Maria>.

4.4 Text-to-Speech System Results and Evaluation

4.4.1 Introduction to the Evaluation Schemes

The evaluation tests usually used in TTS systems consist of subjective evaluation schemes using specific groups of listeners. Subjective quality tests using human listeners are very common in speech systems, since there is no metric or parameter that allows us to classify the quality of speech according to human perception.

There are two common types of TTS evaluation tests, the tests of the intelligibility of speech, which deal with listening comprehension based on open or multiple choice answers, and the tests of the acceptability of speech, which deal with subjective opinions relating to the audible speech quality.

Intelligibility, in speech synthesis, can be defined as the degree of comprehension by the listener, or the extension to which listeners identify words or synthesized sentences. There are some particular features of each language that are critical, because speech comprehension will depend on accurate synthesis, and sometimes their acoustic signals are very similar, which makes them difficult cases to deal with.

Acceptability denotes how well a synthetic voice is accepted by the listener. Acceptability is strongly correlated with intelligibility, but there are situations where there is good intelligibility but the quality is degraded. Some reasons for that can be, for instance, poor prosody, gaps in the concatenation or bad sound quality. The opposite can also happen, if for instance some units are missing but the prosody is pleasant. In fact, prosody is one of the main aspects of speech naturalness and therefore very important for acceptability.

Subjective Evaluation Tests for Intelligibility

Intelligibility tests can use closed or open answers. On the closed answers tests there is a multiple choice of possible answers, while on the open answers tests the listeners have to write down what they had heard.

There are several standard intelligibility evaluation schemes, some using words and others using sentences.

Some of the common schemes using words are [Syrdal, Bennett & Greenspan, 1994] [Dutoit, 1997]:

Words Phonetically Balanced (WPB)

The WPB tests use monosyllabic words that are grouped in lists according to a pattern and are usually presented in carrier sentences. One example of these tests is the PB-50 test [ANSI s3.2-1989, 1989]. It is an open answer test that uses 1000 monosyllabic words with the structure consonant-vowel-consonant, organized in groups of 50 words. All the words use the same carrier sentence.

Diagnostic Rhyme Test (DRT)

The DRT evaluation scheme uses closed answers presented in six groups of rhyming pairs of words. One word from each pair is heard. The word is presented isolated, not in a carrier sentence. In the original test, the groups are according to the following phonetic features: voicing, nasality, sustention, sibilation, graveness and compactness.

Modified Rhyme Test (MRT)

The MRT evaluation scheme also uses closed answers and evaluates the same phonetic features as in DRT, but presents six choices for each answer and usually the words come in carrier sentences.

A common scheme using sentences is the:

Semantically Unpredictable Sentences (SUS)

The SUS evaluation scheme uses open answers. The number of presented sentences is arbitrary, but the sentences must be semantically unpredictable. Although the sentences cannot have an obvious sense, they must have a valid grammatical structure. The use of semantically unpredictable sentences is an advantage to test the system behavior to phonetic contexts that were not covered in the database.

Subjective Evaluation Tests for Acceptability

The tests of the acceptability deal with subjective opinions related to the audio quality of speech. As these tests use very subjective measures, the listeners need to be instructed and the results from the different listeners must be compared in the end.

Two of the common types of acceptability tests are [Syrdal, Bennett & Greenspan, 1994] [Dutoit, 1997]:

Pairwise Comparison

These types of tests compare a system with another one that can be a reference system, a commercial one or even another system with some particularity for the evaluation, like using the same database or the same synthesis approach. The listener is presented with pairs of results from both systems and points out his/her preference. All the pairs should be heard twice, in both orders, to allow a fair judgment of each system: "system1-system2" and "system2-system1".

Evaluation scales

For this type of tests the listeners are presented with one or more sentences from the system to be evaluated and are requested to classify them according to a particular scale that is presented.

Two of the most used schemes for acceptability tests are:

Diagnostic Acceptability Measure

The Diagnostic Acceptability Measure (DAM) uses twelve sentences for each of six listeners, 3 male and 3 female. It uses twenty-one evaluation scores divided into three main groups: 10 for the perceptual quality of the signal; 8 for the background quality; and 3 for the general speech features.

Mean Opinion Score

The MOS usually uses sentences, which are presented to the listeners for evaluation. A scale from one to five is used, meaning excellent, good, sufficient, poor and bad. Usually many listeners are used, but without specific training. This type of test is essentially used for system comparison, as it is very subjective. In this case, the speech material must be presented to the listeners in different orders. For example, when comparing two systems, some sentences must be presented first from "system1" and then from "system2", and other sentences using the opposite order, to allow a fair judgement for each system.

4.4.2 European Portuguese Text-to-Speech System Results

The test scheme chosen to evaluate the implemented system was the test of the acceptability MOS [Dutoit, 1997], in order to perceive how the system would be accepted when compared with existent systems for the same language.

The test utterances were chosen in order to include all the EP phonemes. The test-set is presented in table 4.4.2, where the utterances are translated into

SAMPA [SAMPA, visited in 2010] and the first occurrence of each phoneme appears in bold and underlined.

The TTS systems evaluated in the test were:

- The implemented HMM-based EP TTS;
- An EP concatenative synthesizer implemented based on an adaptation of the Time Domain Pitch Synchronous OverLap and Add (TD-PSOLA) method [Barros, 2002];
- A commercial synthesizer, *RealSpeak*, from Scansoft [Realspeak, visited in 2010].

The test was performed by twenty-seven listeners without previous training, twenty-two male and five female participants, with ages varying from 26 to 60 years old. There were more male participants than female ones as the listeners were from a Faculty of Electrotechnical Engineering, where normally there are mainly men and almost no women.

It was decided to use the score scale from one to five, corresponding to bad, poor, fair, good and excellent, respectively, instead of the usual inverse correspondence to excellent, good, sufficient, poor and bad. This decision was based in the fact that in Portugal it is common to have this scale with the chosen correspondence and having the opposite classification could confuse the persons performing the test.

The test results are presented in figure 4.7. The chart contains seven column sets: the first six with the mean values of all listeners per utterance for each synthesizer, and the last one with the mean values of all listeners and all utterances for each synthesizer.

Analyzing the chart and having in consideration that the listeners had no training and most of them were not familiar with speech synthesizers, it is shown that the HMM-based TTS system has a good acceptability, approaching the commercial system in score, even though the HMM-based system was trained with only 21 minutes of speech.

From the listeners' comments the HMM-based TTS system presents a smooth prosody and has a good acceptability. The fast speaking rate and a buzzy, vocoder like sound are the reasons for the HMM-based TTS being worse when compared to the commercial system. The fast speaking rate is due to the fact that the used database was implemented from long newspaper news readings. As the test was composed of small sentences, the speech rate seems fast. The buzzy, vocoder like sound is due to the excitation signal plus generated spectra. It must be highlighted

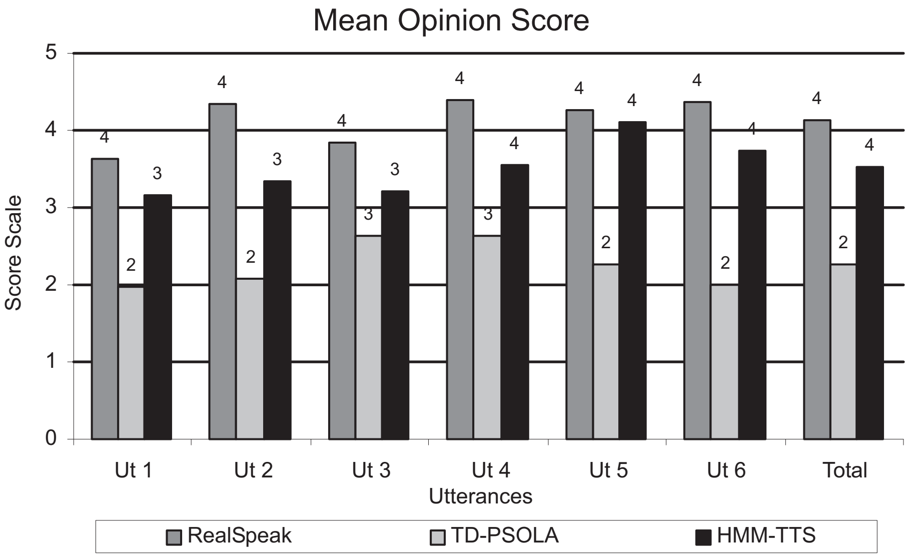


Figure 4.7: Chart with MOS test results.

that the HMM-based system was trained with only twenty-one minutes of speech, what is less than five per cent of the nine hours of speech usually considered necessary for a unit-selection based system to present good results.

These comments reveal that the system could be improved by refining the source-filter model. Section 6.2 presents a possible solution regarding this improvement proposal.

Another improvement could be achieved by enhancing the speech corpus with respect to the language contextual factors. Chapter 5 presents a new speech corpus, especially designed for language context-based systems, which includes as many language contextual factors as possible, in only one sentence per diphone of the language.

As EP suffers from vocalic reduction effects in continuous speech, an informal test of the influence of considering this effect in the G2P conversion was performed. For the test, the sentences from the MOS test were synthesized first according to the G2P rules, without considering the vocalic reduction effect, and then having the vocalic reduction effect taken into consideration. Both results were presented to some listeners and all the listeners choose the sentences that considered the vocalic reduction effect.

All the listeners preferred the results which considered the vocalic reduction effects. This fact shows that considering the vocalic reduction effects in G2P conversion is important for the naturalness of synthetic speech. It must be mentioned the fact that the segmentation and labeling of the used database, as well as the implemented G2P converter, had the vocalic effects taken into consideration. This fact also influenced the system's results and consequently the listener test.

The new speech corpus presented in chapter 5 is phonetically transcribed in two different ways, one considering the vocalic reduction effects and the other following the G2P rules without considering these effects. Both transcriptions are analyzed and compared in the chapter to better understand the influence of the vocalic reduction in the EP.

4.4.3 Text-to-Speech System Results for a Test Utterance

This section presents an example of the file produced by the HTS engine for an input utterance. This file, called "trace", presents the parameters generated by the models for an utterance to be synthesized.

The example is based on the utterance input: <Olá Maria>.

The utterance is sent to the NLP module. In the NLP module, first the phonetic transcription of the utterance is obtained using the G2P converter.

Following a reproduction of the file resulting from the G2P conversion is presented:

o_O l_l á_a

m_m a_6 r_r i_i a_6

After having the phonetic utterance, information about syllabic stress and syllable, word and phrase boundaries is collected.

Following a reproduction of the file resulting from the NLP module is presented:

```
phoneme        syll        stress         word
XX
O               O           0              Ola
l               la          1
a
m               m6          0              m6ri6
6
r               ri          1
i
6               6           0
```

After the NLP tasks have been performed, the collected information is used to create a file with the language context feature information, using the labels for the features considered for the EP HMM-based speech synthesis system, presented in section B.2 from appendix B.

The file with the language context information labels for the utterance <Olá Maria> is presented in section C.4 from appendix C.

This information is the input for the language context decision trees that generate the state durations and the sequence of MFCCs and logF0s values for the synthesis module.

The data generated for this utterance is presented in section C.5, from appendix C. The information presented is the reproduction of the file resulting from the HTS engine, the “trace” file.

The information contained in this file tells us for instance that for the utterance <Olá Maria> there are nine HMMs, one for a silence introduced at the beginning of the utterance to give a more natural result and one for each phoneme in the utterance. The sequence of the nine HMMs produce a sequence of forty five emitting states. The length of the generated speech is 1.422 seconds.

Also presented for each model is the specific information about the generated parameters, like the duration of the phoneme and the leaves of the decision trees, for the spectrum and the frequency, selected to represent the phoneme in the sentence. There is one leaf per state and the choice of the leaf determines the values of the MFCCs and the logF0s. The trace file also gives information about the voiced/unvoiced decision per state and the duration of each state.

In this chapter the implemented TTS was described, explaining the different modules that were developed: the language dependent module involving the contextual information for EP, the NLP tasks and the synthesis module. The chapter ends with the evaluation of the system. Next chapters will propose some system improvements, which will not be integrated in the system during the work of the thesis.

		Symbol *SAMPA*	Symbol *IPA*	Example Word
01	Consonants	p	p	***p***ai
02		b	b	***b***ar
03		t	t	**t**ia
04		d	d	**d**ata
05		k	k	**c**asa
06		g	g	**g**ato
07		f	f	**f**érias
08		v	v	**v**aca
09		s	s	**s**elo
10		z	z	a**z**ul
11		S	ʃ	**ch**ave
12		Z	ʒ	a**g**ir
13		m	m	**m**eta
14		n	n	**n**eta
15		J	ɲ	se**nh**a
16		l	l	**l**ado
17		l∼	ɫ	sa**l**
18		L	ʎ	fo**lh**a
19		r	r	ca**r**o
20		R	R	ca**rr**o
21	Vowels	i	i	f**i**ta
22		e	e	p**ê**ra
23		E	ɛ	s**e**ta
24		a	a	c**a**ro
25		6	ɐ	cam**a**
26		O	ɔ	c**o**rda
27		o	o	s**o**pa
28		u	u	m**u**da
29		@	ɨ	dest**e**
30		i∼	ĩ	p**in**ta
31		e∼	ẽ	m**en**ta
32		6∼	ɐ̃	m**an**ta
33		o∼	õ	p**on**ta
34		u∼	ũ	m**un**dial
35	Semivowels	j	j	pa**i**
36		w	w	pa**u**
37		j∼	j̃	mu**i**to
38		w∼	j̃	pã**o**

Table 4.1: European Portuguese Phonetic Set, in SAMPA and IPA

European Portuguese Phonemes		
	Voiced	Unvoiced
01	a	p
02	6	t
03	E	k
04	O	s
05	e	S
06	@	f
07	i	
08	o	
09	u	
10	j	
11	w	
12	6∼	
13	e∼	
14	i∼	
15	o∼	
16	u∼	
17	j∼	
18	w∼	
19	b	
20	d	
21	g	
22	m	
23	n	
24	J	
25	z	
26	v	
27	Z	
28	l	
29	l∼	
30	L	
31	r	
32	R	

Table 4.2: Voiced and unvoiced European Portuguese Phonemes, in SAMPA.

European Portuguese Phonemes		
	Continuant	Non-continuant
01	a	R
02	6	r
03	E	l
04	O	l~
05	e	L
06	@	p
07	i	b
08	o	t
09	u	d
10	j	k
11	w	g
12	6~	m
13	e~	n
14	i~	J
15	o~	
16	u~	
17	j~	
18	w~	
19	s	
20	S	
21	z	
22	f	
23	v	
24	Z	

Table 4.3: Continuant and non-continuant European Portuguese Phonemes, in SAMPA.

Table 4.4: Portuguese Graphemes Phonetic Transcription

	Grapheme	**Phonemes**
01	ú	u∼, u
02	õ	o∼
03	ô	o
04	ó	O
05	í	i∼, i
06	ê	6∼j∼6∼j∼, e∼, 6j, 6∼j∼, e
07	é	E, 6, 6∼j∼, e
08	ç	s
09	ã	6, 6∼
10	â	6, 6∼
11	á	a, &
12	à	a
13	z	Z, S, z
14	x	S, ks, z, s
15	v	v
16	u	@, u∼, &, w, u, w∼
17	t	t
18	s	Z, S, &, z, s
19	r	R, &, r
20	q	k
21	p	&, p
22	o	O, o∼, @, &, w, u, w∼, o
23	n	J, &, n
24	m	&, m
25	l	L, l∼, l
26	k	k
27	j	Z
28	i	i∼, @, &, j∼, j, i
29	h	&
30	g	Z, g
31	f	f
32	e	i∼, E, 6, @, e∼, 6∼, &, 6j, 6∼j∼, j∼, j, i, e
33	d	&, d
34	c	S, &, s, k
35	b	b
36	a	a, 6, 6∼, &, 6∼,w∼, o

Table 4.5: Prediction results of G2P transcription

	Result
Log likelihood	-12329.53
Performance	88.94 %

Table 4.6: Prediction results of syllable boundary detection

	Result
Log likelihood	-1200.10
Performance	97.64 %

Table 4.7: Prediction results of stress prediction

	Result
Log likelihood	-5465.59
Performance	85.57 %

Table 4.8: Confusion matrix for consonants (in percentage), for Maximum Entropy Model

	ks	S	d	Z	k	g	t	J	v	s	b	&	z	r	l∼	L	f	n	m	l	p	R
ks	87	10	-	-	-	-	-	-	-	-	-	-	3	-	-	-	-	-	-	-	-	-
S	1	96	-	-	-	-	-	-	-	2	-	-	1	-	-	-	-	-	-	-	-	-
d	-	-	99	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Z	-	4	-	93	-	-	-	-	-	-	-	-	3	-	-	-	-	-	-	-	-	-
k	-	-	-	-	97	-	-	-	-	-	1	2	-	-	-	-	-	-	-	-	-	-
g	-	-	-	2	-	98	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
t	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
J	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-
v	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-
s	1	4	-	-	-	-	-	-	-	93	-	1	1	-	-	-	-	-	-	-	-	-
b	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-
z	-	3	-	-	-	-	-	-	-	3	-	-	94	-	-	-	-	-	-	-	-	-
r	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-
l∼	-	-	-	-	-	-	-	-	-	-	-	-	-	-	83	-	-	-	-	17	-	-
L	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-
f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-	-	-	2	-	-	-	-	-	98	-	-	-	-
m	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-
l	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2	-	-	-	-	98	-	-
p	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	99	-
R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100

Table 4.9: Confusion matrix for vowels (in percentage), for Maximum Entropy Model

	e∼	a	E	j	j∼	u	*	e	**	u∼	&	i∼	w	w∼	@	i	6	O	o∼	6∼	6j	o
e∼	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
a	-	83	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16	-	-	-	-	-
E	-	-	78	1	-	-	-	8	-	-	4	-	-	-	5	4	-	-	-	-	-	-
j	-	-	-	77	2	-	-	4	-	-	6	-	-	-	-	11	-	-	-	-	-	-
j∼	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
u	-	-	-	-	-	63	-	-	-	-	12	-	17	-	4	-	-	-	-	-	-	4
*	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
e	-	-	-	-	-	-	-	66	-	-	27	-	-	-	7	-	-	-	-	-	-	-
**	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-	-
u∼	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-	-	-
&	-	-	1	-	-	1	-	1	-	-	90	-	3	-	2	2	-	-	-	-	-	-
i∼	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-	-	-
w	-	-	-	-	-	18	-	-	-	-	17	-	59	-	-	-	-	1	-	-	-	5
w∼	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-	-	-	-	-	-
@	-	-	-	-	-	-	-	4	-	-	40	-	12	-	36	4	-	-	-	-	-	4
i	-	-	-	-	-	-	-	3	-	-	-	-	-	-	-	97	-	-	-	-	-	-
6	-	8	-	-	-	-	-	-	-	-	3	-	-	-	-	-	89	-	-	-	-	-
O	-	-	-	-	-	3	-	-	-	-	3	-	5	-	-	-	-	75	-	-	-	14
o∼	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-	-
6∼	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-	-
6j	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	-
o	-	-	-	-	-	3	-	-	-	-	2	-	2	-	-	-	-	13	-	-	-	80

* stands for 6∼j∼ and ** stands for 6∼j∼6∼j∼

Sampling frequency	8 KHz	10 KHz	12 KHz	16 KHz
Mel scale	0,31	0,35	0,37	0,42
Bark scale	0,42	0,47	0,50	0,55

Table 4.10: Examples of frequency warping factor to approximate human auditory frequency scales

Table 4.11: Test-set for MOS test, with sentences presented in *SAMPA*

Sentence Index	*SAMPA* Translation
Mói-te a mão.	**mOjt6**m**6~w**
Põe da cor do céu.	**po~j~d**6**kor**dw**sEw**
Ao lado mora um velho senhor.	**awl**adwm**O**r**w~v**E**L**ws**@J**or
A Joana tem a chave do carro!	6**Zu**a**n**6t6j~6**S**av@dwka**R**w
Eu pintei a manta com muita magenta!	**e**wp**i**~t6j6m**6**~t6k**o**~m**u**~j~t6m6Z**e**~t6
O pai foi à casa da tia buscar gotas e selos.	upajfojaka**z**6d6t**i**6**b**uSkar**g**ot6ziselwS

Chapter 5

Speech Corpus for Context-Based Text-to-Speech Systems

In a TTS system, the speech synthesis quality is strongly affected by the quality of the speech corpus, the quality of the NLP module and the quality of the speech generation module. The speech corpus quality, on its turn, depends on the quality of the corpus design, the quality of the annotations and transcriptions and the quality of the recording system.

Nowadays TTS systems are usually data-driven statistical systems. This means that they rely on the quality of the corpus used to produce better or worse results. Besides that, they are often systems using the language contexts for choosing the best speech units for a particular target. It is of extremely importance that the speech corpus design includes as many contexts as possible. Usually, statistical methods are used to search the intended speech units, within the considered language contexts, in large amounts of texts. For this reason the speech corpora for generic vocabulary systems are very large, sometimes too large to be considered.

There are many context features that may be considered. As an example, section B.2 from appendix B presents the list of features considered for the EP HMM-based speech synthesis system [Barros et al., 2005].

As a solution to improve the results of the implemented TTS system, which depends on the coverage of EP phonetic units in different language contexts, efforts were concentrated on the corpus design quality. A new speech corpus especially designed for language context-based systems with small footprint engines was implemented.

This manually designed speech corpus intends to achieve, within a limited size, a large amount of important language contexts. To do so, one sentence was constructed targeting each diphone of the language. These sentences included as many contexts as possible. Through statistical searches in different text corpora it is possible to design a speech corpus that considers as many features as needed, but for that a very large corpus is needed or some compromises have to be made in order to limit its size.

This speech corpus is designed for EP, although the idea can be extrapolated to other languages. The same data should be used to train the NLP tasks and the synthesis engine, as this way a better synchronization between the units selected by the NLP and the units that can be found in the synthesizer database is achieved.

5.1 Speech Corpus Methodology

The methodology used to design the corpus took the 38 phonemes from EP and the silence unit into consideration, to construct one sentence dedicated to each of these 39 units combined with each other and themselves. Excluding the combination of silence with itself there were 1520 possible sentences to be constructed. The result was a speech corpus with 1436 sentences, as the rest of the combinations are not possible due to language rules that will be explained later in the chapter.

The sentences were all manually constructed, trying to take into account a number of language context factors and considering, but not limiting to, the most frequently used words in the lists relating to EP usage of the words, collected from the first Morpholympics for Portuguese [Morfolimpiadas, visited in 2010]. These lists were extracted from a set of 613 different texts, from different fields (literature, politics, advertising, informal chats, general news, etc.), collected through different sources (newspapers, books, net, advertising, etc.), containing 80.903 text words corresponding to 17.128 different units. More information about the description of the texts and the lists, with statistics, is available on the website [Morfolimpiadas, visited in 2010].

The contextual factors that were considered when building the sentences are:

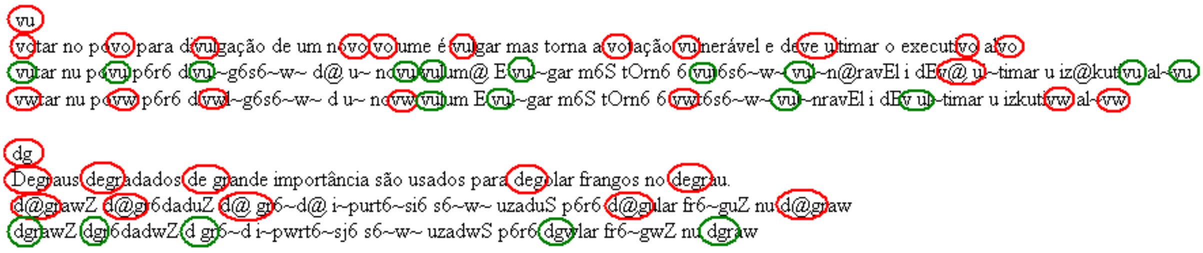
- At the diphone level, occurrences of the diphone:
  - positioned at the beginning/end of the sentence;
  - positioned at the beginning/middle/end of the word;
  - between two words.

- At the word level, occurrences of the:
  - Words containing the target diphone at the beginning/middle/end of the sentence;
  - Combinations with the target diphone between two words positioned at the beginning/middle/end of the sentence.

The speech corpus is presented in three forms:

- The orthographic sentences;
- A phonetic transcription considering the word coarticulation (that is a natural effect in the EP continuous speech), following the G2P rules for the EP language;
- A phonetic transcription considering the word coarticulation and the vocalic reduction effects, common in the EP language.

Appendix E presents a representative subset of the complete corpus. Figure 5.1 shows two examples of sentences, for the phonetic sequences /vu/ and /dg/, where all the possibilities of the sequence occurrences are marked with circles, independently of being realized or not. In the example, the difference in the number of the sequence occurrences for both types of phonetic transcription can be observed, and the fact that some diphones do not exist in the language but may appear when considering vocalic reduction, as it will be explained below, is possible to be realized.



vu
votar no povo para divulgação de um novo volume é vulgar mas torna a votação vulnerável e deve ultimar o executivo alvo
vutar nu povu p6r6 divul~g6s6~w~ d@ u~ novu vulum@ E vul~gar m6S tOrn6 6 vut6s6~w~ vul~n@ravEl i dEv@ ul~timar u iz@kutivu al~vu
vwtar nu povw p6r6 divwl~g6s6~w~ d u~ novw vulum E vul~gar m6S tOrn6 6 vwt6s6~w~ vul~nravEl i dEv ul~timar u izkutivw al~vw

dg
Degraus degradados de grande importância são usados para degolar frangos no degrau.
d@grawZ d@gr6daduZ d@ gr6~d@ i~purt6~si6 s6~w~ uzaduS p6r6 d@gular fr6~guZ nu d@graw
dgrawZ dgr6dadwZ d gr6~d i~pwrt6~sj6 s6~w~ uzadwS p6r6 dgwlar fr6~gwZ nu dgraw

Figure 5.1: Example of the sentences in the Speech Corpus.

For the phonetic transcription considering vocalic reduction, the NLP tool developed in this thesis and described in section 4.2 was used. It was then fully manually corrected. The phonetic transcription following the G2P rules for the EP language was manually developed.

For a better understanding of the G2P rules for EP by non-speakers, some grapheme constructions that can produce each of the phonemes are listed in table D.1, from appendix D. The list from table D.1 does not intend to be a complete list that covers all the cases, but a reference list with common examples. In table 4.4, from section 4.2, a list with the Portuguese graphemes' phonetic transcriptions is presented.

5.2 Speech Corpus Description

The speech corpus presented here has a total of 1436 sentences, comprising 5853 different words, in a total of 21500 words occurrences.

Two phonetic transcriptions for the orthographic sentences are provided, one following the G2P rules for the EP language and the other considering the vocalic reduction effect that is common in the EP language. The number of distinct words in the phonetic speech corpus transcribed following the G2P rules is 6264 and in the one taking into consideration the vocalic reduction the total is 6415. The difference in the number of words is due to the different transcriptions for some words when considering the vocalic reduction effect.

The effect of coarticulation between words is present in the EP continuous speech. Due to this effect, the same word can have a different transcription depending on the context, because there are graphemes that have different phonetic transcriptions depending on the following one. For example, the grapheme <s> is transcribed as a /S/ if at the end of a sentence or followed by an unvoiced consonant (/p, t, k, s, S, f/ for EP); as a /Z/ if followed by a voiced consonant (/b, d, g, m, n, J, z, v, Z, l, l~, L, r, R/ for EP); or as a /z/, if followed by a vowel. Another example is the grapheme <l>, which is transcribed as a /l~/ if at the end of a sentence or followed by a consonant; or as a /l/ if followed by a vowel. One example of these situations in EP words is the word <vai**s**>, which is transcribed as /vaj**S**/ if it is followed by a word starting with an unvoiced consonant or if it is at the end of a sentence; as /vaj**Z**/ if it is followed by a voiced consonant; and as /vaj**z**/ if it is followed by a vowel. Another example is the word <ma**l**>, which is transcribed as /ma**l~**/ if it is followed by a word starting with a consonant or if it is at the end of a sentence; and as /ma**l**/ if it is followed by a vowel.

There are other coarticulation effects in continuous speech, which are associated with faster rhythms of speech, which were not considered in the phonetic transcriptions, although the orthographic sentences include cases that lead to examples of these. The effect is related to words ending with a vowel and followed by another word also starting with a vowel. In these cases, both vowels are concatenated in a third vowel. One of these cases is when a word ends with a vowel <a> and the following word starts with an <a> as well. When this happens, instead of having the transcription /6 6/, the two vowels can be concatenated and substituted by the vowel /a/. E.g. <par**a** **a**> would be transcribed as /p6r **a**/ and not as /p6r**6** **6**/. A similar result is achieved when a word ends with a vowel <o> and the following word starts with a <a>. Instead of having the transcription /u 6/, the two vowels can be concatenated and substituted by the vowel /a/. E.g. <como a> would be transcribed as /kum a/ and not as /kumu 6/. Also, when a word ends with a vowel <a> and the following word starts with a <o>, instead of having the transcription /6 u/ the two vowels can be concatenated and substituted by the vowel /O/. E.g. <par**a** **o**> would be transcribed as /p6r **O**/ and not as /p6r**6** **u**/.

Vocalic reduction is another effect present in the EP continuous speech. It can be reflected by suppression of the phoneme /@/, by reduction or suppression of the phoneme /u/, or by reduction of the phonemes /u∼/, /i/ and /i∼/. Considering this effect, some language contextual factors are present that otherwise would not be found.

The suppression of the phoneme /@/ can happen in the middle of the words or at the end of the words or sentences, due to vocalic reduction depending on the rhythm of speech. The decision was made to include sentences for diphones that are not existent in the EP but could appear if this effect is considered in the G2P transcription. Some examples of this phenomenon are found in the words: <**de**> that would be transcribed as /**d**/ instead of /**d@**/; <**ded**ica> that would be transcribed as /**dd**ik6/ instead of /**d@d**ik6/; <po**te**> that would be transcribed as /pO**t**/ instead of /pO**t@**/; and <a**pet**e**ce**> that would be transcribed as /6**pt**E**s**/ instead of /6**p@t**E**s@**.

The reduction or suppression of the phoneme /u/ happens in the middle of the words or at the end of the words or sentences, due to vocalic reduction depending on the rhythm of speech. When the vowel /u/ suffers reduction it gives place to the semivowel /w/. An example of a word with a phonetic transcription involving /u/ two times, which usually presents /u/ suppression and reduction, is the word <sé**culo**s>. Following the G2P rules this word is transcribed as /sE**kulu**S/, but commonly is found as /sE**klw**S/ due to vocalic reduction. The paper "Backclose Nonsyllabic Vowel /u/ Behavior in EP: Reduction or Supression" [Barros et al., 2001] covers this phenomenon.

The phonemes /u∼/, /i/ and /i∼/ can suffer reduction, giving place to the semivowels /w∼/, /j/ and /j∼/, respectively, but they cannot suffer suppression. Examples of these situations can be found for instance in words like <conjuntura>, which would be transcribed as /ko∼Zw∼tur6/ instead of /ko∼Zu∼tur6/, or <l**i**t**o**ral>, which would be transcribed as /l**j**t**w**ral∼/ instead of /l**i**t**u**ral∼/, or <**in**venta>, which would be transcribed as /**j**∼ve∼t6/ instead of /**i**∼ve∼t6/.

In the EP many consonants cannot be found next to each other in the same word. Considering the case of the phoneme /@/ vocalic reduction means having to include almost all the combinations between consonants that in another way would not exist.

5.2.1 The Speech Corpus by Graphemes

The 1436 sentences speech corpus comprises 36 graphemes (considering the graphemes with accents - à, á, â, ã, é, ê, í, ó, ô, õ and ú - as individual units) and a total of 101750 grapheme occurrences. The total number of occurrences of each grapheme is shown in figure 5.2.

Although <k> is not a Portuguese grapheme it can appear in some foreign words adopted in Portuguese vocabulary. In this case it appears in the word <snack>, what justifies its presence in the speech corpus.

Figure 5.2: Total Number of Graphemes Occurrences.

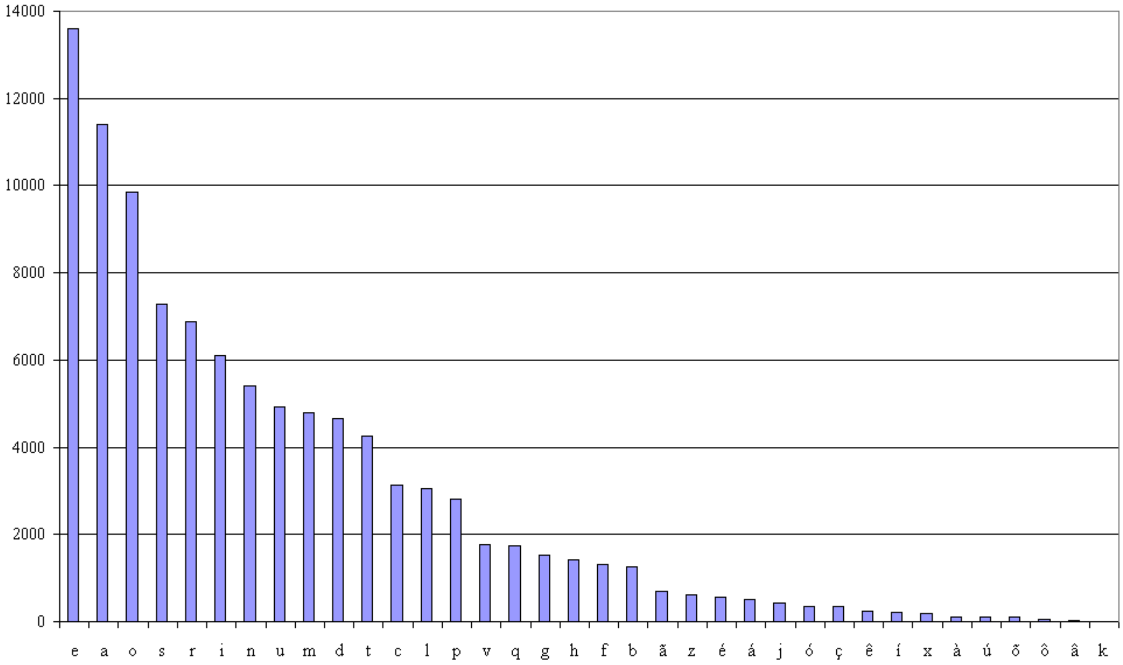
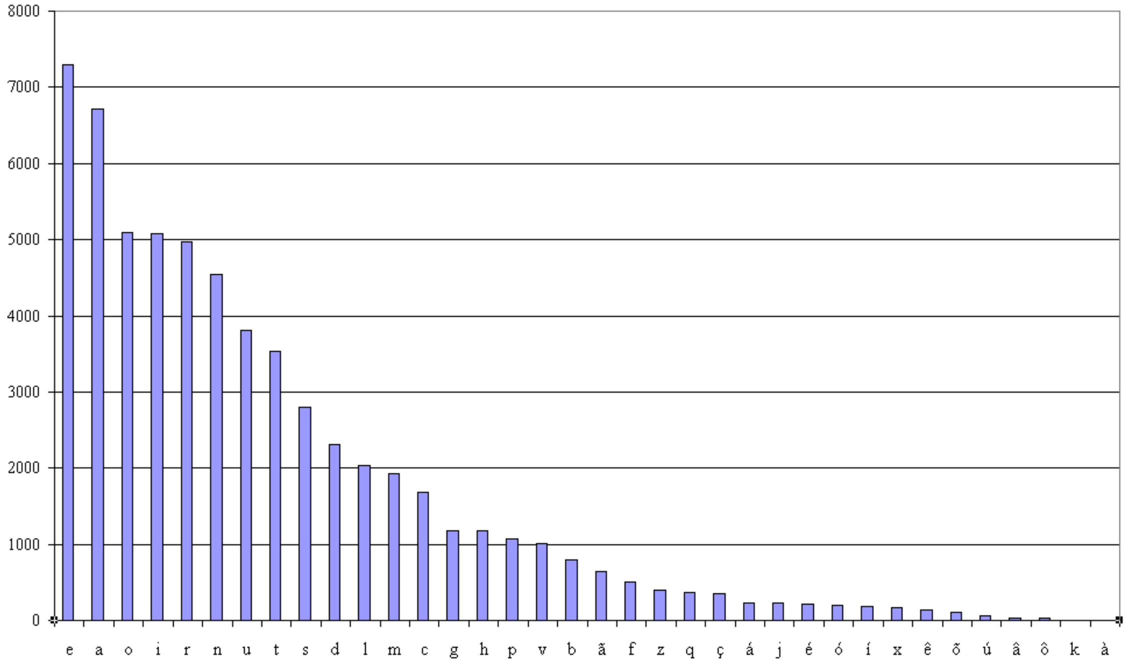


Figure 5.3 presents only the number of occurrences of each grapheme in the middle of a word.

The only graphemes that do not present any occurrence is the <k>, which is not a Portuguese grapheme as explained above, and the <à>, which is a particular case in Portuguese language because it only appears with the contraction between <a a>, so it is not possible in the middle of a word, and represents the contraction of the preposition "a" with the definite article "a" or a pronoun like "aquele", "aquela", or “aquilo”. For example, the sentence <Dou **à** Joana.> (meaning <*I give* ***to*** *Joana.*>) has a grammar construction that comes from <Dou **a a** Joana.> (that would mean: <*I give* ***to the*** *Joana.*>). The examples present in the speech corpus are enumerated below, when describing the number of occurrences of graphemes at the beginning of a word.

Figure 5.3: Number of Graphemes Occurrences in the middle of a Word.



The number of occurrences of each grapheme at the beginning of a sentence and at the beginning of a word are shown in figure 5.4 and in figure 5.5, respectively.

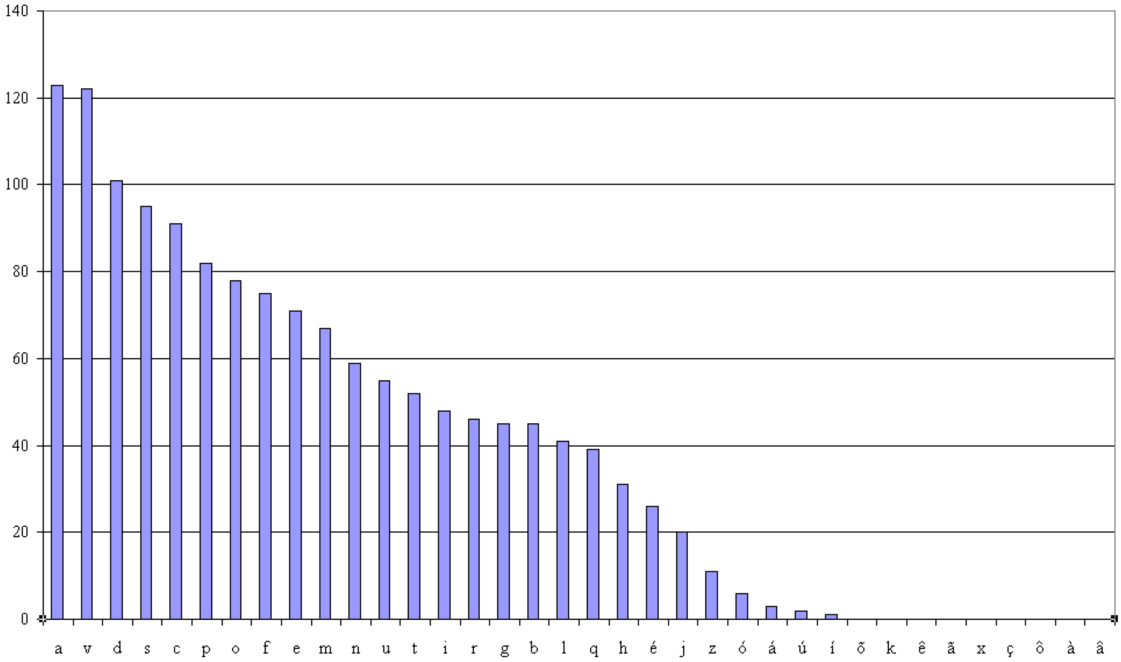
The <ç> is the only consonant that cannot appear at the beginning of a Portuguese word. This grapheme is always phonetically transcribed as /s/, which is produced by <s>, or <c> if followed by <(e,i)>, when at the beginning of a word.

The <ô>, <õ> or <ã> are the vowels that cannot appear at the beginning of a word. Relating to the last two, the nasal vowels, the <˜> is used to nasalize the vowel and in cases with the vowel at the beginning of the word, the vowel is nasalized by using the consonants <(n,m)> in front of it.

The fact that <x> does not show up at the beginning of a sentence, although it would be possible, does not have any influence on the phonetic results because it is always phonetically transcribed as /S/ and the same result can be produced by words started by <ch>. The speech corpus includes sentences started for instance by the words <chuva>, <chefe>, or <chegou>.

The graphemes <á> and <à> do not appear at the beginning of a sentence, but their phonetic transcription is /a/ and the speech corpus includes sentences started with for instance <**há**>, which has the same result as the <h> in Portuguese is mute. As a substitution of the result of <ê>, which is /e/ at the beginning of a sentence, the speech corpus includes for instance sentences started with <**e**le>, producing the transcription /**e**l@/.

Figure 5.4: Number of Graphemes Occurrences at the beginning of a Sentence.



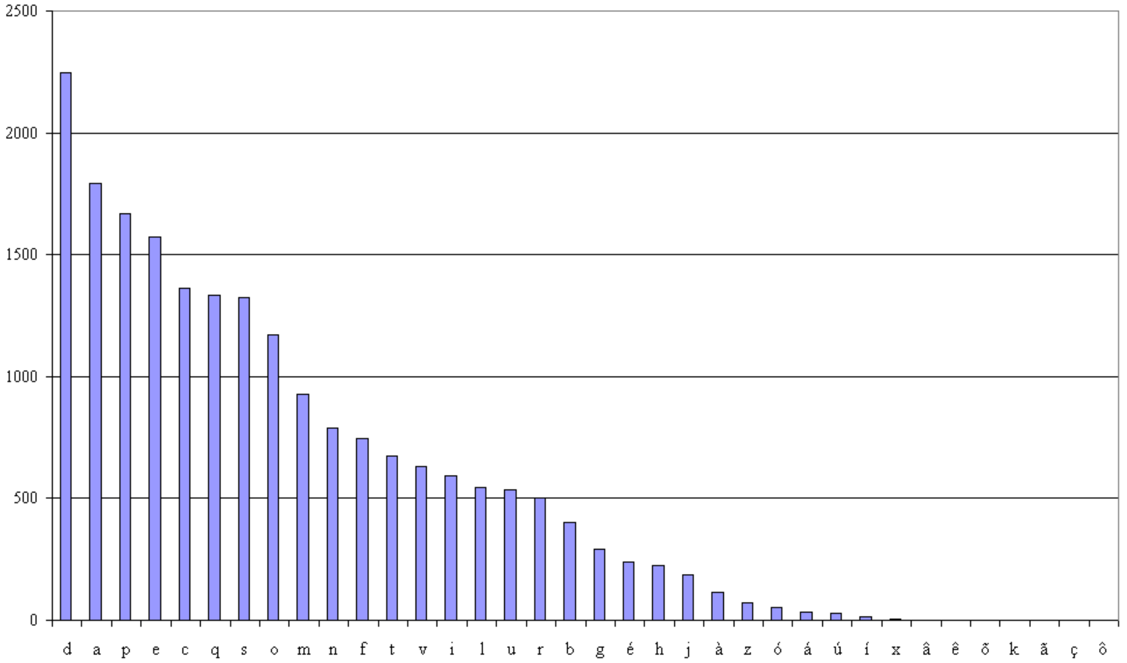
The number of occurrences of each grapheme at the end of a sentence and at the end of a word are shown in figure 5.6 and in figure 5.7, respectively.

The <o>, <a> and <s> are the graphemes that at the end of a word or sentence occurred most. These graphemes are the most common endings for masculine words, feminine words and plurals, respectively. The infinitive form of Portuguese verbs always ending with <r> is what justifies the large amount of this graphemes occurrences at the end of words and sentences.

As it was previously mentioned, <k> is not a Portuguese grapheme and only appears in some foreign words adopted in Portuguese vocabulary.

The <à> is a particular case in Portuguese language because it only appears with the contraction between <a a>, as explained before when presenting the number of graphemes in the middle of a word. In the speech corpus these cases have examples in the following words: <à>, <àquela>, <àquelas>, <àquele> and <àqueles>.

Figure 5.5: Number of Graphemes Occurrences at the beginning of a Word.



From the Portuguese grammar rules, besides the consonants <s> and <r>, already mentioned above, only the consonants <m>, <z>, <l> and <b> can appear at the end of a word. The other cases come from foreign words adopted in the Portuguese vocabulary or acronyms. The cases of imported words present in the speech corpus are: <internet>, <slot>, <camping>, <sporting>, <Herman>, <Félix> and <slogan>. the acronyms are: <PC> (*/pese/*), <TV> (*/teve/*) and <etc.> (*/EtsEt@r6/*).

The case <ô> does not appear at the end of any sentence but its phonetic transcription, /o/, is also obtained with <ou>, which is present in the speech corpus with words like <pagou> and <vou> appearing at the end of sentences. The case <í> has the same phonetic transcription result as <i> in a stressed syllable. In the speech corpus there are sentences ending, for example, with the words <ti>, <si>, <li> and <mexi>.

There are no cases of words ending with <õ>, as the nasal sound at the end of words is achieved by following the vowel with <m>. There are no cases of words ending with <â> in the EP.

Tables F.1 and F.2, from appendix F, present the total number of occurrences of each sequence of two graphemes. The first presents the sequences starting with vowel and the second presents the sequences starting with consonant.

Tables F.3 and F.4, from appendix F, present the number of occurrences of each sequence of two graphemes, only when it appears within the same word. The first presents the sequences starting with vowel and the second presents the sequences starting with consonant.

Tables F.5 and F.6, from appendix F, present the number of occurrences of each sequence of two graphemes, only when it appears between words. The first presents the sequences starting with vowel and the second presents the sequences starting with consonant.

The <h> is a special consonant in the EP, as is described in section 4.1.1. It is followed only by vowels, not by consonants, and it is mute, as was mentioned before. Besides its use at the beginning of some words, preceding a vowel, it is used for the combinations <ch>, <nh> and <lh> that produce the phonemes /S/, /J/ and /L/, respectively.

No other considerations are made to the number of occurrences of sequences of two graphemes, because the phonetic results, from the phonetic transcriptions, are the ones intended to be discussed in this chapter.

Figure 5.6: Number of Graphemes Occurrences at the end of a Sentence.

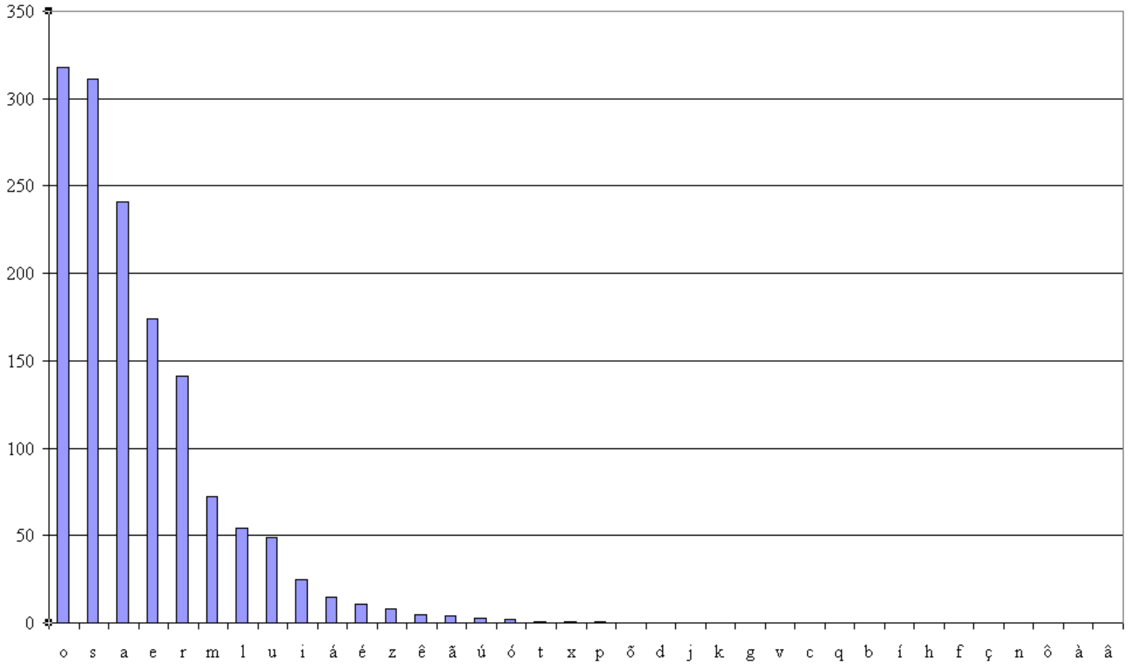
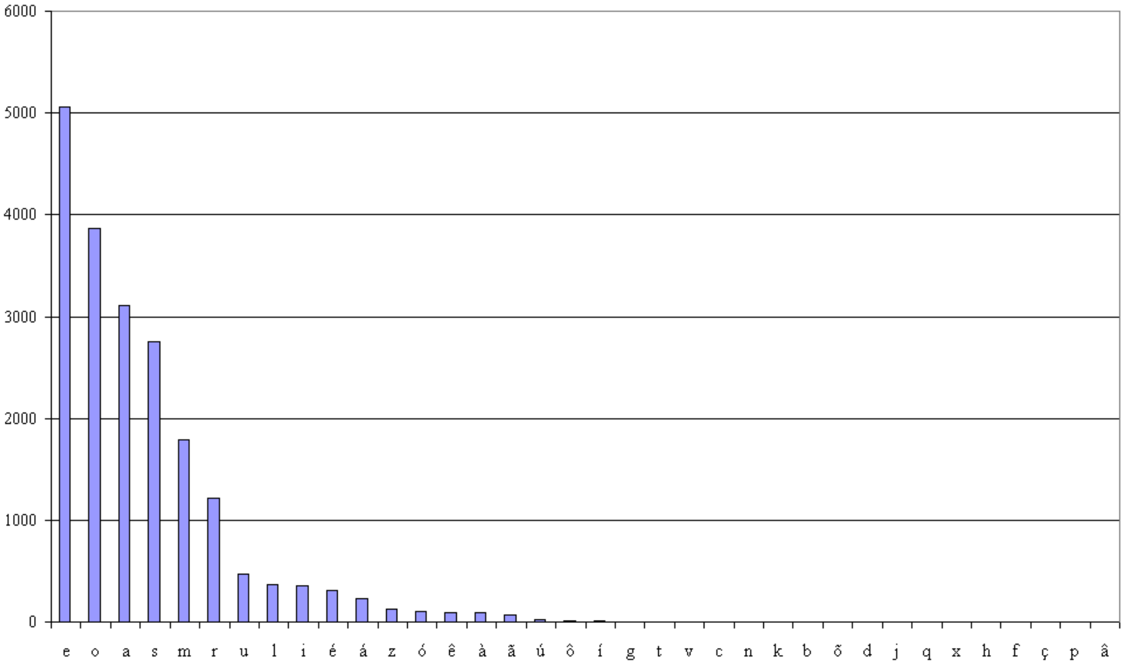


Figure 5.7: Number of Graphemes Occurrences at the end of a Word.



5.2.2 The Phonetic Transcription by Rules

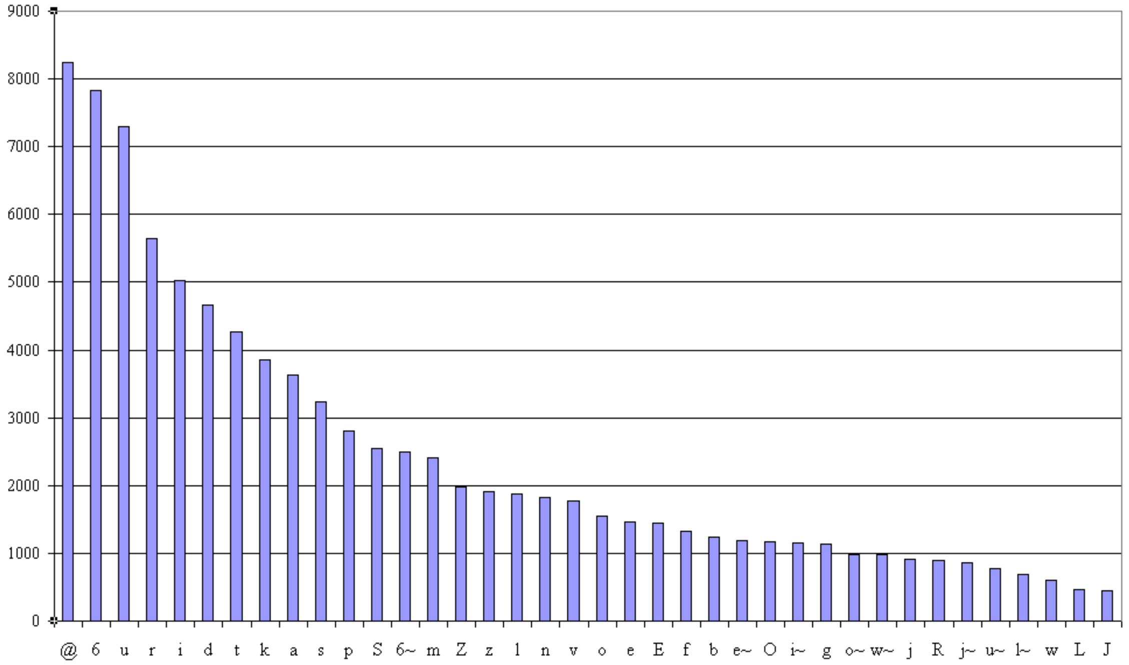
The first phonetic transcription presented follows the G2P rules for the Portuguese language taking into consideration the word coarticulation natural in the EP continuous speech, but not the vocalic reduction effect.

The 1436 sentence speech corpus comprises 38 phonemes and 92618 phoneme occurrences. The total number of occurrences of each phoneme is shown in figure 5.8.

Figures 5.9 to 5.13 present separate charts with statistics related to the language contexts that were considered. The number of occurrences of each phoneme in the middle of a word are presented in figure 5.9. The number of occurrences of each phoneme at the beginning of a sentence and at the beginning of a word are presented in figure 5.10 and in figure 5.11. And the number of occurrences of each phoneme at the end of a sentence and at the end of a word are presented in figure 5.12 and in figure 5.13, respectively.

In the EP language it is not possible to have words starting with the phonemes /l∼/, /r/ or /J/, even considering phonetic effects inherent to continuous speech, unless it is some particular word like words imported from other Portuguese language dialects. This is the case with the word <nhangue>, present in the speech corpus at the beginning of a sentence. Nhangue was imported from the Angolan Portuguese (AP) to represent the name of an African bird or a place in Angola. Section 4.1.1 of chapter 4 lists the countries that use the Portuguese language.

Figure 5.8: Total Number of Phonemes Occurrences, by Rules.



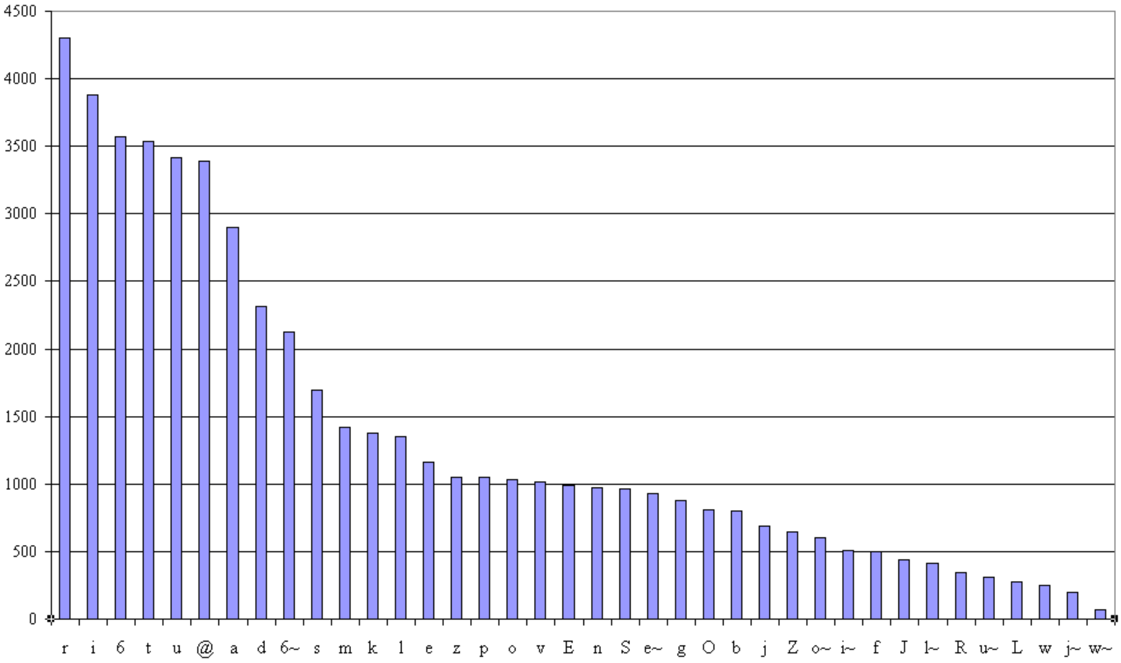
When considering the phonetic transcription following the G2P rules it is not possible to have words starting with a semivowel, although this can happen when considering the vocalic reduction effect as it can be seen in figures 5.16 and 5.17.

All the vowels except /e∼/ can be found at the end of a word or sentence. The consonants /d/, /k/, /g/, /t/, /J/, /v/, /s/, /L/, /f/, /n/, /m/, /l/, /p/ and /R/ are never found at the end of a word or sentence, although due to certain particular cases of foreign words imported into the EP vocabulary, cases of /t/ and /s/ at the end of sentence, and /k/, /g/, /t/, /s/ and /n/ at the end of word can be found in the speech corpus. Such words are <internet>, <Félix>, <snack>, <camping>, <sporting>, <slot>, <homeless>, <Herman> and <slogan>. Also the particular case of the acronym <INEM>, transcribed as /inEm/ gives the case of /m/ at the end of word.

There are almost no cases of Portuguese words ending with a <b>, which explains why there are no cases for sentences ending with /b/. The case of a word ending with <b> present in the speech corpus is the word <sob>.

The consonants /Z/, /z/ and /l/ cannot be found at the end of a sentence or of an isolated word, but they can be found at the end of a word in continuous speech due to the effect of coarticulation of words. The consonants /S/, /l∼/ and /r/ can be found at the end of a sentence or isolated word.

Figure 5.9: Number of Phonemes Occurrences in the middle of a Word, by Rules.



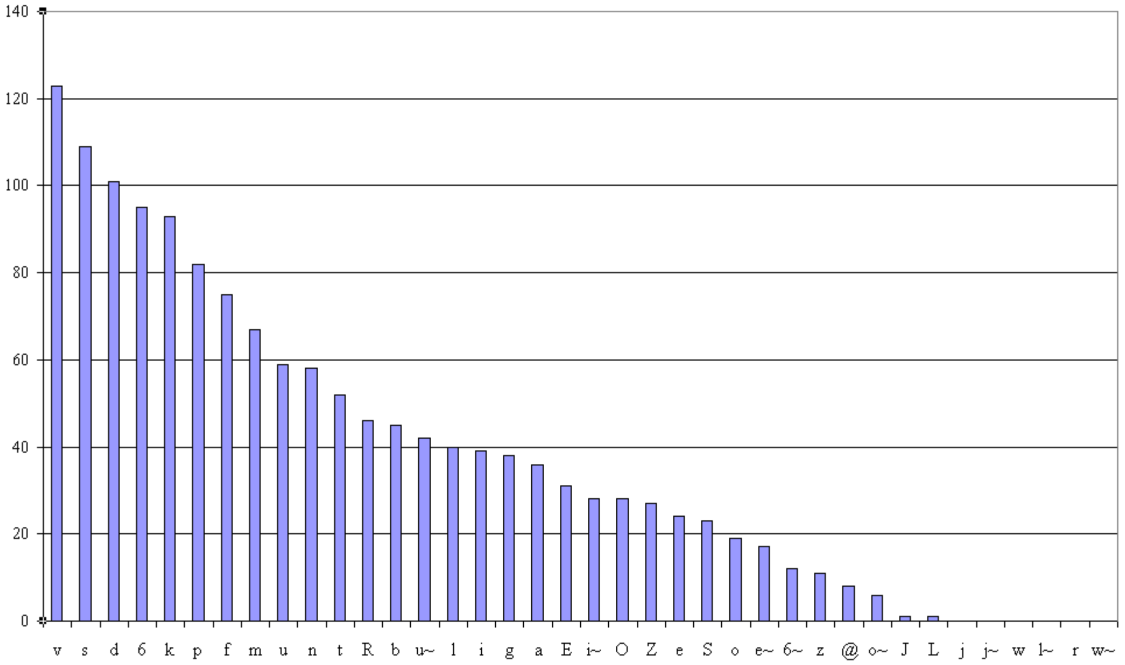
Tables F.7 and F.8, from appendix F, present the total number of occurrences of each diphone by rules. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

Any non-nasal vowel can only be next to a nasal vowel, or vice versa, if there is a small silence between words, in cases of a slow rhythm of speech. Due to the word coarticulation effects, a word that starts with a nasal vowel nasalizes the vowel from the preceding word and vice versa.

Most of the diphones which are composed of two consonants are not present in the EP language, but there are some cases that are common: /pr/, /pl/, /br/, /bl/, /fl/ and /fr/; /Z/ followed by a voiced consonant; /S/ followed by an unvoiced consonant; /ks/ that transcribes the <x> in some cases, as it can be seen in section 4.1.1; /ps/, /pn/, /pt/, /kt/, /kn/, /gn/, /tn/, /gm/ and /tm/; /bt/, /bS/, /bZ/, /bz/, /bs/, /bv/ and /bm/; /dZ/, /dv/, /dm/ and /dr/; and /l~S/. In the speech corpus there are other cases, /sn/, /sl/, /sp/, /gS/, /nS/, /tS/, /gz/, /pz/ and /tz/, due to the foreign words imported to EP vocabulary, <snack>, <slide>, <sloganS>, <slot>, <sporting>, <holdings>, <gangs>, <vips> and <Hertz>. The diphone /Rt/ is also not a EP diphone, but it is present in the acronym <RTP>, /ERtepe/.

The <l> followed by a vowel is always transcribed as /l/, even between words, unless there is a small silence in between the words, caused for instance by a comma, when it is then transcribed by a /l~/. It is either not possible to have /l/ or /l~/ followed by /r/, not because of the construction, which does exist, but because in

Figure 5.10: Number of Phonemes Occurrences at the beginning of a Sentence, by Rules.

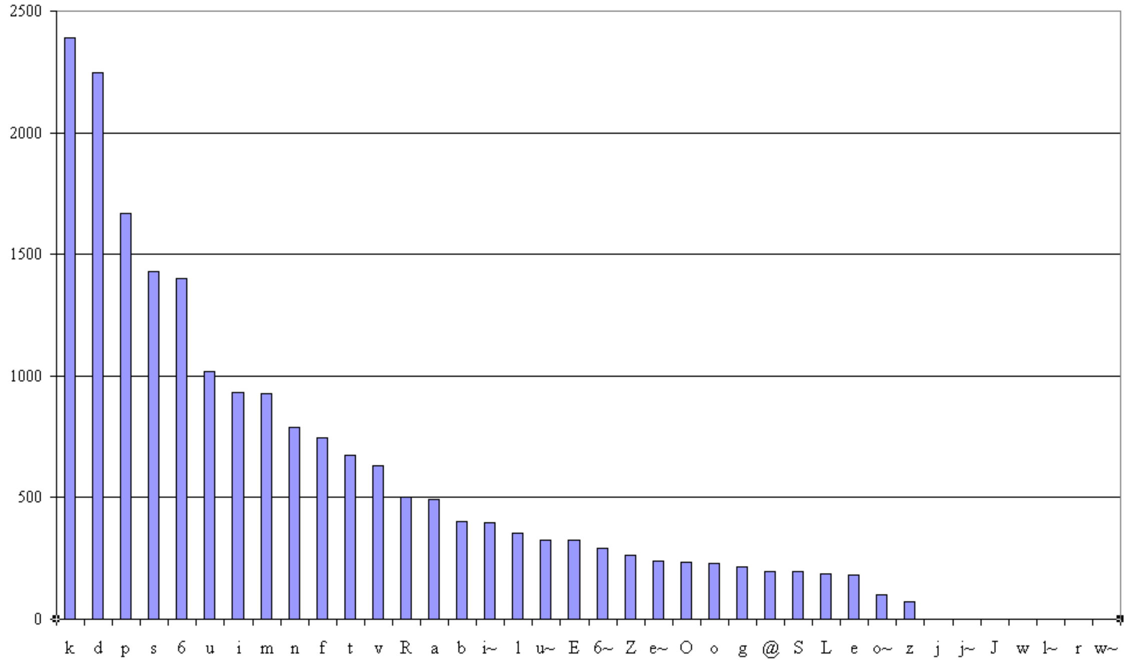


this case the <r> is transcribed by /R/ if in the same word, and there are no words starting with /r/. Another case related to <l> not possible in the EP is having /l/ or /l∼/ followed by /J/, because <lnh> is not an allowed construction and there are no words starting with a /J/. Double "l", <ll>, does not exist in EP words, which makes the diphones among /l∼/, /l/ and /L/ not possible in the same word. The /l∼/ and the /L/ can never be preceded by a consonant or a nasal vowel in the same word, as this would implicate a construction of three following consonants which is not possible in the EP language. Having a nasal vowel before /l∼/ implicates <(m,n)> after the vowel to nasalize it. So, it would be needed to have a consonant before the <l> and another thereafter, to turn it into /l∼/ on these cases, as well as on the consonant cases. It is either not possible to find these cases between words because there are no words starting with /l∼/.

It is not possible to have /J/ preceded by a consonant, as it would lead to another type of construction of three consonants that is not allowed in the EP language, the <nh> that gives the /J/ and any other consonant before. The same applies to nasal vowels preceding a /J/, because to have a nasal vowel before a <nh> would have to be constructed by following the vowel with <m> or <n>.

It is not possible to have /r/ after another /r/ or /R/ in the same word, because the double <r> is always read as /R/. It is either not possible to have the /r/ after a nasal vowel in the same word, because it is the <n>, or the <m>, after a vowel

Figure 5.11: Number of Phonemes Occurrences at the beginning of a Word, by Rules.



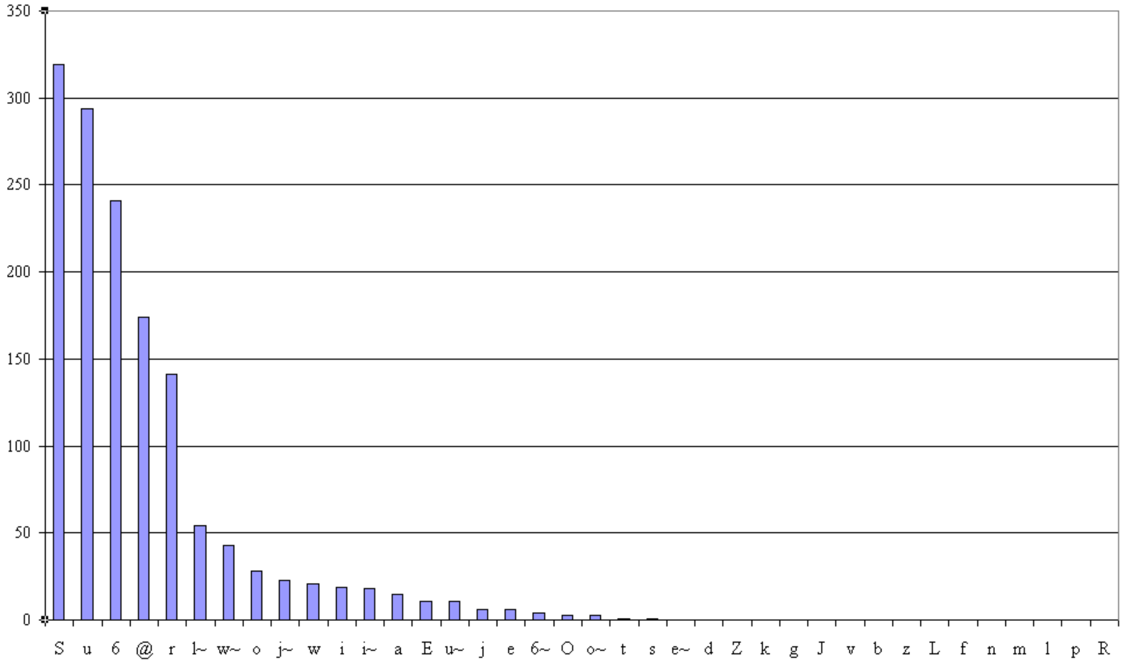
that makes the nasal sound, but after a consonant the <r> is always read as /R/. There is no possibility of having these cases between words because there are no words starting with /r/.

It is not possible to have /n/ or /m/ preceded by a vowel and followed by a consonant, because in these cases the <m> and the <n> are used to nasalize the precedent vowel. Following this rule, it is not possible to have /n/ or /m/ followed by a consonant, because a word construction of three consonants which is not allowed in EP would be needed. There is one exception to these statements, which is the case of <m> followed by <n> and preceded by <a> or <o>, for example in the word <amnistia> or any word using the Portuguese prefix <omni>, which are transcribed as /6mniSti6/ and /Omni/, respectively.

The phoneme /e∼/ is not found at the end of a word or after /e∼/, /6∼/ and /o∼/, unless influenced by some regional accents/dialects, for instance in <lêem>, <têm> and <voem>, which are transcribed as /le∼6∼j∼/, /t6∼j∼6∼j∼/ and /vo∼6∼j∼/, but could be found in some regional accents as /le∼e∼/, /t6∼e∼/ and /vo∼e∼/, respectively.

In the EP language the diphones /Ou/, /Ej/, /@j/, /aE/, /@E/, /aa/, /6a/, /ea/, /Ea/, /oa/ and /Oa/ are not found in the same word, but can be found between words.

Figure 5.12: Number of Phonemes Occurrences at the end of a Sentence, by Rules.



Tables F.9 and F.10, from appendix F, present the number of occurrences of each diphone, by rules, only when it appears within the same word. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

Tables F.11 and F.12, from appendix F, present the number of occurrences of each diphone, by rules, only when it appears between words. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

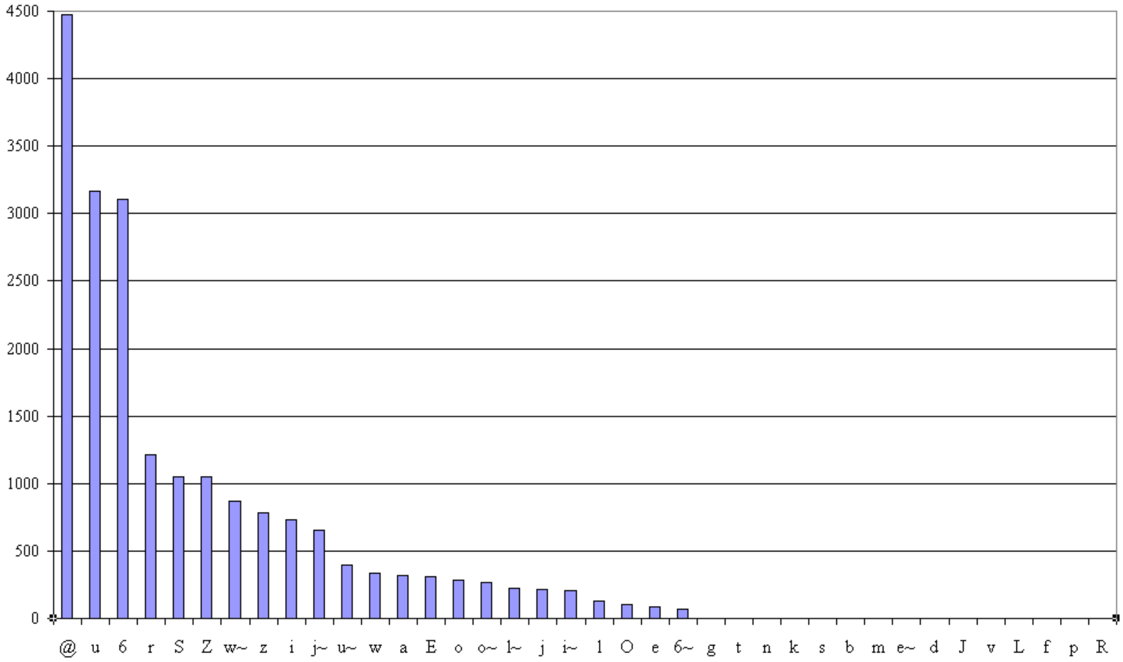
The same tables are presented below, in tables F.19, F.20, F.23 and F.24, with the cases that are not possible according to the G2P rules for EP highlighted with colors.

5.2.3 The Phonetic Transcription with Vocalic Reduction

The other phonetic transcription implemented to present the speech corpus considers the vocalic reduction effect that is common in the EP language, besides the word coarticulation. Due to the vocalic reduction there are fewer phoneme occurrences in this transcription than in the phonetic transcription by rules. The 1436 sentences speech corpus comprises 38 phonemes and 84846 phoneme occurrences.

The total number of occurrences of each phoneme is shown in figure 5.14.

Figure 5.13: Number of Phonemes Occurrences at the end of a Word, by Rules.



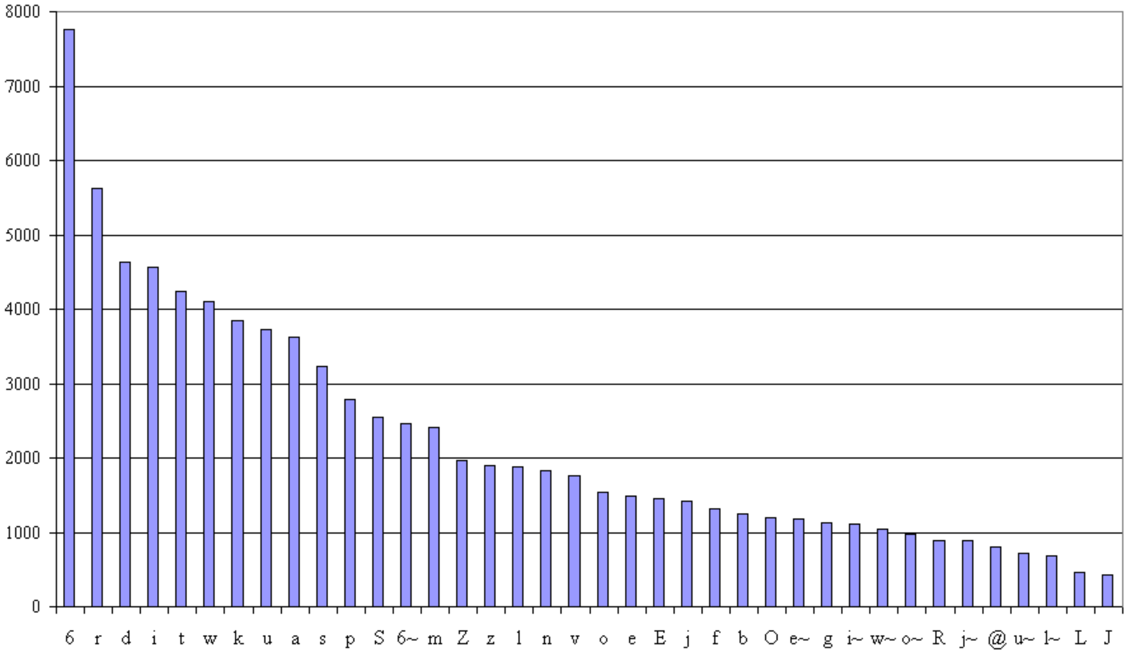
Figures 5.15 to 5.19 present separate charts with statistics related to the language contexts that were considered. The number of occurrences of each phoneme at the middle of a word are shown in figure 5.15. The number of occurrences of each phoneme at the beginning of a sentence and at the beginning of a word are shown in figure 5.16 and in figure 5.17. And the number of occurrences of each phoneme at the end of a sentence and at the end of a word are shown in figure 5.18 and in figure 5.19, respectively.

As it was explained earlier, in section 5.2.2, in the EP language it is not possible to have words starting with the phonemes /l∼/, /r/ or /J/, even considering phonetic effects inherent to continuous speech, unless it is some particular word such as words imported from other Portuguese language dialects, as is the case with the word <nhangue>, imported from the AP dialect, present in the speech corpus at the beginning of a sentence.

Following the G2P rules it is not possible to have words starting with a semivowel, although this can happen when considering the vocalic reduction effect, which is the reason for finding some cases for each semivowel at the beginning of a word, and for /j/ and /j∼/ at the beginning of sentences.

Also due to the vocalic reduction effect, the vowel /@/ can be suppressed. This makes it possible to find almost all consonant combinations. Also the consonant /S/ can be found before a voiced consonant from the transcription of <che> or <xe> that becomes /S/ instead of /S@/.

Figure 5.14: Total Number of Phonemes Occurrences, considering vocalic reduction.



Tables F.13 and F.14, from appendix F, present the total number of occurrences of each diphone considering vocalic reduction. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

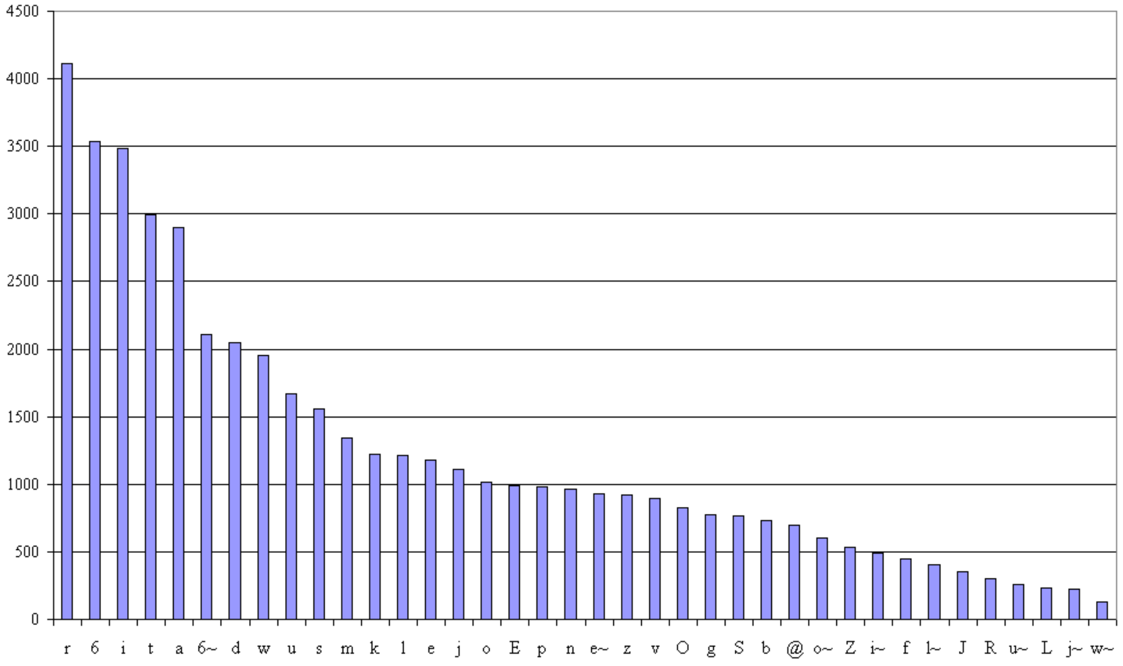
As explained above, in section 5.2.2, a non-nasal vowel can only be next to a nasal vowel, or vice versa, if between words and depending on the rhythm of speech. The phoneme /e∼/ represents a particular case that is not found at the end of a word or after /e∼/, /6∼/ and /o∼/, unless by influence of some regional accents/dialects. The diphones /Ou/, /Ej/, /@j/, /aE/, /@E/, /aa/, /6a/, /ea/, /Ea/, /oa/ and /Oa/ cannot be found in the same word, but can be found between words.

As it was explained before, almost all of the diphones composed by two consonants which are not possible when following the G2P rules, can appear in continuous speech, due to the vocalic reduction effect of the vowel /@/ suppression.

As explained in section 5.2.2, the <l> followed by a vowel is always transcribed as /l/, even if between words, unless there is a small silence in between the words, caused for instance by a comma, when it is then transcribed by a /l∼/. In section 5.2.2 it was also explained that it is not possible to have <l> followed by /r/, /J/, or /l∼/ in the same word and between words; and by /l/ or /L/ in the same word and the /l∼/ can never be preceded by a consonant or by a nasal vowel.

The /@/ is suppressed in most cases of continuous speech, if considering the vocalic reduction effect, in any context of word or sentence.

Figure 5.15: Number of Phonemes Occurrences in the middle of a Word, considering vocalic reduction.



Tables F.15 and F.16, from appendix F, present the number of occurrences of each diphone, considering vocalic reduction, only when it appears within the same word. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

Tables F.17 and F.18, from appendix F, present the number of occurrences of each diphone, considering vocalic reduction, only when it appears between words. The first presents the diphones starting with vowel and the second presents the diphones starting with consonant.

5.3 The vocalic reduction influence

To better understand the influence of the vocalic reduction in the EP continuous speech, a comparison between both phonetic transcriptions, following the G2P rules and considering the vocalic reduction, in the different contexts at word and sentence levels is presented.

Figure 5.16: Number of Phonemes Occurrences at the beginning of a Sentence, considering vocalic reduction.

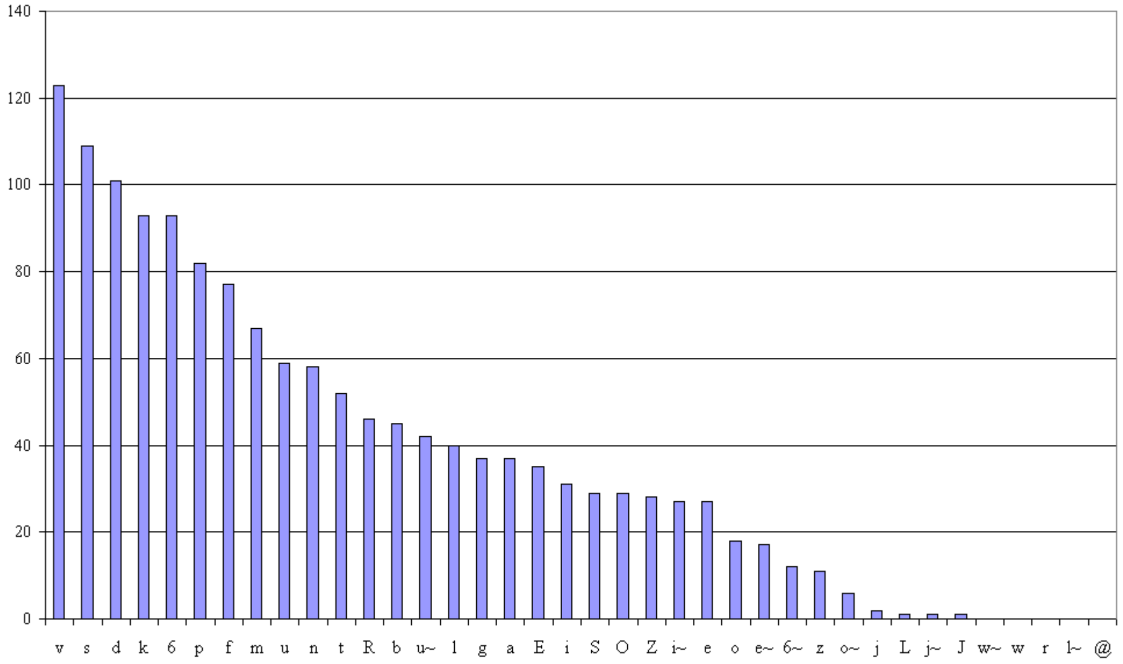
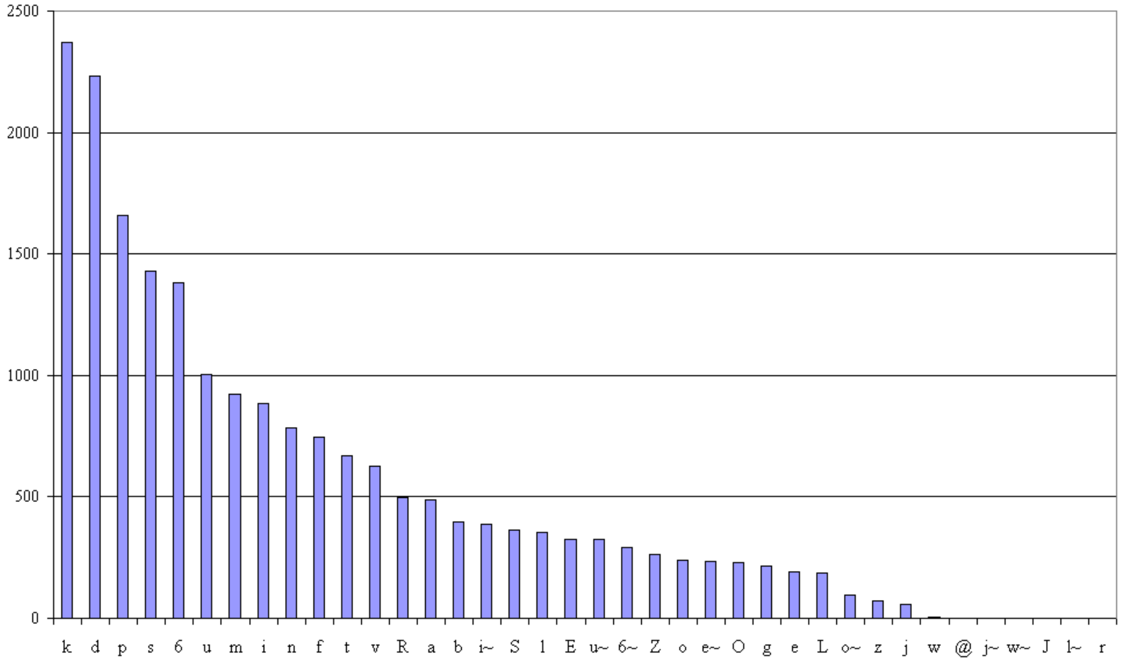


Figure 5.20 presents a chart with the number of occurrences of each phoneme when in the middle of a word, for both phonetic transcriptions, following the G2P rules and considering the vocalic reduction. The same kind of chart is presented for the phonemes at the beginning of a sentence, in figure 5.21, at the beginning of a word, in figure 5.22, at the end of a sentence, in figure 5.23, and at the end of a word, in figure 5.24.

From the figures it is possible to see that the biggest differences between considering or not considering the vocalic reduction are in the number of occurrences of the phoneme /@/, and of the semivowels and corresponding vowels. It is also observed that the number of occurrences of some consonants in a particular context change when considering the vocalic reduction.

Tables F.19 and F.22, from appendix F, present the same tables as tables F.9 and F.10, with the diphones in the same word, from the phonetic transcription following the rules, but highlighting with colors the situations that are not accepted when following the G2P rules. The first table presents the diphones starting with a vowel and the second presents those starting with a consonant. In yellow, the diphones involving the combinations of two consonants that are not possible in the EP are highlighted. In grey and green are the diphones involving nasal and non-nasal vowels together. In grey the diphones starting with a nasal and followed by a non-nasal. In green the diphones starting with a non-nasal and followed by a nasal.

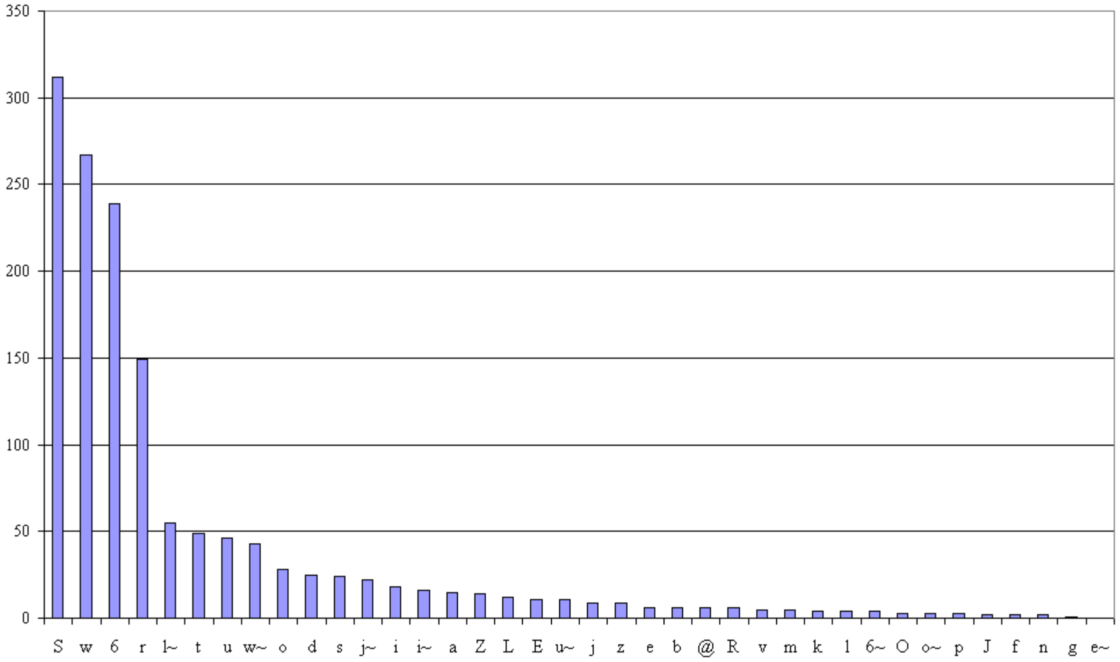
Figure 5.17: Number of Phonemes Occurrences at the beginning of a Word, considering vocalic reduction.



In pink, the other diphones with vowel combinations that are not allowed in the EP are presented. In blue, the cases of diphones that cannot be followed by /J/, /l∼/ or /L/ are shown. Finally, in orange, the diphones starting with /l∼/ and followed by vowel, which are not permitted, are shown.

Tables F.21 and F.22, from appendix F, present the same tables as tables F.15 and F.16, with the diphones in the same word, from the phonetic transcription considering vocalic reduction, but highlighting with colors the situations that are not accepted when following the G2P rules, some of those being possible to happen when considering vocalic reduction. Again, the first table presents the diphones starting with a vowel and the second presents those starting with a consonant. In yellow, the diphones involving the combinations of two consonants that are not possible in the EP are highlighted. Almost all of them are achieved when considering vocalic reduction. In grey and green are the diphones involving nasal and non-nasal vowels together. In grey the diphones starting with a nasal and followed by a non-nasal. In green the diphones starting with a non-nasal and followed by a nasal. In pink, the other diphones with vowel combinations that are not allowed in the EP are presented. In blue, the cases of diphones that cannot be followed by /J/, /l∼/ or /L/ are shown. Finally, in orange, the diphones starting with /l∼/ and followed by vowel, which are not permitted, are shown.

Figure 5.18: Number of Phonemes Occurrences at the end of a Sentence, considering vocalic reduction.



Tables F.23, F.23, F.24 and F.24, present the same kind of color highlights explained in the previous paragraph, but for the diphones between words. These tables were already presented in tables F.11, F.12, F.17 and F.18. In yellow, the diphones involving words ending with /S/, /Z/, /z/ or /l/, which are not permitted, are highlighted. In green are the diphones involving consonants that are not allowed at the end of a word. In pink, the diphones involving the non existent words ending with the vowel /e∼/ and the diphones with words ending in /l∼/ followed by words beginning with a vowel are highlighted. In orange are the diphones involving phonemes that are not allowed at the beginning of words.

The speech corpus here presented was specially designed for context based TTS systems as the EP phonetic units appear in as much different language contexts as possible. The corpus includes one sentence per each diphone of the language. The corpus was phonetically transcribed in order to be possible to infer some conclusions. The speech corpus is the first proposal for improvement of the system, two other proposals are analyzed in the next chapter for future work.

Figure 5.19: Number of Phonemes Occurrences at the end of a Word, considering vocalic reduction.

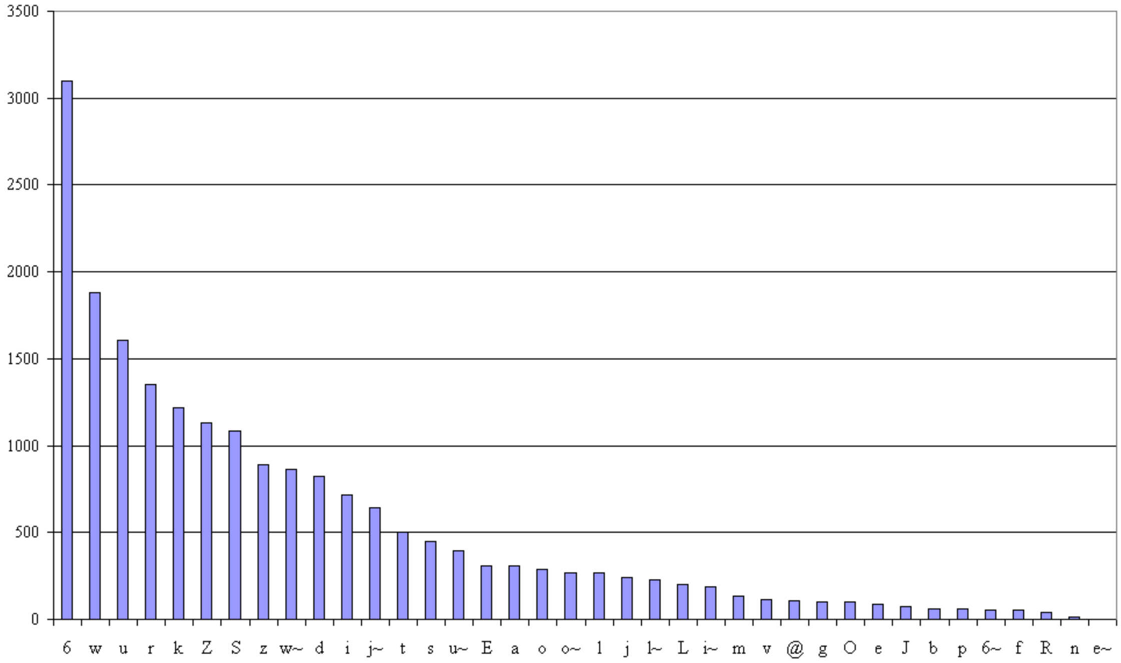


Figure 5.20: Number of Phonemes Occurrences in the middle of a Word, for both phonetic transcriptions.

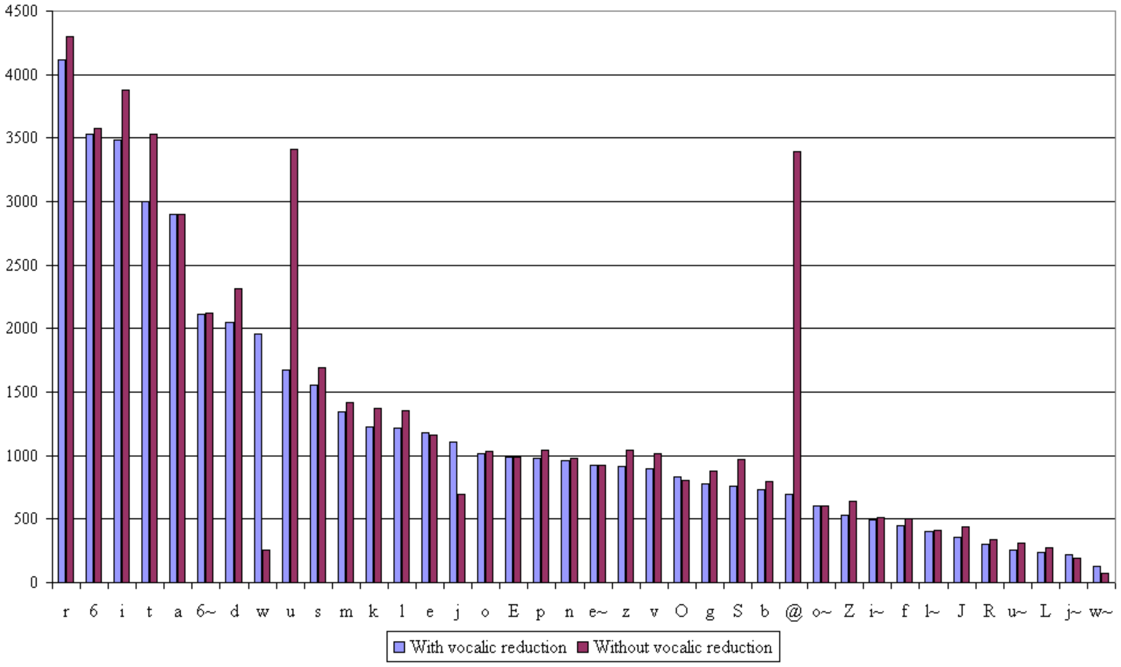


Figure 5.21: Number of Phonemes Occurrences at the beginning of a Sentence, for both phonetic transcriptions.

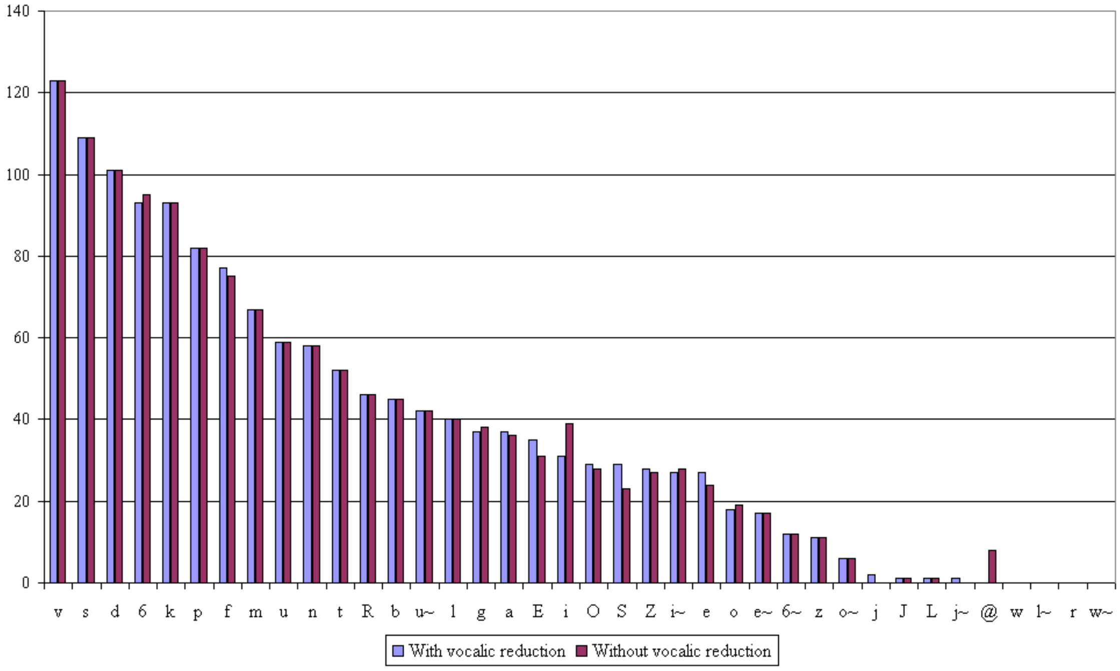


Figure 5.22: Number of Phonemes Occurrences at the beginning of a Word, for both phonetic transcriptions.

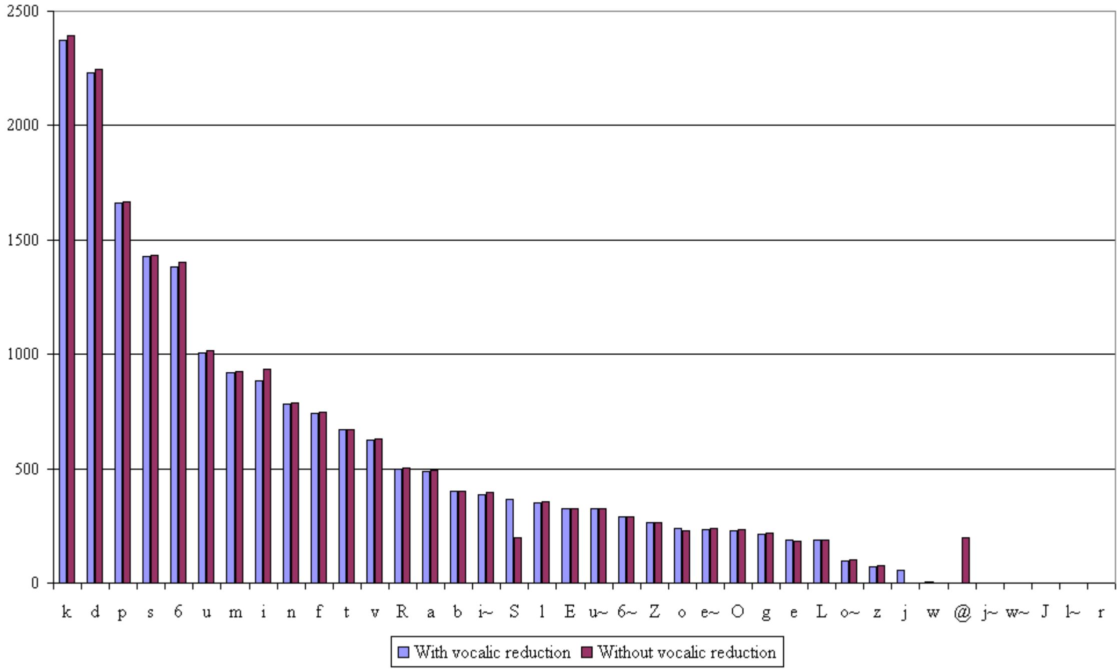


Figure 5.23: Number of Phonemes Occurrences at the end of a Sentence, for both phonetic transcriptions.

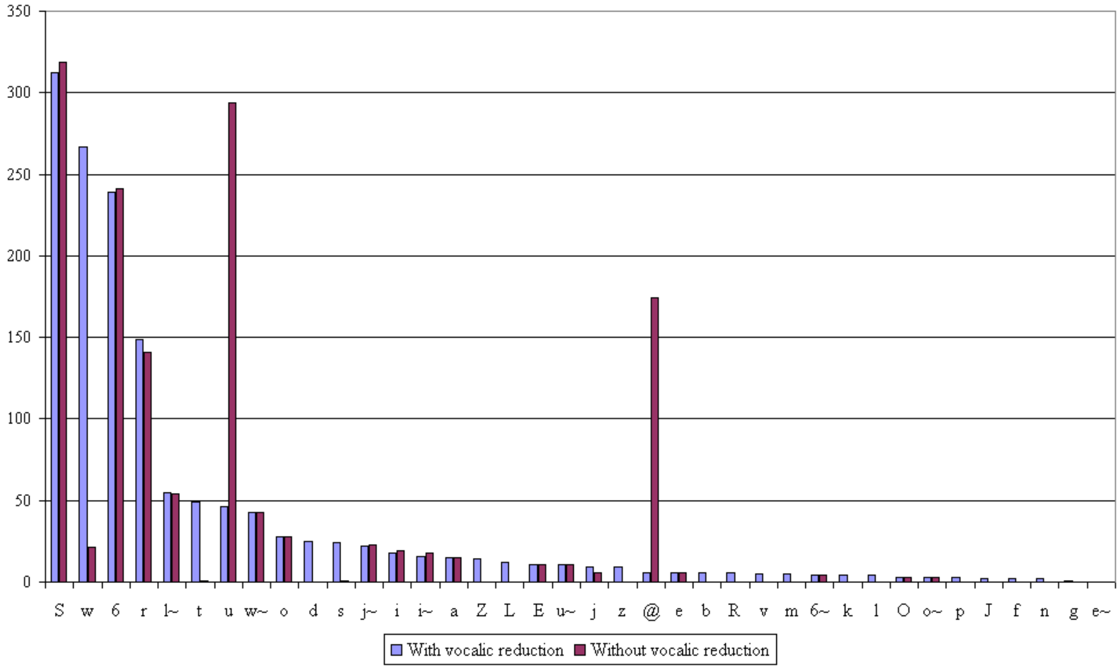
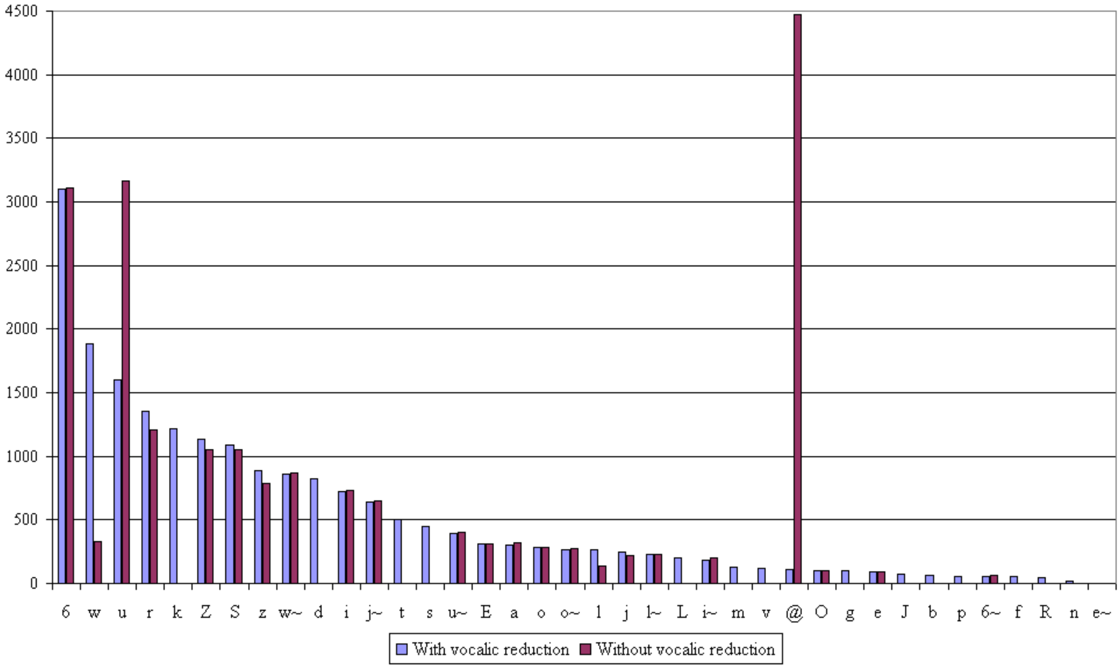


Figure 5.24: Number of Phonemes Occurrences at the end of a Word, for both phonetic transcriptions.



Chapter 6

Analysis for System Improvements

This chapter presents two possible improvements to the system, besides the use of the new speech corpus, rich in language contexts, as presented in chapter 5. First, a module to resolve foreign words is proposed, in order to synthesize multilingual texts, e.g. a Portuguese text including an English name, using the same voice and maintaining the original prosody. The other proposal is to use the residual signal from the acoustic units used for training as the excitation of the speech reconstruction filter. For this proposal a study was carried on that shows how the residual signals could be used in order to overcome the vocoder like synthetic speech that results from the source-filter model based on the pulse train/Gaussian noise excitation of the MLSA filter.

6.1 Polyglot Synthesis to resolve Foreign Words

This section presents a solution to synthesize multilingual text entries, using the same voice for different languages. It consists of mapping the phoneme sets of the foreign languages onto the phoneme set of the original one.

The solution is based on the HMM-based speech synthesis, allowing to maintain the original prosody of the voice or switch to a target prosody, by changing the F0 and duration models. The fact that the system uses monophone models, and not diphone or triphone models, is important for the presented solution.

Nowadays it is inevitable to find multilingual texts to synthesize. TTS synthesis is confronted more and more with the language mixing phenomenon, e.g., in documents with heavily mixed languages, in e-mail readers, in audio web browsers, or even in automatic cinema program announcements. The need of synthesizing a text in which several languages are present is very common. Multilingual synthesis systems help deal with this problem, but only to a certain point, as these are systems that need language detection and change the systems settings to produce the appropriate output voice in the required language, and for some applications this is not practical.

Beside the problem with different languages, most of the languages have many dialects or different accents. For each region it is desirable to synthesize speech according to that region's accent, but the work, time and resources to be spent on different voices databases is huge.

If one could exploit the fact that most of the phoneme units of one language are the same as the ones for the others, it would be much easier to implement new system languages. When dealing with languages with good resources, this does not constitute a problem. But there are many languages, like minor languages and even some languages that are spoken all over the world, with few shared resources or speech experts. Many languages have specific speech properties, e.g., phonological and prosodic parameters, which have not been studied yet, and this constitutes a problem when building a speech system for such a language. Beside this, there is the problem, even for languages with resources, of building systems with the same voice for all languages.

Another issue when dealing with multilingual systems is how to deal with prosody. There are applications that need to speak each language with its own prosody, but there are others for which switching the prosody every time that the language is switched would not seem natural. For instance, if in a sentence a foreign name is spoken, it would not be natural to use a different prosody when pronouncing this name.

Polyglot synthesis systems are a new approach to this problem. Being systems capable of synthesizing several languages using the same voice with appropriate pronunciations, these systems give response to lots of applications for what multilingual systems are not appropriate. But it is not easy to find polyglot speakers, they are expensive and speak a limited number of languages and with non-native speaker accents [Traber, 1995].

A solution would be to adapt the phoneme set of a language to the one of a target language, which could be very easy when the phoneme sets are similar or one is a subset of the other. Another solution would be a system that could convert different voices in a common target voice.

Many of the synthesis systems today are language context-based, which makes it difficult to synthesize one language using databases from others with different contexts. HMM-based systems are easier to adapt to this problem, because the databases are much smaller and due to the language context decision trees for unit clustering it deals efficiently with unknown contexts.

The solution presented here for text input from other languages than the main language of the text is to apply the phonetic units of the main language to the language contexts of the others, in order to synthesize different languages, using for that a direct mapping of the phoneme sets.

The first issue to consider is the fact that most of the times the phoneme sets from different languages differ at least in some phonemes. In this matter, two kinds of situations are possible:

1. The phoneme set used has phonemes that are not needed, but this situation brings no conflicts;
2. The phoneme set used lacks some phonemes, which is the difficult case to deal with.

The solution presented here is to solve the situation of the phoneme set lacking some phonemes of the languages needed to synthesize by finding the closest, most suitable, phoneme, or even a combination of phonemes, from the original language set. This method was called “direct phoneme mapping”.

The next sections show the study performed for direct mapping between EP and BP and between the Portuguese and German languages.

6.1.1 Polyglot Synthesis

The term “multilingual” derives from Latin and the term “polyglot” derives from Greek, but both express the capability of speaking several languages.

Already in 1999, Traber et al. [Traber et al., 1999] wrote about the importance of multilingual systems using only one voice, identifying them as polyglot systems, and made an effort in the sense of defining what is multilingual and what is polyglot. In this work these definitions are followed. They distinguished between polyglot and multilingual speech synthesis, defining:

- polyglot synthesis systems as those capable of synthesizing several languages using the same voice with appropriate pronunciations;
- multilingual synthesis systems as those that need language detection and to change the system settings to produce an appropriate output voice in a required language.

Multilingual speech systems are not new, there are several known names in the speech field that already explored the problem of multilingual synthesis [Traber et al., 1999] [Campbell, 1998] [Badino, Barolo & Quazza, 2004] [Black & Lenzo, 2004], and for several years had addressed the problem.

What is a relatively new approach to the multilingual problem is a speech system implemented for many languages using the same voice [Campbell, 1998] [Latorre, Iwano & Furui, 2005a] [Latorre, Iwano & Furui, 2005b].

This new approach has been explored in two different ways. One way is using databases recorded with the same speaker, having foreign pronunciations in some of the languages. These systems use polyglot synthesis with foreign pronunciations for some languages. The Loquendo TTS approach to mixed language speech synthesis is an example of a system using polyglot speakers. The other way is using databases recorded with several speakers, creating then an artificial common voice. These systems use speaker adaptation techniques to create an average voice, like the proposal of Javier Latorre, Koji Iwano and Sadaoki Furui's polyglot system, based on an HMM-based synthesis technique.

The HTS flexibility in voice conversion represents an advantage for polyglot systems, not only because it deals independently with the prosody models and the choice of the units, but also in the sense that it is possible to adapt different language voices to a common target voice.

The method presented here constitutes a new approach, which consists of using the database of one language to synthesize different languages, through the use of a mapping between the phoneme sets of the original language and the language to synthesize.

Treating foreign languages as if they were the original one requires the system to recognize all the graphemes, e.g. the "umlaut", and that the G2P conversion module knows how to deal with grapheme sequences unobserved in the original language.

There are some issues that must be considered when dealing with polyglot systems. One is the input language detection, which can be performed by text analysis or by tags. Another issue is at the NLP level, in which the G2P conversion module has to consider the graphemes from all the considered languages, all possible grapheme sequences conversion and the phonemes from all languages.

Regarding the speech database, it is required:

- To include all the phonemes of all the considered languages;
- To decide if the system will use different voices that must be modified, or the same voice;
- To consider the different languages' phoneme contexts, as much as possible.

Related to the prosody models, the decision between using only one prosody model or one model for each language has to be made, according to what sounds more natural in a particular application.

6.1.2 European and Brazilian Portuguese Direct Mapping

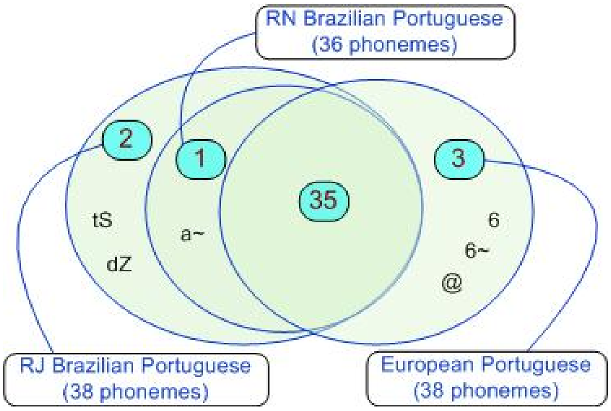
There is a big similarity between BP and EP phoneme sets. EP has only one dialect besides the official one, which is spoken in Azores island. BP has many dialects, from different states. Here, the official EP and the BP dialects from Rio de Janeiro (RJ) and Rio Grande do Norte (RN) were considered.

Table 6.1.2 summarizes both the BP and the EP phoneme sets used for this work. BP has 38 models, including two kinds of silences, short and long, and EP has 41 models, including three kinds of silences, short, pause and aspiration.

Table 6.1: Brazilian and European Portuguese Phoneme Sets

Brazilian Portuguese	European Portuguese
Oral Vowels and semivowels	
a, E, e, O, o, u, i, j, w	6, a, E, e, @, O, o, u, i, j, w
Nasal Vowels and semivowels	
a∼, e∼,o∼, u∼, i∼, j∼, w∼	6∼, e∼, o∼, u∼, i∼, j∼, w∼
Fricative Consonants	
v, f, z, s, S, Z	v, f, z, s, S, Z
Liquid Consonants	
L, l, r, R, X	L, l, l∼, r, R
Plosive Consonants	
b, p, t, k, g, d	b, p, t, k, g, d
Nasal Consonants	
m, n, J	m, n, J
Silences	
Sil, Pau	Asp, SSil, LSil

Figure 6.1: European and Brazilian, from Rio Janeiro and Rio Grande do Norte, Portuguese Phonemes Comparison.



From the analysis of figure 6.1 it can be observed that there are only some differences between EP and the considered BP dialects:

- Two nasal vowels, one from the BP, /a∼/, and another from the EP, /6∼/;
- Two oral vowels that exist in the EP do not exist in BP: /@/ and /6/;
- Two affricate consonants in BP from RJ that do not exist in the other dialects: /tS/ and /dZ/;
- The existence of a consonant in EP that does not exist in BP: /l∼/ and another consonant in BP that does not exist in EP: /X/.

Regarding the two nasal vowels, as it can be observed in figures 6.2 and 6.3, using the direct mapping the /6∼/ from EP and the /a∼/ from the BP can substitute each other, as they have common spectral values.

Regarding the vowels /6/ and /a/, these are quite similar in certain situations. From experimental observations, many occurrences of the /a/ in BP can be considered the EP /6/, when at the end of a word.

As the difference between EP and BP vowels is not significant, another possibility for mapping non-existent vowels in one of the databases would be to use a scheme of EP and BP vowel clustering, as it is shown in figure 6.4. In this scheme a distance metric could be used to cluster EP and BP vowels together, according to their Linear Spectral Frequency (LSF) averages. This way, in case of non-existence of some vowel in a database the closest one would be identified and could be used.

Another difference between the phoneme sets is the schwa, /@/. This phoneme is ignored in some EP synthesis systems with good results, as this phoneme is often

Figure 6.2: Spectrogram of European Portuguese /6∼/

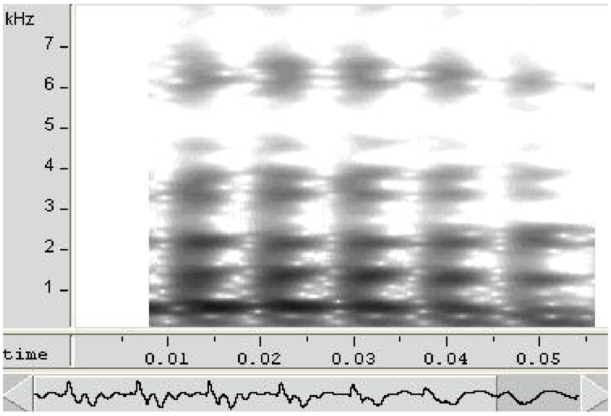
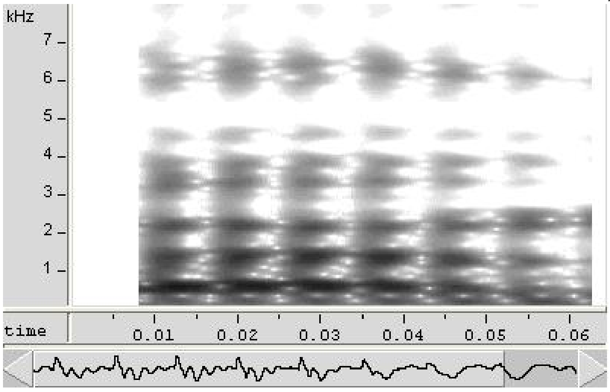


Figure 6.3: Spectrogram of Brazilian Portuguese /a∼/



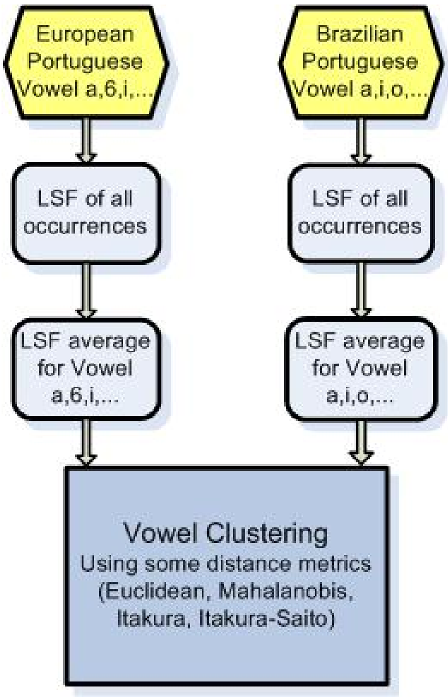
suppressed in continuous speech as it was shown in chapter 5. The same approach is suggested here.

Regarding the affricate consonant sounds /tS/ and /dZ/, they can be approximated with the use of the plosives /t/ and /d/, followed by the fricatives /S/ and /Z/, respectively. This solution did not show reasonable results, mainly because these sequences do not appear in the database.

A test of the direct mapping between EP and BP from RN, using the EP system, was performed. For the test, the input sentence <Isto representa um teste para o sintetizador em Português Europeu> (meaning *“This represents a test of the synthesizer in European Portuguese”*) was synthesized with the EP system. The phonetic transcription result was:

/iStwR@pr@ze∼t6u∼tESt@p6rosi∼t@tiz6dor6∼jpwrtwgezeurpew/

Figure 6.4: Scheme of European and Brazilian Portuguese Vowel Clustering.


European Portuguese Vowel a,6,i,...
Brazilian Portuguese Vowel a,i,o,...
LSF of all occurrences
LSF of all occurrences
LSF average for Vowel a,6,i,...
LSF average for Vowel a,i,o,...
Vowel Clustering
Using some distance metrics (Euclidean, Mahalanobis, Itakura, Itakura-Saito)

A second input sentence, <Isto representa um teste para o sintetizador em Português do Brasil> (meaning *"This represents a test of the synthesizer in Portuguese from Brazil"*) was synthesized with a BP system. The phonetic transcription result was:

/iStuREprEze∼tau tEStiparausi∼tetizadore∼j∼portugejsdubraziw/

It should be noticed that besides the differences in the phonemes of both dialects, in BP the <l> at the end of a sentence is transcribed as /w/ and not /l∼/ as it would be for EP.

Both sentences were presented to the listeners. Afterwards, the second input sentence was synthesized by the EP synthesizer using the phonetic transcription for BP and the algorithm for direct mapping between both dialects of the Portuguese language, although in this case the BP from the RN phonetic set is a subset of the EP one. This last sentence was then presented to the listeners for the algorithm evaluation. The test was performed with both Portuguese and Brazilian listeners.

Interestingly, the listeners preferred the result of the second sentence when synthesized by the EP system than by the BP one. This was due to the less vocalic sound produced by this system, but showed that the result of the phonetic mapping sounded natural even to the Brazilian listeners.

6.1.3 Portuguese and German Languages Direct Mapping

The German and EP phoneme sets can be observed in the table 6.2, using the SAMPA [SAMPA, visited in 2010] alphabet. Table G.1, from appendix G, presents a more detailed comparison between the phonemes of both languages, using examples from both languages, for better understanding the differences.

Table 6.2: German and European Portuguese Phoneme Sets.

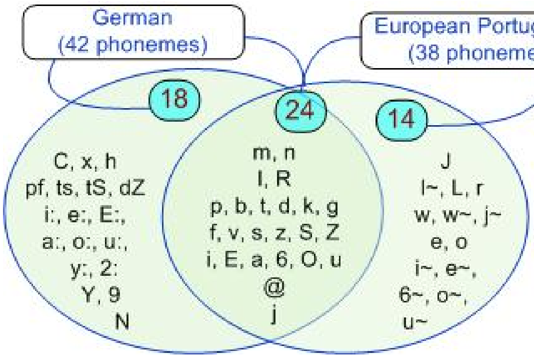
German	European Portuguese
Long Oral Vowels	
a:, E:, e:, o:, u:, i:, y:, 2:	
Short Oral Vowels	
6, a, E, O, U, I, Y, 9	6, a, E, e, O, o, u, i
Schwa Vowel	
@	@
semivowels	
j	j, w
Nasal Vowels	
	6∼, e∼, o∼, u∼, i∼
Nasal semivowels	
	j∼, w∼
Affricate Consonants	
pf, ts, tS, dZ	
Fricative Consonants	
v, f, z, s, S, Z, C, x, h	v, f, z, s, S, Z
Liquid Consonants	
l, R	L, l, l∼, r, R
Plosive Consonants	
b, p, t, k, g, d	b, p, t, k, g, d
Nasal Consonants	
m, n, N	m, n, J

Figure 6.5 schematizes the differences between the German and EP phoneme sets. As can be seen in the figure, EP has 14 phonemes non-existent in German, and German has 18 phonemes non-existent in EP.

To map the German phonemes non-existent in EP, 18 phonemes must be considered.

EP does not have the affricate sounds: /pf/, /ts/, /tS/ and /dZ/. The suggestion for synthesizing these sounds using a Portuguese speech database is to use the plosive sounds, /pf/, /ts/, /tS/, /dZ/ followed by the fricatives /f/, /s/, /S/ and /Z/, as it was suggested for the BP affricates.

Figure 6.5: German and European Portuguese Phonetic Sets Comparison.



Regarding the fricatives, the following mapping is proposed:

- /C/ can be substituted by /S/;
- /x/ can be substituted by /R/;
- /h/ can be substituted by /R/.

The nasal sound /N/ can be achieved by nasalizing its preceding vowel and following it with a /g/.

For the checked (short) vowels, the substitutions proposed are:

- The /Y/ can be substituted by /i/ pronounced with the lips shape for the /u/;
- The /9/ can be substituted by /e/ pronounced with the lips shape for the /o/.

For the free (Long) Vowels: /i:/, /e:/, /E:/, /a:/, /o:/, /u:/, /y:/, /2:/, it is proposed to lengthen the duration of the short vowels /I/, /e/,/E/, /a/, /o/, /u/, /Y/, /9/, respectively.

To map the EP phonemes non-existent in German, 14 phonemes must be considered.

For the glides, or semivowels, the following mapping is proposed:

- The /w/ can be substituted by /u/ with short duration;
- The /w∼/ can be substituted by /u/ with short duration followed by /N/ to give the nasal sound;
- The /j∼/ can be achieved by the /j/ followed by /N/ to give the nasal sound, which means substituting it by /jN/.

For the checked vowels, or short vowels, /e/ and /o/, it is proposed to substitute them by the long vowels /e:/ and /o:/, respectively, with short duration.

For the nasal vowels, /i~/, /e~/, /6~/, /o~/ and /u~/, the proposed substitution is to follow the vowels /i/, /e/, /6/, /o/ and /u/, respectively, by /N/ to give the nasal sound.

The nasal consonant /J/ can be substituted by /N/.

The liquids are the most difficult cases. A mapping solution for the /L/ and the /r/ was not found.

The proposed mapping for the /l~/ is substituting it with /l/, a situation that is found in some regional accents of EP.

A test of the direct mapping between EP and German, using the EP system, was performed. For the test, three semantically unpredictable input sentences were created by using isolated words, covering the affricate sounds, the fricative ones and the short vowels. The use of semantically unpredictable sentences made the test more difficult for the listeners, but the intention was to include as many study cases as possible.

The first input sentence, for the affricate sounds, was the sentence <Pfahl Zahlts Deutsch Dschungel> (meaning *"Stake paid German jungle"*). The phonetic transcription result would be:

/pfa:l tsa:lts dOYtS dZuN6l/

The approximation result from the direct mapping algorithm was the phonetic sentence:

/pfal tsal dOjtS dZu~g6l/

The second input sentence, for the fricative sounds, was the sentence <Fast was Tasse Hase waschen Genie sicher Buch Hand> (meaning *"Almost what cup rabbit wash genius certainly book hand"*). The phonetic transcription result would be:

/fast vas tas@ ha:z@ vaS6n Ze:ni: zIC6 bu:x hant/

The approximation result from the direct mapping algorithm was the phonetic sentence:

/fast vas tas@ az@ vaS6n Zenii ziS6 buR ant/

The third input sentence, for the short vowels, was the sentence <Sitz Gesetz Satz besser trotz Schutz hübsch plötzlich bitte> (meaning *"Seat law sentence better despite a better protection beautiful suddenly please"*). The phonetic transcription result would be:

/zIts g@zEts zats bEs6 trOts SUts hYpS pl9tslIC bIt@/

The approximation result from the direct mapping algorithm was the phonetic sentence:

/zits g@zEts zats bEs6 trOts Suts jwpS pl6otsliS bit@/

The listeners had difficulties in understanding several words, which is common for any test using semantically unpredictable sentences as long as the second and the last ones. But when having the written sentences available the listeners were satisfied with most of the results, saying that it resembles German with a foreign accent, but it is possible to understand the meaning of most of the words. The most difficult cases were the words involving affricate sounds and the phoneme /C/.

The proposal to use polyglot synthesis to resolve foreign input text was tested with some samples, but not integrated in the implemented system.

6.2 Hidden Markov Models based Speech System using Residual Signal

The main purpose of this section is to present an approach suitable for overcoming the vocoder quality of the output sound from HMM-based speech synthesis. This approach is theoretically presented and analyzed, but it was not implemented during the work of this thesis.

HMM-based speech synthesis, as presented in the previous chapters, is a data-driven approach, implemented by applying statistical learning algorithms and using HMMs to reproduce voice characteristics of the original speaker. It can be automatically trained to generate natural synthetic speech, but it is still vocoder like. The speech is generated from the concatenation of HMMs that represent the statistics of the acoustic features of the phonetic units, usually monophones or triphones. These models output the parameters that simulate the vocal tract and use them as the coefficients of a speech reconstruction filter. Usually, the excitation of these filters is composed of pulse trains for the voiced regions and Gaussian noise for the unvoiced ones. In the new approach the residual signal from the acoustic units used for the training substitutes the pulse train and Gaussian noise as the excitation of the speech reconstruction filter.

The residual signal is the error between the original speech signal and a predicted signal obtained through some spectral coefficients, like the MFCCs or the LPCs. These signals are flat enough for most of the cases, at least at lower frequencies.

The synthesis engine would need a larger bandwidth than the original version using the pulse trains and Gaussian noise as input, as the residual signals have more or less the same size as the original sound wave signals, but this is not a problem for HMM-based speech systems because the amount of speech needed is very small, e.g., the EP system implemented in this thesis showed good results with 21 minutes of speech. So, if the size of the engine is not a constraint, as in most of the desktop speech applications, the approach is feasible.

The next sections explain how the residual signal can be obtained and how to integrate the residual signal in the speech synthesis system.

6.2.1 Obtaining the Residual Signal

The residual signal can be obtained by a method known as inverse filtering, which consists of exciting the inverse model used to extract the MFCCs with the original speech signal. This process can be part of the HMM-based training procedures, as it is shown in figure 6.8.

There are several techniques to extract the MFCCs from a speech signal. For this task the procedure shown in figure 6.6 is used.

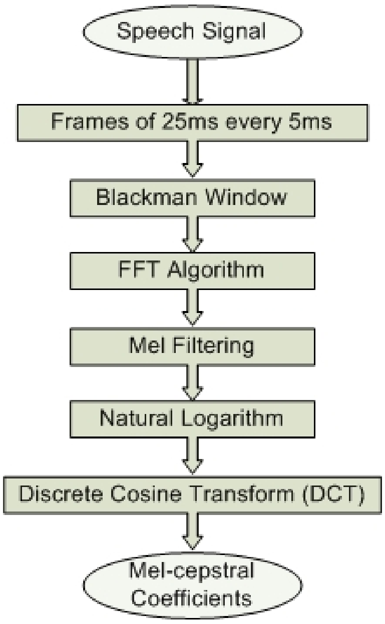


Figure 6.6: Mel-cepstral coefficients extraction

The speech signal, $x(n)$, is analyzed in short-time signals, of 25ms, in this case, with an overlap of 20ms and weighted by a Blackman window:

$$y(n) = x(n) \cdot b(n) \tag{6.1}$$

An FFT is used to calculate the magnitude spectrum of the windowed speech frames:

$$|Y(f)| = |\sum_{n=0}^{N-1} y(n) \exp^{\frac{-j2\pi fn}{N}}| \tag{6.2}$$

When we take the magnitude of $Y(f)$ we lose the phase information. $Y(f)$ then goes into a filter bank, uniformly spaced in the mel-scale, to obtain the mel-spectrum:

$$Y_k = \sum_{f=0}^{N/2-1} |Y(f)| w_k(f) \tag{6.3}$$

A natural logarithm is applied to Y_k:

$$lnY_k \tag{6.4}$$

This procedure is used as a smoothing function to make the signal more suitable for spectral representation, separating the convolved signal components. The MFCCs will be the terms of the cosine expansion of the logarithmic magnitude spectrum expressed on the mel-scale:

$$C_y(n) = u_n \sum_{k=0}^{K-1} (\log Y_k) \cos(\frac{(2k+1)n\pi}{2K}) \tag{6.5}$$

Where:

$$u_n = \begin{cases} 1/\sqrt{K} & n = 0 \\ \sqrt{2/K} & n > 0 \end{cases} \tag{6.6}$$

The Discrete Cosine Transform (DCT) encodes the mel logarithmic magnitude spectrum into the MFCC [Milner & Shao, 2002] [Tychtl & Psutka, 2000].

It is known that much of the information is lost during the MFCC extraction to allow it to be inverted into a time-domain signal again, but it is possible to recover a smoothed estimate of the log magnitude spectrum [Tychtl & Psutka, 2000]:

$$\log \hat{Y}_k = \sum_{n=0}^{N-1} u_n C_y(n) \cos(\frac{(2k+1)n\pi}{2K}) \tag{6.7}$$

To invert the log operation the use the inverse exponential operation is needed. An estimate of the vocal tract filter coefficients can be achieved from the obtained magnitude spectrum. This way, it is possible to create an inverse filter that can be excited with the original speech wave in order to get the estimated residual.

To integrate the residual signal in the HMM-based speech synthesis, some procedures have to be taken during training and also during synthesis, which will be described in the following section.

6.2.2 Integration of the residual signal in the System

Reconstructing a speech signal from MFCC vectors has been widely used through different techniques [Fukada et al., 1992] [Milner & Shao, 2002]. The filter used here is the MLSA filter [Fukada et al., 1992], which models the vocal tract using MFCC sequences as coefficients for the filter transfer function. The speech waveform is synthesized directly from the MFCC values and the excitation signal. The excitation signal is usually a pulse train for the voiced regions and Gaussian noise for the unvoiced ones, but can be substituted by other excitation signals such as the residual signal. Figure 6.7 shows a scheme of the synthesis filter with two different excitations: a) the typical pulse train and Gaussian noise filter excitation and b) the residual signal one.

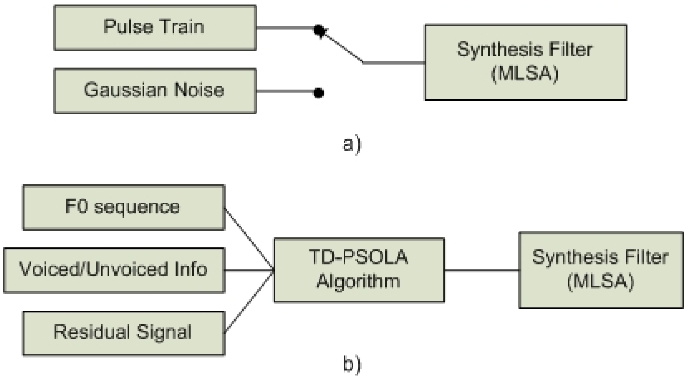


Figure 6.7: Synthesis filter excitations. a) typical pulse train and Gaussian noise excitation; b) residual signal excitation

During the training part, first the speech parameters used to train the models, in this case MFCCs, F0 and durations, will be extracted. In this stage, the inverse filtering can be used to obtain the residual signals and store them in a database whose indices will be passed on to the feature vector to be used in the HMM training. This procedure can be seen in figure 6.8.

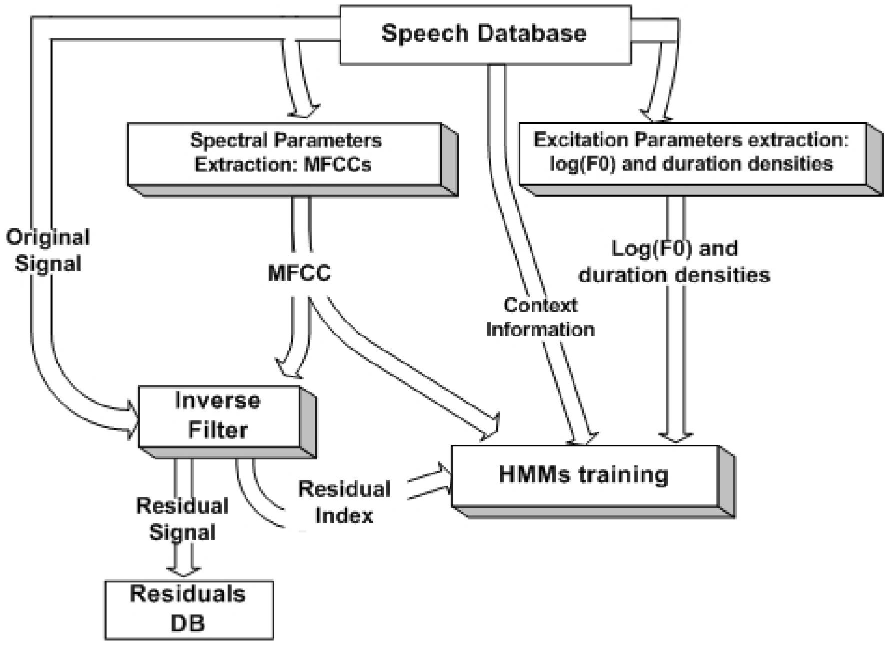

Speech Database
Spectral Parameters Extraction: MFCCs
Excitation Parameters extraction: log(F0) and duration densities
Original Signal
MFCC
Log(F0) and duration densities
Context Information
Inverse Filter
HMMs training
Residual Signal
Residual Index
Residuals DB

Figure 6.8: Schematic diagram of HMM-based training module with residual signal

Figure 6.9 describes the HMM-based synthesis stages as in the original version implemented in this thesis. It consists of inputting the text into a G2P converter in order to get the phoneme sequence, and analyzing the context of each phoneme to build the context label sequences.

The context information consists of phonetic information, as explained in the previous chapters. With the context labels, the system goes through the context-based decision trees until reaching a leaf. Spectrum, F0 and durations are clustered independently, as the contextual factors that affect them are different.

After this, an HMM sequence is constructed by concatenating context-dependent HMMs. State durations of the HMM sequence are determined in order to maximize the output probability of state durations. A sequence of MFCCs and logF0 values, including voiced/unvoiced decisions, is determined in a way that the sequence output probability for the HMM is maximized, using the speech parameter generation algorithm described in [Tokuda et al., 2000a].

In the typical HMM-based speech synthesis system, the speech waveform is synthesized by using the MLSA filter directly from the generated MFCCs, which will model the vocal tract, and the voiced/unvoiced information and F0 values are used to construct the excitation signal. It is at this stage where the new approach is different. Until this stage the residual signal approach brings no modification.

In the residual approach, the HMM output will give an index to a residual signal in the residual database. The F0 information will be used in a TD-PSOLA algorithm to modify the residual F0 accordingly, keeping the residual intact for the unvoiced regions. The output signal from the TD-PSOLA algorithm will be the input of the MLSA filter.

Figure 6.9 shows all the stages in a typical HMM-based system, but outlines the stages that will undergo changes for the residual approach. These changes are presented in figure 6.10, which represents the system modifications to be performed in order to use the residual signal in the synthesis module.

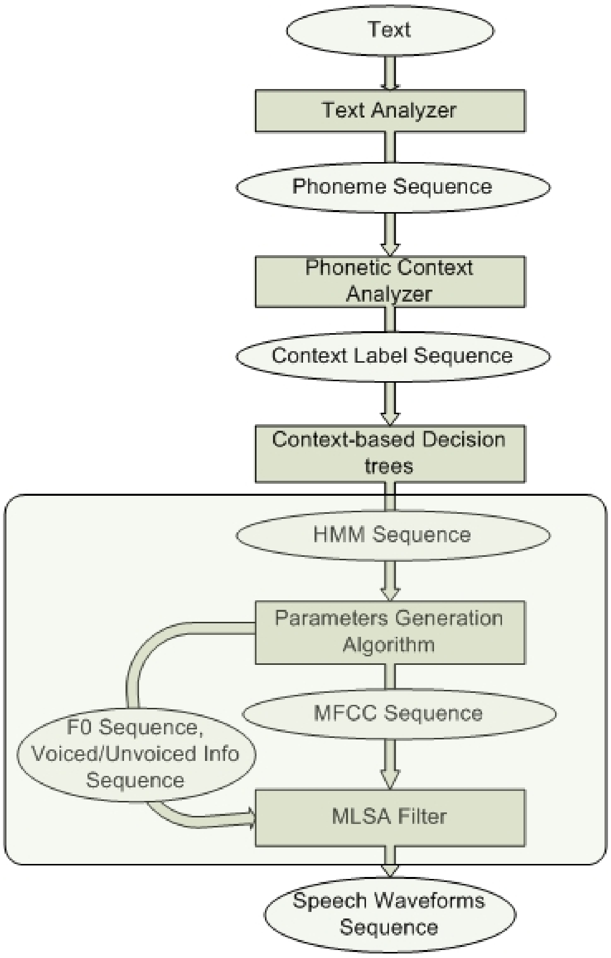

Text
Text Analyzer
Phoneme Sequence
Phonetic Context Analyzer
Context Label Sequence
Context-based Decision trees
HMM Sequence
Parameters Generation Algorithm
MFCC Sequence
F0 Sequence, Voiced/Unvoiced Info Sequence
MLSA Filter
Speech Waveforms Sequence

Figure 6.9: Stages of the HMM-based synthesis module

The speech, in this approach as well as in the original approach, is generated from the concatenation of context-dependent HMM models. The difference is that in this approach these models represent the statistics of the phonetic units' spectral features, to be used as the synthesis filter transfer function coefficients, and index their corresponding residual signals, to be used as the excitation signals for the synthesis filter.

It is commonly accepted that when using speech parameters to model the vocal tract, the residual signal is the excitation that gives the best quality. Applying this

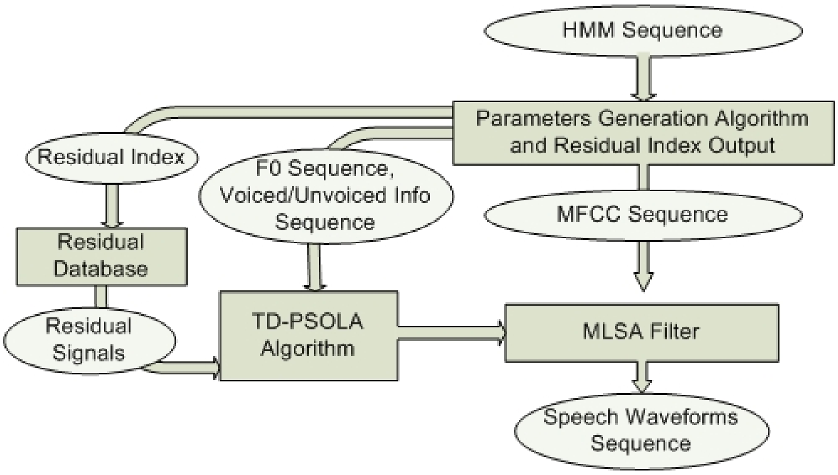

HMM Sequence
Parameters Generation Algorithm and Residual Index Output
Residual Index
F0 Sequence, Voiced/Unvoiced Info Sequence
MFCC Sequence
Residual Database
Residual Signals
TD-PSOLA Algorithm
MLSA Filter
Speech Waveforms Sequence

Figure 6.10: Changes to the HMM-based synthesis module for using the residual signal

concept to HMM-based speech synthesis is the reasonable solution to improve the quality of the implemented system. The implementation of an excitation model for HMM-based speech synthesis based on residual modeling was presented by Maia et al. at SSW6, in 2007 [Maia et al., 2007].

Using the residual signal for HMM based synthesis was here analyzed as a proposal for improvement of the implemented system. This analysis was theoretical and the proposal was not implemented for the work of this thesis.

Chapter 7

Conclusions

In this thesis an HMM-based synthesis system for EP [Barros et al., 2005] was presented. It was developed by using the speech synthesis framework HTS [HTS, visited in 2010], which is based on HMM modeling of speech signal parameters and language dependent context-based information. The synthesis system was integrated in a completely automatic TTS system that is capable of converting any incoming EP text into a speech audio file.

According to state-of-the-art research, this thesis covers the most prominent speech synthesis approaches and gives an overview of the techniques as well as the pitfalls and challenges of TTS synthesis systems. Today's two major speech synthesis approaches are data-driven methods, more precisely an HMM-based approach, and a unit-selection-based approach.

The tasks integrating the NLP module implemented for the TTS system were based on a statistical method, the ME algorithm [Barros & Weiss, 2006]. The tasks are G2P conversion, syllable boundary detection and syllable stress prediction.

When listening to speech waveforms synthesized by the implemented HMM-based synthesis system, the main conclusion is that the prosody is good and pleasant, speech is smooth and does not present discontinuities. These results are good in comparison with the other systems. It must be highlighted that the system was trained with only 21 minutes of speech, which brings an extra value to the results and demonstrates that the HMM-based synthesis approach is the ideal choice for lesser resources systems. The results can be improved with a speech corpus that is richer in language context features, but the main drawback of the approach is an output sound quality that is always a bit metallic, vocoder like. This effect is due to the source-filter model used to generate speech from the estimated parameters. The results from the evaluation tests performed were in agreement with these observations.

A new speech corpus for EP was designed, as, in a language context-based system, it is very important to have as many linguistic, prosodic and syntactic features in the database as possible. The new corpus consists of one sentence per diphone, in which as many contexts of the language as possible were integrated. The corpus is intended to be used to train both the synthesizer and the NLP module. It is phonetically transcribed taking into consideration the words' coarticulation effects. Two different phonetic transcriptions were performed, one considering and the other not considering the vocalic reduction effects. These two transcriptions allow a better understanding of the influence of the vocalic reduction effect in the EP language.

Two improvements were proposed for the implemented system. To overcome the vocoder like sound of the HMM-based speech synthesis a different source-filter model was proposed, as there is still a strong need for improvement in the source-filter in the HTS approach. The approach proposed consists of using the residual signals as the excitation of the speech reconstruction filter, as these signals are flat enough for most of the cases. This approach represents an improvement to the original pulse train/ Gaussian noise model. The residual signal is obtained by inverse filtering, which consists of exciting the inverse model used to extract the MFCCs with the original speech signal. The rest of the filter is the same as in the original approach.

The other improvement consists of a method for dealing with foreign words when implementing an HMM-based TTS system, which is based on the mapping of phonemes from EP onto phonemes of other languages or dialects.

Appendix A

Speech Corpora for Natural Language Processing

A.1 Excerpt of the Corpus for Grapheme-to-Phoneme Conversion

```
a_6 b_b c_s i_i s_s s_& a_6
    a_a u_w t_t o_w m_m a_a t_t i_i c_k a_6 m_m e_e~ n_& t_t e_&
    b_b a_a ú_u
    c_k a_a b_b e_@
    c_k l_l a_6 s_s s_& i_i f_f i_i c_k a_6 ç_s ã_6~ o_w~
    d_d i_i v_v e_E r_r s_s a_6 s_S
    g_g l_l o_o b_b o_w
    m_m a_a i_j s_S
    m_m a_a l_l~
    m_m e_@ l_L h_& o_O r_r
    n_n a_6 s_z a_a l_l~
    n_n ã_6~ o_w~
    n_n e_@ n_J h_& u_u~ m_&
    n_n e_6~j~ m_&
    r_R e_E g_g r_r a_6 s_S
    s_s u_u a_6
    t_t a_6~ m_& b_b é_6~j~ m_&
    x_S a_6 d_d r_r e_e z_S
    z_z o_o n_n a_6
```

A.2 Excerpt of the Corpus for Syllable Boundaries Detection

6_1 f_1 i_0 n_1 a_0 l~_0

b_1 O_0 l_1 6_0

E_1 r_1 6_0

f_1 a_0 b_1 r_0 i_0 k_1 6_0 S_0

g_1 r_0 6~_0 d_1 S_0

i_1 g_1 w_0 a_0 l~_0 d_1 a_0 d_1 @_0

k_1 o~_0 s_1 i_0 d_1 E_0 r_1 6_0

l_1 i_0 J_1 6_0

m_1 a_0 j_0 S_0

O_1 r_1 6_0 S_0

p_1 O_0 b_1 r_0 @_0

u_1 m_1 6_0

A.3 Excerpt of the Corpus for Stress Prediction

6_0 b_0 6_0 n_1 a_1 r_1

d_1 E_1 S_1 t_0 6_0 S_0

f_0 6_0 l_1 a_1 r_1

g_1 E_1 R_0 6_0

k_0 o~_0 t_1 r_1 a_1 r_0 i_0 w_0

m_1 e_1 n_0 u_0 S_0

p_0 @_0 r_0 d_1 e_1 r_1

s_0 6_0 k_0 u_0 d_1 i_1 w_1

t_1 6~_1 t_0 6_0

t_1 i_1 J_0 6~_0 w~_0

v_1 i_1 v_0 @_0

Z_0 u_0 g_1 a_1 d_0 6_0 Z_0

Appendix B

Hidden Markov Models based Speech Synthesis Data Files

B.1 Output File from the Natural Language Processing Module

```
phoneme     syll    stress      word
n           n6       0          n6
6
p           p6       0          p6sad6
6
s           sa       1
a
d           d6       0
6
k           ki~      1          ki~t6
i~
t           t6       0
6
```

B.2 List of the European Portuguese Context-dependent Feature Labels

The context-dependent feature labels file uses the following labels:

m1ˆm2 - m3 + m4 = m5 /M2: m6_m7
/S1: s1_?s2 - s3_?s4 + s5_?s6 /S2: s7_s8 /S3: s9_s10 /S4: s11_s12 /S5: s13_s14 /S6: s15
/W1: #w1 - #w2 + #w3 /W2: w4_w5 /W3: w6
/P1: p1_!p2 - p3_!p4 + p5_!p6 /P2: p7_p8
/U: u1_$u2_&u3

m1, phone before previous phone
m2, previous phone

m3, current phone
m4, next phone
m5, phone after next phone
m6, position of current phone in current syllable (forward)
m7, position of current phone in current syllable (backward)

s1, whether previous syllable is stressed or not (0 → no; 1 → yes)
s2, number of phones in previous syllable
s3, whether current syllable is stressed or not (0 → no; 1 → yes)
s4, number of phones in current syllable
s5, whether next syllable is stressed or not (0 → no; 1 → yes)
s6, number of phones in next syllable
s7, position of current syllable in current word (forward)
s8, position of current syllable in current word (backward)
s9, position of current syllable in current phrase (forward)
s10, position of current syllable in current phrase (backward)
s11, number of stressed syllables before current syllable in current phrase
s12, number of stressed syllables after current syllable in current phrase
s13, number of syllables, counting from the previous stressed syllable to the current syllable in the utterance
s14, number of syllables, counting from the current syllable to the next stressed syllable in the utterance
s15, vowel of current syllable

w1, number of syllables in previous word
w2, number of syllables in current word
w3, number of syllables in next word
w4, position of current word in current phrase (forward)
w5, position of current word in current phrase (backward)
w6, punctuation flag for the work (1 → interrogation, 2 → exclamation, 3 → three points)

p1, number of syllables in previous phrase
p2, number of words in previous phrase
p3, number of syllables in current phrase
p4, number of words in current phrase
p5, number of syllables in next phrase
p6, number of words in next phrase
p7, position of current phrase in the utterance (forward)
p8, position of current phrase in the utterance (backward)

u1, number of syllables in the utterance
u2, number of words in the utterance
u3, number of phrases in the utterance

B.3 Example of a Context-dependent Feature Labels File

Reproduction of File portuguese_euro_008p24.lab

```
y^y-X+e=S/M2:y_y/S1:y_?y-y_?y+1_?2/S2:y_y/S3:y_y/S4:0_4/S5:0_2/S6:y/W1:#y-#y+#1/W2:y_y/W3:y/P1:y_!y-y_!y+11_!5/P2:y_!y/U:13_$7_&2
y^X-e+S=t/M2:1_2/S1:y_?y-1_?2+1_?2/S2:1_1/S3:1_11/S4:0_3/S5:0_2/S6:e/W1:#y-#1+#1/W2:1_5/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
X^e-S+t=a/M2:2_1/S1:y_?y-1_?2+1_?2/S2:1_1/S3:1_11/S4:0_3/S5:0_2/S6:e/W1:#y-#1+#1/W2:1_5/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
e^S-t+a=u~/M2:1_2/S1:1_?2-1_?2+0_?1/S2:1_1/S3:2_10/S4:1_2/S5:2_4/S6:a/W1:#1-#1+#1/W2:2_4/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
S^t-a+u~=b/M2:2_1/S1:1_?2-1_?2+0_?1/S2:1_1/S3:2_10/S4:1_2/S5:2_4/S6:a/W1:#1-#1+#1/W2:2_4/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
t^a-u~+b=k/M2:1_1/S1:1_?2-0_?1+0_?1/S2:1_1/S3:3_9/S4:2_2/S5:2_3/S6:u~/W1:#1-#1+#3/W2:3_3/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
a^u~-b+k=a/M2:1_1/S1:0_?1-0_?1+1_?2/S2:1_3/S3:4_8/S4:2_2/S5:3_2/S6:y/W1:#1-#3+#5/W2:4_2/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
u~^b-k+a=d/M2:1_2/S1:0_?1-1_?2+0_?1/S2:2_2/S3:5_7/S4:2_1/S5:4_6/S6:a/W1:#1-#3+#5/W2:4_2/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
b^k-a+d=s/M2:2_1/S1:0_?1-1_?2+0_?1/S2:2_2/S3:5_7/S4:2_1/S5:4_6/S6:a/W1:#1-#3+#5/W2:4_2/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
k^a-d+s=b/M2:1_1/S1:1_?2-0_?1+0_?1/S2:3_1/S3:6_6/S4:3_1/S5:2_5/S6:y/W1:#1-#3+#5/W2:4_2/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
a^d-s+b=r/M2:1_1/S1:0_?1-0_?1+0_?2/S2:1_5/S3:7_5/S4:3_1/S5:3_4/S6:y/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
d^s-b+r=s/M2:1_2/S1:0_?1-0_?2+0_?2/S2:2_4/S3:8_4/S4:3_1/S5:4_3/S6:y/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
s^b-r+s=a/M2:2_1/S1:0_?1-0_?2+0_?2/S2:2_4/S3:8_4/S4:3_1/S5:4_3/S6:y/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
b^r-s+a=t/M2:1_2/S1:0_?2-0_?2+1_?2/S2:3_3/S3:9_3/S4:3_1/S5:5_2/S6:a/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
r^s-a+t=a/M2:2_1/S1:0_?2-0_?2+1_?2/S2:3_3/S3:9_3/S4:3_1/S5:5_2/S6:a/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
s^a-t+a=d/M2:1_2/S1:0_?2-1_?2+0_?1/S2:4_2/S3:10_2/S4:3_0/S5:6_0/S6:a/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
a^t-a+d=X/M2:2_1/S1:0_?2-1_?2+0_?1/S2:4_2/S3:10_2/S4:3_0/S5:6_0/S6:a/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
t^a-d+X=m/M2:1_1/S1:1_?2-0_?1+0_?3/S2:5_1/S3:11_1/S4:4_0/S5:2_0/S6:y/W1:#3-#5+#1/W2:5_1/W3:0/P1:y_!y-11_!5+2_!2/P2:1_2/U:13_$7_&2
a^d-X+m=6/M2:y_y/S1:0_?1-y_?y+0_?3/S2:y_y/S3:y_y/S4:4_0/S5:3_0/S6:y/W1:#5-#y+#1/W2:y_y/W3:y/P1:11_!5-y_!y+2_!2/P2:y_!y/U:13_$7_&2
d^X-m+6=Z/M2:1_3/S1:0_?1-0_?3+0_?3/S2:1_1/S3:1_2/S4:0_0/S5:3_0/S6:6/W1:#5-#1+#1/W2:1_2/W3:0/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
X^m-6+Z=v/M2:2_2/S1:0_?1-0_?3+0_?3/S2:1_1/S3:1_2/S4:0_0/S5:3_0/S6:6/W1:#5-#1+#1/W2:1_2/W3:0/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
m^6-Z+v=a/M2:3_1/S1:0_?1-0_?3+0_?3/S2:1_1/S3:1_2/S4:0_0/S5:3_0/S6:6/W1:#5-#1+#1/W2:1_2/W3:0/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
6^Z-v+a=j/M2:1_3/S1:0_?3-0_?3+y_?y/S2:1_1/S3:2_1/S4:0_0/S5:4_0/S6:a/W1:#1-#1+#y/W2:2_1/W3:1/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
Z^v-a+j=y/M2:2_2/S1:0_?3-0_?3+y_?y/S2:1_1/S3:2_1/S4:0_0/S5:4_0/S6:a/W1:#1-#1+#y/W2:2_1/W3:1/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
v^a-j+y=y/M2:3_1/S1:0_?3-0_?3+y_?y/S2:1_1/S3:2_1/S4:0_0/S5:4_0/S6:a/W1:#1-#1+#y/W2:2_1/W3:1/P1:11_!5-2_!2+y_!y/P2:2_1/U:13_$7_&2
```

B.4 Context related questions file for European Portuguese

Phoneme position = {Left Left, Left, Center, Right, Right Right}

"pos_Pause" {X, XX}

"pos_Sil_Pause" {X}

"pos_Asp_Pause" {XX}

"pos_Voiced" {a, 6, E, O, e, h, i, o, u, 6~, e~, i~, o~, u~, w, j, w~, j~, b, d, g, m, n, J, z, v, Z, l, l~, L, r, R}

"pos_Continuant" {a, 6, E, O, e, h, i, o, u, j, w, 6~, e~, i~, o~, u~, j~, w~, s, S, z, f, v, Z }

"pos_Noncontinuant" {R, r, l, l~, L, p, b, t, d, k, g, m, n, J}

"pos_Vowel" {a, 6, E, O, e, h, i, o, u, 6~, e~, i~, o~, u~}

"pos_Anterior_vowel" {E, e, i, e~, i~}

"pos_Central_vowel" {h, a, 6, 6~}

"pos_Posterior_vowel" {O, o, u, o~, u~}

"pos_High_vowel" {i, u, i~, u~, h}

"pos_Middle_vowel" {e, o, e~, o~}

"pos_Nonrounded_vowel" {a, 6, E, e, h, i, 6~, e~, i~}

"pos_Closed_vowel" {h, i, u, i~, u~}

"pos_Semi_open_vowel" {6, 6~, E, O}

"pos_Open_vowel" {a}

"pos_Reduced_vowel" {a, i, u, 6~, i~, u~}

"pos_Oral_vowel" {a, 6, E, O, e, h, i, o, u}

"pos_Nasal_vowel" {6~, e~, i~, o~, u~}

"pos_Anterior_and_closed_vowel" {i, i~}

"pos_Anterior_and_oral_vowel" {E, e, i}

"pos_Anterior_and_nasal_vowel" {e~, i~}

"pos_Posterior_and_closed_vowel" {u, u~}

"pos_Posterior_and_oral_vowel" {O, o, u}

"pos_Posterior_and_nasal_vowel" {o~, u~}

"pos_Closed_and_oral_vowel" {h, i, u}

"pos_Semi_open_and_oral_vowel" {6, E, O}

"pos_Closed_and_nasal_vowel" {i~, u~}

"pos_Semi_open_and_nasal_vowel" {6~}

"pos_Reduced_and_oral_vowel" {a, i, u}

"pos_Reduced_and_nasal_vowel" {6~, i~, u~}

"pos_Semivowel" {w, j, w~, j~}

"pos_Oral_semivowel" {w, j}

"pos_Nasal_semivowel" {w~, j~}

“pos_Consonant” {p, b, t, d, k, g, m, n, J, s, S, z, f, v, Z, l, l∼, L, R, r}

“pos_Stop” {p, b, t, d, k, g, m, n, J}

“pos_Constrictive” {s, S, z, f, v, Z, l, l∼, L, R, r}

“pos_Fricative” {s, S, z, f, v, Z}

“pos_Liquid” {R, r, l, l∼, L}

“pos_Vibrant_liquid” {R, r}

“pos_Lateral_liquid” {l, l∼, L}

“pos_Bilabial” {p, b, m}

“pos_Labiodental” {f, v}

“pos_Dental” {t, d, n}

“pos_Alveolar” {l, l∼, r, s, z}

“pos_Nonconvex_alveolar” {l, l∼, r}

“pos_Convex_alveolar” {s, z}

“pos_Palatal” {J, S, Z, L}

“pos_Concave_palatal” {S, Z}

“pos_Nonconcave_Palatal” {J, L}

“pos_Velar” {k, g, R}

“pos_Voiced_consonant” {b, d, g, m, n, J, z, v, Z, l, l∼, L, r, R}

“pos_Unvoiced_consonant” {p, t, k, s, S, f}

“pos_Oral_consonant” {p, b, t, d, k, g, s, S, z, f, v, Z, l, l∼, L, r, R}

“pos_Nasal_consonant” {m, n, J}

“pos_Bilabial_stop” {p, b}

“pos_Dental_stop” {t, d}

“pos_Velar_stop” {k, g}

“pos_Unvoiced_stop” {p, t, k}

“pos_Voiced_stop” {b, d, g, m, n, J}

“pos_Voiced_fricative” {z, v, Z}

“pos_Unvoiced_fricative” {s, S, f}

“pos_Voiced_bilabial” {b, m}

“pos_Voiced_dental” {d, n}

“pos_Voiced_alveolar” {l, l∼, r, z}

“pos_Voiced_palatal” {J, Z, L}

“pos_Oral_palatal” {S, Z, L}

“pos_Voiced_velar” {g, R}

“pos_Voiced_Oral_consonant” {b, d, g, z, v, Z, l, l∼, L, r, R}

“pos_a_or_6_or_6∼” {a, 6, 6∼}

“pos_a_or_6∼” {a, 6∼}

“pos_6_or_6∼” {6, 6∼}

“pos_E_or_e_or_e∼” {E, e, e∼}

“pos_E_or_e” {E, e}

“pos_E_or_e∼” {E, e∼}

“pos_e_or_e∼” {e, e∼}

“pos_O_or_o_or_o∼” {O, o, o∼}

“pos_O_or_o” {O, o}

“pos_O_or_o∼” {O, o∼}

“pos_o_or_o∼” {o, o∼}

“pos_i_or_i∼” {i, i∼}

“pos_u_or_u∼_or_h” {u, u∼, h}

“pos_u_or_u∼” {u, u∼}

“pos_h_or_u∼” {h, u∼}

“pos_h_or_u” {h, u}

“pos_w_or_w∼” {w, w∼}

“pos_j_or_j∼” {j, j∼}

“pos_l_or_l∼” {l, l∼}

“pos_6” {6}

“pos_E” {E}

“pos_e” {e}

“pos_h” {h}

“pos_i” {i}

“pos_O” {O}

“pos_o” {o}

“pos_u” {u}

“pos_6∼” {6∼}

“pos_e∼” {e∼}

“pos_i∼” {i∼}

“pos_o∼” {o∼}

“pos_u∼” {u∼}

“pos_f” {f}

“pos_s” {s}

“pos_S” {S}

“pos_z” {z}

“pos_v” {v}

“pos_Z” {Z}

“pos_b” {b}

“pos_d” {d}

“pos_t” {t}

“pos_k” {k}

“pos_g” {g}

"pos_p" {p}

"pos_w" {w}

"pos_j" {j}

"pos_w~" {w~}

"pos_j~" {j~}

"pos_L" {L}

"pos_R" {R}

"pos_r" {r}

"pos_l~" {l~}

"pos_l" {l}

"pos_m" {m}

"pos_n" {n}

"pos_J" {J}

"Right_context_Diphthong" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, wa, we, wi, wo, ja, je, jo, ju, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~, w6~, we~}

"Right_context_Diphthong-D" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~}

"Right_context_Diphthong-C" {wa, we, wi, wo, ja, je, jo, ju, w6~, we~}

"Right_context_Diphthong-open" {aj, aw, wa, ja}

"Right_context_Diphthong-semi_open" {6j, Ew, Oj, Ow, wE, wO, jE, jO, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, w6~}

"Right_context_Diphthong-closed" {iw, uj, wi, ju, iw~, uj~, w~i, j~u, i~w, u~j, wi~, ju~, i~w~, u~j~, w~i~, j~u~}

"Right_context_Diphthong-oral" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, wa, we, wi, wo, ja, je, jo, ju}

"Right_context_Diphthong-nasal" {6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~, w6~, we~}

"Right_context_Diphthong-D_open" {aj, aw}

"Right_context_Diphthong-D_semi_open" {6j, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, 6j, Ew, Oj}

"Right_context_Diphthong-D_closed" {iw, i~w, iw~, i~w~, uj, u~j~, u~j, uj~}

"Right_context_Diphthong-C_open" {wa, ja}

"Right_context_Diphthong-C-semi_open" {wE, wO, jE, jO, w6~}

"Right_context_Diphthong-C_closed" {wi, w~i, wi~, w~i~, ju, j~u, ju~, j~u~}

"Right_context_Diphthong-D_oral" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj}

"Right_context_Diphthong-D_nasal" {6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~}

"Right_context_Diphthong-C_oral" {wa, we, wi, wo, ja, je, jo, ju}

"Right_context_Diphthong-C_nasal" {w6~, we~, w~6~, w~e~}

"Left_context_Diphthong" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, wa, we, wi, wo, ja, je, jo, ju, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~, w6~, we~}

"Left_context_Diphthong-D" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, 6~j~, 6~j, 6j~, 6~w~, 6~w, 6w~, o~j~, o~j, oj~, u~j~, u~j, uj~}

"Left_context_Diphthong-C" {wa, we, wi, wo, ja, je, jo, ju, w6∼, we∼}

"Left_context_Diphthong-open" {aj, aw, wa, ja}

"Left_context_Diphthong-semi_open" {6j, Ew, Oj, Ow, wE, wO, jE, jO, 6∼j∼, 6∼j, 6j∼, 6∼w∼, 6∼w, 6w∼, w6∼}

"Left_context_Diphthong-closed" {iw, uj, wi, ju, iw∼, uj∼, w∼i, j∼u, i∼w, u∼j, wi∼, ju∼, i∼w∼, u∼j∼, w∼i∼, j∼u∼}

"Left_context_Diphthong-oral" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj, wa, we, wi, wo, ja, je, jo, ju}

"Left_context_Diphthong-nasal" {6∼j∼, 6∼j, 6j∼, 6∼w∼, 6∼w, 6w∼, o∼j∼, o∼j, oj∼, u∼j∼, u∼j, uj∼, w6∼, we∼}

"Left_context_Diphthong-D_open" {aj, aw}

"Left_context_Diphthong-D_semi_open" {6j, 6∼j∼, 6∼j, 6j∼, 6∼w∼, 6∼w, 6w∼, 6j, Ew, Oj}

"Left_context_Diphthong-D_closed" {iw, i∼w, iw∼, i∼w∼, uj, u∼j∼, u∼j, uj∼}

"Left_context_Diphthong-C_open" {wa, ja}

"Left_context_Diphthong-C-semi_open" {wE, wO, jE, jO, w6∼}

"Left_context_Diphthong-C_closed" {wi, w∼i, wi∼, w∼i∼, ju, j∼u, ju∼, j∼u∼}

"Left_context_Diphthong-D_oral" {aj, aw, 6j, ew, oj, ow, Ew, Oj, iw, uj}

"Left_context_Diphthong-D_nasal" {6∼j∼, 6∼j, 6j∼, 6∼w∼, 6∼w, 6w∼, o∼j∼, o∼j, oj∼, u∼j∼, u∼j, uj∼}

"Left_context_Diphthong-C_oral" {wa, we, wi, wo, ja, je, jo, ju}

"Left_context_Diphthong-C_nasal" {w6∼, we∼, w∼6∼, w∼e∼}

"End_of_utterance_Vowel" {a, 6, E, O, e, h, i, o, u, 6∼, e∼, i∼, o∼, u∼}

"End_of_utterance_Anterior_vowel" {E, e, i, e∼, i∼}

"End_of_utterance_Central_vowel" {h, a, 6, 6∼}

"End_of_utterance_Posterior_vowel" {O, o, u, o∼, u∼}

"End_of_utterance_High_vowel" {i, u, i∼, u∼, h}

"End_of_utterance_Middle_vowel" {e, o, e∼, o∼}

"End_of_utterance_Nonrounded_vowel" {a, 6, E, e, h, i, 6∼, e∼, i∼}

"End_of_utterance_Closed_vowel" {h, i, u, i∼, u∼}

"End_of_utterance_Semi_open_vowel" {6, 6∼, E, O}

"End_of_utterance_Open_vowel" {a}

"End_of_utterance_Reduced_vowel" {a, i, u, 6∼, i∼, u∼}

"End_of_utterance_Oral_vowel" {a, 6, E, O, e, h, i, o, u}

"End_of_utterance_Nasal_vowel" {6∼, e∼, i∼, o∼, u∼}

"End_of_utterance_Anterior_and_closed_vowel" {i, i∼}

"End_of_utterance_Anterior_and_oral_vowel" {E, e, i}

"End_of_utterance_Anterior_and_nasal_vowel" {e∼, i∼}

"End_of_utterance_Posterior_and_closed_vowel" {u, u∼}

"End_of_utterance_Posterior_and_oral_vowel" {O, o, u}

"End_of_utterance_Posterior_and_nasal_vowel" {o∼, u∼}

"End_of_utterance_Closed_and_oral_vowel" {h, i, u}

"End_of_utterance_Semi_open_and_oral_vowel" {6, E, O}

"End_of_utterance_Closed_and_nasal_vowel" {i∼, u∼}

"End_of_utterance_Semi_open_and_nasal_vowel" {6∼}

"End_of_utterance_Reduced_and_oral_vowel" {a, i, u}

"End_of_utterance_Reduced_and_nasal_vowel" {6∼, i∼, u∼}

"End_of_utterance_vowel_with_pause" {aX, 6X, EX, OX, eX, hX, iX, oX, uX, 6∼X, e∼X, i∼X, o∼X, u∼X}

"End_of_utterance_Anterior_vowel_with_pause" {EX, eX, iX, e∼X, i∼X}

"End_of_utterance_Central_vowel_with_pause" {hX, aX, 6X, 6∼X}

"End_of_utterance_Posterior_vowel_with_pause" {OX, oX, uX, o∼X, u∼X}

"End_of_utterance_High_vowel_with_pause" {iX, uX, i∼X, u∼X, hX}

"End_of_utterance_Middle_vowel_with_pause" {eX, oX, e∼X, o∼X}

"End_of_utterance_Nonrounded_vowel_with_pause" {aX, 6X, EX, eX, hX, iX, 6∼X, e∼X, i∼X}

"End_of_utterance_Closed_vowel_with_pause" {hX, iX, uX, i∼X, u∼X}

"End_of_utterance_Semi_open_vowel_with_pause" {6X, 6∼X, EX, OX}

"End_of_utterance_Open_vowel_with_pause" {aX}

"End_of_utterance_Reduced_vowel_with_pause" {aX, iX, uX, 6∼X, i∼X, u∼X}

"End_of_utterance_Oral_vowel_with_pause" {aX, 6X, EX, OX, eX, hX, iX, oX, uX}

"End_of_utterance_Nasal_vowel_with_pause" {6∼X, e∼X, i∼X, o∼X, u∼X}

"End_of_utterance_Anterior_and_closed_vowel_with_pause" {iX, i∼X}

"End_of_utterance_Anterior_and_oral_vowel_with_pause" {EX, eX, iX}

"End_of_utterance_Anterior_and_nasal_vowel_with_pause" {e∼X, i∼X}

"End_of_utterance_Posterior_and_closed_vowel_with_pause" {uX, u∼X}

"End_of_utterance_Posterior_and_oral_vowel_with_pause" {OX, oX, uX}

"End_of_utterance_Posterior_and_nasal_vowel_with_pause" {o∼X, u∼X}

"End_of_utterance_Closed_and_oral_vowel_with_pause" {hX, iX, uX}

"End_of_utterance_Semi_open_and_oral_vowel_with_pause" {6X, EX, OX}

"End_of_utterance_Closed_and_nasal_vowel_with_pause" {i∼X, u∼X}

"End_of_utterance_Semi_open_and_nasal_vowel_with_pause" {6∼X}

"End_of_utterance_Reduced_and_oral_vowel_with_pause" {aX, iX, uX}

"End_of_utterance_Reduced_and_nasal_vowel_with_pause" {6∼X, i∼X, u∼X}

Information about number of syllables before.

Information about number of syllables after.

Information about number of words before.

Information about number of words after.

Information about number of phrases before.

Information about number of phrases after.

Position, counting forward, of the current monophone in the current syllable: possible values range from 1 to 8 and less than 2 to less than 7.

Position, counting backwards, of the current monophone in the current syllable: possible values range from 1 to 8 and less than 2 to less than 7.

Left Syllable Stressed or Left Syllable Not Stressed.

Left Syllable number of monophones: possible values range from 1 to 8 and less than 2 to less than 7.

Center Syllable Stressed or Center Syllable Not Stressed.

Center Syllable number of monophones: possible values range from 1 to 8 and less than 2 to less than 7.

Right Syllable Stressed or Right Syllable Not Stressed.

Right Syllable number of monophones: possible values range from 1 to 8 and less than 2 to less than 8.

Position, counting forward, of the current Syllable in the current Word: possible values range from 1 to 10 and less than 2 to less than 9.

Position, counting backwards, of the current Syllable in the current Word: possible values range from 1 to 10 and less than 2 to less than 9.

Position, counting forward, of the current Syllable in the current Phrase: possible values range from 1 to 50 and less than 2 to less than 49.

Position, counting backwards, of the current Syllable in the current Phrase: possible values range from 1 to 50 and less than 2 to less than 49.

Number of Stressed Syllables before current Syllable in current Phrase: possible values range from 0 to 30 and less than 1 to less than 29.

Number of Stressed Syllables after current Syllable in current Phrase: possible values range from 0 to 30 and less than 1 to less than 29.

Number of Syllables from previous Stressed to current Syllable: possible values range from 0 to 30 and less than 1 to less than 29.

Number of Syllables from current Syllable to next Stressed Syllable: possible values range from 0 to 30 and less than 1 to less than 29.

"Vowel_of_the_syllableAnterior" {E, e, h, i, e∼, i∼}

"Vowel_of_the_syllableCentral" {a, 6, 6∼}

"Vowel_of_the_syllablePosterior" {O, o, u, o∼, u∼}

"Vowel_of_the_syllableHigh" {i, u, i∼, u∼, h}

"Vowel_of_the_syllableMiddle" {e, o, e∼, o∼}

"Vowel_of_the_syllableNonrounded" {a, 6, E, e, h, i, 6∼, e∼, i∼}

"Vowel_of_the_syllableClosed" {h, i, u, i∼, u∼}

"Vowel_of_the_syllableSemi_open" {6, 6∼, E, O}

"Vowel_of_the_syllableOpen" {a}

"Vowel_of_the_syllableReduced" {a, i, u, 6∼, i∼, u∼}

"Vowel_of_the_syllableOral" {a, 6, E, O, e, h, i, o, u}

"Vowel_of_the_syllableNasal" {6∼, e∼, i∼, o∼, u∼}

"Vowel_of_the_syllableAnterior_and_closed" {i, i∼}

"Vowel_of_the_syllableAnterior_and_oral" {E, e, i}

"Vowel_of_the_syllableAnterior_and_nasal" {e∼, i∼}

"Vowel_of_the_syllablePosterior_and_closed" {u, u∼}

"Vowel_of_the_syllablePosterior_and_oral" {O, o, u}

“Vowel_of_the_syllablePosterior_and_nasal” {o∼, u∼}

“Vowel_of_the_syllableClosed_and_oral” {h, i, u}

“Vowel_of_the_syllableSemi_open_and_oral” {6, E, O}

“Vowel_of_the_syllableClosed_and_nasal” {i∼, u∼}

“Vowel_of_the_syllableSemi_open_and_nasal” {6∼}

“Vowel_of_the_syllableReduced_and_oral” {a, i, u}

“Vowel_of_the_syllableReduced_and_nasal” {6∼, i∼, u∼}

“Vowel_of_the_syllablea_or_6_or_6∼” {a, 6, 6∼}

“Vowel_of_the_syllablea_or_6∼” {a, 6∼}

“Vowel_of_the_syllable6_or_6∼” {6, 6∼}

“Vowel_of_the_syllableE_or_e_or_e∼” {E, e, e∼}

“Vowel_of_the_syllableE_or_e” {E, e}

“Vowel_of_the_syllableE_or_e∼” {E, e∼}

“Vowel_of_the_syllablee_or_e∼” {e, e∼}

“Vowel_of_the_syllableO_or_o_or_o∼” {O, o, o∼}

“Vowel_of_the_syllableO_or_o” {O, o}

“Vowel_of_the_syllableO_or_o∼” {O, o∼}

“Vowel_of_the_syllableo_or_o∼” {o, o∼}

“Vowel_of_the_syllablei_or_i∼” {i, i∼}

“Vowel_of_the_syllableu_or_u∼_or_h” {u, u∼, h}

“Vowel_of_the_syllableu_or_u∼” {u, u∼}

“Vowel_of_the_syllableh_or_u∼” {h, u∼}

“Vowel_of_the_syllableh_or_u” {h, u}

“Vowel_of_the_syllablel_or_l∼” {l, l∼}

“Vowel_of_the_syllable6” {6}

“Vowel_of_the_syllableE” {E}

“Vowel_of_the_syllablee” {e}

“Vowel_of_the_syllableh” {h}

“Vowel_of_the_syllablei” {i}

“Vowel_of_the_syllableO” {O}

“Vowel_of_the_syllableo” {o}

“Vowel_of_the_syllableu” {u}

“Vowel_of_the_syllable6∼” {6∼}

“Vowel_of_the_syllablee∼” {e∼}

“Vowel_of_the_syllablei∼” {i∼}

“Vowel_of_the_syllableo∼” {o∼}

“Vowel_of_the_syllableu∼” {u∼}

“No_vowel_in_the_syllable” {}

Left Word number of Syllables: possible values range from 1 to 10 and less than 2 to less than 9.

Center Word number of Syllables: possible values range from 1 to 10 and less than 2 to less than 9.

Right Word number of Syllables: possible values range from 1 to 10 and less than 2 to less than 9.

Position, counting forward, of the current Word in the current Phrase: possible values range from 1 to 30 and less than 2 to less than 29.

Position, counting backwards, of the current Word in the current Phrase: possible values range from 1 to 30 and less than 2 to less than 29.

Left Phrase number of Syllables: possible values range from 1 to 50 and less than 2 to less than 49.

Left Phrase number of Words: possible values range from 1 to 30 and less than 2 to less than 29.

Center Phrase number of Syllables: possible values range from 1 to 50 and less than 2 to less than 49.

Center Phrase number of Words: possible values range from 1 to 30 and less than 2 to less than 29.

Right Phrase number of Syllables: possible values range from 1 to 50 and less than 2 to less than 49.

Right Phrase number of Words: possible values range from 1 to 30 and less than 2 to less than 29.

Position, counting forward, of the current Phrase in the current Utterance: possible values range from 1 to 30 and less than 2 to less than 29.

Position, counting backwards, of the current Phrase in the current Utterance: possible values range from 1 to 30 and less than 2 to less than 29.

Number of Syllables in the Utterance: possible values range from 1 to 150; less than 2 to less than 70; between 71 and 80 to between 111 to 150, in ranges of 10 values; and between 71 and 90 to between 131 to 150, in ranges of 20 values.

Number of Words in the Utterance: possible values range from 1 to 70 and less than 2 to less than 69.

Number of Phrases in the Utterance: possible values range from 1 to 30 and less than 2 to less than 29.

B.5 Phonemes Boundaries File Format Example

Reproduction of File portuguese_euro_008p24.lab

```
0 9927000 X
9927000 10520000 e
10520000 11595000 S
11595000 12209000 t
12209000 13019000 a
13019000 13315000 u~
13315000 13966000 b
13966000 14775000 k
14775000 15414000 a
15414000 15843000 d
15843000 16694000 s
16694000 17053000 b
17053000 17512000 r
```

```
17512000 18409000 s
18409000 19394000 a
19394000 20316000 t
20316000 21371000 a
21371000 23511000 d
23511000 26361000 X
26361000 26807000 m
26807000 27111000 6
27111000 27579000 Z
27579000 27946000 v
27946000 28934000 a
```

B.6 Fundamental Frequency File Example

Reproduction of Part of the File portuguese_euro_001p1.lf0

```
0   -1e+10
1   -1e+10
2   -1e+10
3   -1e+10
4   -1e+10
5   -1e+10
6   -1e+10

(...)

285 4.33533
286 4.35984
287 4.30606
288 4.3059
289 4.30544
290 4.32617
291 4.29774
292 4.29244
293 4.2894
294 4.31451
295 4.27389
296 4.27577
297 4.2982

(...)

1160    4.21546
1161    4.12322
1162    4.1705
1163    4.16976
1164    4.20295
1165    4.20387
1166    4.12945
1167    4.13
1168    4.14555
1169    4.22547
1170    4.2488
1171    -1e+10
1172    -1e+10
1173    -1e+10
```

```
1174    -1e+10
1175    -1e+10
1176    -1e+10
1177    -1e+10
1178    -1e+10
1179    -1e+10
1180    -1e+10
1181    -1e+10
1182    -1e+10
1183    -1e+10
```

B.7 Mel-cepstral File Example

Reproduction of Part of the File portuguese_euro_001p1.mcp

```
0   3.96856
1   1.15359
2   -0.514667
3   0.296488
4   0.0503076
5   -0.0322024
6   0.231352
7   -0.170016
8   -0.337306
9   0.136926
10  0.302986
11  0.122616
12  0.0483132
13  0.0196315
14  -0.0232513
15  0.134954
16  0.284009
17  -0.0330638
18  0.0713024
19  0.0172048
20  0.0433208

(...)

29528   0.455613
29529   0.31268
29530   0.653731
29531   0.335845
29532   0.0241619
29533   -0.0312129
29534   0.101763
29535   0.095027
29536   0.0582945
29537   0.12413
29538   -0.0395192
29539   -0.140123
29540   0.0812458
29541   -0.277785
29542   -0.0939404
29543   0.0506658
29544   -0.00858268
```

```
29545   0.0436031
29546   -0.0205608
29547   0.19525
29548   0.161155
29549   0.201396
```

B.8 Data Format File Example for the Data Training

Reproduction of Part of the File portuguese_euro_008p24.cmp

```
------------------------------------ Samples: 0->-1 -------------------------
0:       3.744   1.729   0.512   0.743   0.361   0.217   0.072  -0.191  -0.191
         0.292  -0.316  -0.097  -0.008  -0.022   0.011  -0.014  -0.033   0.084
         0.204  -0.158  -0.038   0.076  -0.016   0.024  -0.047   2.076   0.950
         0.023   0.440   0.336   0.178  -0.004  -0.233   0.111  -0.110  -0.115
        -0.014  -0.046   0.031   0.045   0.052  -0.025  -0.017   0.077  -0.062
         0.003  -0.024  -0.025   0.019  -0.005  -3.336  -1.559  -0.978  -0.607
        -0.050  -0.076  -0.151  -0.084  -0.362   0.437   0.404   0.167  -0.075
         0.104   0.067   0.131   0.016  -0.203  -0.253   0.191   0.082  -0.199
        -0.017  -0.009   0.084   4.646   2.335  -4.621
1:       4.152   1.900   0.047   0.879   0.672   0.357  -0.007  -0.465   0.222
        -0.220  -0.229  -0.028  -0.091   0.061   0.089   0.104  -0.050  -0.034
         0.155  -0.125   0.007  -0.047  -0.050   0.038  -0.010   0.537   0.131
        -0.364   0.105   0.194   0.076  -0.074  -0.191  -0.095   0.085  -0.006
         0.066  -0.081   0.090  -0.013   0.131  -0.075  -0.026  -0.024   0.008
         0.040  -0.102  -0.020   0.008  -0.020   0.259  -0.080   0.203  -0.063
        -0.234  -0.129   0.010   0.167  -0.050  -0.048  -0.186  -0.007   0.005
         0.014  -0.181   0.027  -0.116   0.185   0.050  -0.050  -0.009   0.042
         0.028  -0.014  -0.114   4.670  -0.061  -0.171
2:       4.819   1.991  -0.216   0.953   0.749   0.368  -0.077  -0.573   0.102
        -0.159  -0.328   0.034  -0.170   0.158  -0.014   0.249  -0.183   0.032
         0.156  -0.141   0.043  -0.128  -0.056   0.039  -0.087   0.287   0.086
        -0.092   0.022   0.015  -0.018  -0.040  -0.110  -0.039   0.044  -0.045
         0.043  -0.030   0.033  -0.007   0.074  -0.040   0.040   0.024   0.005
         0.025  -0.049   0.008   0.006  -0.015  -0.760  -0.009   0.341  -0.102
        -0.125  -0.059   0.060  -0.004   0.163  -0.033   0.107  -0.039   0.097
        -0.128   0.193  -0.142   0.186  -0.052   0.046   0.043  -0.022   0.063
         0.028   0.011   0.125   4.524  -0.071   0.150

(...)

388:     4.677   2.925  -0.015   0.349   0.274   0.213  -0.276  -0.478   0.180
        -0.041   0.007   0.368  -0.239   0.234   0.054   0.302  -0.172  -0.137
        -0.210  -0.038  -0.093  -0.213  -0.001   0.062   0.056  -0.266  -0.076
         0.164   0.055   0.085   0.082   0.024  -0.024   0.040  -0.050  -0.077
         0.024   0.035   0.060   0.023  -0.061   0.031  -0.022   0.036  -0.043
        -0.071   0.107   0.061  -0.006  -0.012   0.012  -0.002   0.063   0.086
        -0.103   0.063   0.086  -0.080   0.037  -0.093  -0.208   0.103   0.073
        -0.049  -0.033  -0.230  -0.014   0.087   0.086   0.028   0.019   0.144
         0.092  -0.162  -0.122   4.775   0.016   0.004
389:     4.417   2.848   0.181   0.447   0.307   0.326  -0.209  -0.542   0.238
        -0.138  -0.174   0.444  -0.167   0.269   0.061   0.126  -0.149  -0.116
        -0.130  -0.067  -0.155  -0.034   0.106  -0.026  -0.017  -2.338  -1.463
         0.007  -0.175  -0.137  -0.106   0.138   0.239  -0.090   0.021  -0.004
```

```
      -0.184   0.120  -0.117  -0.027  -0.151   0.086   0.069   0.105   0.019
       0.047   0.107   0.001  -0.031  -0.028  -4.157  -2.770  -0.376  -0.545
      -0.341  -0.439   0.142   0.606  -0.297   0.234   0.354  -0.519   0.095
      -0.305  -0.068   0.050   0.125   0.095   0.051   0.096   0.216  -0.145
      -0.213   0.113   0.091   4.792   0.018  -0.000
-------------------------------------- END -------------------------------
```

B.9 Window Files for Static and Dynamic Features

Window files contain the window coefficients to calculate the static and dynamic features:

- "win1" contains the static features window coefficients;
- "win2" contains the velocity, delta, dynamic features window coefficients;
- "win3" contains the acceleration, delta delta, dynamic features window coefficients.

Values from file "lf0.win1"

1 1.0

Values from file "lf0.win2"

3 -0.5 0.0 0.5

Values from file "lf0.win3"

3 1.0 -2.0 1.0

Values from file "mcp.win1"

1 1.0

Values from file "mcp.win2"

3 -0.5 0.0 0.5

Values from file "mcp.win3"

3 1.0 -2.0 1.0

Appendix C

Hidden Markov Models Files

C.1 Model Prototype Definition File

The HMM prototype definition file for the European Portuguese system is the following:

```
~o
<VecSize> 78 <USER><DIAGC>
<MSDInfo> 4  0  1  1  1
<StreamInfo> 4 75 1 1 1
<BeginHMM>
  <NumStates> 7
  <State> 2
  <SWeights> 4 1.0 1.0 1.0 1.0
  <Stream> 1
    <Mean> 75
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0
    <Variance> 75
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0
  <Stream> 2
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
```

```
    <Mean> 0
    <Variance> 0
  <Stream> 3
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 4
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <State> 3
  <SWeights> 4 1.0 1.0 1.0 1.0
  <Stream> 1
    <Mean> 75
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0
    <Variance> 75
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0
  <Stream> 2
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 3
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 4
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <State> 4
  <SWeights> 4 1.0 1.0 1.0 1.0
  <Stream> 1
    <Mean> 75
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0
    <Variance> 75
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0
  <Stream> 2
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 3
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 4
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
```

```
    0.0
  <Variance> 1
    1.0
<Mixture> 2 0.5000
  <Mean> 0
  <Variance> 0
<State> 5
<SWeights> 4 1.0 1.0 1.0 1.0
<Stream> 1
  <Mean> 75
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
    0.0 0.0 0.0 0.0 0.0
  <Variance> 75
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0 1.0 1.0
<Stream> 2
<NumMixes> 2
<Mixture> 1 0.5000
  <Mean> 1
    0.0
  <Variance> 1
    1.0
<Mixture> 2 0.5000
  <Mean> 0
  <Variance> 0
<Stream> 3
<NumMixes> 2
<Mixture> 1 0.5000
  <Mean> 1
    0.0
  <Variance> 1
    1.0
<Mixture> 2 0.5000
  <Mean> 0
  <Variance> 0
<Stream> 4
<NumMixes> 2
<Mixture> 1 0.5000
  <Mean> 1
    0.0
  <Variance> 1
    1.0
<Mixture> 2 0.5000
  <Mean> 0
  <Variance> 0
<State> 6
<SWeights> 4 1.0 1.0 1.0 1.0
```

```
  <Stream> 1
    <Mean> 75
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
      0.0 0.0 0.0 0.0 0.0
    <Variance> 75
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
      1.0 1.0 1.0 1.0 1.0
  <Stream> 2
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 3
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <Stream> 4
  <NumMixes> 2
  <Mixture> 1 0.5000
    <Mean> 1
      0.0
    <Variance> 1
      1.0
  <Mixture> 2 0.5000
    <Mean> 0
    <Variance> 0
  <TransP> 7
    0.000e+0 1.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0
    0.000e+0 6.000e-1 4.000e-1 0.000e+0 0.000e+0 0.000e+0 0.000e+0
    0.000e+0 0.000e+0 6.000e-1 4.000e-1 0.000e+0 0.000e+0 0.000e+0
    0.000e+0 0.000e+0 0.000e+0 6.000e-1 4.000e-1 0.000e+0 0.000e+0
    0.000e+0 0.000e+0 0.000e+0 0.000e+0 6.000e-1 4.000e-1 0.000e+0
    0.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0 6.000e-1 4.000e-1
    0.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0 0.000e+0
<EndHMM>
```

C.2 Examples of a Phoneme's Hidden Markov Model Files

Examples of the Phoneme /a/ Hidden Markov Model Files, before and after re-estimation, are following presented.

Hidden Markov Model File of the Phoneme /a/, before re-estimation

```
~o
<STREAMINFO> 4 75 1 1 1
<MSDINFO> 4 0 1 1 1
<VECSIZE> 78<NULLD><USER><DIAGC>
~h "a"
<BEGINHMM>
<NUMSTATES> 7
<STATE> 2
<STREAM> 1
<MEAN> 75
 5.680285e+000 2.869029e+000 -3.158306e-001 6.255158e-001 -2.582239e-001
 1.453627e-001 -1.347974e-001 2.974754e-002 -1.584741e-001 -4.983088e-002
 -2.429484e-001 2.629248e-001 -9.217250e-002 2.223544e-002 -1.846510e-002
 1.065742e-001 -1.012778e-001 -7.110933e-002 -6.082824e-002 -1.417648e-001
 -7.496329e-002 -9.719331e-002 -6.743717e-002 -3.150209e-002 -1.074062e-001
 2.686367e-001 1.973642e-001 -1.197651e-001 -2.948458e-002 -5.674085e-002
 -6.790268e-002 1.012902e-002 -4.434020e-002 -1.111826e-002 4.001113e-003
 -2.691035e-002 6.563016e-002 1.162961e-002 1.823530e-002 5.485525e-003
 2.024699e-003 -2.806091e-002 -3.044690e-002 -9.911239e-003 -2.846553e-002
 3.941567e-003 -1.187937e-002 5.444724e-003 2.448096e-004 -5.605025e-003
 -7.642567e-002 -1.196761e-001 7.901477e-003 -1.338834e-002 1.490660e-002
 3.332086e-002 4.432013e-003 1.915248e-002 1.669097e-002 3.185470e-003
 3.379107e-002 -2.270242e-005 6.752308e-004 2.204371e-003 -1.164406e-002
 -1.127228e-002 9.867410e-003 3.320295e-003 7.118180e-003 9.370820e-003
 4.995848e-003 4.274378e-003 4.408400e-003 5.392765e-003 4.340503e-004
<VARIANCE> 75
 4.089649e-001 2.816712e-001 1.497134e-001 8.086012e-002 1.525544e-001
 7.909854e-002 3.897391e-002 4.631180e-002 5.641913e-002 2.363160e-002
 4.601870e-002 2.139665e-002 1.794054e-002 1.659164e-002 1.336639e-002
 1.620437e-002 1.574464e-002 1.294516e-002 1.196254e-002 1.283176e-002
 8.989383e-003 7.485199e-003 7.073106e-003 6.349994e-003 1.087544e-002
 5.354426e-002 6.007685e-002 1.383886e-002 9.250308e-003 1.000660e-002
 8.029110e-003 5.629198e-003 5.449177e-003 4.935551e-003 4.869333e-003
 6.100937e-003 4.626961e-003 4.691097e-003 3.271718e-003 3.538322e-003
 3.278442e-003 3.432890e-003 2.492084e-003 2.616451e-003 2.983556e-003
 2.337500e-003 2.159499e-003 2.065930e-003 1.928001e-003 2.538200e-003
 1.234125e-001 6.504867e-002 4.884181e-002 2.372944e-002 2.437549e-002
 2.039140e-002 1.550167e-002 1.657197e-002 1.577055e-002 1.536598e-002
 1.936136e-002 1.313478e-002 1.089107e-002 1.044247e-002 1.137670e-002
 1.242220e-002 9.584584e-003 9.079142e-003 9.077965e-003 9.106769e-003
 8.160944e-003 8.045697e-003 7.944401e-003 7.010192e-003 8.890267e-003
<GCONST> -1.880147e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 8.420473e-001
<MEAN> 1 4.669183e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
```

```
<MIXTURE> 2 1.579527e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 6.978536e-001
<MEAN> 1 -3.650823e-003
<VARIANCE> 1 7.039504e-004
<GCONST> -5.420926e+000
<MIXTURE> 2 3.021464e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 6.978536e-001
<MEAN> 1 -5.253255e-003
<VARIANCE> 1 4.643095e-003
<GCONST> -3.534497e+000
<MIXTURE> 2 3.021464e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 3
<STREAM> 1
<MEAN> 75
 6.283080e+000 3.096561e+000 -6.613206e-001 4.909454e-001 -3.565168e-001
 9.279284e-002 -1.457840e-001 -4.387248e-002 -1.439868e-001 -1.043513e-002
 -1.695406e-001 4.397422e-001 -4.854864e-002 3.975075e-002 -9.206934e-002
 6.648490e-002 -1.794838e-001 -1.481623e-001 -7.856791e-002 -1.816700e-001
 -2.685716e-002 -1.041740e-001 -1.970671e-002 1.082832e-003 -1.189393e-001
 7.634778e-002 -8.112167e-003 -4.695000e-002 -2.766069e-002 -4.306486e-002
 9.096184e-003 1.393224e-003 2.549304e-003 2.273092e-002 -5.056483e-003
 4.405711e-002 2.302707e-002 3.981893e-003 -1.282181e-002 -3.089828e-002
 -1.721774e-002 -1.838849e-002 5.741579e-003 -4.313791e-003 8.180684e-003
 1.149883e-002 4.888603e-003 1.602126e-002 2.131438e-003 1.356443e-004
 -3.065348e-002 -1.109950e-002 1.281863e-002 8.267423e-003 2.292708e-003
 8.256826e-003 -1.487919e-003 6.262337e-003 4.447843e-003 -3.166094e-003
 1.890869e-003 -1.065250e-002 -7.665734e-003 -8.157125e-003 -6.685391e-004
 2.654897e-003 3.267192e-003 1.010487e-002 7.876453e-004 4.409556e-003
 -1.963078e-003 2.276087e-003 9.862548e-005 -4.595688e-003 1.352785e-004
<VARIANCE> 75
 2.122324e-001 1.208374e-001 8.113578e-002 5.436461e-002 1.094496e-001
 4.710227e-002 4.191863e-002 4.998121e-002 4.663548e-002 2.460325e-002
 3.608933e-002 1.848086e-002 2.056140e-002 1.569933e-002 1.666712e-002
 1.778935e-002 1.949053e-002 1.550722e-002 1.119884e-002 1.137778e-002
 9.510711e-003 8.357611e-003 8.881397e-003 7.202502e-003 1.137411e-002
 8.087551e-003 6.061737e-003 5.520017e-003 4.292023e-003 4.474618e-003
 3.785433e-003 3.422555e-003 3.317485e-003 3.214924e-003 2.537954e-003
 2.470329e-003 2.195281e-003 2.256812e-003 2.293161e-003 1.987732e-003
 2.037635e-003 1.996980e-003 2.186917e-003 1.901079e-003 1.798409e-003
 1.477216e-003 1.395602e-003 1.320645e-003 1.315077e-003 1.901173e-003
 6.817517e-002 2.328966e-002 1.881267e-002 1.139076e-002 1.757151e-002
 1.202207e-002 1.004893e-002 1.056494e-002 1.240019e-002 8.144977e-003
 1.536188e-002 1.016119e-002 9.658707e-003 7.407085e-003 8.599474e-003
 9.044888e-003 6.379372e-003 7.388057e-003 6.276374e-003 6.517994e-003
 5.732707e-003 5.086421e-003 5.270642e-003 5.387155e-003 7.083282e-003
<GCONST> -2.179039e+002
```

```
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 9.991854e-001
<MEAN> 1 4.639900e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 8.145534e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 9.975563e-001
<MEAN> 1 -9.876596e-003
<VARIANCE> 1
 2.194553e-004
<GCONST> -6.586485e+000
<MIXTURE> 2 2.443660e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.975563e-001
<MEAN> 1 6.552959e-004
<VARIANCE> 1 2.204393e-003
<GCONST> -4.279426e+000
<MIXTURE> 2 2.443660e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 4
<STREAM> 1
<MEAN> 75
 6.484803e+000 3.062266e+000 -8.171138e-001 3.802651e-001 -4.735110e-001
 1.730113e-001 -1.425388e-001 -1.907701e-002 -5.975538e-002 -5.052727e-002
 1.750381e-002 5.310808e-001 -1.264496e-001 -4.254993e-002 -2.089831e-001
 3.327885e-002 -2.250636e-001 -6.868704e-002 -1.001156e-001 -1.430876e-001
 2.096891e-003 -7.097451e-002 5.606905e-002 -4.832263e-002 -1.289876e-001
 -1.143475e-002 3.574357e-003 3.538510e-003 -2.655954e-003 -3.314569e-003
 7.749069e-003 -3.604469e-003 -2.570627e-004 5.583893e-003 -7.147311e-003
 6.026736e-003 1.147330e-003 -3.860628e-003 9.316143e-004 1.514014e-003
 1.100568e-003 -1.846349e-003 3.512351e-003 -3.086037e-003 2.662490e-003
 -2.110878e-003 1.333437e-003 2.363695e-003 -3.128774e-003 3.783629e-003
 -1.108133e-002 7.143524e-003 6.369598e-003 2.083891e-003 4.735759e-003
 -2.502184e-003 2.939537e-004 -2.199365e-003 -4.050154e-003 -6.106487e-004
 -6.581097e-003 -1.182115e-003 3.518057e-003 4.009466e-003 5.932585e-003
 8.394792e-004 1.212851e-003 -3.978339e-003 1.263093e-004 -2.303828e-003
 -1.238671e-003 -1.487660e-003 -1.541377e-003 1.966089e-003 1.696325e-003
<VARIANCE> 75
 1.771052e-001 7.562875e-002 5.060370e-002 3.545594e-002 4.808841e-002
 3.552059e-002 3.044456e-002 3.256055e-002 2.364551e-002 1.552028e-002
 1.787508e-002 1.211884e-002 1.516401e-002 1.156203e-002 9.131740e-003
 1.456044e-002 9.088043e-003 1.497142e-002 9.834149e-003 8.229506e-003
 6.917315e-003 6.506952e-003 6.591529e-003 6.690333e-003 8.525670e-003
 4.000148e-003 3.900229e-003 3.404244e-003 2.646899e-003 2.955719e-003
 2.248634e-003 2.248585e-003 2.250704e-003 1.810680e-003 1.614877e-003
 1.546610e-003 1.286924e-003 1.293684e-003 1.341127e-003 1.142535e-003
 1.046004e-003 9.998409e-004 1.088937e-003 8.793372e-004 8.567912e-004
 8.658202e-004 8.135129e-004 7.878919e-004 6.953944e-004 1.068753e-003
 6.846434e-002 2.015058e-002 1.358686e-002 8.613971e-003 1.286878e-002
 9.922920e-003 7.992115e-003 8.817165e-003 1.112618e-002 6.534432e-003
 1.544250e-002 1.957450e-002 8.821790e-003 5.333550e-003 4.428985e-003
 5.129130e-003 3.882912e-003 4.833535e-003 4.023375e-003 4.001094e-003
 4.248135e-003 4.226505e-003 6.179841e-003 3.659453e-003 4.490431e-003
<GCONST> -2.473642e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 9.994920e-001
<MEAN> 1 4.628002e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 5.079580e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3 <NUMMIXES> 2
<MIXTURE> 1 9.994920e-001
<MEAN> 1 -2.620848e-003
<VARIANCE> 1 4.945463e-005
<GCONST> -8.076578e+000
<MIXTURE> 2 5.079580e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.994920e-001
<MEAN> 1 -2.807400e-004
<VARIANCE> 1 1.484119e-004
<GCONST> -6.977642e+000
<MIXTURE> 2 5.079580e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 5
<STREAM> 1
<MEAN> 75
 6.172116e+000 3.115268e+000 -4.717699e-001 4.950087e-001 -7.494054e-001
 2.794445e-002 -5.529208e-002 8.709366e-002 3.700882e-002 -1.201547e-001
 -4.777139e-002 3.518775e-001 -6.500734e-002 5.329799e-002 -6.346886e-002
 -1.233291e-002 -2.476731e-001 -5.881895e-002 -1.234928e-001 -1.442156e-001
 -1.386336e-004 -8.741182e-002 2.348843e-002 -4.299700e-002 -5.315135e-002
 -7.646725e-002 1.888121e-002 5.709797e-002 1.520324e-002 -8.734064e-005
 -4.912035e-003 -1.992404e-003 -2.359827e-003 -3.346562e-004 -3.886102e-003
 -1.009537e-002 -1.178695e-002 -3.040735e-003 1.448888e-002 1.745068e-002
 9.962967e-003 1.092402e-002 -1.973451e-003 -3.692624e-003 -7.820646e-003
 -4.532821e-003 -6.499815e-003 -2.244740e-003 -8.233513e-004 7.853754e-003
 -2.854955e-002 -1.648896e-003 1.152971e-002 3.075943e-003 4.966778e-003
 2.260783e-003 -1.272010e-003 1.664775e-004 -5.087157e-004 -3.164764e-004
 2.356320e-004 3.891429e-004 -1.382057e-003 2.339106e-003 9.299759e-004
 3.541875e-003 4.893335e-003 1.044269e-003 9.612349e-004 1.145145e-004
 -1.945179e-003 -1.794683e-003 -1.157956e-003 -8.069407e-004 -7.197033e-004
<VARIANCE> 75
 2.899052e-001 1.191433e-001 1.256315e-001 5.719721e-002 9.985071e-002
 8.112300e-002 4.140688e-002 4.600666e-002 2.890667e-002 2.390043e-002
 2.587129e-002 3.536739e-002 1.915744e-002 2.149249e-002 2.013053e-002
 1.818011e-002 1.446581e-002 1.339542e-002 1.405390e-002 1.211570e-002
 1.254129e-002 1.004722e-002 8.319393e-003 8.345520e-003 1.266630e-002
 7.368339e-003 5.286343e-003 4.877661e-003 3.844562e-003 3.673277e-003
 3.524277e-003 3.443154e-003 2.970148e-003 2.387640e-003 2.215949e-003
 2.347940e-003 1.830755e-003 1.790074e-003 1.753662e-003 1.619819e-003
 1.447564e-003 1.452024e-003 1.353800e-003 1.467978e-003 1.327325e-003
 1.272672e-003 1.216161e-003 1.160769e-003 1.010550e-003 1.483326e-003
 7.456575e-002 2.390350e-002 1.974105e-002 1.163197e-002 1.315003e-002
 1.457281e-002 1.179670e-002 1.127549e-002 1.009303e-002 7.561693e-003
 1.289061e-002 1.892443e-002 1.112749e-002 6.906633e-003 7.242564e-003
 6.077180e-003 6.028195e-003 6.298085e-003 5.472540e-003 5.900606e-003
 5.711789e-003 5.077768e-003 6.493759e-003 5.140565e-003 5.774564e-003
<GCONST> -2.213594e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 9.990596e-001
<MEAN> 1 4.608026e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 9.404389e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 9.978057e-001
<MEAN> 1 -4.839413e-003
<VARIANCE> 1 7.183420e-005
<GCONST> -7.703273e+000
<MIXTURE> 2 2.194357e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.978057e-001
<MEAN> 1 -6.063694e-004
<VARIANCE> 1 2.349775e-004
<GCONST> -6.518144e+000
<MIXTURE> 2 2.194357e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 6
<STREAM> 1
<MEAN> 75
 5.304475e+000 3.040450e+000 -2.856365e-001 5.282518e-001 -4.516693e-001
 2.807854e-001 -1.667855e-001 2.348327e-002 -3.970765e-002 -1.582609e-001
 8.071586e-003 4.121417e-001 -7.490131e-002 4.959830e-002 -8.583564e-002
 9.868980e-002 -1.635699e-001 -6.695002e-002 -1.133527e-001 -1.043271e-001
 -7.923343e-002 -6.811576e-002 7.244040e-003 -4.917356e-002 -5.123671e-002
 -1.785906e-001 -5.558249e-002 9.137577e-002 2.762049e-002 3.984374e-002
 1.017126e-002 1.029417e-002 2.511593e-002 -1.285183e-002 8.221339e-003
 -9.081028e-003 -2.859096e-002 5.491662e-003 2.571850e-004 1.426945e-002
 3.434399e-004 1.907313e-002 -1.544459e-004 1.331505e-002 7.837088e-003
 -4.689711e-003 4.434191e-003 -1.286122e-002 1.216869e-003 -1.591385e-004
 4.608543e-003 -2.325957e-002 -4.398187e-003 1.648929e-004 1.117323e-002
 -1.040830e-003 8.277005e-003 9.776711e-003 -8.712521e-005 1.186788e-002
 -1.538839e-003 -1.057655e-002 -1.863560e-003 -9.439423e-003 -9.320381e-003
 -9.417804e-003 -3.943326e-003 1.688886e-003 3.652957e-003 3.655393e-003
 6.422806e-003 4.651651e-003 7.710585e-004 1.310163e-003 -3.519061e-003
<VARIANCE> 75
 5.709512e-001 1.774016e-001 1.806531e-001 4.844262e-002 7.408302e-002
 5.765606e-002 3.029956e-002 5.198014e-002 3.786019e-002 2.829556e-002
 4.601412e-002 2.732973e-002 2.366690e-002 1.672540e-002 1.955198e-002
 1.650136e-002 1.741969e-002 1.301701e-002 1.505990e-002 1.480491e-002
 1.243707e-002 1.230802e-002 9.743297e-003 8.655710e-003 1.149798e-002
 4.353357e-002 3.470933e-002 1.430440e-002 7.279304e-003 7.141347e-003
 5.641914e-003 6.070070e-003 6.038132e-003 4.500529e-003 5.781045e-003
 5.092887e-003 5.263933e-003 4.096777e-003 3.654578e-003 3.383604e-003
 3.251637e-003 2.765874e-003 2.744458e-003 2.768171e-003 2.654761e-003
 2.626398e-003 2.495540e-003 2.233177e-003 2.163998e-003 2.582867e-003
 1.411367e-001 7.044913e-002 3.884758e-002 2.369957e-002 2.227083e-002
 1.915058e-002 2.213786e-002 1.927857e-002 1.619962e-002 1.549782e-002
 2.500152e-002 3.276065e-002 2.044817e-002 1.477911e-002 1.467697e-002
 1.302884e-002 1.013010e-002 1.092122e-002 1.075404e-002 1.103982e-002
 1.087954e-002 1.088815e-002 1.066800e-002 9.311851e-003 1.057165e-002
<GCONST> -1.834110e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 8.230119e-001
<MEAN> 1 4.501863e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 1.769880e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 7.712878e-001
<MEAN> 1 -1.033593e-002
<VARIANCE> 1 5.636676e-004
<GCONST> -5.643169e+000
<MIXTURE> 2 2.287122e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 7.712878e-001
<MEAN> 1 2.408203e-003
<VARIANCE> 1 3.172550e-003
<GCONST> -3.915342e+000
<MIXTURE> 2 2.287122e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<TRANSP> 7
 0.000000e+000 1.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 5.751238e-001 4.248762e-001 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 7.903883e-001 2.096117e-001 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 8.692855e-001 1.307145e-001
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 7.579938e-001
 2.420063e-001 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 7.283603e-001 2.716397e-001
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
<ENDHMM>
```

Hidden Markov Model File of the Phoneme /a/, after re-estimation

```
~o
<STREAMINFO> 4 75 1 1 1
<MSDINFO> 4 0 1 1 1
<VECSIZE> 78<NULLD><USER><DIAGC>
~h "a"
<BEGINHMM>
<NUMSTATES> 7
<STATE> 2
<STREAM> 1
<MEAN> 75
 5.681834e+000 2.870263e+000 -3.160225e-001 6.242437e-001 -2.586395e-001
 1.461518e-001 -1.351238e-001 2.977822e-002 -1.589978e-001 -4.998430e-002
 -2.427037e-001 2.633354e-001 -9.226663e-002 2.279400e-002 -1.917341e-002
 1.065384e-001 -1.010693e-001 -7.122381e-002 -6.093309e-002 -1.417712e-001
 -7.469261e-002 -9.729404e-002 -6.714995e-002 -3.155933e-002 -1.071108e-001
 2.682337e-001 1.959620e-001 -1.192103e-001 -2.951328e-002 -5.666073e-002
 -6.751867e-002 9.988864e-003 -4.425409e-002 -1.085920e-002 4.063886e-003
 -2.669773e-002 6.550436e-002 1.158872e-002 1.822025e-002 5.236183e-003
 1.913034e-003 -2.789413e-002 -3.039538e-002 -1.001422e-002 -2.841410e-002
 4.016284e-003 -1.179106e-002 5.591167e-003 8.161552e-005 -5.775399e-003
 -7.690504e-002 -1.191025e-001 7.935163e-003 -1.327656e-002 1.443478e-002
 3.316778e-002 4.242208e-003 1.896814e-002 1.676349e-002 3.379855e-003
 3.387947e-002 1.199140e-005 4.560882e-004 1.990271e-003 -1.130770e-002
 -1.112358e-002 9.926850e-003 3.113391e-003 6.876754e-003 9.050235e-003
 5.063588e-003 4.417765e-003 4.425318e-003 5.485985e-003 7.901793e-004
<VARIANCE> 75
 4.090358e-001 2.811737e-001 1.499366e-001 8.083030e-002 1.525167e-001
 7.883903e-002 3.898547e-002 4.625502e-002 5.628227e-002 2.358621e-002
 4.596424e-002 2.141077e-002 1.801004e-002 1.657575e-002 1.341861e-002
 1.621129e-002 1.587055e-002 1.294327e-002 1.196736e-002 1.286448e-002
 8.978319e-003 7.484601e-003 7.089382e-003 6.359507e-003 1.085551e-002
 5.336566e-002 6.016097e-002 1.383476e-002 9.230866e-003 9.985825e-003
 8.030841e-003 5.635656e-003 5.448000e-003 4.912145e-003 4.851738e-003
 6.098524e-003 4.603765e-003 4.686568e-003 3.267995e-003 3.537019e-003
 3.267354e-003 3.436069e-003 2.488808e-003 2.614300e-003 2.981762e-003
 2.334780e-003 2.159404e-003 2.063937e-003 1.922961e-003 2.523865e-003
 1.231528e-001 6.491653e-002 4.853828e-002 2.366610e-002 2.431221e-002
 2.040050e-002 1.549685e-002 1.648839e-002 1.563333e-002 1.529045e-002
 1.922953e-002 1.301898e-002 1.083414e-002 1.045241e-002 1.132687e-002
 1.235423e-002 9.601763e-003 9.080986e-003 9.046299e-003 9.096003e-003
 8.122768e-003 8.022893e-003 7.874446e-003 6.976502e-003 8.834847e-003
<GCONST> -1.881372e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 8.428089e-001
<MEAN> 1 4.669188e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 1.571911e-001
```

```
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 6.997184e-001
<MEAN> 1 -3.630670e-003
<VARIANCE> 1 7.052926e-004
<GCONST> -5.419021e+000
<MIXTURE> 2 3.002816e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 6.997184e-001
<MEAN> 1 -4.951731e-003
<VARIANCE> 1 4.629061e-003
<GCONST> -3.537524e+000
<MIXTURE> 2 3.002816e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 3
<STREAM> 1
<MEAN> 75
 6.281786e+000 3.096737e+000 -6.615632e-001 4.912871e-001 -3.559738e-001
 9.220063e-002 -1.459711e-001 -4.382425e-002 -1.441207e-001 -9.762003e-003
 -1.695344e-001 4.397747e-001 -4.871844e-002 3.973676e-002 -9.157673e-002
 6.591451e-002 -1.792608e-001 -1.485711e-001 -7.817274e-002 -1.818579e-001
 -2.655359e-002 -1.043268e-001 -1.998605e-002 1.162986e-003 -1.193063e-001
 7.568783e-002 -7.331334e-003 -4.709464e-002 -2.763750e-002 -4.295988e-002
 8.887836e-003 1.554982e-003 2.680501e-003 2.268174e-002 -5.102057e-003
 4.384529e-002 2.321889e-002 3.910416e-003 -1.271872e-002 -3.072287e-002
 -1.736694e-002 -1.843275e-002 5.654078e-003 -4.104272e-003 8.192975e-003
 1.144284e-002 4.822708e-003 1.597455e-002 2.322531e-003 2.742308e-004
 -3.098943e-002 -1.113947e-002 1.295125e-002 7.930491e-003 2.182014e-003
 8.509076e-003 -1.414604e-003 6.248754e-003 4.533367e-003 -3.275249e-003
 2.033486e-003 -1.041877e-002 -7.425859e-003 -7.918653e-003 -6.689258e-004
 2.865716e-003 3.017799e-003 1.028238e-002 5.632619e-004 4.514380e-003
 -2.071298e-003 2.283974e-003 2.987702e-004 -4.459871e-003 4.813899e-005
<VARIANCE> 75
 2.124188e-001 1.209822e-001 8.064368e-002 5.425137e-002 1.097980e-001
 4.681705e-002 4.176019e-002 5.022150e-002 4.664669e-002 2.457483e-002
 3.612355e-002 1.844078e-002 2.037552e-002 1.568385e-002 1.664910e-002
 1.784151e-002 1.939059e-002 1.560139e-002 1.116512e-002 1.136923e-002
 9.556616e-003 8.338192e-003 8.851596e-003 7.141263e-003 1.138485e-002
 8.097156e-003 6.066304e-003 5.523978e-003 4.271845e-003 4.483160e-003
 3.776648e-003 3.416347e-003 3.306345e-003 3.238044e-003 2.536093e-003
 2.466198e-003 2.197998e-003 2.256609e-003 2.281971e-003 1.992162e-003
 2.030402e-003 1.981989e-003 2.177109e-003 1.899057e-003 1.797510e-003
 1.479266e-003 1.395683e-003 1.328994e-003 1.320218e-003 1.894296e-003
 6.845179e-002 2.313896e-002 1.890451e-002 1.137860e-002 1.762483e-002
 1.193684e-002 9.997338e-003 1.064313e-002 1.250356e-002 8.145099e-003
 1.538362e-002 1.018085e-002 9.688055e-003 7.393805e-003 8.637632e-003
 9.037647e-003 6.347512e-003 7.399092e-003 6.271173e-003 6.542471e-003
 5.761724e-003 5.128207e-003 5.306771e-003 5.402497e-003 7.099148e-003
<GCONST> -2.179065e+002
<STREAM> 2
```

```
<NUMMIXES> 2
<MIXTURE> 1 9.991808e-001
<MEAN> 1 4.639595e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 8.191915e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 9.973387e-001
<MEAN> 1 -9.893203e-003
<VARIANCE> 1 2.180553e-004
<GCONST> -6.592885e+000
<MIXTURE> 2 2.661339e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.973387e-001
<MEAN> 1 5.940453e-004
<VARIANCE> 1 2.206851e-003
<GCONST> -4.278312e+000
<MIXTURE> 2 2.661339e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 4
<STREAM> 1
<MEAN> 75
 6.482461e+000 3.063078e+000 -8.165830e-001 3.801554e-001 -4.739920e-001
 1.732078e-001 -1.422095e-001 -1.858994e-002 -5.957159e-002 -5.081468e-002
 1.729134e-002 5.310423e-001 -1.259133e-001 -4.246381e-002 -2.090566e-001
 3.378211e-002 -2.252805e-001 -6.861358e-002 -1.001907e-001 -1.433229e-001
 1.993065e-003 -7.083096e-002 5.616352e-002 -4.815598e-002 -1.288016e-001
 -1.122439e-002 3.224408e-003 3.499091e-003 -2.687719e-003 -3.469467e-003
 7.814971e-003 -3.510369e-003 -3.199013e-004 5.575954e-003 -7.175893e-003
 6.307884e-003 1.028461e-003 -3.768743e-003 8.538565e-004 1.343784e-003
 1.108550e-003 -1.858175e-003 3.586431e-003 -3.130197e-003 2.713470e-003
 -2.088020e-003 1.388798e-003 2.394273e-003 -3.158440e-003 3.764831e-003
 -9.616865e-003 6.841691e-003 6.138051e-003 2.320740e-003 4.855923e-003
 -2.539357e-003 1.176451e-004 -1.948304e-003 -4.324961e-003 -5.039772e-004
 -6.912203e-003 -1.625511e-003 3.089170e-003 3.999568e-003 5.765498e-003
 6.556856e-004 1.479604e-003 -3.938728e-003 3.290472e-004 -2.137027e-003
 -1.252447e-003 -1.666797e-003 -1.849851e-003 1.783297e-003 1.574246e-003
<VARIANCE> 75
 1.777479e-001 7.524596e-002 5.067957e-002 3.549240e-002 4.782310e-002
 3.569947e-002 3.060350e-002 3.245657e-002 2.369553e-002 1.552047e-002
 1.789595e-002 1.213513e-002 1.529348e-002 1.158910e-002 9.160016e-003
 1.460143e-002 9.083215e-003 1.486653e-002 9.835752e-003 8.201371e-003
 6.891892e-003 6.493105e-003 6.588398e-003 6.712490e-003 8.545943e-003
 4.089139e-003 3.924599e-003 3.434570e-003 2.662519e-003 2.969735e-003
 2.264690e-003 2.259044e-003 2.263877e-003 1.809653e-003 1.629084e-003
 1.564153e-003 1.289506e-003 1.299015e-003 1.356501e-003 1.151440e-003
 1.054820e-003 1.013830e-003 1.095364e-003 8.854067e-004 8.618239e-004
 8.648942e-004 8.143429e-004 7.885551e-004 6.953860e-004 1.078863e-003
 6.857520e-002 2.028679e-002 1.359851e-002 8.641722e-003 1.281644e-002
 9.962530e-003 8.044823e-003 8.824890e-003 1.107107e-002 6.572527e-003
 1.546959e-002 1.961664e-002 8.836279e-003 5.344774e-003 4.412952e-003
 5.158548e-003 3.903924e-003 4.831396e-003 4.043814e-003 3.993853e-003
 4.244475e-003 4.222135e-003 6.180879e-003 3.667429e-003 4.504606e-003
<GCONST> -2.471483e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 9.994930e-001
<MEAN> 1 4.627734e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 5.070123e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 9.994930e-001
<MEAN> 1 -2.625411e-003
<VARIANCE> 1 4.955040e-005
<GCONST> -8.074643e+000
<MIXTURE> 2 5.070124e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.994930e-001
<MEAN> 1 -2.881667e-004
<VARIANCE> 1 1.505093e-004
<GCONST> -6.963608e+000
<MIXTURE> 2 5.070124e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 5
<STREAM> 1
<MEAN> 75
 6.179216e+000 3.113612e+000 -4.726636e-001 4.954067e-001 -7.489555e-001
 2.648538e-002 -5.512673e-002 8.575257e-002 3.721920e-002 -1.198505e-001
 -4.837276e-002 3.514281e-001 -6.545287e-002 5.353885e-002 -6.301785e-002
 -1.284840e-002 -2.480742e-001 -5.904308e-002 -1.239187e-001 -1.437601e-001
 1.109599e-004 -8.783783e-002 2.328959e-002 -4.323861e-002 -5.336457e-002
 -7.595820e-002 1.882801e-002 5.686376e-002 1.524707e-002 3.887515e-005
 -4.940409e-003 -2.284426e-003 -2.329028e-003 -3.104353e-004 -3.701423e-003
 -1.013594e-002 -1.185046e-002 -3.213048e-003 1.440804e-002 1.748576e-002
 1.011420e-002 1.086945e-002 -2.028380e-003 -3.809888e-003 -7.833997e-003
 -4.493971e-003 -6.529021e-003 -2.238197e-003 -8.680180e-004 7.846222e-003
 -3.094869e-002 -7.343845e-004 1.207586e-002 3.514014e-003 4.921120e-003
 2.261963e-003 -7.747948e-004 -2.202834e-004 -4.053343e-004 -4.178717e-004
 4.681206e-004 7.020264e-004 -5.937886e-004 1.815098e-003 9.880972e-004
 3.671335e-003 4.573561e-003 9.808287e-004 1.159386e-003 -2.317836e-004
 -1.948865e-003 -1.358380e-003 -8.289472e-004 -5.500803e-004 -5.024732e-004
<VARIANCE> 75
 2.899835e-001 1.198180e-001 1.264321e-001 5.748516e-002 1.006726e-001
 8.094563e-002 4.130880e-002 4.631608e-002 2.888552e-002 2.392798e-002
 2.585697e-002 3.531872e-002 1.934628e-002 2.148340e-002 2.002551e-002
 1.807971e-002 1.450088e-002 1.345597e-002 1.408312e-002 1.212750e-002
 1.253752e-002 1.003285e-002 8.306883e-003 8.398900e-003 1.270954e-002
 7.322491e-003 5.232832e-003 4.812048e-003 3.808926e-003 3.633750e-003
 3.514432e-003 3.421105e-003 2.960601e-003 2.386573e-003 2.200675e-003
 2.341905e-003 1.831293e-003 1.774877e-003 1.752257e-003 1.617932e-003
 1.443232e-003 1.446296e-003 1.363399e-003 1.455737e-003 1.317269e-003
 1.278035e-003 1.220283e-003 1.153926e-003 1.007337e-003 1.476598e-003
 7.351016e-002 2.368446e-002 1.968682e-002 1.153251e-002 1.316755e-002
 1.452519e-002 1.171362e-002 1.126522e-002 1.015192e-002 7.501457e-003
 1.280623e-002 1.877295e-002 1.113744e-002 6.937289e-003 7.267938e-003
 6.071156e-003 6.020104e-003 6.285935e-003 5.469806e-003 5.907860e-003
 5.719463e-003 5.065391e-003 6.445678e-003 5.123921e-003 5.762279e-003
<GCONST> -2.214929e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 1 9.990559e-001
<MEAN> 1 4.609253e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 9.441054e-004
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 9.978873e-001
<MEAN> 1 -4.885087e-003
<VARIANCE> 1 7.224264e-005
<GCONST> -7.697603e+000
<MIXTURE> 2 2.112741e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 9.978873e-001
<MEAN> 1 -6.588325e-004
<VARIANCE> 1 2.348379e-004
<GCONST> -6.518738e+000
<MIXTURE> 2 2.112741e-003
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STATE> 6
<STREAM> 1
<MEAN> 75
 5.306767e+000 3.040741e+000 -2.865146e-001 5.282863e-001 -4.527594e-001
 2.807230e-001 -1.667705e-001 2.384920e-002 -3.974725e-002 -1.583529e-001
 8.308741e-003 4.123441e-001 -7.476599e-002 4.920102e-002 -8.604240e-002
 9.850488e-002 -1.635197e-001 -6.674971e-002 -1.131841e-001 -1.045328e-001
 -7.948874e-002 -6.790553e-002 7.373705e-003 -4.900702e-002 -5.109692e-002
 -1.783404e-001 -5.526533e-002 9.145223e-002 2.749050e-002 3.956765e-002
 1.026487e-002 1.024689e-002 2.502814e-002 -1.274629e-002 8.023261e-003
 -9.130953e-003 -2.842181e-002 5.580971e-003 3.008700e-004 1.430185e-002
 3.361402e-004 1.904526e-002 -5.645475e-005 1.327579e-002 7.763710e-003
 -4.682489e-003 4.397902e-003 -1.283915e-002 1.204302e-003 -1.424717e-004
 4.908997e-003 -2.361260e-002 -4.611872e-003 -3.511679e-004 1.134523e-002
 -1.213768e-003 8.051402e-003 9.774146e-003 2.038893e-004 1.169570e-002
 -1.409153e-003 -1.034961e-002 -2.085522e-003 -9.032607e-003 -9.202087e-003
 -9.442679e-003 -3.833493e-003 1.652891e-003 3.426388e-003 3.760859e-003
 6.492001e-003 4.416240e-003 7.613553e-004 1.104933e-003 -3.619201e-003
```

```
<VARIANCE> 75
 5.717232e-001 1.774071e-001 1.803465e-001 4.829086e-002 7.405405e-002
 5.759233e-002 3.037431e-002 5.195399e-002 3.778894e-002 2.829009e-002
 4.597893e-002 2.737632e-002 2.351891e-002 1.675854e-002 1.958008e-002
 1.648557e-002 1.734422e-002 1.302158e-002 1.502864e-002 1.483845e-002
 1.242313e-002 1.235159e-002 9.775540e-003 8.636122e-003 1.143841e-002
 4.338938e-002 3.465805e-002 1.428898e-002 7.307805e-003 7.166008e-003
 5.633261e-003 6.064909e-003 6.021063e-003 4.492021e-003 5.766408e-003
 5.084055e-003 5.246174e-003 4.092483e-003 3.641231e-003 3.374591e-003
 3.239603e-003 2.759330e-003 2.731787e-003 2.768973e-003 2.652083e-003
 2.618236e-003 2.485274e-003 2.227371e-003 2.158346e-003 2.583209e-003
 1.412641e-001 7.041377e-002 3.879442e-002 2.373471e-002 2.229280e-002
 1.918591e-002 2.214457e-002 1.918761e-002 1.616396e-002 1.548674e-002
 2.501478e-002 3.279744e-002 2.037843e-002 1.471376e-002 1.465122e-002
 1.300149e-002 1.011180e-002 1.090671e-002 1.072281e-002 1.100547e-002
 1.084025e-002 1.084338e-002 1.068710e-002 9.303171e-003 1.055525e-002
<GCONST> -1.835005e+002
<STREAM> 2
<NUMMIXES> 2
<MIXTURE> 18.237487e-001
<MEAN> 1 4.501828e+000
<VARIANCE> 1 5.274333e-002
<GCONST> -1.104441e+000
<MIXTURE> 2 1.762513e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 3
<NUMMIXES> 2
<MIXTURE> 1 7.721398e-001
<MEAN> 1 -1.027523e-002
<VARIANCE> 1 5.609470e-004
<GCONST> -5.648007e+000
<MIXTURE> 2 2.278603e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<STREAM> 4
<NUMMIXES> 2
<MIXTURE> 1 7.721398e-001
<MEAN> 1 2.444448e-003
<VARIANCE> 1 3.156502e-003
<GCONST> -3.920413e+000
<MIXTURE> 2 2.278603e-001
<MEAN> 0
<VARIANCE> 0
<GCONST> 0.000000e+000
<TRANSP> 7
 0.000000e+000 1.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 5.771661e-001 4.228339e-001 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 7.892829e-001 2.107171e-001 0.000000e+000
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 8.695285e-001 1.304715e-001
 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 7.570530e-001
 2.429470e-001 0.000000e+000
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 7.294922e-001 2.705078e-001
 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
 0.000000e+000 0.000000e+000
<ENDHMM>
```

C.3 Contextual Information Decision Trees

The following figures show, as example, parts of the decisions trees for the durations, figure C.1, for the fourth state of the logF0 distribution, figure C.2, and for the fourth state of the MFCC distribution, figure C.3.

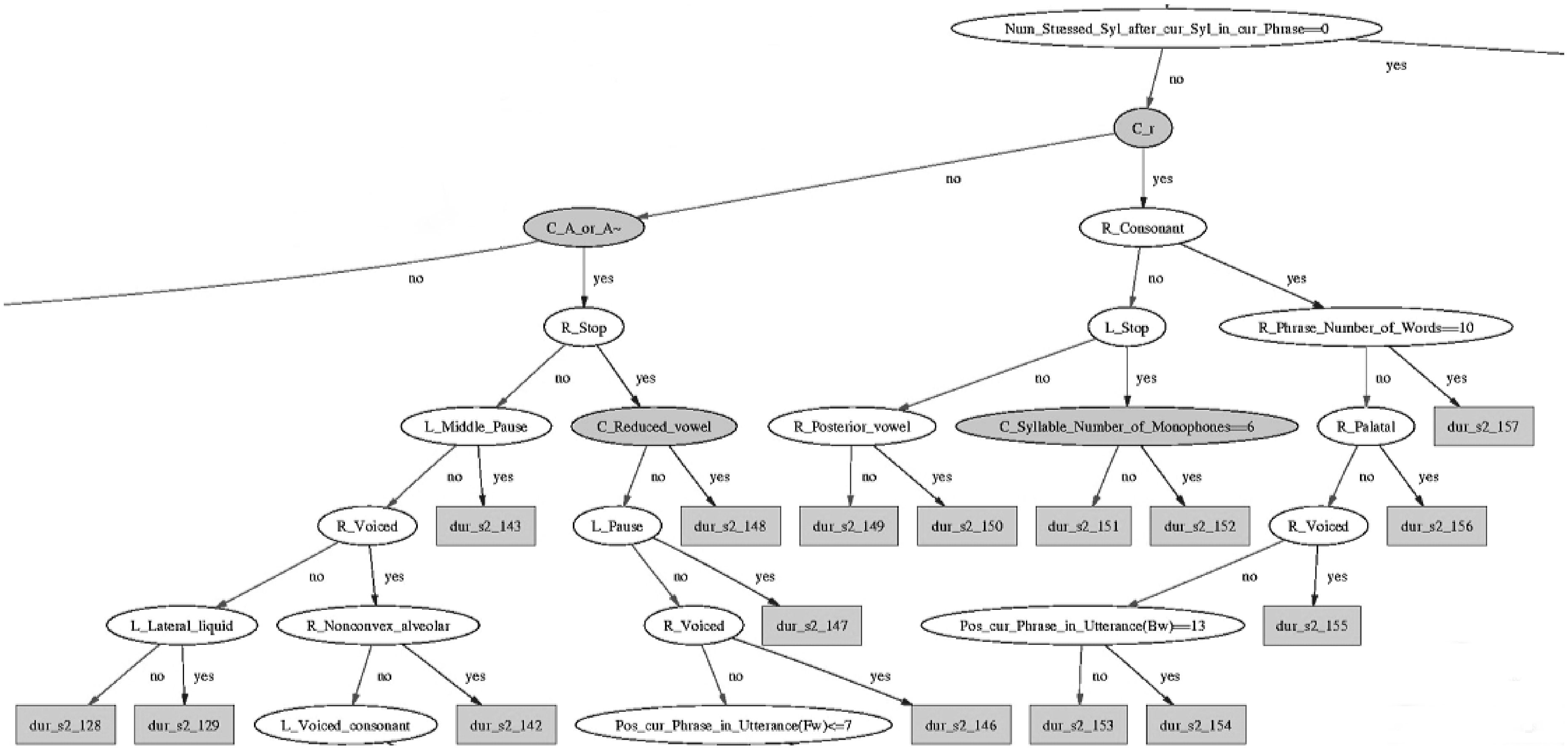


Figure C.1: Example of part of one of the decision trees for the durations.

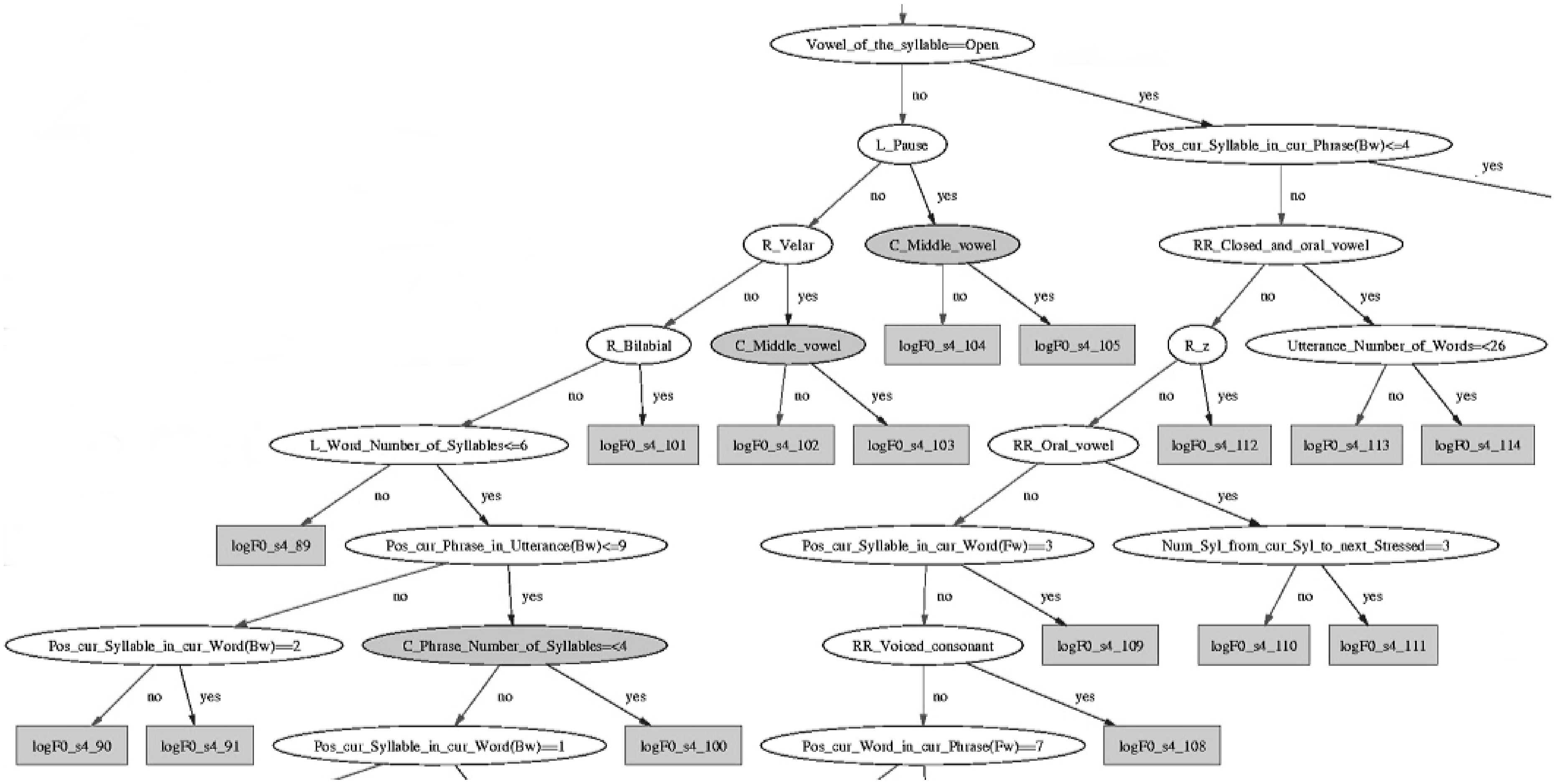


Figure C.2: Example of part of one of the decision trees for the fourth state of the F0 logarithm distribution.

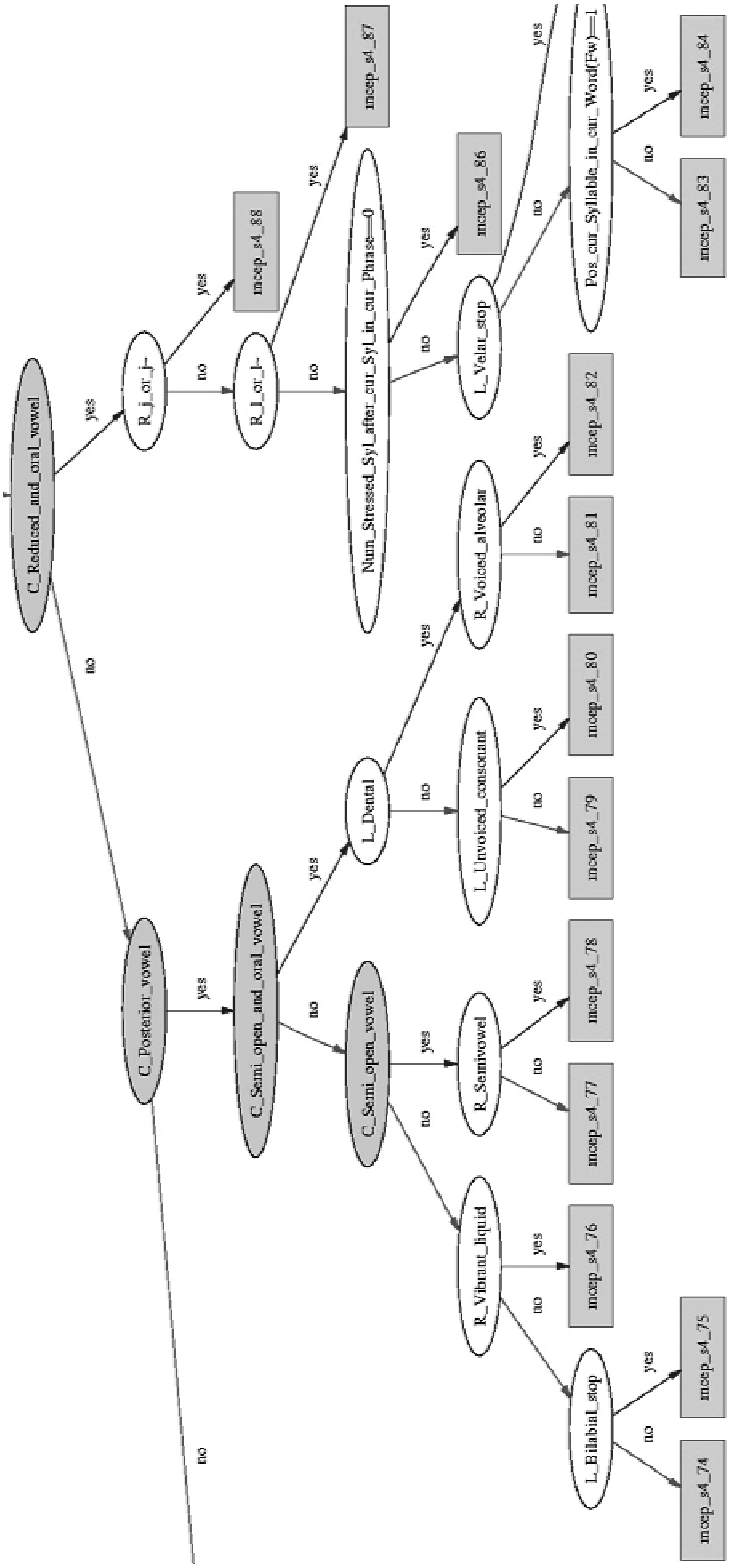


Figure C.3: Example of part of one of the decision trees for the fourth state of the mel-cepstral coefficients distribution.

C.4 Language Context Information Labels File Example

Language Context Information Labels for Input <Olá Maria>

```
y^y-XX+O=l/M2:y_y/S1:y_?y-y_?y+0_?1/S2:y_y/S3:y_y/S4:0_1/S5:0_4/S6:y/W1:#y-#y+#2/W2:y_y/W3:y/P1:y_!y-y_!y+5_!2/P2:y_!y/U:5_$2_&1
y^XX-O+l=a/M2:1_1/S1:y_?y-0_?1+0_?2/S2:1_2/S3:1_5/S4:0_1/S5:0_4/S6:O/W1:#y-#2+#3/W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
XX^O-l+a=m/M2:1_2/S1:0_?1-0_?2+0_?2/S2:2_1/S3:2_4/S4:0_1/S5:0_3/S6:a/W1:#y-#2+#3/W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
O^l-a+m=6/M2:2_1/S1:0_?1-0_?2+0_?2/S2:2_1/S3:2_4/S4:0_1/S5:0_3/S6:a/W1:#y-#2+#3/W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
l^a-m+6=r/M2:1_2/S1:0_?2-0_?2+1_?2/S2:1_3/S3:3_3/S4:0_1/S5:0_2/S6:6/W1:#2-#3+#y/W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
a^m-6+r=i/M2:2_1/S1:0_?2-0_?2+1_?2/S2:1_3/S3:3_3/S4:0_1/S5:0_2/S6:6/W1:#2-#3+#y/W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
m^6-r+i=6/M2:1_2/S1:0_?2-1_?2+0_?1/S2:2_2/S3:4_2/S4:0_0/S5:0_0/S6:i/W1:#2-#3+#y/W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
6^r-i+6=y/M2:2_1/S1:0_?2-1_?2+0_?1/S2:2_2/S3:4_2/S4:0_0/S5:0_0/S6:i/W1:#2-#3+#y/W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
r^i-6+y=y/M2:1_1/S1:1_?2-0_?1+y_?y/S2:3_1/S3:5_1/S4:1_0/S5:2_0/S6:6/W1:#2-#3+#y/W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
```

C.5 Example of a File Resulting from the Synthesis Engine

This section shows an example of a file resulting from the HTS engine, the “trace” file. The HTS engine used was version 2.0.

```
sampling frequency                     -> 16000(Hz)
frame period                           -> 112(point) 7.00(msec)
use state alignment for duration       -> 0
use phoneme alignment for duration     -> 0
all-pass constant                      -> 0.420000
postfiltering coefficient              -> 0.400000
control duration parameter             -> 0.000000
multiply f0                           -> 1.000000
add f0                                 -> 0.000000
voiced/unvoiced threshold              -> 0.500000
specified utterance length             -> 0.000000(sec.)

number of HMMs        -> 9
number of HMM states  -> 45
length of this speech -> 1.442 sec. (206 frames)

1:
y^y-XX+O=l/M2:y_y/S1:y_?y-y_?y+0_?1/S2:y_y/S3:y_y/S4:0_1/S5:0_4/S6:y/W1:#y-#y+#2/
W2:y_y/W3:y/P1:y_!y-y_!y+5_!2/P2:y_!y/U:5_$2_&1
            duration -> 130
  2-state : spectrum -> 80    f0 -> 137    0.000--0.049(sec)   7(frame)   unvoiced
  3-state : spectrum -> 13    f0 -> 117    0.049--0.210(sec)  23(frame)   unvoiced
  4-state : spectrum -> 53    f0 -> 4      0.210--0.455(sec)  35(frame)   unvoiced
  5-state : spectrum -> 47    f0 -> 51     0.455--0.567(sec)  16(frame)   unvoiced
  6-state : spectrum -> 95    f0 -> 16     0.567--0.637(sec)  10(frame)   unvoiced
2:
y^XX-O+l=a/M2:1_1/S1:y_?y-0_?1+0_?2/S2:1_2/S3:1_5/S4:0_1/S5:0_4/S6:O/W1:#y-#2+#3/
W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 29
  2-state : spectrum -> 84    f0 -> 122    0.637--0.658(sec)   3(frame)   unvoiced
  3-state : spectrum -> 91    f0 -> 10     0.658--0.679(sec)   3(frame)   voiced
  4-state : spectrum -> 59    f0 -> 91     0.679--0.714(sec)   5(frame)   voiced
  5-state : spectrum -> 68    f0 -> 34     0.714--0.735(sec)   3(frame)   voiced
  6-state : spectrum -> 105   f0 -> 72     0.735--0.763(sec)   4(frame)   voiced
3:
XX^O-l+a=m/M2:1_2/S1:0_?1-0_?2+0_?2/S2:2_1/S3:2_4/S4:0_1/S5:0_3/S6:a/W1:#y-#2+#3/
W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 149
  2-state : spectrum -> 23    f0 -> 72     0.763--0.777(sec)   2(frame)   voiced
  3-state : spectrum -> 15    f0 -> 88     0.777--0.812(sec)   5(frame)   voiced
  4-state : spectrum -> 26    f0 -> 120    0.812--0.840(sec)   4(frame)   voiced
  5-state : spectrum -> 11    f0 -> 15     0.840--0.854(sec)   2(frame)   voiced
  6-state : spectrum -> 18    f0 -> 90     0.854--0.868(sec)   2(frame)   voiced
4:
O^l-a+m=6/M2:2_1/S1:0_?1-0_?2+0_?2/S2:2_1/S3:2_4/S4:0_1/S5:0_3/S6:a/W1:#y-#2+#3/
W2:1_2/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 228
  2-state : spectrum -> 5     f0 -> 103    0.868--0.882(sec)   2(frame)   voiced
  3-state : spectrum -> 68    f0 -> 54     0.882--0.903(sec)   3(frame)   voiced
  4-state : spectrum -> 63    f0 -> 130    0.903--0.931(sec)   4(frame)   voiced
  5-state : spectrum -> 58    f0 -> 68     0.931--0.959(sec)   4(frame)   voiced
  6-state : spectrum -> 72    f0 -> 61     0.959--0.987(sec)   4(frame)   voiced
5:
l^a-m+6=r/M2:1_2/S1:0_?2-0_?2+1_?2/S2:1_3/S3:3_3/S4:0_1/S5:0_2/S6:6/W1:#2-#3+#y/
W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 183
  2-state : spectrum -> 83    f0 -> 89     0.987--1.001(sec)   2(frame)   voiced
  3-state : spectrum -> 98    f0 -> 83     1.001--1.022(sec)   3(frame)   voiced
  4-state : spectrum -> 80    f0 -> 90     1.022--1.050(sec)   4(frame)   voiced
  5-state : spectrum -> 71    f0 -> 54     1.050--1.057(sec)   1(frame)   voiced
  6-state : spectrum -> 26    f0 -> 60     1.057--1.071(sec)   2(frame)   voiced
6:
a^m-6+r=i/M2:2_1/S1:0_?2-0_?2+1_?2/S2:1_3/S3:3_3/S4:0_1/S5:0_2/S6:6/W1:#2-#3+#y/
W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 24
  2-state : spectrum -> 12    f0 -> 9      1.071--1.085(sec)   2(frame)   voiced
  3-state : spectrum -> 10    f0 -> 14     1.085--1.099(sec)   2(frame)   voiced
  4-state : spectrum -> 17    f0 -> 84     1.099--1.127(sec)   4(frame)   voiced
  5-state : spectrum -> 50    f0 -> 10     1.127--1.141(sec)   2(frame)   voiced
  6-state : spectrum -> 52    f0 -> 80     1.141--1.148(sec)   1(frame)   voiced
7:
m^6-r+i=6/M2:1_2/S1:0_?2-1_?2+0_?1/S2:2_2/S3:4_2/S4:0_0/S5:0_0/S6:i/W1:#2-#3+#y/
W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 54
  2-state : spectrum -> 60    f0 -> 73     1.148--1.162(sec)   2(frame)   voiced
  3-state : spectrum -> 71    f0 -> 136    1.162--1.169(sec)   1(frame)   voiced
  4-state : spectrum -> 113   f0 -> 144    1.169--1.176(sec)   1(frame)   voiced
  5-state : spectrum -> 65    f0 -> 46     1.176--1.183(sec)   1(frame)   voiced
  6-state : spectrum -> 15    f0 -> 34     1.183--1.197(sec)   2(frame)   unvoiced
8:
6^r-i+6=y/M2:2_1/S1:0_?2-1_?2+0_?1/S2:2_2/S3:4_2/S4:0_0/S5:0_0/S6:i/W1:#2-#3+#y/
W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 245
  2-state : spectrum -> 110   f0 -> 139    1.197--1.211(sec)   2(frame)   voiced
  3-state : spectrum -> 63    f0 -> 42     1.211--1.232(sec)   3(frame)   voiced
  4-state : spectrum -> 83    f0 -> 20     1.232--1.246(sec)   2(frame)   voiced
  5-state : spectrum -> 8     f0 -> 79     1.246--1.295(sec)   7(frame)   voiced
  6-state : spectrum -> 5     f0 -> 49     1.295--1.337(sec)   6(frame)   voiced
9:
r^i-6+y=y/M2:1_1/S1:1_?2-0_?1+y_?y/S2:3_1/S3:5_1/S4:1_0/S5:2_0/S6:6/W1:#2-#3+#y/
W2:2_1/W3:0/P1:y_!y-5_!2+y_!y/P2:1_1/U:5_$2_&1
            duration -> 232
  2-state : spectrum -> 107   f0 -> 158    1.337--1.351(sec)   2(frame)   voiced
  3-state : spectrum -> 49    f0 -> 13     1.351--1.358(sec)   1(frame)   voiced
  4-state : spectrum -> 48    f0 -> 134    1.358--1.365(sec)   1(frame)   voiced
  5-state : spectrum -> 7     f0 -> 14     1.365--1.407(sec)   6(frame)   voiced
  6-state : spectrum -> 65    f0 -> 85     1.407--1.442(sec)   5(frame)   unvoiced
```

Appendix D

European Portuguese Phonemes and Corresponding Graphemes

Table D.1: List of European Portuguese phonemes and corresponding graphemes.

Phonemes	**Corresponding graphemes**
S	<**ch**>, <**s**(p,t,q,c,f)>, <**s** (p,t,q,c,f)>, <**z** (p,t,q,c,f)>, < **x**>, <n**x**>, <e**x**(p,t,q,c,f)>, <e**x-**(p,t,q,c,f)>, <**s**(.,!,?)>, <**z**(.,!,?)>.
e∼	<**e(m,n)**(b,c,d,f,g,l,p,q,r,s,t,v,x,z)>, <**ê(m,n)**(b,c,d,f,g,l,p,q,r,s,t,v,x,z)>.
a	<**á**>, <**à**>, <**a**l(b,c,d,f,g,l,p,q,r,s,t,v,x,z)>, <**a**(i,o,u)>, <**a**(cc,cç,ct,pt)>, <**a**r >, and some other situations of <**a**> between consonants.
d	<**d**>.
E	<**é**>, <**e**(cc,cç,ct,gn,pç,pt)>, <**e**(l,n,r) >, <a**e**>, <**e**l(h,b,c,d,f,g,l,p,q,r,s,t,v,x,z)>, and other situations of <**e**> at the beginning of a word and between consonants.
j	<**e**(a,o)>, <**i**(a,o)>, <(a,o)**i**>, and other situations of <**(e,i)**>.
Z	<**j**>, <**g**(e,i)>, <**s**(b,d,g,m,n,z,v,l,r)>.

Continues on next page

Table D.1 – continued from previous page

Phonemes	**Corresponding graphemes**
j~	<(ã,õ)**e**>, <**(e,é,ê)m**>, <**en**s>.
u	<**u**>, <**o**>.
k	<**c**>, <**q**> <**qu**(e,i)>.
g	<**g**>, <**gu**(e,i)>.
t	<**t**>.
e	<**ê**> and many situations of <**e**>.
J	<**nh**>.
u~	<**o(n,m)**>, <**u(n,m)**>, <**u**(a,e,i)(**n,m**)>.
v	<**v**>.
s	< **s**>, <**ss**>, <**ç**>, <**c**(e,i)>, <(b,c,d,f,g,j,l,m,n,p,q,r,t,v,x,z)**s**>, <**x** > and <**x**> at the middle of a word in several situations.
b	<**b**>.
i~	<**i(m,n)**(b,c,d,f,g,j,l,p,q,r,s,t,v,x,z)>, <**im** >, <**í(m,n)**(b,c,d,f,g,j,l,p,q,r,s,t,v,x,z)>, <**ím** >.
z	<**z**>, <(a,e,i,o,u)**s**(a,e,i,o,u)>, <**s** (a,e,i,o,u)>.
w	<**u**(a,e,i)> <(a,e,i)**u**>.
r	<(a,e,i,o,u)**r**>, <(b,c,d,f,g,j,l,m,n,p,q,s,t,v,x,z)**r**>.
l~	<**l**(b,c,d,f,g,j,m,n,p,q,r,s,t,v,x,z)>, <**l** >.
w~	<ã**o**>, <(a,e,i)**u(n,m)**>, <**u**(a,e,i) **(n,m)**>.
@	<**e**> in many situations at the end of word and between consonants.
L	<**lh**>.
f	<**f**>.
i	<**i**>, <**í**>, < **e** >, and <**e**> in some other situations.
6	<**a**> in many situations.
n	<**n**>.

Continues on next page

Table D.1 – continued from previous page

Phonemes	**Corresponding graphemes**
O	<**ó**>, and <**o**> in some situations.
m	<**m**>.
o~	<**õ**>, and <**o**> in some situations.
6~	<**ã**>, <**a(n,m)**>.
l	<**l**><(a,e,i,o,u)>.
p	<**p**>.
R	< **r**>, <**rr**>.
o	<**ô**>, <**ou**> and <**o**> in some situations.
ks	<**x**> at the middle of a word in several situations.

Appendix E

Speech Corpus for Text-to-Speech Systems

SS

Vais chatear-nos com as tuas bochechinhas ou vais chamá-los.

vajS S6tjar nuS ko∼ 6S tu6Z buS@SiJ6z o vajS S6ma luS

vajS S6tjar nuS ko∼ 6S tu6z bwSSiJ6z o vajS S6ma luS

e∼S

Encher o copo aos que vêem chegar a procissão ainda se faz enchendo a garrafa aos que dêem cheques.

e∼Ser u kOpu awS k@ ve∼6∼j∼ S@gar 6 prusis6∼w∼ 6∼i∼d6 s@ faz e∼Se∼du 6 g6Raf6 awS k@ de∼6∼j∼ SEk@S

e∼Ser u kOpw awS k ve∼6∼j∼ Sgar 6 prwsis6∼w∼ 6∼i∼d6 s faz e∼Se∼dw 6 g6Raf6 awS k de∼6∼j∼ SEkS

ae∼

Há entrefalas e não há entraves.

a e∼tr@fal6z i n6∼w∼ a e∼trav@S

a e∼trfal6z i n6∼w∼ a e∼travS

de∼

Dentro e depois de entrefalas e de entreténs entre dentes dos presidentes foi evidente a identidade dos independentes.

de∼tru i d@pojZ d@ e∼tr@fal6z i d@ e∼tr@t6∼j∼z e∼tr@ de∼t@Z duS pr@zide∼t@S foj ivide∼t@ 6 ide∼tidad@ duz i∼d@pe∼de∼t@S

de∼tru i dpojZ d e∼trfal6z i d e∼trt6∼j∼z e∼tr de∼tZ duS przide∼tS foj ivide∼t 6 ide∼tidad duz i∼dpe∼de∼tS

aa

Há áreas já áridas onde está água e dá álcool que fará árvores crescerem e terá árabes quando lá há.

a arj6Z Za arid6z o∼d@ @Sta agw6 i da al∼kwOl∼ k@ f6ra arvwr@S kr@Sser6∼j∼ i t@ra ar6b@S kw∼6∼du la a

a arj6Z Za arid6z o∼d Sta agw6 i da al∼kuOl∼ k f6ra arvwrS krSser6∼j∼ i t@ra ar6bS kw~6~dw la a

Ea

É à conta e até às vezes ando a pé alguns quilómetros para beber café águia.

E a ko~t6 i 6tE aZ vez@z 6~du 6 pE al~gu~S kilOm@truS p6r6 b@ber k6fE agi6

E a ko~t6 i 6tE aZ vezz 6~d 6 pE al~gu~S kilOmtrwS p6r6 bber k6fE agi6

dd

Dedicou-se ao vendedor mais dedicado de dedicação extrema e daí se deduz.

d@diko s@ aw ve~d@dor majZ d@dikadu d@ d@dik6s6~w~ 6jStrem6 i d6i s@ d@duS

ddiko s aw ve~ddor majZ ddikadw d ddik6s6~w~ 6jStrem6 i d6i s ddwS

j~d

Indicar indústrias independentes, ainda que assim descobrindo indirectamente que o indicador vai subindo.

i~dikar i~duStri6z i~d@pe~de~t@z 6~i~d6 k@ 6si~ d@Skubri~du i~dirEt6me~t@ k@ u i~dik6dor vaj subi~du

i~dikar i~duStri6z i~dpe~de~tz 6~i~d6 k 6si~ dSkubri~dw i~dirEt6me~t k u i~dik6dor vaj subi~dw

aE

Há épocas em que não se dava nenhuma importância ao a-e-i-o-u nem tão pouco à ética.

a Epuk6z 6~j~ k@ n6~w~ s@ dav6 n@Jum6 i~purt6~si6 aw a E i O u n6~j~ t6~w~ poku a Etik6

a Epukaz 6~j~ k n6~w~ s dav6 n@Jum6 i~pwrt6~sj6 aw a E i O u n6~j~ t6~w~ pokw a Etik6

JE

Conhece as regras mas reconhece que para que ganhe é preciso começar quando amanhece.

kuJEs@ 6Z REgr6Z m6Z R@kuJEs@ k@ p6r6 k@ g6J@ E pr@sizu kum@sar kw~6~du 6m6JEs@

kwJEs 6Z REgr6Z m6Z RkwJEs k p6r6 k g6J E prsizw kumsar kw~6~dw 6m6JEs

Sj

Chita e chimpanzé chineses vinham em caixinhas baixinhas e mais coisas em que mexi.

Sit6 i Si~p6~zE Sinez@Z viJ6~w~ 6~j~ kajSiJ6Z bajSiJ6z i majS kojz6z 6~j~ k@ m@Si

Sit6 i Si~p6~zE SinezZ viJ6~w~ 6~j~ kajSiJ6Z bajSiJ6z i majS kojz6z 6~j~ k mSi

l~j

Sal e azeite mal irradie calor real e tudo isso.

sal i 6z6jt@ mal iR6di@ k6lor Rjal i tudu isu

sal i 6z6jt mal iR6dj@ k6lor Rjal i tudu isw

uZ

Ruge o leão cujo pêlo está sujo pelo refúgio de fugir do gel na penugem.

RuZ@ u lj~6~w~ kuZu pelu @Sta suZu pelu R@fuZiu d@ fuZir du ZEl~ n6 p@nuZ6~j~

RuZ u li~6~w~ kuZw pelu Sta suZw pelu RfuZiw d fuZir du ZEl~ n6 pnuZ6~j~

gZ

Se consegue gente a praguejar para que se pague jantares, então entregue já isso hoje.

s@ ko~sEg@ Ze~t@ 6 pr6g@Zar p6r6 k@ s@ pag@ Z6~tar@z e~t6~w~ e~trEg@ Za isu oZ@

s ko~sEg Ze~t 6 pr6gZar p6r6 k s pag Z6~tar@z e~t6~w~ e~trEg Za isw oZ

u~j~

Muitos anos passaram sobre muita coisa sem nenhum interesse que muito marcou épocas e minou muitas outras.

mu~j~tuz 6nuS p6sar6~w~ sobr@ mu~j~t6 kojz6 s6~j~ n@Ju~ i~t@res@ k@ mu~j~tu m6rko Epuk6z i mino mu~j~t6z otr6S

mu~j~twz 6nuS p6sar6~w~ sobr mu~j~t6 kojz6 k mu~j~tw m6rko Epukaz i mino mu~j~t6S

o~j~

Põe milhões nas eleições para lidar com informações em condições junto às populações e associações.

po~j~ miLo~j~Z n6z il6jso~j~S p6r6 lidar ko~ i~furm6so~j~z 6~j~ ko~diso~j~Z Zu~tu aS pupul6so~j~z i 6susi6so~j~S

po~j~ miLo~j~Z n6z jl6jso~j~S p6r6 ter i~fwrm6so~j~z 6~j~ ko~diso~j~Z Zu~tw aS pwpwl6so~j~z i 6swsj6so~j~S

au

Baú com tesouros não é mau para baú.

bau ko~ t@zoruZ n6~w~ E maw p6r6 bau

bau ko~ tzorwZ n6~w~ E maw p6r6 bau

su

Sucesso vem para sua surpresa do processo de consumo de consultas e tudo isso.

susEsu v6~j~ p6r6 su6 surprez6 du prusEsu d@ ko~sumu d@ ko~sul~t6z i tudu isu

susEsw v6~j~ p6r6 su6 swrprez6 du prusEsw d ko~swmw d ko~swl~t6z i tudu isw

ak

Há que combater o acne, mas dá que fazer.

a k@ ko~b6ter u akn@ m6Z da k@ f6zer

a k ko~b6ter u akn m6Z da k f6zer

bk

Bebe sem pensar na rabecada porque sabe quanto recebe cada.

bEb@ s6~j~ pe~sar n6 R6b@kad6 pwrk@ sab@ kw~6~tu R@sEb@ k6d6

bEb@ s6~j~ pe~sar n6 R6bkad6 pwrk sab kw~6~tw R@sEb k6d6

ag

Vaga que não for paga dá grande flagra a quem divaga ao pé do lago em Praga.

vag6 k@ n6~w~ for pag6 da gr6~d@ flagr6 6 k6~j~ divag6 aw pE du lagu 6~j~ prag6

vag6 k n6~w~ for pag6 da gr6~d flagr6 6 k6~j~ divag6 aw pE du lagw 6~j~ prag6

Og

Fotógrafos só garantem por dó globalmente sobre pó gorduroso, mas não as fotógrafas.

futOgr6fuS sO g6r6~t6~j~ pur dO glubal~me~t@ sobr@ pO gurdurozu m6Z n6~w~ 6S futOgr6f6S

fwtOgr6fwS sO g6r6~t6~j~ pur dO glwbal~me~t sobr pO gwrdwrozw m6Z n6~w~ 6S fwtOgr6f6S

et

Preto é a cor escolhida para o gabinete, embora a tinta preta não se vê tanto como mesmo preto.

pretu E 6 kor @SkuLid6 p6r6 u g6binet@ e~bOr6 6 ti~t6 pret6 n6~w~ s@ ve t6~tu komu meZmu pretu

pretw E 6 kor SkwLid6 p6r6 u g6binet e~bOr6 6 ti~t6 pret6 n6~w~ s ve t6~tw komw meZmu pretw

bt

Betão sai da betoneira onde cabe tanto betão quanto sabe tentar o zombeteiro.

b@t6~w~ saj d6 b@tun6jr6 o~d@ kab@ t6~tu b@t6~w~ kw~6~tu sab@ te~tar u zo~b@t6jru

bt6~w~ saj d6 btwn6jr6 o~d kab t6~tw bt6~w~ kw~6~tw sab te~tar u zo~bt6jrw

ae

Vá ele saber o que dá este.

va el@ s6ber u k@ da eSt@

va el s6ber u k da eSt

se

Sê homem e levanta o teu cerdo porque se ele morrer é mais um ser que morre dos seus.

se Om6~j~ i l@v6~t6 u tew sErdu purk@ s@ el@ muRer E majz u~ ser k@ mOR@ duS sewS

se Om6~j~ i lv6~t6 u tew serdw purk s el muRer E majz u~ ser k mOR duS sewS

iJ

Tinha um amigo que vinha buscar vinho aqui às minhas vinhas mas não saía das linhas que tinham no caminho.

tiJ6 u~ 6migu k@ viJ6 buSkar viJu 6ki aZ miJ6Z viJ6Z m6Z n6~w~ s6i6 d6Z liJ6S k@ tiJ6~w~ nu k6miJu

tiJ6 u~ 6migw k viJ6 bwSkar viJu 6ki aZ miJ6Z viJ6Z m6Z n6~w~ s6i6 d6Z liJ6S k tiJ6~w~ nu k6miJu

oJ

Suponho que ponho conhaque e sonho que imponham um menos tristonho e mais risonho.

supoJu k@ poJu kOJak@ i soJu k@ i~poJ6~w~ u~ menuS triStoJu i majZ RizoJu

supoJw k poJw kOJak i soJw k i~poJ6~w~ u~ menwS trjStoJu i majZ RizoJw

Zu~

Juntos vamos em conjunto ter com o adjunto da junção que hoje untara o ajuntamento da junta.

Zu~tuZ v6muz 6~j~ ko~Zu~tu ter ko~ u 6dZu~tu d6 Zu~s6~w~ k@ oZ@ u~tar6 u 6Zu~t6me~tu d6 Zu~t6

Zu~twZ v6mwz 6~j~ ko~Zu~tw ter ko~ u 6dZu~tw d6 Zu~s6~w~ k oZ u~tar6 u 6Zu~t6me~tw d6 Zu~t6

fu~

Funciona às mil maravilhas no que é fundamental, mas lá no fundo há funções que não funcionam.

fu~sion6 aZ mil~ m6r6viL6Z nu k@ E fu~d6me~tal~ m6Z la nu fu~du a fu~so~j~S k@ n6~w~ fu~sion6~w~

fu~sjOn6 aZ mil~ m6r6viL6Z nu k E fu~d6me~tal~ m6Z la nu fu~dw a fu~so~j~S k n6~w~ fu~sjOn6~w~

Ov

Nova prova dos ovos que só vale para jovens não aprova quem a promove.

nOv6 prOv6 duz OvuS k@ sO val@ p6r6 ZOv6~j~Z n6~w~ 6prOv6 k6~j~ 6 prumOv@

nOv6 prOv6 duz OvwS k sO val p6r6 ZOv6~j~Z n6~w~ 6prOv6 k6~j~ 6 prwmOv

lv

Fui levar o carro à oficina e a televisão a levantar o som que é relevante para aquele valor elevado.

fui l@var u kaRu a Ofisin6 i 6 t@l@viz6~w~ 6 l@v6~tar u so~ k@ E R@l@v6~t@ p6r6 6kel@ v6lor il@vadu

fuj lvar u kaRw a Ofisin6 i 6 tl@viz6~w~ 6 lv6~tar u so~ k E Rl@v6~t p6r6 6kel v6lor jl@vadw

as

Passa o passe para a classe económica porque há sempre passaportes que a massa critica ultrapassa.

pas6 u pas@ p6r6 6 klas@ ikunOmik6 purk@ a se~pr@ p6s6pOrt@S k@ 6 mas6 kritik6 ul~tr6pas6

pas6 u pas p6r6 6 klas ikwnOmik6 purk a se~pr pas6pOrtS k 6 mas6 kritjk6 ul~tr6pas6

ss

Assessoria seria se o assessor se se ouvisse na sessão ou sessões da assessora.

6s@suri6 s@ri6 s@ u 6s@sor s@ s@ ovis@ n6 s@s6~w~ o s@so~j~Z d6 6s@sor6

6sswri6 s@ri6 s u 6ssor s s ovis n6 ss6~w~ o sso~j~Z d6 6ssor6

ub

Outubro viu o clube subir na publicidade do boneco sem substituir o tubo.

otubru viw u klub@ subir n6 publisidad@ du bunEku s6~j~ subStituir u tubu

otubrw viw u klub subir n6 publisidad du bwnEkw s6~j~ subStitwir u tubw

nb

Nebulosa neblina sobre um telefone bom que reúne boas características cor tenebrosa e fúnebre.

n@bulOz6 n@blin6 sobr@ u~ t@l@fOn@ bo~ k@ R@wn@ bo6S k6r6t@riStik6S kor t@n@brOz6 i fun@br@

nbwlOz6 nblin6 sobr u~ tl@fOn@ bo~ k Rjun@ bo6S k6r6ktriStik6S kor tnbrOz6 i fun@br bi~

Subindo ao bingo encontra o bimbo que sabe implorar.

subi~du aw bi~gu e~ko~tr6 u bi~bu k@ sab@ i~plurar

subi~dw aw bi~gw e~ko~tr6 u bi~bw k sab i~plwrar

mi~

Dormindo ao domingo famintos fomos assumindo que me interessava e exprimindo o regime imposto por mim.

durmi~du aw dumi~gu f6mi~tuS fomuz 6sumi~du k@ m@ i~t@r@sav6 i 6jSprimi~du u R@Zim@ i~poStu pur mi~

dwrmi~dw aw dumi~gw f6mi~tuS fomwz 6sumi~dw k m i~trsav6 i 6jSprimi~dw u R@Zim i~poStw pur mi~

sz

Cesarianas dos diocesanos e Cesarianos do cesarismo fazem-se zelosamente pelas diocesanas.

s@z6ri6n6Z duZ dius@z6nuz i s@z6ri6nuZ du s@z6riZmu faz6~j~ s@ z@lOz6me~t@ pel6Z dius@z6n6S

sz6ri6n6Z dwZ disz6nuz i sz6ri6nuZ du sz6riZmw faz6~j~ s zlOz6me~t pel6Z disz6n6S

Rz

Rezinga com os resultados de quem varre zonas de resenha e resolução.

R@zi~g6 ko~ uZ R@zul~taduZ d@ k6~j~ vaR@ zon6Z d@ R@z6J6 i R@zulus6~w~

Rzi~g6 ko~ uZ Rzul~tadwZ d k6~j~ vaR zon6Z d R@z6J6 i Rzwlws6~w~

gw

O governo governa pela igualdade, digo.

u guvernu guvErn6 pel6 igwal~dad@ digu

u gwvernw gwvErn6 pel6 igwal~dad digw

ew

Eu sou a favor do seu direito pela eutanásia.

ew so 6 f6vor du sew dir6jtu pel6 ewt6nazi6

ew so 6 f6vor du sew dir6jtw pel6 ewt6nazi6

kr

Criada na sua querida construção crucial, acreditou no lucro da democracia.

kriad6 n6 su6 k@rid6 ko~Strus6~w~ krusial 6kr@dito nu lukru d6 d@mukr6si6

kriad6 n6 su6 k@rid6 ko~Strus6~w~ krwsjal 6krdito nu lukrw d6 dmwkr6si6

ir

Afirmo ao director que tenho de sair para ir abrir o que irá decidir quem deve desistir.

6firmu aw dirEtor k@ t6Ju d@ s6jr p6r6 ir 6brir u k@ ira d@sidir k6~j~ dEv@ d@ziStir

6firmu aw dirEtor k t6Ju d s6ir p6r6 ir 6brir u k ira dsidir k6~j~ dEv dziStir

ul~

Último ultraje ultrapassa todas as ofensivas entre norte e sul.

ul~timu ul~traZ@ ul~tr6pas6 tod6z 6z Ofe~siv6z e~tr@ nOrt@ i sul~

ul~timu ul~traZ ul~tr6pas6 tod6z 6z Ofe~siv6z e~tr nOrt i sul~

il~

Mil crianças de perfil é sempre útil para um filme civil mesmo em Abril.

mil~ kri~6~s6Z d@ p@rfil E se~pr@ util~ p6r6 u~ fil~m@ sivil~ meZmu 6~j~ 6bril~

mil~ kri~6~s6Z d prfil E se~pr util~ p6r6 u~ fil~m sivil~ meZmu 6~j~ 6bril~

bw~

Abundância numa terra abundante em abundâncias.

6bu~d6~si6 num6 tER6 6bu~d6~t@ 6~j~ 6bu~d6~si6S

6bu~d6~sj6 num6 tER6 6bw~d6~t 6~j~ 6bu~d6~sj6S

rw~

Tem o trunfo truncado na manga, porque foi trunfado pelo outro jogador.

t6~j~ u tru~fu tru~kadu n6 m6~g6 purk@ foj tru~fadu pelu otru Zug6dor

t6~j~ u tru~fw trw~kadw n6 m6~g6 purk foj tru~fadw pelu otru Zwg6dor

v@

Vive suavemente a verdade, sem venerar a mentira a vergonha vive.

viv@ suav@me~t@ 6 v@rdad@ s6~j~ v@n@rar 6 me~tir6 6 v@rgoJ6 viv@

viv suavme~t 6 v@rdad s6~j~ vn@rar 6 me~tir6 6 v@rgoJ6 viv

n@

Negociações ao telefone são necsessárias para as toneladas de benefícios monetários sem nenhuma higiene.

n@gusi6so~j~z aw t@l@fOn@ s6~w~ n@s@sari6S p6r6 6S tun@lad6Z d@ b@n@fisiuZ mun@tariuS s6~j~ n@Jum6 eZiEn@

n@gwsj6so~j~z aw tl@fOn@ s6~w~ nssari6S p6r6 6S twnlad6Z d bnfisiwZ mun@tariwS s6~j~ n@Jum6 iZiEn

aL

Alho dá tanto trabalho a descascar que merecia uma medalha pelo detalhe que lá lhe espalha a toalha.

aLu da t6~tu tr6baLu 6 d@k6Skar k@ m@r@si6 um6 m@daL6 pelu d@taL@ k@ la L@ @SpaL6 6 tuaL6

aLw da t6~tw tr6baLw 6 dSk6Skar k m@rsj6 um6 mdaL6 pelu dtaL k la L Sp6L6 6 twaL6

l~L

Mal lhe digo que tal lhe faz mal fica fácil lho levar.

mal~ L@ digu k@ tal~ L@ faZ mal~ fik6 fasil~ Lu l@var

mal~ L digw k tal~ L faZ mal~ fik6 fasil~ Lw lvar

Sf

Chefiar o desfile significa um esforço de transformar em satisfação casos fracos.

S@fiar u d@Sfil@ signifik6 u~ @Sforsu d@ tr6~Sfurmar 6~j~ s6tiSf6s6~w~ kazuS frakuS

Sfiar u dSfil@ signifik6 u~ Sforsw d tr6~Sfwrmar 6~j~ s6tjSf6s6~w~ kazuS frakwS

u~f

Triunfou por triunfar um forte trunfo, mas comportou-se como se triunfasse só pelo triunfo.

tri~u~fo pur tri~u~far u~ fOrt@ tru~fu m6S ko~purto s@ komu s@ tri~u~fas@ sO pelu tri~u~fu

tri~u~fo pur tri~u~far u~ fOrt tru~fw m6S ko~pwrto s komw s tri~u~fas sO pelu tri~u~fw

di

Di-lo com convicção na condição de dizer o não dito neste difícil mundial.

di lu ko~ ko~viks6~w~ n6 ko~dis6~w~ d@ dizer u n6~w~ ditu nESt@ difisil~ mu~dial~

di lu ko~ ko~viks6~w~ n6 ko~dis6~w~ d dizer u n6~w~ ditw neSt difisil~ mu~djal~

ti

Tia Tita, diz-me se é fictício o que tinham sobre ti.

ti6 tit6 diZ m@ s@ E fiktisiu u k@ tiJ6~w~ sobr@ ti

ti6 tit6 diZ m s E fiktisiu u k tiJ6~w~ sobr ti

S6

Chama mas a chamada não paga taxa.

S6m6 m6z 6 S6mad6 n6~w~ pag6 taS6

S6m6 m6z 6 S6mad6 n6~w~ pag6 taS6

p6

Papá que pape a papa é o que parece que dá a pápa.

p6pa k@ pap@ 6 pap6 E u k@ p6rEs@ k@ da 6 pap6

p6pa k pap 6 pap6 E u k p6rEs k da 6 pap6

En

É na Arménia que há milénios o ténis é no género para os arménios um desporto com cenas cénicas.

E n6 6rmEni6 k@ a milEniuz u tEniz E nu ZEn@ru p6r6 uz 6rmEniuz u~ d@Sportu ko~ sen6S sEnik6S

E n6 6rmEni6 k a milEniwz u tEniz E nu ZEnrw p6r6 uz 6rmEniwz u~ dSportw ko~ sen6S sEnik6S

kn

Técnicos da tecnologia moderna que não é tecnologicamente pequenina dizem-se hoje técnicos da luminotécnia.

tEknikuZ d6 tEknuluZi6 mudErn6 k@ n6~w~ E tEknulOZik6me~t@ p@k@nin6 diz6~j~ s@ oZ@ tEknikuZ d6 luminOtEkni6

tEknikwZ d6 tEknwlwZi6 mwdErn6 k n6~w~ E tEknwlOZik6me~t pk@nin6 diz6~j~ s oZ tEknikwZ d6 luminotEkni6

EO

É hora de um Maomé óptimo para o parque eólico e é óbvio que é horóscopo no que é óptica do neo.

E Or6 d@ u~ maOmE Otimu p6r6 u park@ EOliku i E Obviu k@ E OrOSkupu nu k@ E Otik6 du nEO

E Or6 d u~ maOmE Otimu p6r6 u park EOlikw i E Obvju k E OrOskwpw nu k E Otik6 du nEO

gO

Gosto muito de negócios pedagogicamente interessantes porque se consegue óptimo rendimento mesmo que a crise vigore.

gOStu mu~j~tu d@ n@gOsiuS p@d6gOZik6me~t@ i~t@r@s6~t@S purk@ s@ ko~sEg@ Otimu Re~dime~tu meZmu k@ 6 kriz@ vigOr@

gOStw mu~j~tw d ngOsiuS pd6gOZik6me~t i~trs6~tS purk s ko~sEg Otimu Re~dime~tw meZmu k 6 kriz vigOr@

dm

O administrador admitiu que se admirou de me ter dado sem demora em demasia os dados de metade da administração.

u 6dm@niStr6dor 6dmitiw k@ s@ 6dmiro d@ m@ ter dadu s6~j~ d@mOr6 6~j~ d@m6zi6 uZ daduZ d@ m@tad@ d6 6dm@niStr6s6~w~

u 6dmniStr6dor 6dmtiu k s 6dmiro d m ter dadw s6~j~ dmOr6 6~j~ dm6zi6 uZ dadwZ d mtad d6 6dmniStr6s6~w~

tm

Ritmos matemáticos e aritméticos de algoritmos que te meteram na cabeça, massacram-te muito ritmadamente.

RitmuZ m6t@matikuz i 6ritmEtikuZ d@ al~guritmuS k@ t@ m@ter6~w~ n6 k6bes6 m6sakr6~w~ t@ mu~j~tu Ritmad6me~t@

RitmuZ m6tmatikwz i 6ritmEtikwZ d al~gwritmuS k t mter6~w~ n6 k6bes6 m6sakr6~w~ t mu~j~tw Ritmad6me~t

Eo~

É onde todos sabem que o Zé onça fez o pé ondular.

E o~d@ toduS sab6~j~ k@ u zE o~s6 fEz u pE o~dular

E o~d todwS sab6~j~ k u zE o~s6 fez u pE o~dular

go~

Gonçalo era um menino desengonçado, mas agora consegue onze saltos de seguida naquela geringonça.

go~salu Er6 u~ m@ninu d@ze~go~sadu m6z 6gOr6 ko~sEg@ o~z@ sal~tuZ d@ s@gid6 n6kEl6 Z@ri~go~s6

go~salu Er6 u~ mninu dze~go~sadw m6z 6gOr6 ko~sEg unz sal~twZ d sgid6 n6kEl6 Zri~go~s6

s6~

Sangue de santos passando de sandálias não é interessante a não ser que façam ou aconteçam santidades que possam.

s6~g@ d@ s6~tuS p6s6~du d@ s6~dali6Z n6~w~ E i~t@r@s6~t@ 6 n6~w~ ser k@ fas6~w~ o 6ko~tes6~w~ s6~tidad@S k@ pOs6~w~

s6~g d s6~twS p6s6~dw d s6~dalj6Z n6~w~ E i~trs6~t 6 n6~w~ ser k fas6~w~ o

6ko~tes6~w~ s6~tidadS k pOs6~w~

f6~

Fantasias infantis sobre fantasmas e fadas fizeram o chefe anterior ser fantástico na infantaria.

f6~t6zi6z i~f6~tiS sobr@ f6~taZm6z i fad6S fizEr6~w~ u SEf@ 6~t@rior ser f6~taStiku n6 i~f6~t6ri6

f6~t6zi6z i~f6~tiS sobr f6~taZm6z i fad6S fizEr6~w~ u SEf 6~trior ser f6~taStikw n6 i~f6~t6ri6

El

Célebres células evangélicas disseram que ela é lutadora e maquiavélica.

sEl@br@S sElul6z iv6~ZElik6Z disEr6~w~ k@ El6 E lut6dor6 i m6ki6vElik6

sElbrS sElul6z iv6~ZElik6Z disEr6~w~ k El6 E lut6dor6 i m6ki6vElik6

gl

Globalmente aglomerado pelos Ingleses com glória negligentemente dada a quem consegue ler Inglês.

glubal~me~t@ 6glum@radu peluz i~glez@S ko~ glOri6 n@gliZe~t@me~t@ dad6 6 k6~j~ ko~sEg@ ler i~gleS

glwbal~me~t 6glumradw peluz i~glezS ko~ glOri6 n@gliZe~tme~t dad6 6 k6~j~ ko~sEg ler i~gleS

ap

Rapidamente passou as etapas do mapa indo à pressa como quem se esapa e adapta ao colapso.

Rapid6me~t@ p6so 6z etap6Z du map6 i~du a prEs6 komu k6~j~ s@ izap6 i 6dapt6 aw kulapsu

Rapid6me~t p6so 6z etap6Z du map6 i~dw a prEs6 komw k6~j~ s ezap6 i 6dapt6 aw kwlapsw

Rp

Representantes repetem o irrepreensível arrepio de arrependimento que varre penas sem arrepio.

R@pr@ze~t6~t@Z R@pEt6~j~ u iR@pri~e~sivEl 6R@piu d@ 6R@pe~dime~tu k@ vaR@ pen6S s6~j~ 6R@piu

Rprze~t6~tZ RpEt6~j~ u iRprj~e~sivEl 6Rpiu d 6Rpe~dime~tw k vaR pen6S s6~j~ 6Rpiu

eR

Cerro que aparece por trás do aterro deixa o bezerro no desterro de quem vê recriar um enterro.

seRu k@ 6p6rEs@ pur traZ du 6teRu d6jS6 u b@zeRu nu d@StERu d@ k6~j~ ve R@kriar u~ e~teRu

seRw k 6p6rEs pur traZ du 6teRw d6jS6 u bzeRw nu dSteRw d k6~j~ ve Rkrjar u~ e~teRw

JR

Ganhe razões para champanhe rebuscado e acompanhe ritmos.

g6J@ R6zo~j~S p6r6 S6~p6J@ R@buSkadu i 6ko~p6J@ RitmuS

g6J R6zo~j~S p6r6 S6~p6J RbwSkadw i 6ko~p6J RitmwS

do

Dores de costas levaram-me ao doutor dos jogadores de outras equipas de trabalhadores.

dor@Z d@ kOSt6Z l@var6∼w∼ m@ aw dotor duZ Zug6dor@Z d@ otr6z ikip6Z d@ tr6b6L6dor@S

dorZ d kOSt6Z lvar6∼w∼ m aw dotor duZ Zwg6dorZ d otr6z ikip6Z d tr6b6L6dorS

ko

De acordo com o que ficou por dizer, deve-se evitar que outra coisa ocorra pela mesma escolha que criticou.

d@ 6kordu ko∼ u k@ fiko pur dizer dEv@ s@ ivitar k@ otr6 kojz6 OkoR6 pel6 meZm6 @SkoL6 k@ kritiko

d 6kordw ko∼ u k fiko pur dizer dEv s ivitar k otr6 kojz6 okoR6 pel6 meZm6 SkoL6 k kritiko

Xo

Ou ficas ou vais!

o fik6z o vajS

o fik6z o vajS

SX

Quero que o mimes.

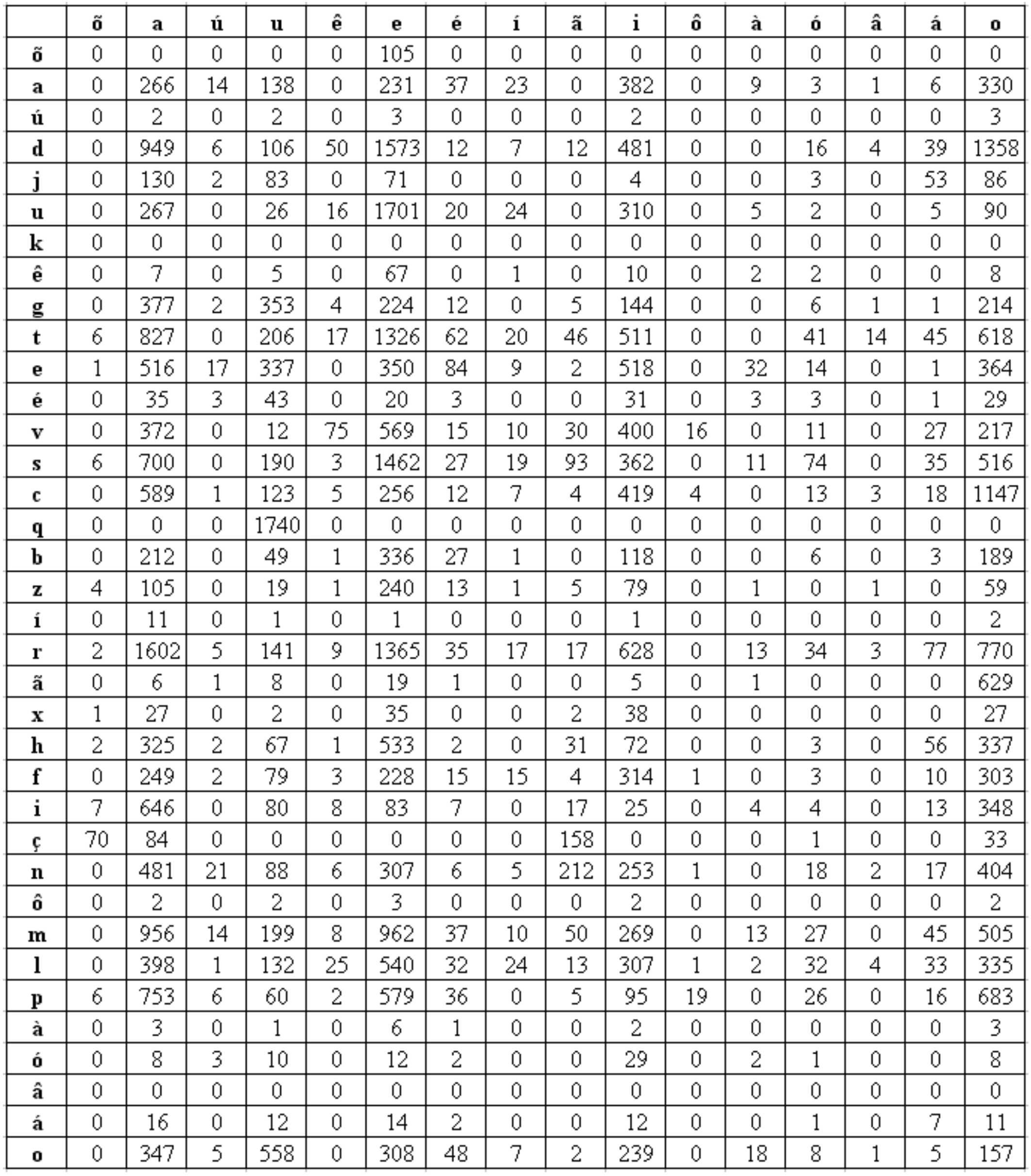
kEru k@ u mim@S

kErw k u mimS

Appendix F

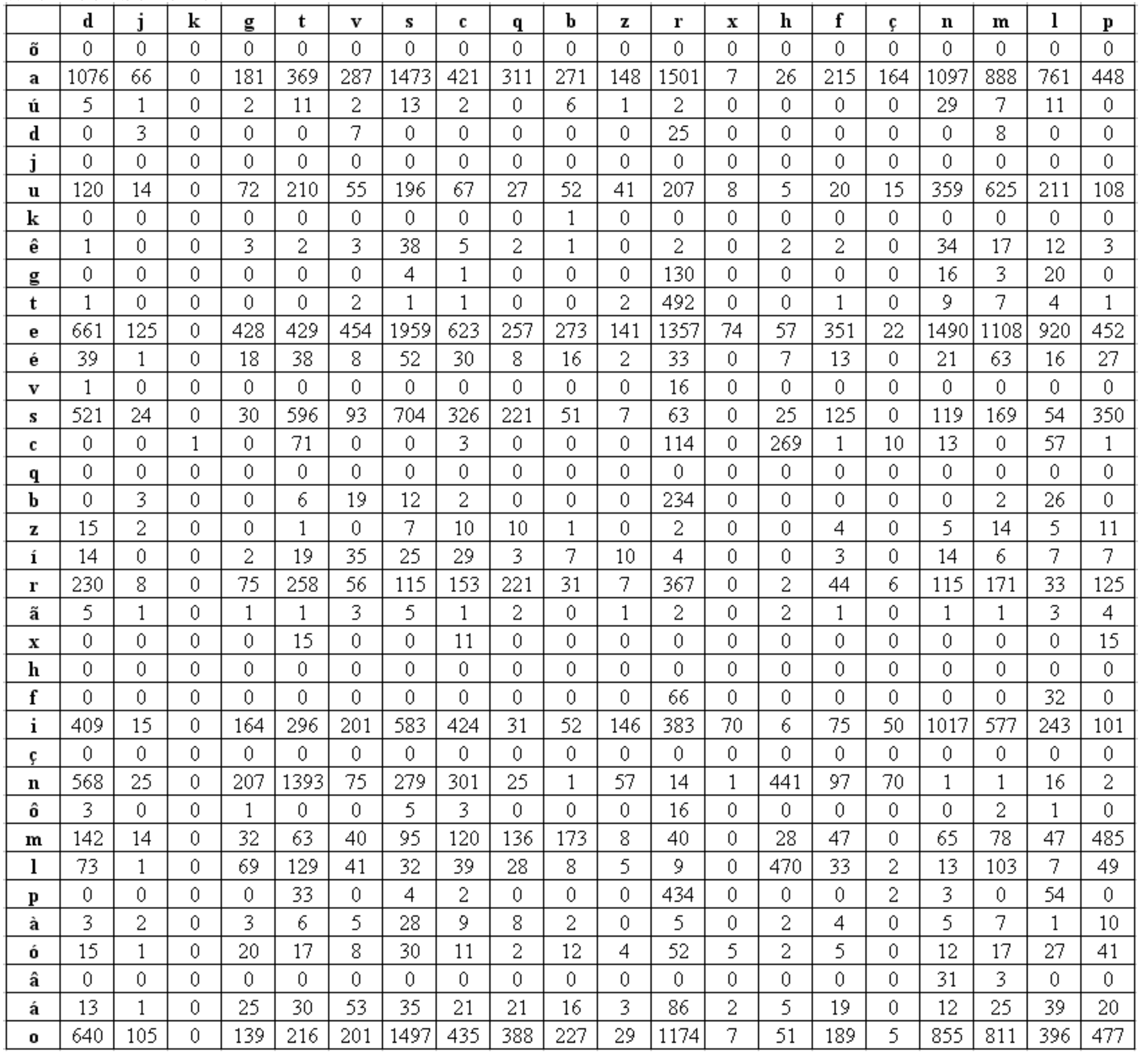
Diphones in the Speech Corpus

Table F.1: Table with total number of two grapheme occurrences' sequences, starting with vowel



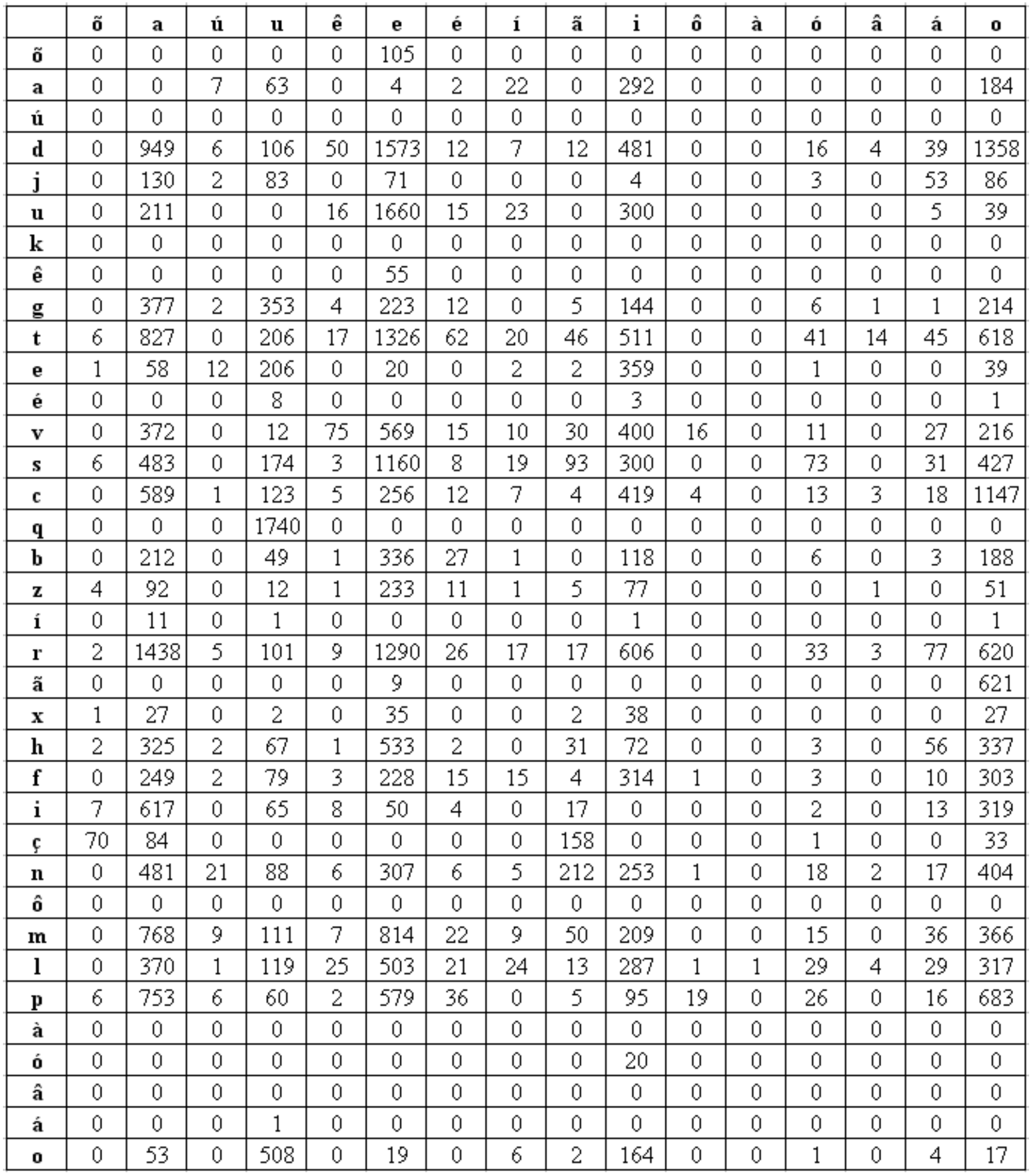
	õ	**a**	**ú**	**u**	**ê**	**e**	**é**	**í**	**ã**	**i**	**ô**	**à**	**ó**	**â**	**á**	**o**
õ	0	0	0	0	0	105	0	0	0	0	0	0	0	0	0	0
a	0	266	14	138	0	231	37	23	0	382	0	9	3	1	6	330
ú	0	2	0	2	0	3	0	0	0	2	0	0	0	0	0	3
d	0	949	6	106	50	1573	12	7	12	481	0	0	16	4	39	1358
j	0	130	2	83	0	71	0	0	0	4	0	0	3	0	53	86
u	0	267	0	26	16	1701	20	24	0	310	0	5	2	0	5	90
k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ê	0	7	0	5	0	67	0	1	0	10	0	2	2	0	0	8
g	0	377	2	353	4	224	12	0	5	144	0	0	6	1	1	214
t	6	827	0	206	17	1326	62	20	46	511	0	0	41	14	45	618
e	1	516	17	337	0	350	84	9	2	518	0	32	14	0	1	364
é	0	35	3	43	0	20	3	0	0	31	0	3	3	0	1	29
v	0	372	0	12	75	569	15	10	30	400	16	0	11	0	27	217
s	6	700	0	190	3	1462	27	19	93	362	0	11	74	0	35	516
c	0	589	1	123	5	256	12	7	4	419	4	0	13	3	18	1147
q	0	0	0	1740	0	0	0	0	0	0	0	0	0	0	0	0
b	0	212	0	49	1	336	27	1	0	118	0	0	6	0	3	189
z	4	105	0	19	1	240	13	1	5	79	0	1	0	1	0	59
í	0	11	0	1	0	1	0	0	0	1	0	0	0	0	0	2
r	2	1602	5	141	9	1365	35	17	17	628	0	13	34	3	77	770
ã	0	6	1	8	0	19	1	0	0	5	0	1	0	0	0	629
x	1	27	0	2	0	35	0	0	2	38	0	0	0	0	0	27
h	2	325	2	67	1	533	2	0	31	72	0	0	3	0	56	337
f	0	249	2	79	3	228	15	15	4	314	1	0	3	0	10	303
i	7	646	0	80	8	83	7	0	17	25	0	4	4	0	13	348
ç	70	84	0	0	0	0	0	0	158	0	0	0	1	0	0	33
n	0	481	21	88	6	307	6	5	212	253	1	0	18	2	17	404
ô	0	2	0	2	0	3	0	0	0	2	0	0	0	0	0	2
m	0	956	14	199	8	962	37	10	50	269	0	13	27	0	45	505
l	0	398	1	132	25	540	32	24	13	307	1	2	32	4	33	335
p	6	753	6	60	2	579	36	0	5	95	19	0	26	0	16	683
à	0	3	0	1	0	6	1	0	0	2	0	0	0	0	0	3
ó	0	8	3	10	0	12	2	0	0	29	0	2	1	0	0	8
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
á	0	16	0	12	0	14	2	0	0	12	0	0	1	0	7	11
o	0	347	5	558	0	308	48	7	2	239	0	18	8	1	5	157

Table F.2: Table with total number of two grapheme occurrences' sequences, starting with consonant



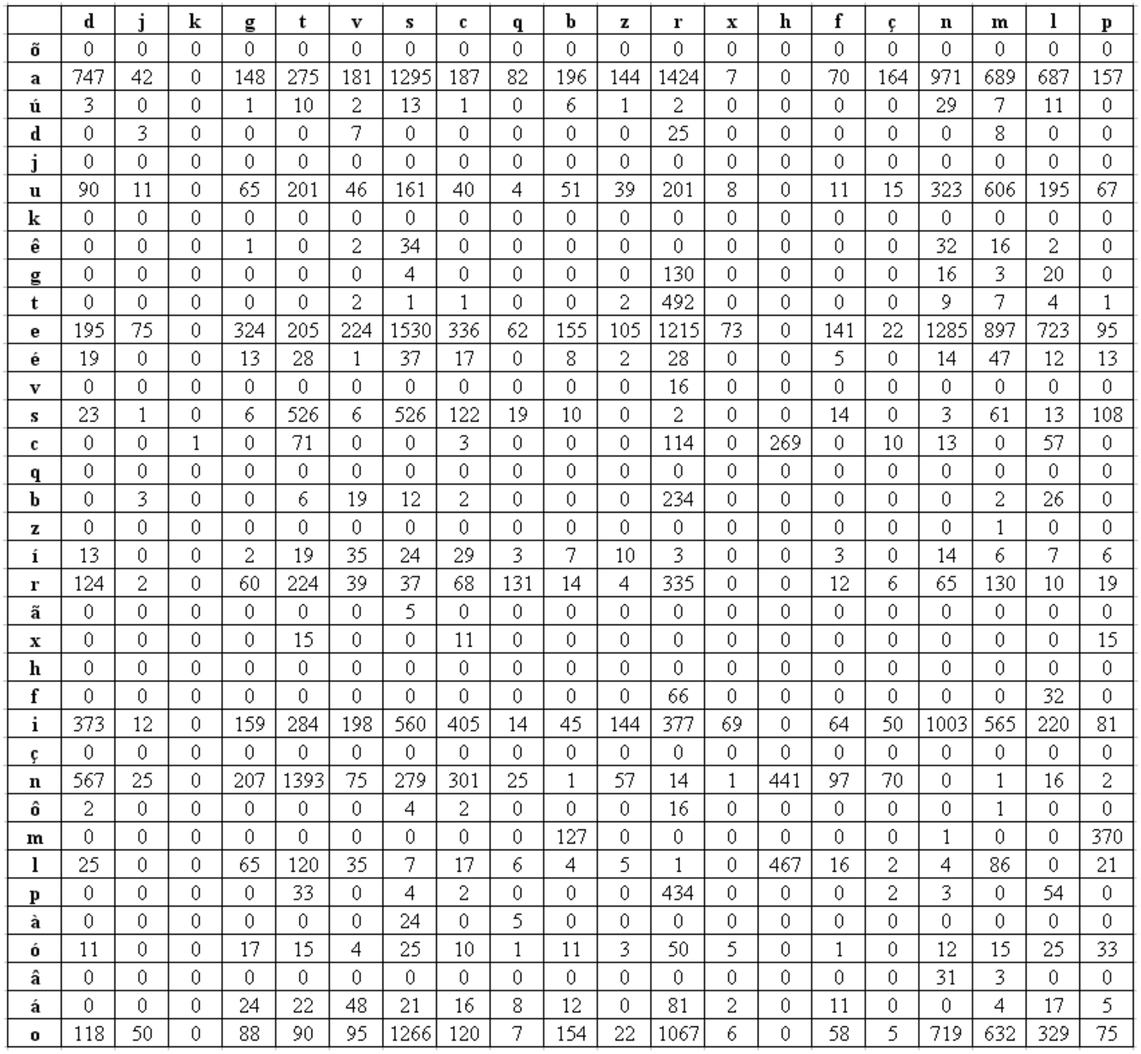
	d	**j**	**k**	**g**	**t**	**v**	**s**	**c**	**q**	**b**	**z**	**r**	**x**	**h**	**f**	**ç**	**n**	**m**	**l**	**p**
õ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	1076	66	0	181	369	287	1473	421	311	271	148	1501	7	26	215	164	1097	888	761	448
ú	5	1	0	2	11	2	13	2	0	6	1	2	0	0	0	0	29	7	11	0
d	0	3	0	0	0	7	0	0	0	0	0	25	0	0	0	0	0	8	0	0
j	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u	120	14	0	72	210	55	196	67	27	52	41	207	8	5	20	15	359	625	211	108
k	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
ê	1	0	0	3	2	3	38	5	2	1	0	2	0	2	2	0	34	17	12	3
g	0	0	0	0	0	0	4	1	0	0	0	130	0	0	0	0	16	3	20	0
t	1	0	0	0	0	2	1	1	0	0	2	492	0	0	1	0	9	7	4	1
e	661	125	0	428	429	454	1959	623	257	273	141	1357	74	57	351	22	1490	1108	920	452
é	39	1	0	18	38	8	52	30	8	16	2	33	0	7	13	0	21	63	16	27
v	1	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0
s	521	24	0	30	596	93	704	326	221	51	7	63	0	25	125	0	119	169	54	350
c	0	0	1	0	71	0	0	3	0	0	0	114	0	269	1	10	13	0	57	1
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	3	0	0	6	19	12	2	0	0	0	234	0	0	0	0	0	2	26	0
z	15	2	0	0	1	0	7	10	10	1	0	2	0	0	4	0	5	14	5	11
í	14	0	0	2	19	35	25	29	3	7	10	4	0	0	3	0	14	6	7	7
r	230	8	0	75	258	56	115	153	221	31	7	367	0	2	44	6	115	171	33	125
ã	5	1	0	1	1	3	5	1	2	0	1	2	0	2	1	0	1	1	3	4
x	0	0	0	0	15	0	0	11	0	0	0	0	0	0	0	0	0	0	0	15
h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	66	0	0	0	0	0	0	32	0
i	409	15	0	164	296	201	583	424	31	52	146	383	70	6	75	50	1017	577	243	101
ç	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n	568	25	0	207	1393	75	279	301	25	1	57	14	1	441	97	70	1	1	16	2
ô	3	0	0	1	0	0	5	3	0	0	0	16	0	0	0	0	0	2	1	0
m	142	14	0	32	63	40	95	120	136	173	8	40	0	28	47	0	65	78	47	485
l	73	1	0	69	129	41	32	39	28	8	5	9	0	470	33	2	13	103	7	49
p	0	0	0	0	33	0	4	2	0	0	0	434	0	0	0	2	3	0	54	0
à	3	2	0	3	6	5	28	9	8	2	0	5	0	2	4	0	5	7	1	10
ó	15	1	0	20	17	8	30	11	2	12	4	52	5	2	5	0	12	17	27	41
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31	3	0	0
á	13	1	0	25	30	53	35	21	21	16	3	86	2	5	19	0	12	25	39	20
o	640	105	0	139	216	201	1497	435	388	227	29	1174	7	51	189	5	855	811	396	477

Table F.3: Table with the number of two grapheme occurrences' sequences, starting with vowel, within the same word



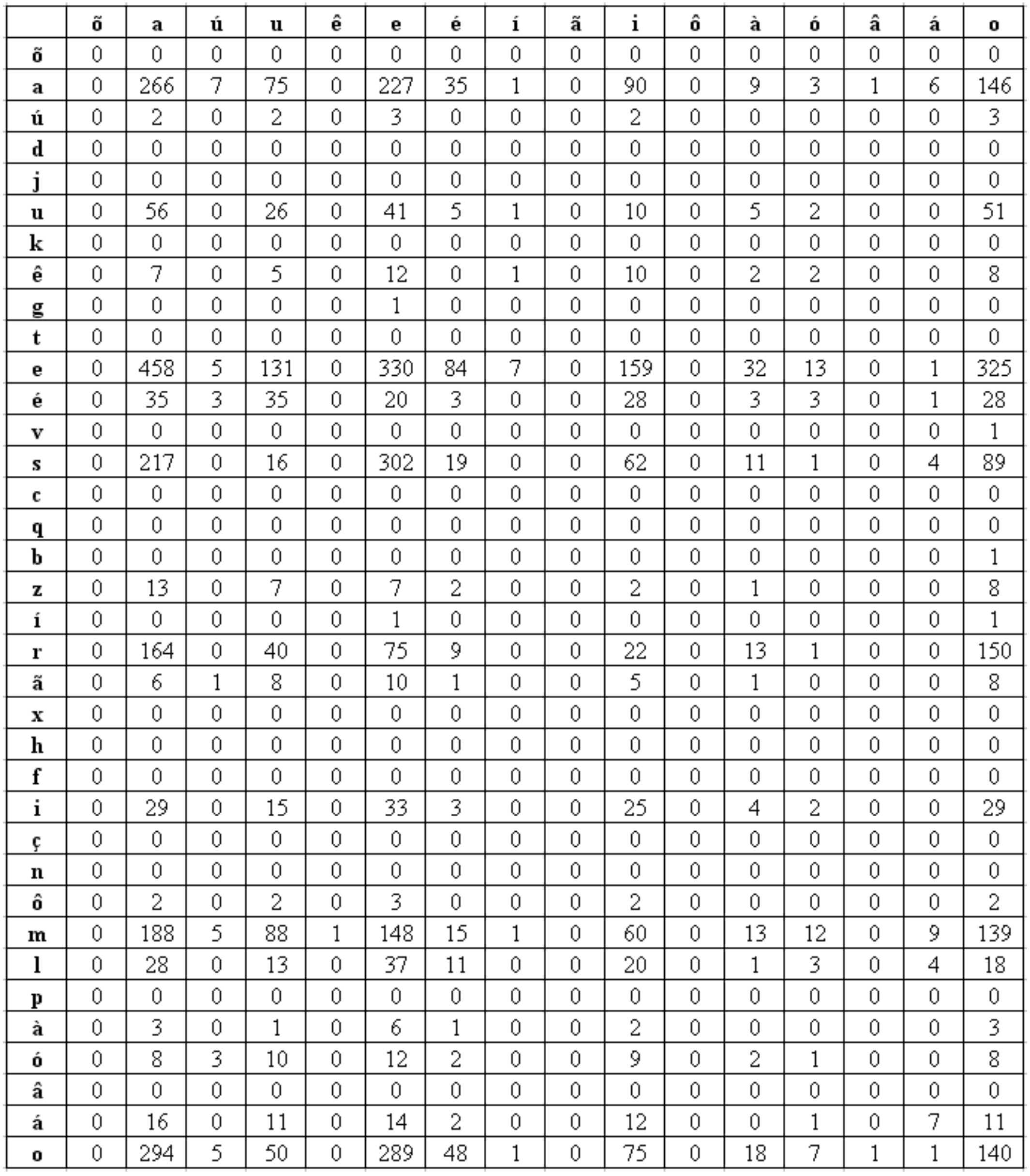
	õ	a	ú	u	ê	e	é	í	ã	i	ô	à	ó	â	á	o
õ	0	0	0	0	0	105	0	0	0	0	0	0	0	0	0	0
a	0	0	7	63	0	4	2	22	0	292	0	0	0	0	0	184
ú	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d	0	949	6	106	50	1573	12	7	12	481	0	0	16	4	39	1358
j	0	130	2	83	0	71	0	0	0	4	0	0	3	0	53	86
u	0	211	0	0	16	1660	15	23	0	300	0	0	0	0	5	39
k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ê	0	0	0	0	0	55	0	0	0	0	0	0	0	0	0	0
g	0	377	2	353	4	223	12	0	5	144	0	0	6	1	1	214
t	6	827	0	206	17	1326	62	20	46	511	0	0	41	14	45	618
e	1	58	12	206	0	20	0	2	2	359	0	0	1	0	0	39
é	0	0	0	8	0	0	0	0	0	3	0	0	0	0	0	1
v	0	372	0	12	75	569	15	10	30	400	16	0	11	0	27	216
s	6	483	0	174	3	1160	8	19	93	300	0	0	73	0	31	427
c	0	589	1	123	5	256	12	7	4	419	4	0	13	3	18	1147
q	0	0	0	1740	0	0	0	0	0	0	0	0	0	0	0	0
b	0	212	0	49	1	336	27	1	0	118	0	0	6	0	3	188
z	4	92	0	12	1	233	11	1	5	77	0	0	0	1	0	51
í	0	11	0	1	0	0	0	0	0	1	0	0	0	0	0	1
r	2	1438	5	101	9	1290	26	17	17	606	0	0	33	3	77	620
ã	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	621
x	1	27	0	2	0	35	0	0	2	38	0	0	0	0	0	27
h	2	325	2	67	1	533	2	0	31	72	0	0	3	0	56	337
f	0	249	2	79	3	228	15	15	4	314	1	0	3	0	10	303
i	7	617	0	65	8	50	4	0	17	0	0	0	2	0	13	319
ç	70	84	0	0	0	0	0	0	158	0	0	0	1	0	0	33
n	0	481	21	88	6	307	6	5	212	253	1	0	18	2	17	404
ô	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
m	0	768	9	111	7	814	22	9	50	209	0	0	15	0	36	366
l	0	370	1	119	25	503	21	24	13	287	1	1	29	4	29	317
p	6	753	6	60	2	579	36	0	5	95	19	0	26	0	16	683
à	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ó	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
á	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
o	0	53	0	508	0	19	0	6	2	164	0	0	1	0	4	17

Table F.4: Table with the number of two grapheme occurrences' sequences, starting with consonant, within the same word



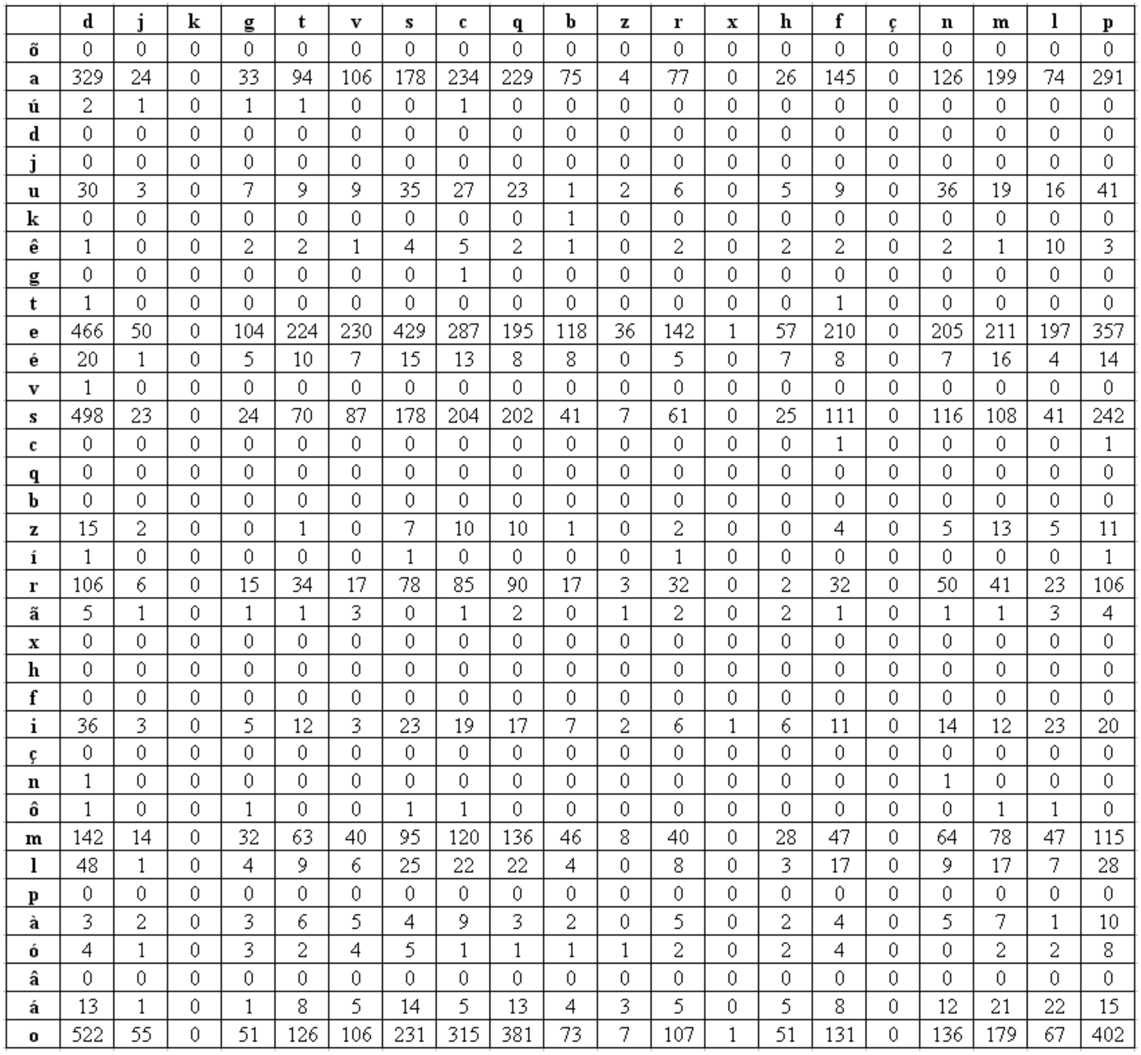
	d	**j**	**k**	**g**	**t**	**v**	**s**	**c**	**q**	**b**	**z**	**r**	**x**	**h**	**f**	**ç**	**n**	**m**	**l**	**p**
õ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	747	42	0	148	275	181	1295	187	82	196	144	1424	7	0	70	164	971	689	687	157
ú	3	0	0	1	10	2	13	1	0	6	1	2	0	0	0	0	29	7	11	0
d	0	3	0	0	0	7	0	0	0	0	0	25	0	0	0	0	0	8	0	0
j	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u	90	11	0	65	201	46	161	40	4	51	39	201	8	0	11	15	323	606	195	67
k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ê	0	0	0	1	0	2	34	0	0	0	0	0	0	0	0	0	32	16	2	0
g	0	0	0	0	0	0	4	0	0	0	0	130	0	0	0	0	16	3	20	0
t	0	0	0	0	0	2	1	1	0	0	2	492	0	0	0	0	9	7	4	1
e	195	75	0	324	205	224	1530	336	62	155	105	1215	73	0	141	22	1285	897	723	95
é	19	0	0	13	28	1	37	17	0	8	2	28	0	0	5	0	14	47	12	13
v	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0
s	23	1	0	6	526	6	526	122	19	10	0	2	0	0	14	0	3	61	13	108
c	0	0	1	0	71	0	0	3	0	0	0	114	0	269	0	10	13	0	57	0
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	3	0	0	6	19	12	2	0	0	0	234	0	0	0	0	0	2	26	0
z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
í	13	0	0	2	19	35	24	29	3	7	10	3	0	0	3	0	14	6	7	6
r	124	2	0	60	224	39	37	68	131	14	4	335	0	0	12	6	65	130	10	19
ã	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0
x	0	0	0	0	15	0	0	11	0	0	0	0	0	0	0	0	0	0	0	15
h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	66	0	0	0	0	0	0	32	0
i	373	12	0	159	284	198	560	405	14	45	144	377	69	0	64	50	1003	565	220	81
ç	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n	567	25	0	207	1393	75	279	301	25	1	57	14	1	441	97	70	0	1	16	2
ô	2	0	0	0	0	0	4	2	0	0	0	16	0	0	0	0	0	1	0	0
m	0	0	0	0	0	0	0	0	0	127	0	0	0	0	0	0	1	0	0	370
l	25	0	0	65	120	35	7	17	6	4	5	1	0	467	16	2	4	86	0	21
p	0	0	0	0	33	0	4	2	0	0	0	434	0	0	0	2	3	0	54	0
à	0	0	0	0	0	0	24	0	5	0	0	0	0	0	0	0	0	0	0	0
ó	11	0	0	17	15	4	25	10	1	11	3	50	5	0	1	0	12	15	25	33
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31	3	0	0
á	0	0	0	24	22	48	21	16	8	12	0	81	2	0	11	0	0	4	17	5
o	118	50	0	88	90	95	1266	120	7	154	22	1067	6	0	58	5	719	632	329	75

Table F.5: Table with the number of two grapheme occurrences' sequences, starting with vowel, between words



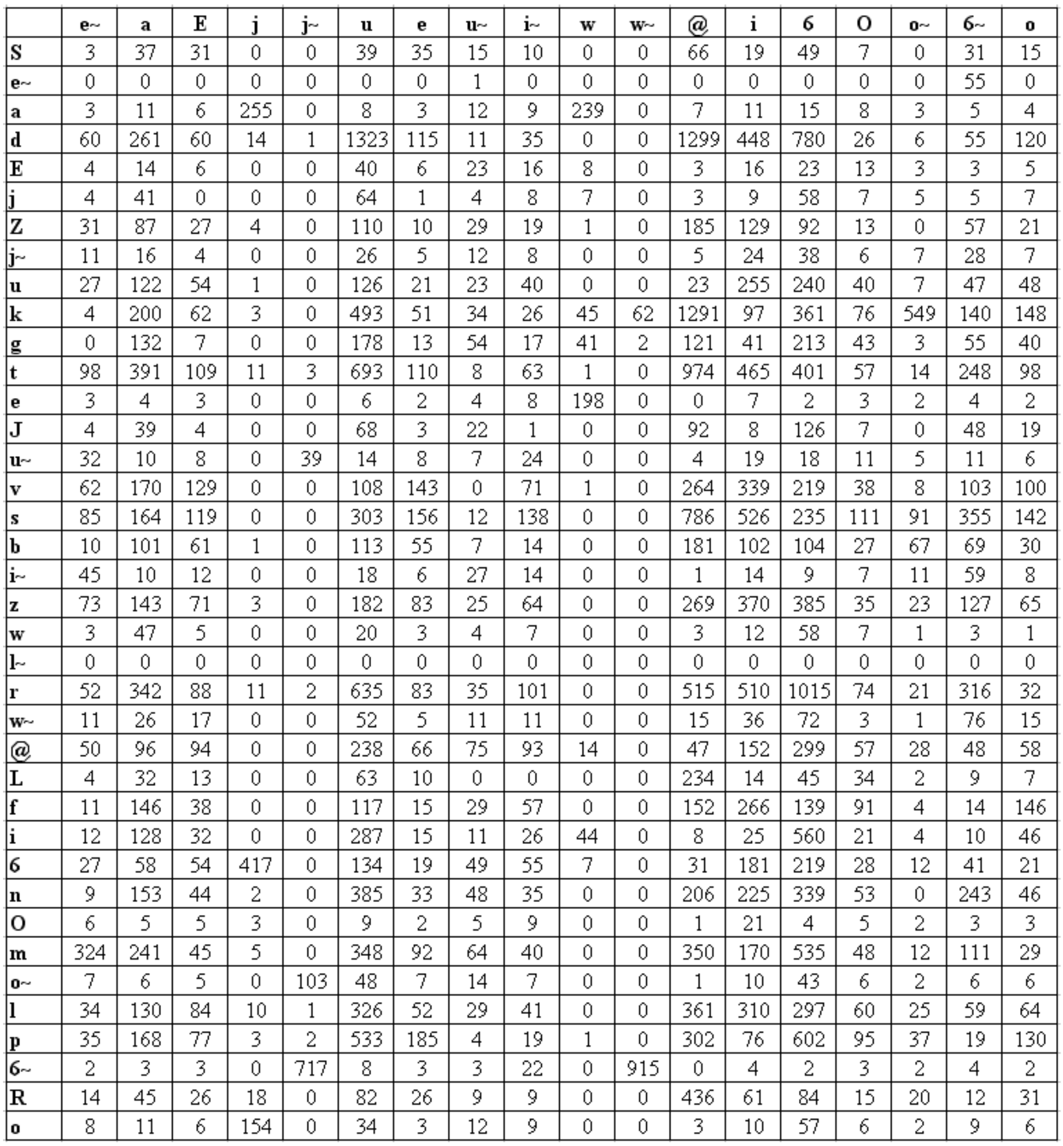
	õ	a	ú	u	ê	e	é	í	ã	i	ô	à	ó	â	á	o
õ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	0	266	7	75	0	227	35	1	0	90	0	9	3	1	6	146
ú	0	2	0	2	0	3	0	0	0	2	0	0	0	0	0	3
d	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
j	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u	0	56	0	26	0	41	5	1	0	10	0	5	2	0	0	51
k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ê	0	7	0	5	0	12	0	1	0	10	0	2	2	0	0	8
g	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	0	458	5	131	0	330	84	7	0	159	0	32	13	0	1	325
é	0	35	3	35	0	20	3	0	0	28	0	3	3	0	1	28
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
s	0	217	0	16	0	302	19	0	0	62	0	11	1	0	4	89
c	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
z	0	13	0	7	0	7	2	0	0	2	0	1	0	0	0	8
í	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
r	0	164	0	40	0	75	9	0	0	22	0	13	1	0	0	150
ã	0	6	1	8	0	10	1	0	0	5	0	1	0	0	0	8
x	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	0	29	0	15	0	33	3	0	0	25	0	4	2	0	0	29
ç	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ô	0	2	0	2	0	3	0	0	0	2	0	0	0	0	0	2
m	0	188	5	88	1	148	15	1	0	60	0	13	12	0	9	139
l	0	28	0	13	0	37	11	0	0	20	0	1	3	0	4	18
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
à	0	3	0	1	0	6	1	0	0	2	0	0	0	0	0	3
ó	0	8	3	10	0	12	2	0	0	9	0	2	1	0	0	8
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
á	0	16	0	11	0	14	2	0	0	12	0	0	1	0	7	11
o	0	294	5	50	0	289	48	1	0	75	0	18	7	1	1	140

Table F.6: Table with the number of two grapheme occurrences' sequences, starting with consonant, between words



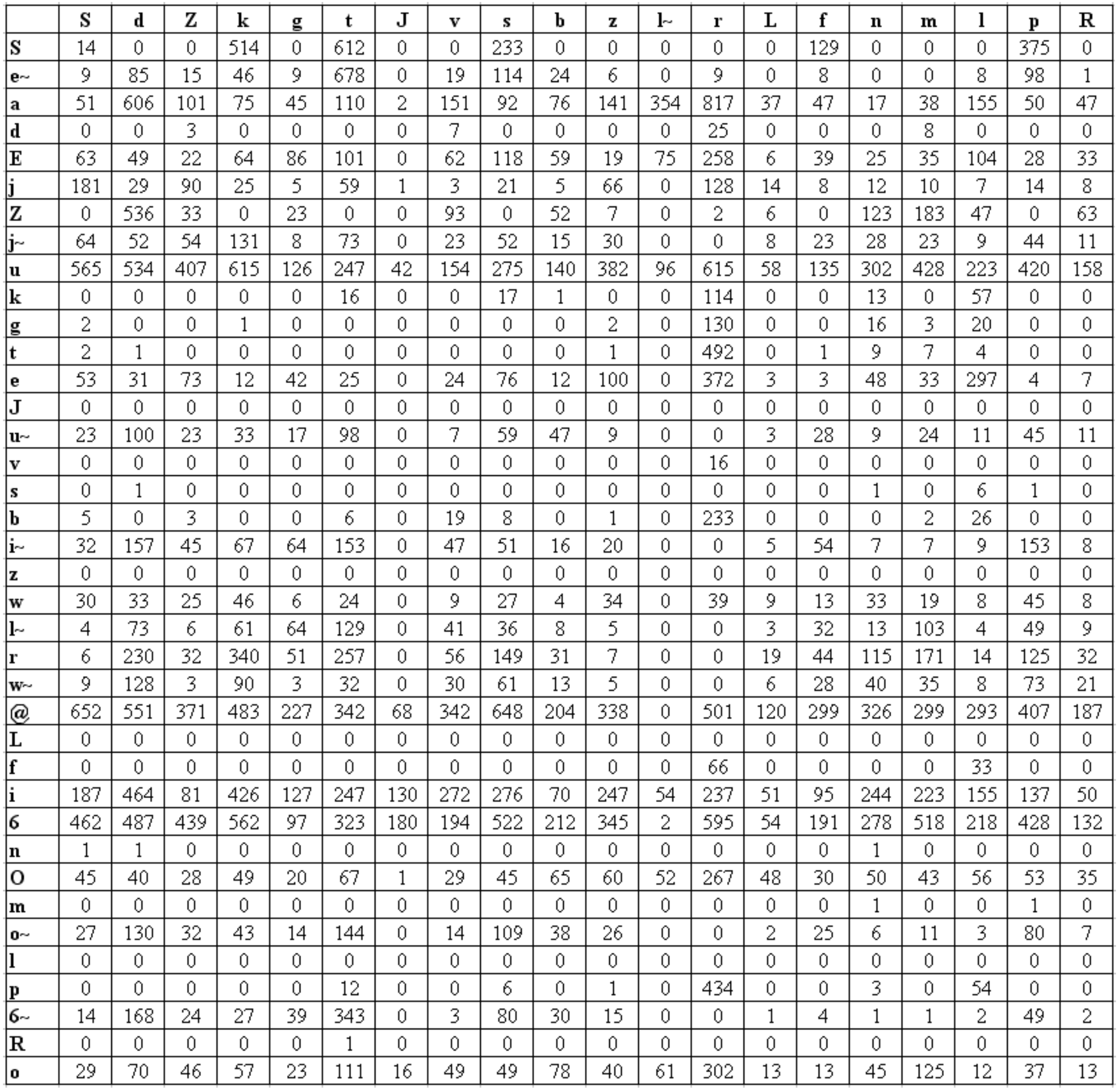
	d	j	k	g	t	v	s	c	q	b	z	r	x	h	f	ç	n	m	l	p
õ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	329	24	0	33	94	106	178	234	229	75	4	77	0	26	145	0	126	199	74	291
ú	2	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
d	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
j	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u	30	3	0	7	9	9	35	27	23	1	2	6	0	5	9	0	36	19	16	41
k	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
ê	1	0	0	2	2	1	4	5	2	1	0	2	0	2	2	0	2	1	10	3
g	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
t	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
e	466	50	0	104	224	230	429	287	195	118	36	142	1	57	210	0	205	211	197	357
é	20	1	0	5	10	7	15	13	8	8	0	5	0	7	8	0	7	16	4	14
v	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s	498	23	0	24	70	87	178	204	202	41	7	61	0	25	111	0	116	108	41	242
c	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
z	15	2	0	0	1	0	7	10	10	1	0	2	0	0	4	0	5	13	5	11
í	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1
r	106	6	0	15	34	17	78	85	90	17	3	32	0	2	32	0	50	41	23	106
ã	5	1	0	1	1	3	0	1	2	0	1	2	0	2	1	0	1	1	3	4
x	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	36	3	0	5	12	3	23	19	17	7	2	6	1	6	11	0	14	12	23	20
ç	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
n	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ô	1	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	1	1	0
m	142	14	0	32	63	40	95	120	136	46	8	40	0	28	47	0	64	78	47	115
l	48	1	0	4	9	6	25	22	22	4	0	8	0	3	17	0	9	17	7	28
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
à	3	2	0	3	6	5	4	9	3	2	0	5	0	2	4	0	5	7	1	10
ó	4	1	0	3	2	4	5	1	1	1	1	2	0	2	4	0	0	2	2	8
â	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
á	13	1	0	1	8	5	14	5	13	4	3	5	0	5	8	0	12	21	22	15
o	522	55	0	51	126	106	231	315	381	73	7	107	1	51	131	0	136	179	67	402

Table F.7: Table with the total number of diphone occurrences, starting with vowel, by rules



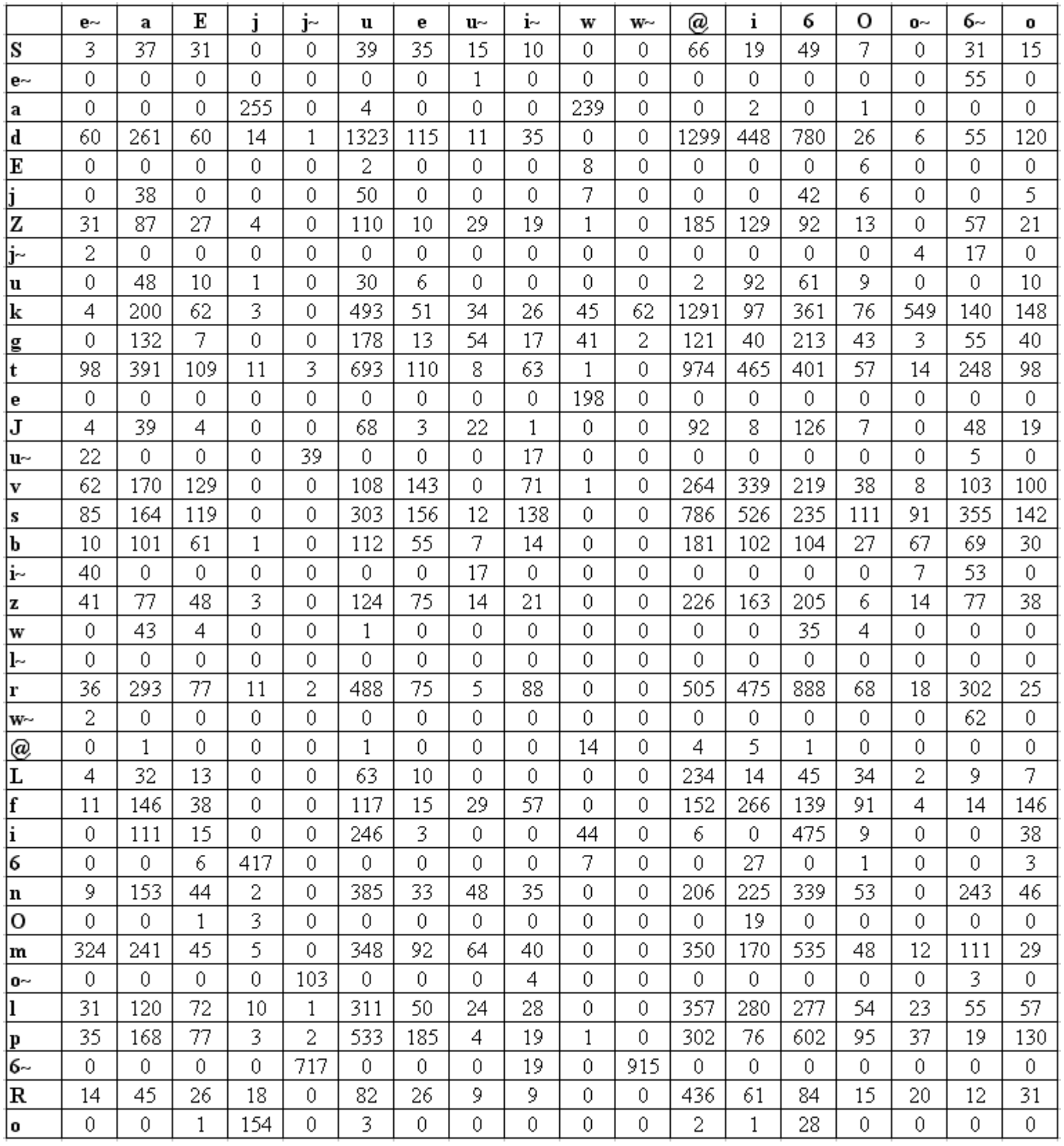
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	3	37	31	0	0	39	35	15	10	0	0	66	19	49	7	0	31	15
e~	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	55	0
a	3	11	6	255	0	8	3	12	9	239	0	7	11	15	8	3	5	4
d	60	261	60	14	1	1323	115	11	35	0	0	1299	448	780	26	6	55	120
E	4	14	6	0	0	40	6	23	16	8	0	3	16	23	13	3	3	5
j	4	41	0	0	0	64	1	4	8	7	0	3	9	58	7	5	5	7
Z	31	87	27	4	0	110	10	29	19	1	0	185	129	92	13	0	57	21
j~	11	16	4	0	0	26	5	12	8	0	0	5	24	38	6	7	28	7
u	27	122	54	1	0	126	21	23	40	0	0	23	255	240	40	7	47	48
k	4	200	62	3	0	493	51	34	26	45	62	1291	97	361	76	549	140	148
g	0	132	7	0	0	178	13	54	17	41	2	121	41	213	43	3	55	40
t	98	391	109	11	3	693	110	8	63	1	0	974	465	401	57	14	248	98
e	3	4	3	0	0	6	2	4	8	198	0	0	7	2	3	2	4	2
J	4	39	4	0	0	68	3	22	1	0	0	92	8	126	7	0	48	19
u~	32	10	8	0	39	14	8	7	24	0	0	4	19	18	11	5	11	6
v	62	170	129	0	0	108	143	0	71	1	0	264	339	219	38	8	103	100
s	85	164	119	0	0	303	156	12	138	0	0	786	526	235	111	91	355	142
b	10	101	61	1	0	113	55	7	14	0	0	181	102	104	27	67	69	30
i~	45	10	12	0	0	18	6	27	14	0	0	1	14	9	7	11	59	8
z	73	143	71	3	0	182	83	25	64	0	0	269	370	385	35	23	127	65
w	3	47	5	0	0	20	3	4	7	0	0	3	12	58	7	1	3	1
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	52	342	88	11	2	635	83	35	101	0	0	515	510	1015	74	21	316	32
w~	11	26	17	0	0	52	5	11	11	0	0	15	36	72	3	1	76	15
@	50	96	94	0	0	238	66	75	93	14	0	47	152	299	57	28	48	58
L	4	32	13	0	0	63	10	0	0	0	0	234	14	45	34	2	9	7
f	11	146	38	0	0	117	15	29	57	0	0	152	266	139	91	4	14	146
i	12	128	32	0	0	287	15	11	26	44	0	8	25	560	21	4	10	46
6	27	58	54	417	0	134	19	49	55	7	0	31	181	219	28	12	41	21
n	9	153	44	2	0	385	33	48	35	0	0	206	225	339	53	0	243	46
O	6	5	5	3	0	9	2	5	9	0	0	1	21	4	5	2	3	3
m	324	241	45	5	0	348	92	64	40	0	0	350	170	535	48	12	111	29
o~	7	6	5	0	103	48	7	14	7	0	0	1	10	43	6	2	6	6
l	34	130	84	10	1	326	52	29	41	0	0	361	310	297	60	25	59	64
p	35	168	77	3	2	533	185	4	19	1	0	302	76	602	95	37	19	130
6~	2	3	3	0	717	8	3	3	22	0	915	0	4	2	3	2	4	2
R	14	45	26	18	0	82	26	9	9	0	0	436	61	84	15	20	12	31
o	8	11	6	154	0	34	3	12	9	0	0	3	10	57	6	2	9	6

Table F.8: Table with the total number of diphone occurrences, starting with consonant, by rules



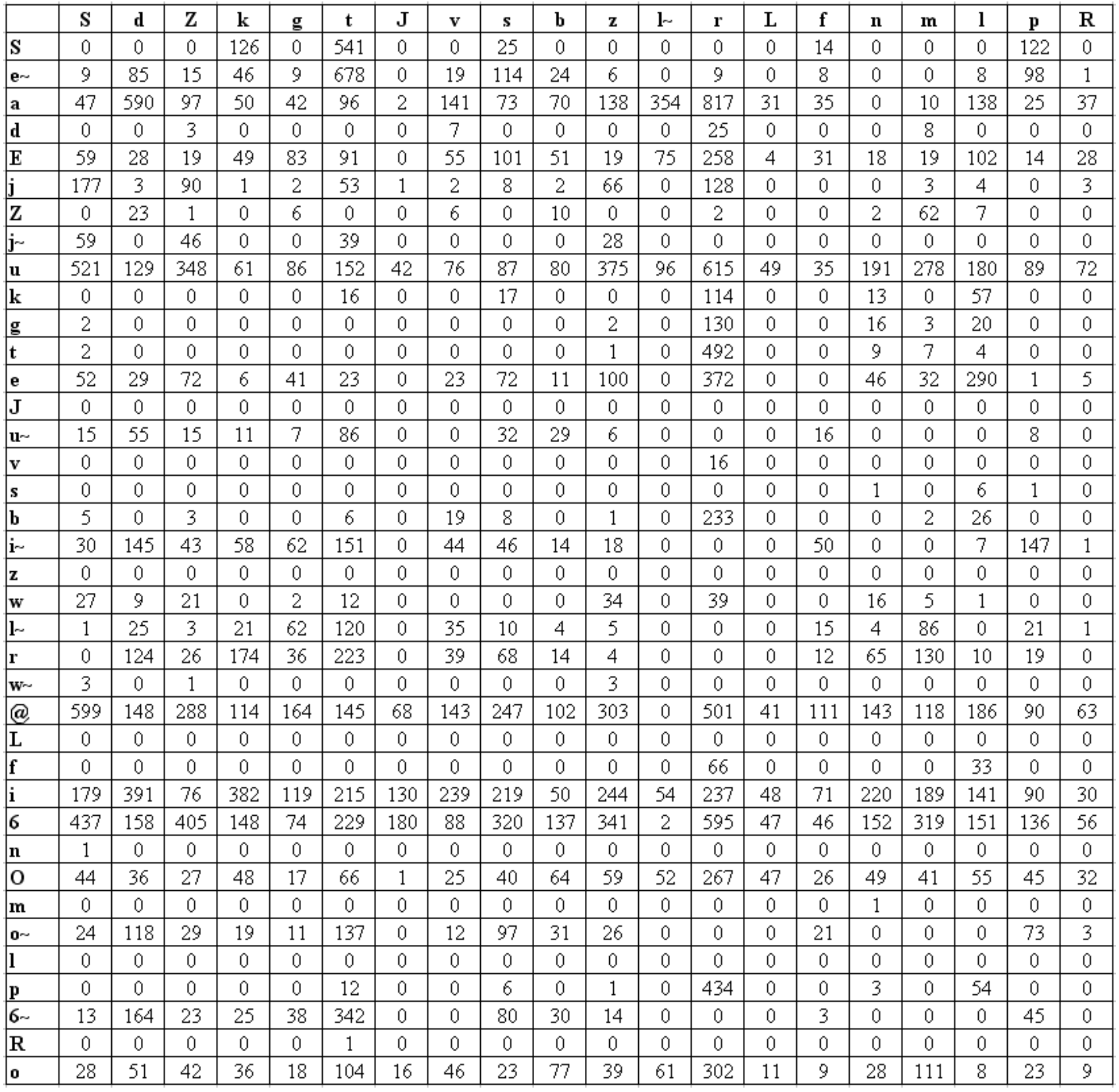
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	14	0	0	514	0	612	0	0	233	0	0	0	0	0	129	0	0	0	375	0
e~	9	85	15	46	9	678	0	19	114	24	6	0	9	0	8	0	0	8	98	1
a	51	606	101	75	45	110	2	151	92	76	141	354	817	37	47	17	38	155	50	47
d	0	0	3	0	0	0	0	7	0	0	0	0	25	0	0	0	8	0	0	0
E	63	49	22	64	86	101	0	62	118	59	19	75	258	6	39	25	35	104	28	33
j	181	29	90	25	5	59	1	3	21	5	66	0	128	14	8	12	10	7	14	8
Z	0	536	33	0	23	0	0	93	0	52	7	0	2	6	0	123	183	47	0	63
j~	64	52	54	131	8	73	0	23	52	15	30	0	0	8	23	28	23	9	44	11
u	565	534	407	615	126	247	42	154	275	140	382	96	615	58	135	302	428	223	420	158
k	0	0	0	0	0	16	0	0	17	1	0	0	114	0	0	13	0	57	0	0
g	2	0	0	1	0	0	0	0	0	0	2	0	130	0	0	16	3	20	0	0
t	2	1	0	0	0	0	0	0	0	0	1	0	492	0	1	9	7	4	0	0
e	53	31	73	12	42	25	0	24	76	12	100	0	372	3	3	48	33	297	4	7
J	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u~	23	100	23	33	17	98	0	7	59	47	9	0	0	3	28	9	24	11	45	11
v	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0
s	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	6	1	0
b	5	0	3	0	0	6	0	19	8	0	1	0	233	0	0	0	2	26	0	0
i~	32	157	45	67	64	153	0	47	51	16	20	0	0	5	54	7	7	9	153	8
z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	30	33	25	46	6	24	0	9	27	4	34	0	39	9	13	33	19	8	45	8
l~	4	73	6	61	64	129	0	41	36	8	5	0	0	3	32	13	103	4	49	9
r	6	230	32	340	51	257	0	56	149	31	7	0	0	19	44	115	171	14	125	32
w~	9	128	3	90	3	32	0	30	61	13	5	0	0	6	28	40	35	8	73	21
@	652	551	371	483	227	342	68	342	648	204	338	0	501	120	299	326	299	293	407	187
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	66	0	0	0	0	33	0	0
i	187	464	81	426	127	247	130	272	276	70	247	54	237	51	95	244	223	155	137	50
6	462	487	439	562	97	323	180	194	522	212	345	2	595	54	191	278	518	218	428	132
n	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
O	45	40	28	49	20	67	1	29	45	65	60	52	267	48	30	50	43	56	53	35
m	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
o~	27	130	32	43	14	144	0	14	109	38	26	0	0	2	25	6	11	3	80	7
l	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	12	0	0	6	0	1	0	434	0	0	3	0	54	0	0
6~	14	168	24	27	39	343	0	3	80	30	15	0	0	1	4	1	1	2	49	2
R	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
o	29	70	46	57	23	111	16	49	49	78	40	61	302	13	13	45	125	12	37	13

Table F.9: Table with the number of diphone occurrences, starting with vowel, within the same word, by rules



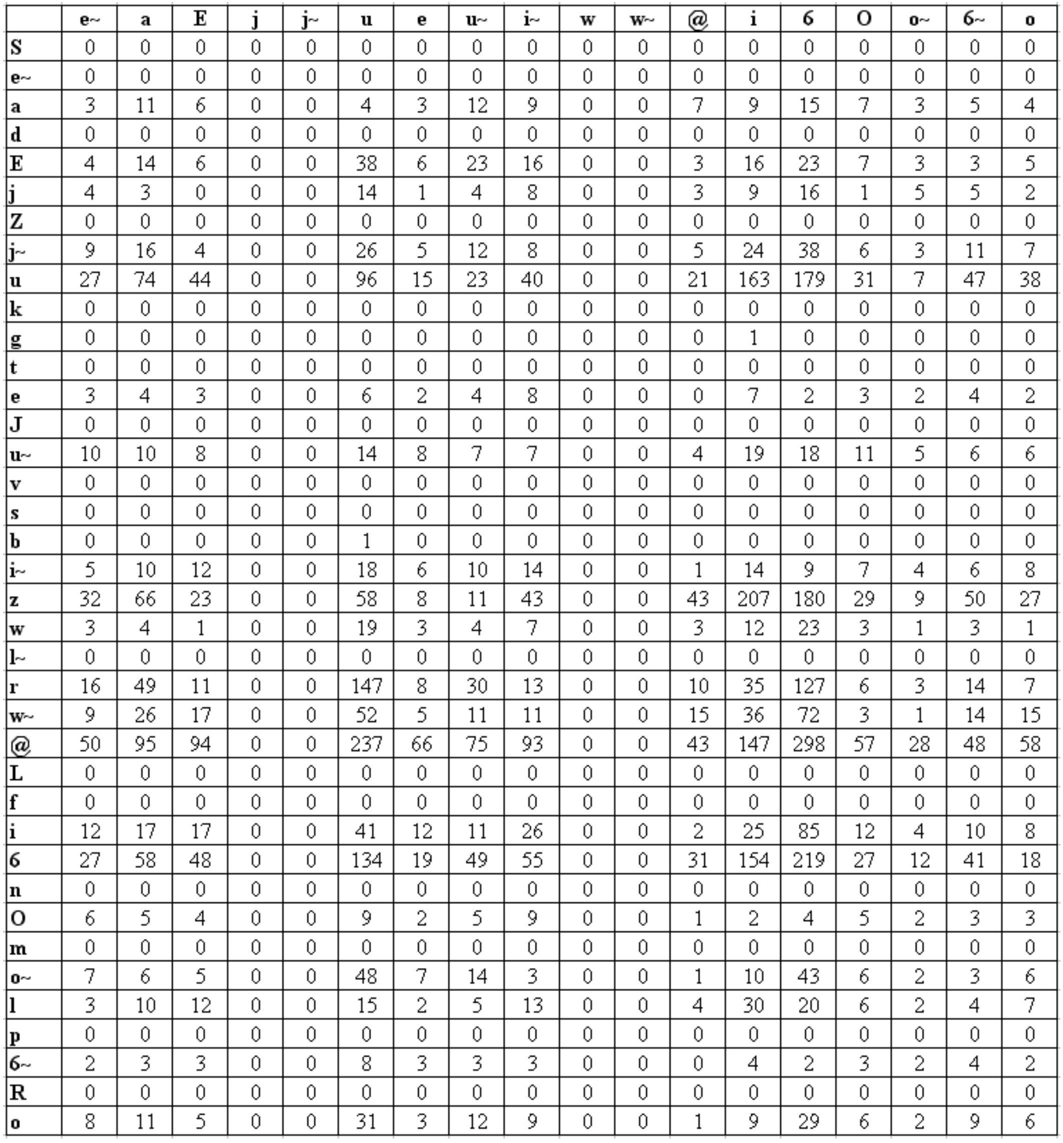
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	3	37	31	0	0	39	35	15	10	0	0	66	19	49	7	0	31	15
e~	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	55	0
a	0	0	0	255	0	4	0	0	0	239	0	0	2	0	1	0	0	0
d	60	261	60	14	1	1323	115	11	35	0	0	1299	448	780	26	6	55	120
E	0	0	0	0	0	2	0	0	0	8	0	0	0	0	6	0	0	0
j	0	38	0	0	0	50	0	0	0	7	0	0	0	42	6	0	0	5
Z	31	87	27	4	0	110	10	29	19	1	0	185	129	92	13	0	57	21
j~	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	17	0
u	0	48	10	1	0	30	6	0	0	0	0	2	92	61	9	0	0	10
k	4	200	62	3	0	493	51	34	26	45	62	1291	97	361	76	549	140	148
g	0	132	7	0	0	178	13	54	17	41	2	121	40	213	43	3	55	40
t	98	391	109	11	3	693	110	8	63	1	0	974	465	401	57	14	248	98
e	0	0	0	0	0	0	0	0	0	198	0	0	0	0	0	0	0	0
J	4	39	4	0	0	68	3	22	1	0	0	92	8	126	7	0	48	19
u~	22	0	0	0	39	0	0	0	17	0	0	0	0	0	0	0	5	0
v	62	170	129	0	0	108	143	0	71	1	0	264	339	219	38	8	103	100
s	85	164	119	0	0	303	156	12	138	0	0	786	526	235	111	91	355	142
b	10	101	61	1	0	112	55	7	14	0	0	181	102	104	27	67	69	30
i~	40	0	0	0	0	0	0	17	0	0	0	0	0	0	0	7	53	0
z	41	77	48	3	0	124	75	14	21	0	0	226	163	205	6	14	77	38
w	0	43	4	0	0	1	0	0	0	0	0	0	0	35	4	0	0	0
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	36	293	77	11	2	488	75	5	88	0	0	505	475	888	68	18	302	25
w~	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	62	0
@	0	1	0	0	0	1	0	0	0	14	0	4	5	1	0	0	0	0
L	4	32	13	0	0	63	10	0	0	0	0	234	14	45	34	2	9	7
f	11	146	38	0	0	117	15	29	57	0	0	152	266	139	91	4	14	146
i	0	111	15	0	0	246	3	0	0	44	0	6	0	475	9	0	0	38
6	0	0	6	417	0	0	0	0	0	7	0	0	27	0	1	0	0	3
n	9	153	44	2	0	385	33	48	35	0	0	206	225	339	53	0	243	46
O	0	0	1	3	0	0	0	0	0	0	0	0	19	0	0	0	0	0
m	324	241	45	5	0	348	92	64	40	0	0	350	170	535	48	12	111	29
o~	0	0	0	0	103	0	0	0	4	0	0	0	0	0	0	0	3	0
l	31	120	72	10	1	311	50	24	28	0	0	357	280	277	54	23	55	57
p	35	168	77	3	2	533	185	4	19	1	0	302	76	602	95	37	19	130
6~	0	0	0	0	717	0	0	0	19	0	915	0	0	0	0	0	0	0
R	14	45	26	18	0	82	26	9	9	0	0	436	61	84	15	20	12	31
o	0	0	1	154	0	3	0	0	0	0	0	2	1	28	0	0	0	0

Table F.10: Table with the number of diphone occurrences, starting with consonant, within the same word, by rules



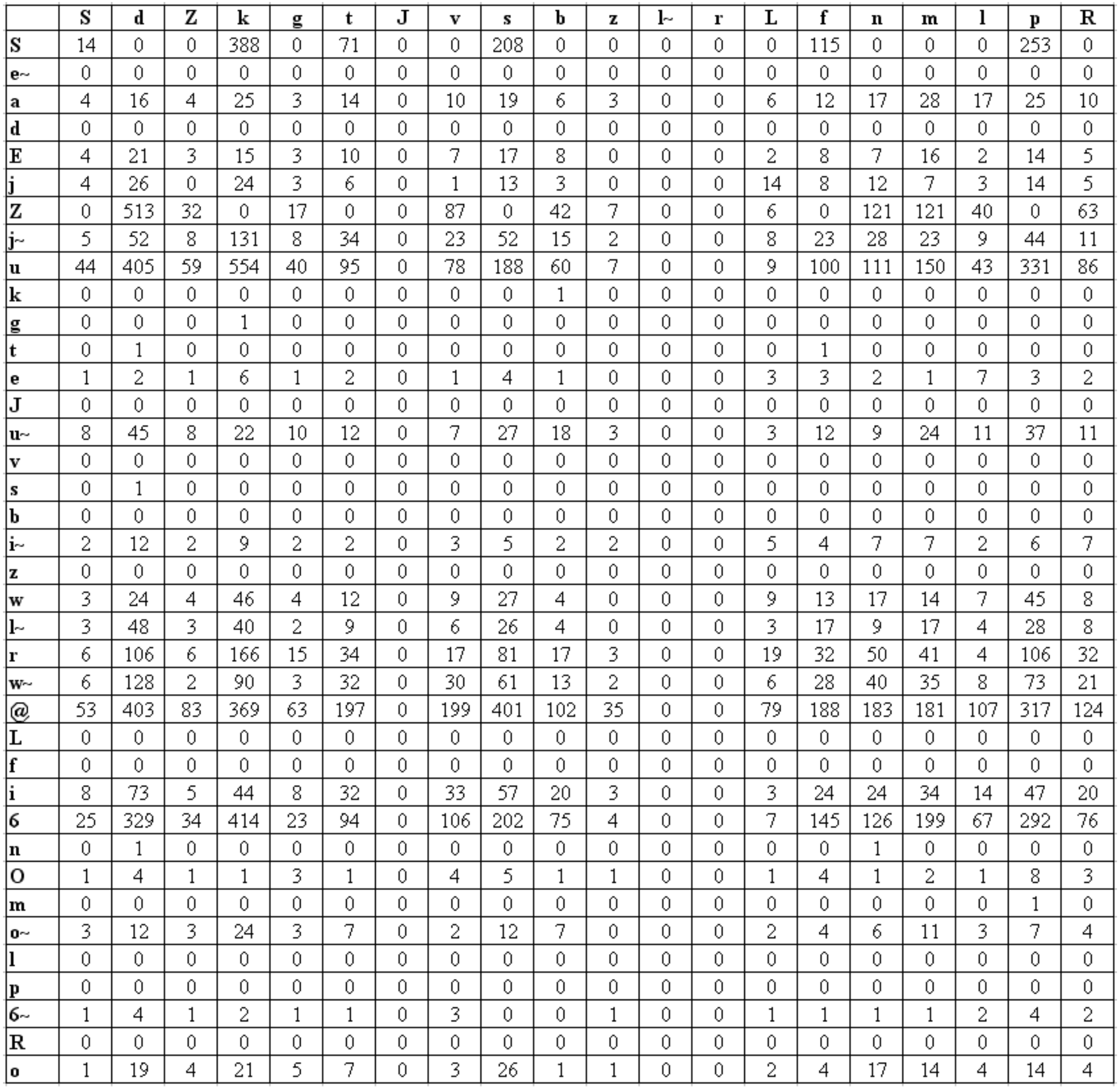
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	0	0	0	126	0	541	0	0	25	0	0	0	0	0	14	0	0	0	122	0
e~	9	85	15	46	9	678	0	19	114	24	6	0	9	0	8	0	0	8	98	1
a	47	590	97	50	42	96	2	141	73	70	138	354	817	31	35	0	10	138	25	37
d	0	0	3	0	0	0	0	7	0	0	0	0	25	0	0	0	8	0	0	0
E	59	28	19	49	83	91	0	55	101	51	19	75	258	4	31	18	19	102	14	28
j	177	3	90	1	2	53	1	2	8	2	66	0	128	0	0	0	3	4	0	3
Z	0	23	1	0	6	0	0	6	0	10	0	0	2	0	0	2	62	7	0	0
j~	59	0	46	0	0	39	0	0	0	0	28	0	0	0	0	0	0	0	0	0
u	521	129	348	61	86	152	42	76	87	80	375	96	615	49	35	191	278	180	89	72
k	0	0	0	0	0	16	0	0	17	0	0	0	114	0	0	13	0	57	0	0
g	2	0	0	0	0	0	0	0	0	0	2	0	130	0	0	16	3	20	0	0
t	2	0	0	0	0	0	0	0	0	0	1	0	492	0	0	9	7	4	0	0
e	52	29	72	6	41	23	0	23	72	11	100	0	372	0	0	46	32	290	1	5
J	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u~	15	55	15	11	7	86	0	0	32	29	6	0	0	0	16	0	0	0	8	0
v	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	6	1	0
b	5	0	3	0	0	6	0	19	8	0	1	0	233	0	0	0	2	26	0	0
i~	30	145	43	58	62	151	0	44	46	14	18	0	0	0	50	0	0	7	147	1
z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	27	9	21	0	2	12	0	0	0	0	34	0	39	0	0	16	5	1	0	0
l~	1	25	3	21	62	120	0	35	10	4	5	0	0	0	15	4	86	0	21	1
r	0	124	26	174	36	223	0	39	68	14	4	0	0	0	12	65	130	10	19	0
w~	3	0	1	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
@	599	148	288	114	164	145	68	143	247	102	303	0	501	41	111	143	118	186	90	63
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	66	0	0	0	0	33	0	0
i	179	391	76	382	119	215	130	239	219	50	244	54	237	48	71	220	189	141	90	30
6	437	158	405	148	74	229	180	88	320	137	341	2	595	47	46	152	319	151	136	56
n	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O	44	36	27	48	17	66	1	25	40	64	59	52	267	47	26	49	41	55	45	32
m	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
o~	24	118	29	19	11	137	0	12	97	31	26	0	0	0	21	0	0	0	73	3
l	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	12	0	0	6	0	1	0	434	0	0	3	0	54	0	0
6~	13	164	23	25	38	342	0	0	80	30	14	0	0	0	3	0	0	0	45	0
R	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
o	28	51	42	36	18	104	16	46	23	77	39	61	302	11	9	28	111	8	23	9

Table F.11: Table with the number of diphone occurrences, starting with vowel, between words, by rules



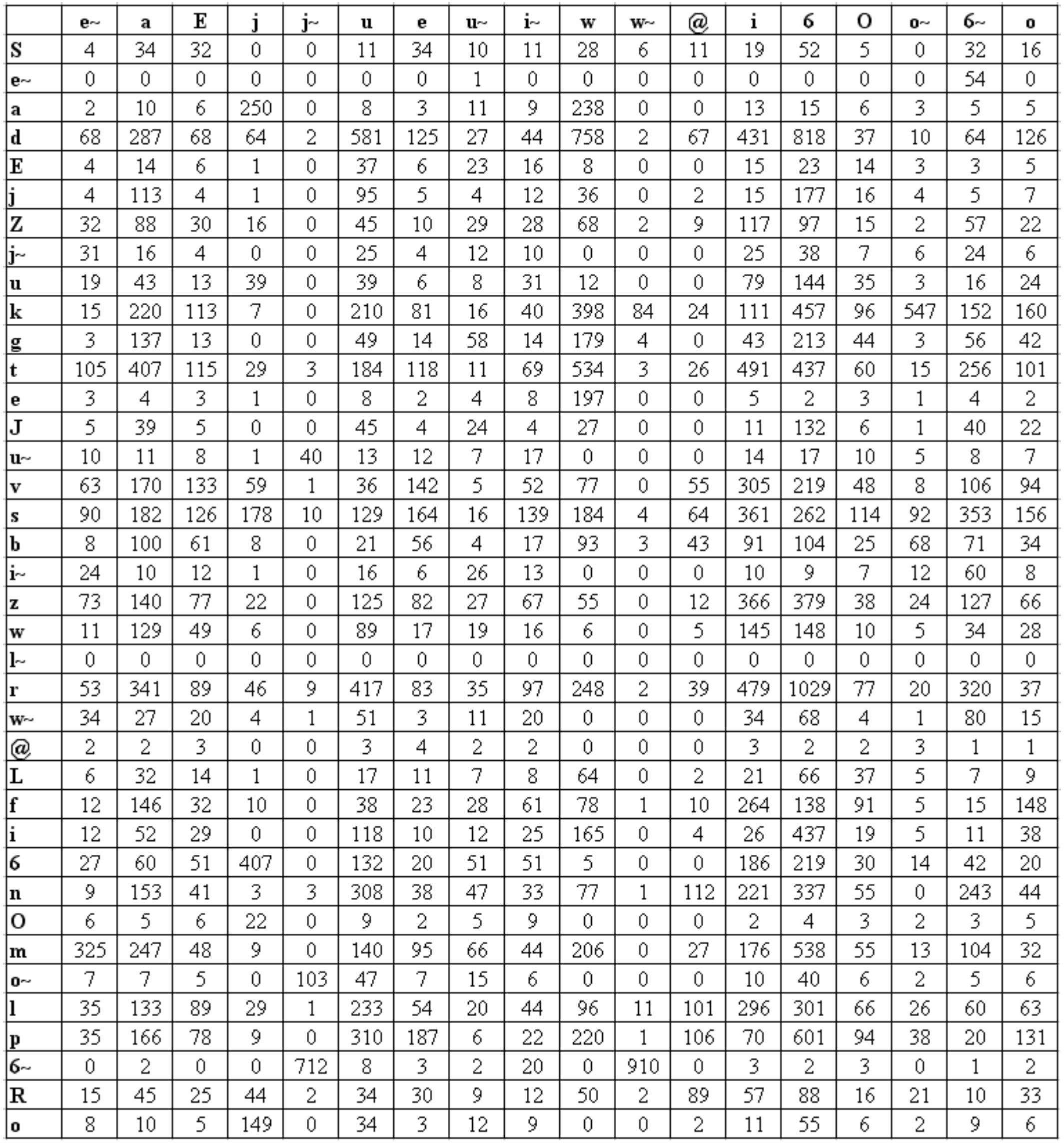
	e~	**a**	**E**	**j**	**j~**	**u**	**e**	**u~**	**i~**	**w**	**w~**	**@**	**i**	**6**	**O**	**o~**	**6~**	**o**
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	3	11	6	0	0	4	3	12	9	0	0	7	9	15	7	3	5	4
d	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	4	14	6	0	0	38	6	23	16	0	0	3	16	23	7	3	3	5
j	4	3	0	0	0	14	1	4	8	0	0	3	9	16	1	5	5	2
Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
j~	9	16	4	0	0	26	5	12	8	0	0	5	24	38	6	3	11	7
u	27	74	44	0	0	96	15	23	40	0	0	21	163	179	31	7	47	38
k	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e	3	4	3	0	0	6	2	4	8	0	0	0	7	2	3	2	4	2
J	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u~	10	10	8	0	0	14	8	7	7	0	0	4	19	18	11	5	6	6
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
i~	5	10	12	0	0	18	6	10	14	0	0	1	14	9	7	4	6	8
z	32	66	23	0	0	58	8	11	43	0	0	43	207	180	29	9	50	27
w	3	4	1	0	0	19	3	4	7	0	0	3	12	23	3	1	3	1
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	16	49	11	0	0	147	8	30	13	0	0	10	35	127	6	3	14	7
w~	9	26	17	0	0	52	5	11	11	0	0	15	36	72	3	1	14	15
@	50	95	94	0	0	237	66	75	93	0	0	43	147	298	57	28	48	58
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	12	17	17	0	0	41	12	11	26	0	0	2	25	85	12	4	10	8
6	27	58	48	0	0	134	19	49	55	0	0	31	154	219	27	12	41	18
n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O	6	5	4	0	0	9	2	5	9	0	0	1	2	4	5	2	3	3
m	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
o~	7	6	5	0	0	48	7	14	3	0	0	1	10	43	6	2	3	6
l	3	10	12	0	0	15	2	5	13	0	0	4	30	20	6	2	4	7
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6~	2	3	3	0	0	8	3	3	3	0	0	0	4	2	3	2	4	2
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
o	8	11	5	0	0	31	3	12	9	0	0	1	9	29	6	2	9	6

Table F.12: Table with the number of diphone occurrences, starting with consonant, between words, by rules



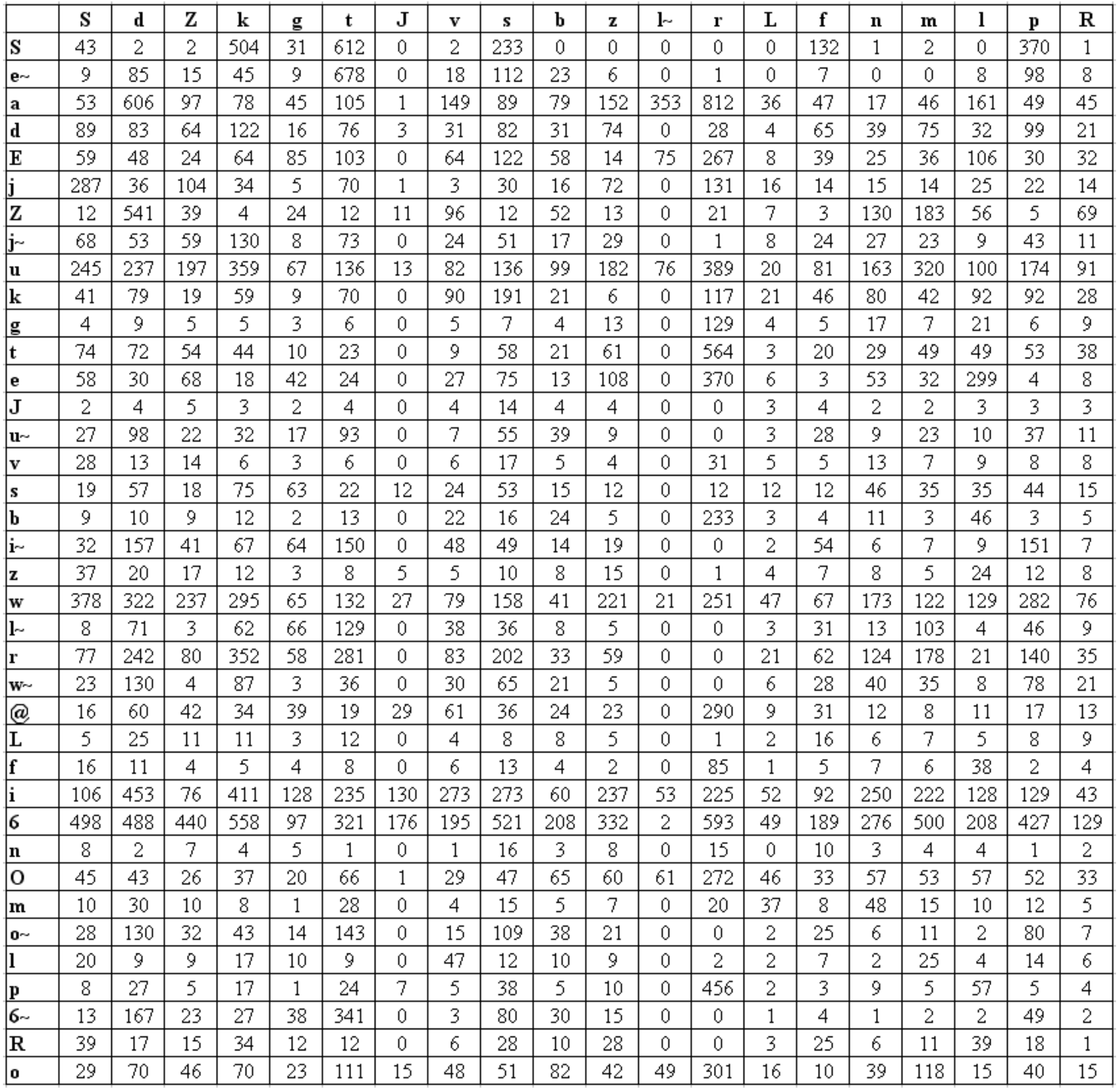
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	14	0	0	388	0	71	0	0	208	0	0	0	0	0	115	0	0	0	253	0
e~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	4	16	4	25	3	14	0	10	19	6	3	0	0	6	12	17	28	17	25	10
d	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	4	21	3	15	3	10	0	7	17	8	0	0	0	2	8	7	16	2	14	5
j	4	26	0	24	3	6	0	1	13	3	0	0	0	14	8	12	7	3	14	5
Z	0	513	32	0	17	0	0	87	0	42	7	0	0	6	0	121	121	40	0	63
j~	5	52	8	131	8	34	0	23	52	15	2	0	0	8	23	28	23	9	44	11
u	44	405	59	554	40	95	0	78	188	60	7	0	0	9	100	111	150	43	331	86
k	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
g	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
t	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
e	1	2	1	6	1	2	0	1	4	1	0	0	0	3	3	2	1	7	3	2
J	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
u~	8	45	8	22	10	12	0	7	27	18	3	0	0	3	12	9	24	11	37	11
v	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i~	2	12	2	9	2	2	0	3	5	2	2	0	0	5	4	7	7	2	6	7
z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
w	3	24	4	46	4	12	0	9	27	4	0	0	0	9	13	17	14	7	45	8
l~	3	48	3	40	2	9	0	6	26	4	0	0	0	3	17	9	17	4	28	8
r	6	106	6	166	15	34	0	17	81	17	3	0	0	19	32	50	41	4	106	32
w~	6	128	2	90	3	32	0	30	61	13	2	0	0	6	28	40	35	8	73	21
@	53	403	83	369	63	197	0	199	401	102	35	0	0	79	188	183	181	107	317	124
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
i	8	73	5	44	8	32	0	33	57	20	3	0	0	3	24	24	34	14	47	20
6	25	329	34	414	23	94	0	106	202	75	4	0	0	7	145	126	199	67	292	76
n	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
O	1	4	1	1	3	1	0	4	5	1	1	0	0	1	4	1	2	1	8	3
m	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
o~	3	12	3	24	3	7	0	2	12	7	0	0	0	2	4	6	11	3	7	4
l	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6~	1	4	1	2	1	1	0	3	0	0	1	0	0	1	1	1	1	2	4	2
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
o	1	19	4	21	5	7	0	3	26	1	1	0	0	2	4	17	14	4	14	4

Table F.13: Table with the total number of diphone occurrences, starting with vowel, considering vocalic reduction



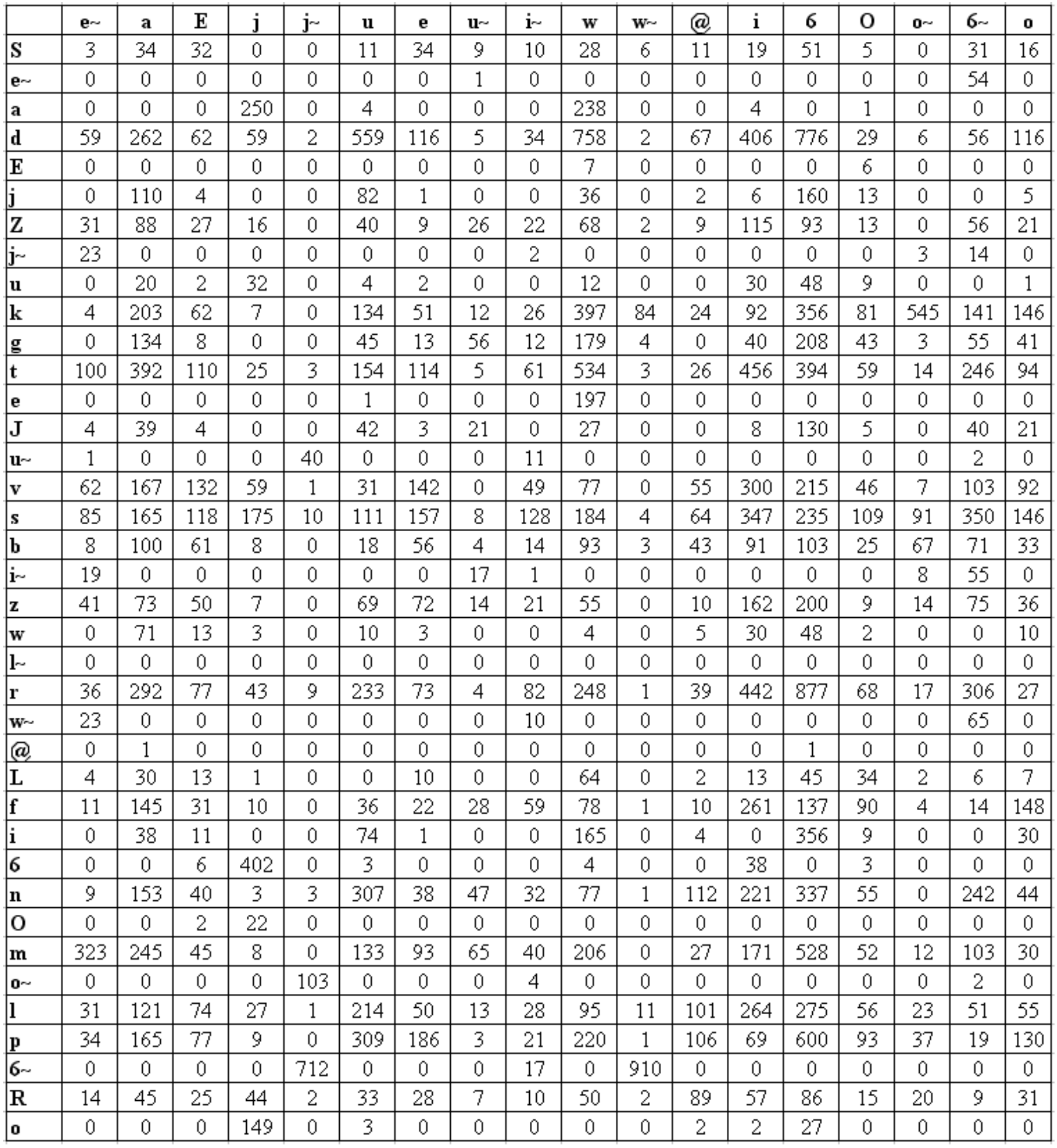
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	4	34	32	0	0	11	34	10	11	28	6	11	19	52	5	0	32	16
e~	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	54	0
a	2	10	6	250	0	8	3	11	9	238	0	0	13	15	6	3	5	5
d	68	287	68	64	2	581	125	27	44	758	2	67	431	818	37	10	64	126
E	4	14	6	1	0	37	6	23	16	8	0	0	15	23	14	3	3	5
j	4	113	4	1	0	95	5	4	12	36	0	2	15	177	16	4	5	7
Z	32	88	30	16	0	45	10	29	28	68	2	9	117	97	15	2	57	22
j~	31	16	4	0	0	25	4	12	10	0	0	0	25	38	7	6	24	6
u	19	43	13	39	0	39	6	8	31	12	0	0	79	144	35	3	16	24
k	15	220	113	7	0	210	81	16	40	398	84	24	111	457	96	547	152	160
g	3	137	13	0	0	49	14	58	14	179	4	0	43	213	44	3	56	42
t	105	407	115	29	3	184	118	11	69	534	3	26	491	437	60	15	256	101
e	3	4	3	1	0	8	2	4	8	197	0	0	5	2	3	1	4	2
J	5	39	5	0	0	45	4	24	4	27	0	0	11	132	6	1	40	22
u~	10	11	8	1	40	13	12	7	17	0	0	0	14	17	10	5	8	7
v	63	170	133	59	1	36	142	5	52	77	0	55	305	219	48	8	106	94
s	90	182	126	178	10	129	164	16	139	184	4	64	361	262	114	92	353	156
b	8	100	61	8	0	21	56	4	17	93	3	43	91	104	25	68	71	34
i~	24	10	12	1	0	16	6	26	13	0	0	0	10	9	7	12	60	8
z	73	140	77	22	0	125	82	27	67	55	0	12	366	379	38	24	127	66
w	11	129	49	6	0	89	17	19	16	6	0	5	145	148	10	5	34	28
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	53	341	89	46	9	417	83	35	97	248	2	39	479	1029	77	20	320	37
w~	34	27	20	4	1	51	3	11	20	0	0	0	34	68	4	1	80	15
@	2	2	3	0	0	3	4	2	2	0	0	0	3	2	2	3	1	1
L	6	32	14	1	0	17	11	7	8	64	0	2	21	66	37	5	7	9
f	12	146	32	10	0	38	23	28	61	78	1	10	264	138	91	5	15	148
i	12	52	29	0	0	118	10	12	25	165	0	4	26	437	19	5	11	38
6	27	60	51	407	0	132	20	51	51	5	0	0	186	219	30	14	42	20
n	9	153	41	3	3	308	38	47	33	77	1	112	221	337	55	0	243	44
O	6	5	6	22	0	9	2	5	9	0	0	0	2	4	3	2	3	5
m	325	247	48	9	0	140	95	66	44	206	0	27	176	538	55	13	104	32
o~	7	7	5	0	103	47	7	15	6	0	0	0	10	40	6	2	5	6
l	35	133	89	29	1	233	54	20	44	96	11	101	296	301	66	26	60	63
p	35	166	78	9	0	310	187	6	22	220	1	106	70	601	94	38	20	131
6~	0	2	0	0	712	8	3	2	20	0	910	0	3	2	3	0	1	2
R	15	45	25	44	2	34	30	9	12	50	2	89	57	88	16	21	10	33
o	8	10	5	149	0	34	3	12	9	0	0	2	11	55	6	2	9	6

Table F.14: Table with the total number of diphone occurrences, starting with consonant, considering vocalic reduction



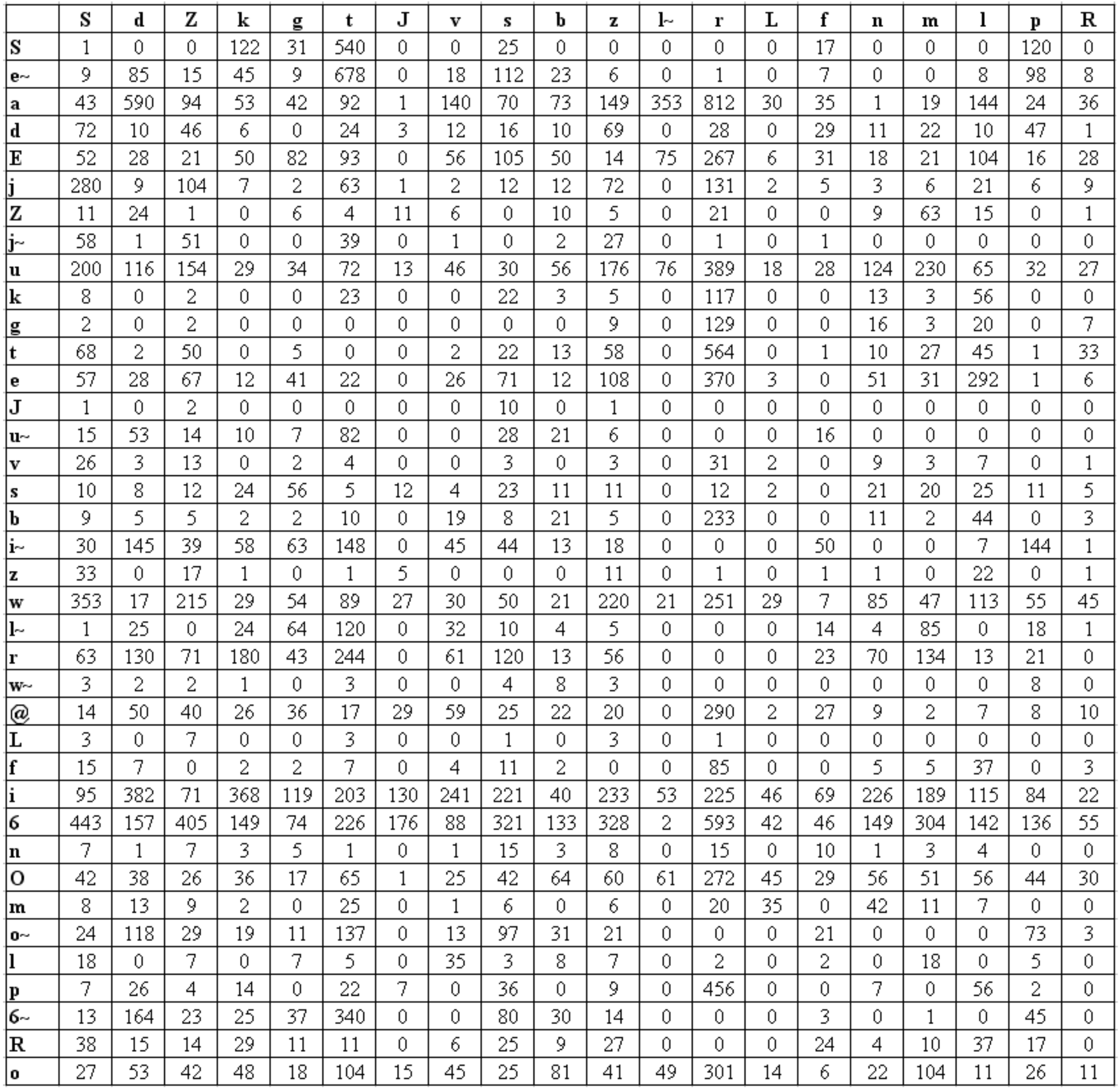
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	43	2	2	504	31	612	0	2	233	0	0	0	0	0	132	1	2	0	370	1
e~	9	85	15	45	9	678	0	18	112	23	6	0	1	0	7	0	0	8	98	8
a	53	606	97	78	45	105	1	149	89	79	152	353	812	36	47	17	46	161	49	45
d	89	83	64	122	16	76	3	31	82	31	74	0	28	4	65	39	75	32	99	21
E	59	48	24	64	85	103	0	64	122	58	14	75	267	8	39	25	36	106	30	32
j	287	36	104	34	5	70	1	3	30	16	72	0	131	16	14	15	14	25	22	14
Z	12	541	39	4	24	12	11	96	12	52	13	0	21	7	3	130	183	56	5	69
j~	68	53	59	130	8	73	0	24	51	17	29	0	1	8	24	27	23	9	43	11
u	245	237	197	359	67	136	13	82	136	99	182	76	389	20	81	163	320	100	174	91
k	41	79	19	59	9	70	0	90	191	21	6	0	117	21	46	80	42	92	92	28
g	4	9	5	5	3	6	0	5	7	4	13	0	129	4	5	17	7	21	6	9
t	74	72	54	44	10	23	0	9	58	21	61	0	564	3	20	29	49	49	53	38
e	58	30	68	18	42	24	0	27	75	13	108	0	370	6	3	53	32	299	4	8
J	2	4	5	3	2	4	0	4	14	4	4	0	0	3	4	2	2	3	3	3
u~	27	98	22	32	17	93	0	7	55	39	9	0	0	3	28	9	23	10	37	11
v	28	13	14	6	3	6	0	6	17	5	4	0	31	5	5	13	7	9	8	8
s	19	57	18	75	63	22	12	24	53	15	12	0	12	12	12	46	35	35	44	15
b	9	10	9	12	2	13	0	22	16	24	5	0	233	3	4	11	3	46	3	5
i~	32	157	41	67	64	150	0	48	49	14	19	0	0	2	54	6	7	9	151	7
z	37	20	17	12	3	8	5	5	10	8	15	0	1	4	7	8	5	24	12	8
w	378	322	237	295	65	132	27	79	158	41	221	21	251	47	67	173	122	129	282	76
l~	8	71	3	62	66	129	0	38	36	8	5	0	0	3	31	13	103	4	46	9
r	77	242	80	352	58	281	0	83	202	33	59	0	0	21	62	124	178	21	140	35
w~	23	130	4	87	3	36	0	30	65	21	5	0	0	6	28	40	35	8	78	21
@	16	60	42	34	39	19	29	61	36	24	23	0	290	9	31	12	8	11	17	13
L	5	25	11	11	3	12	0	4	8	8	5	0	1	2	16	6	7	5	8	9
f	16	11	4	5	4	8	0	6	13	4	2	0	85	1	5	7	6	38	2	4
i	106	453	76	411	128	235	130	273	273	60	237	53	225	52	92	250	222	128	129	43
6	498	488	440	558	97	321	176	195	521	208	332	2	593	49	189	276	500	208	427	129
n	8	2	7	4	5	1	0	1	16	3	8	0	15	0	10	3	4	4	1	2
O	45	43	26	37	20	66	1	29	47	65	60	61	272	46	33	57	53	57	52	33
m	10	30	10	8	1	28	0	4	15	5	7	0	20	37	8	48	15	10	12	5
o~	28	130	32	43	14	143	0	15	109	38	21	0	0	2	25	6	11	2	80	7
l	20	9	9	17	10	9	0	47	12	10	9	0	2	2	7	2	25	4	14	6
p	8	27	5	17	1	24	7	5	38	5	10	0	456	2	3	9	5	57	5	4
6~	13	167	23	27	38	341	0	3	80	30	15	0	0	1	4	1	2	2	49	2
R	39	17	15	34	12	12	0	6	28	10	28	0	0	3	25	6	11	39	18	1
o	29	70	46	70	23	111	15	48	51	82	42	49	301	16	10	39	118	15	40	15

Table F.15: Table with the number of diphone occurrences, starting with vowel, within the same word, considering vocalic reduction



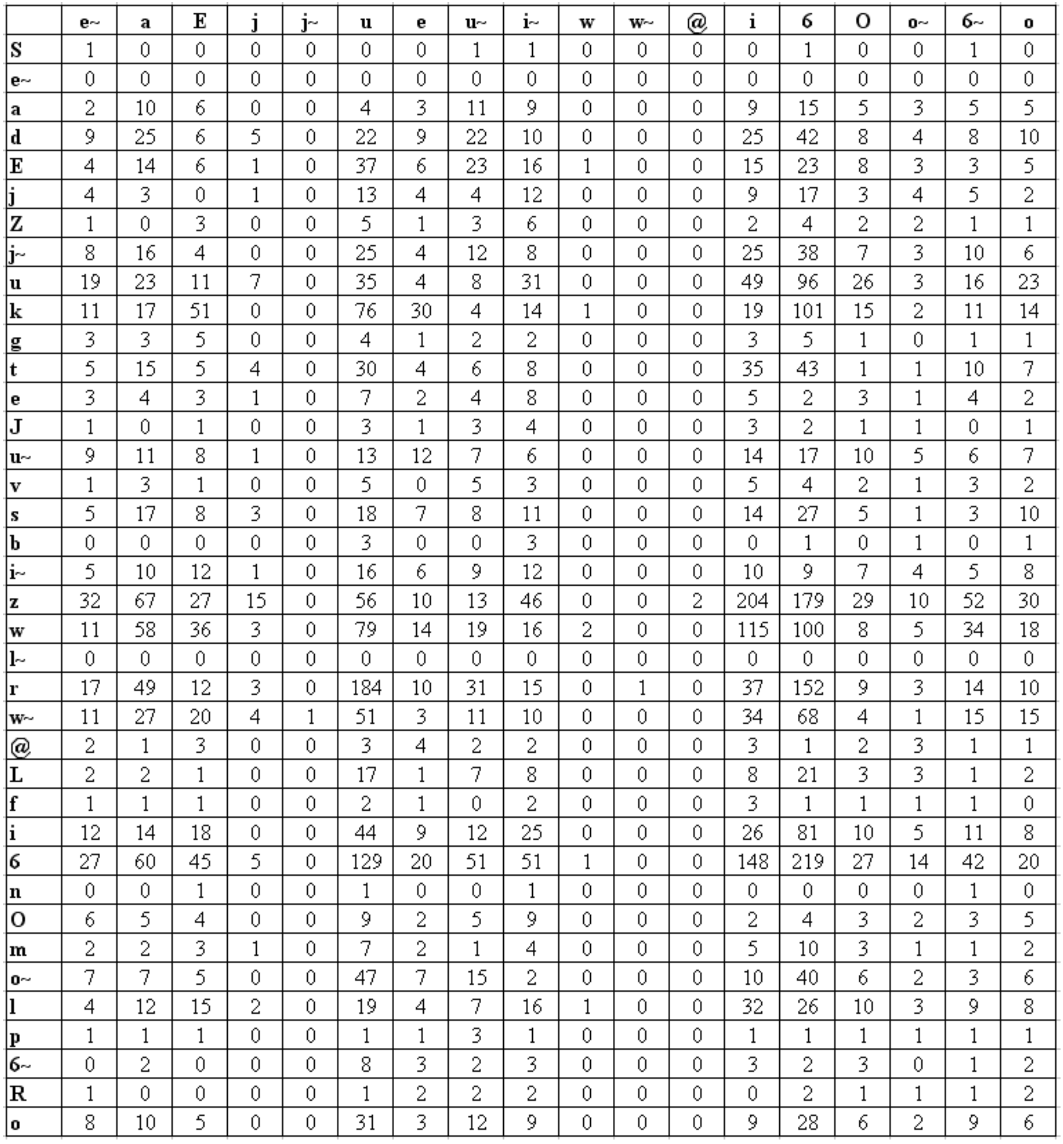
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	3	34	32	0	0	11	34	9	10	28	6	11	19	51	5	0	31	16
e~	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	54	0
a	0	0	0	250	0	4	0	0	0	238	0	0	4	0	1	0	0	0
d	59	262	62	59	2	559	116	5	34	758	2	67	406	776	29	6	56	116
E	0	0	0	0	0	0	0	0	0	7	0	0	0	0	6	0	0	0
j	0	110	4	0	0	82	1	0	0	36	0	2	6	160	13	0	0	5
Z	31	88	27	16	0	40	9	26	22	68	2	9	115	93	13	0	56	21
j~	23	0	0	0	0	0	0	0	2	0	0	0	0	0	0	3	14	0
u	0	20	2	32	0	4	2	0	0	12	0	0	30	48	9	0	0	1
k	4	203	62	7	0	134	51	12	26	397	84	24	92	356	81	545	141	146
g	0	134	8	0	0	45	13	56	12	179	4	0	40	208	43	3	55	41
t	100	392	110	25	3	154	114	5	61	534	3	26	456	394	59	14	246	94
e	0	0	0	0	0	1	0	0	0	197	0	0	0	0	0	0	0	0
J	4	39	4	0	0	42	3	21	0	27	0	0	8	130	5	0	40	21
u~	1	0	0	0	40	0	0	0	11	0	0	0	0	0	0	0	2	0
v	62	167	132	59	1	31	142	0	49	77	0	55	300	215	46	7	103	92
s	85	165	118	175	10	111	157	8	128	184	4	64	347	235	109	91	350	146
b	8	100	61	8	0	18	56	4	14	93	3	43	91	103	25	67	71	33
i~	19	0	0	0	0	0	0	17	1	0	0	0	0	0	0	8	55	0
z	41	73	50	7	0	69	72	14	21	55	0	10	162	200	9	14	75	36
w	0	71	13	3	0	10	3	0	0	4	0	5	30	48	2	0	0	10
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	36	292	77	43	9	233	73	4	82	248	1	39	442	877	68	17	306	27
w~	23	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	65	0
@	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
L	4	30	13	1	0	0	10	0	0	64	0	2	13	45	34	2	6	7
f	11	145	31	10	0	36	22	28	59	78	1	10	261	137	90	4	14	148
i	0	38	11	0	0	74	1	0	0	165	0	4	0	356	9	0	0	30
6	0	0	6	402	0	3	0	0	0	4	0	0	38	0	3	0	0	0
n	9	153	40	3	3	307	38	47	32	77	1	112	221	337	55	0	242	44
O	0	0	2	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0
m	323	245	45	8	0	133	93	65	40	206	0	27	171	528	52	12	103	30
o~	0	0	0	0	103	0	0	0	4	0	0	0	0	0	0	0	2	0
l	31	121	74	27	1	214	50	13	28	95	11	101	264	275	56	23	51	55
p	34	165	77	9	0	309	186	3	21	220	1	106	69	600	93	37	19	130
6~	0	0	0	0	712	0	0	0	17	0	910	0	0	0	0	0	0	0
R	14	45	25	44	2	33	28	7	10	50	2	89	57	86	15	20	9	31
o	0	0	0	149	0	3	0	0	0	0	0	2	2	27	0	0	0	0

Table F.16: Table with the number of diphone occurrences, starting with consonant, within the same word, considering vocalic reduction



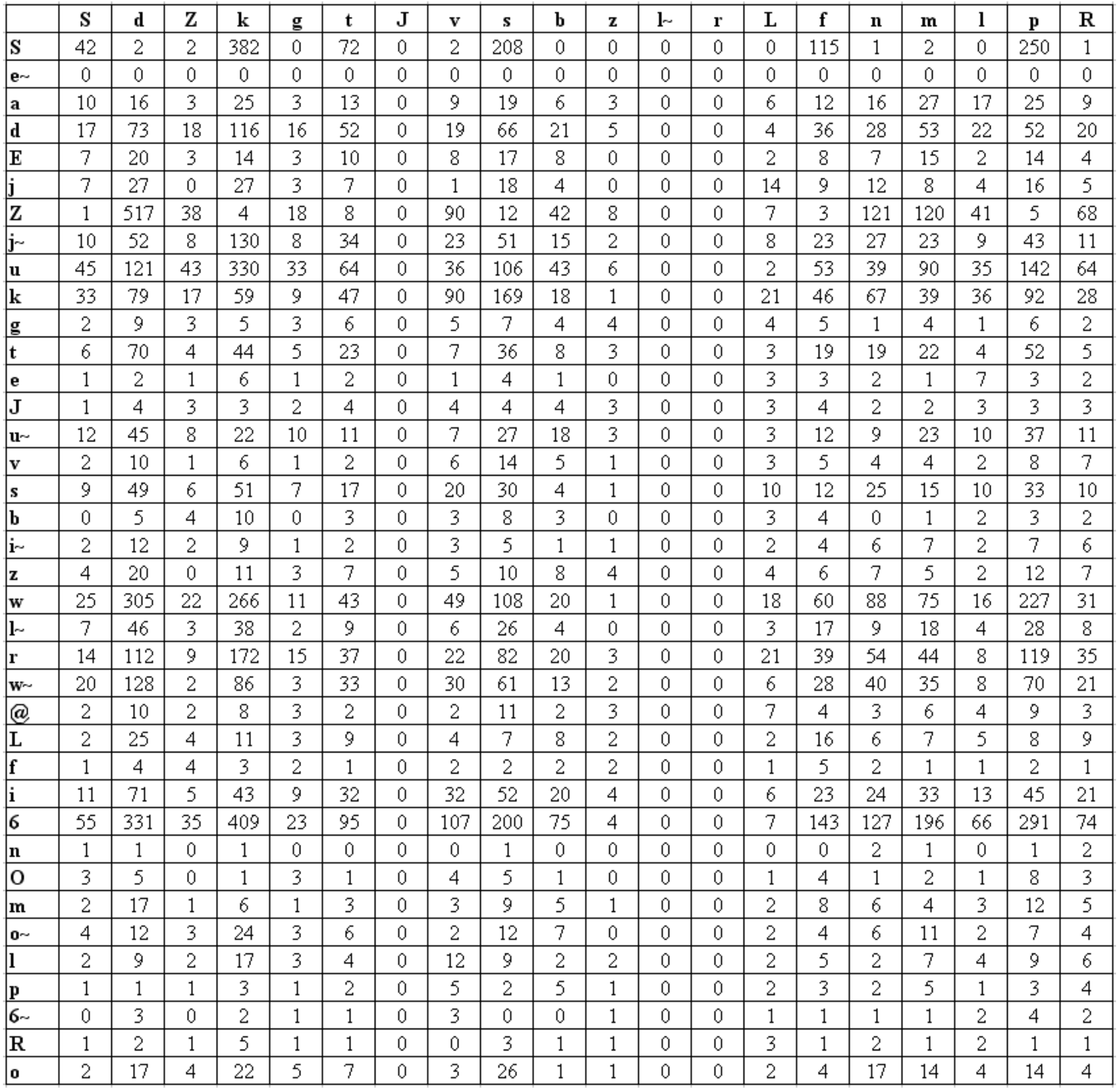
	S	**d**	**Z**	**k**	**g**	**t**	**J**	**v**	**s**	**b**	**z**	**l~**	**r**	**L**	**f**	**n**	**m**	**l**	**p**	**R**
S	1	0	0	122	31	540	0	0	25	0	0	0	0	0	17	0	0	0	120	0
e~	9	85	15	45	9	678	0	18	112	23	6	0	1	0	7	0	0	8	98	8
a	43	590	94	53	42	92	1	140	70	73	149	353	812	30	35	1	19	144	24	36
d	72	10	46	6	0	24	3	12	16	10	69	0	28	0	29	11	22	10	47	1
E	52	28	21	50	82	93	0	56	105	50	14	75	267	6	31	18	21	104	16	28
j	280	9	104	7	2	63	1	2	12	12	72	0	131	2	5	3	6	21	6	9
Z	11	24	1	0	6	4	11	6	0	10	5	0	21	0	0	9	63	15	0	1
j~	58	1	51	0	0	39	0	1	0	2	27	0	1	0	1	0	0	0	0	0
u	200	116	154	29	34	72	13	46	30	56	176	76	389	18	28	124	230	65	32	27
k	8	0	2	0	0	23	0	0	22	3	5	0	117	0	0	13	3	56	0	0
g	2	0	2	0	0	0	0	0	0	0	9	0	129	0	0	16	3	20	0	7
t	68	2	50	0	5	0	0	2	22	13	58	0	564	0	1	10	27	45	1	33
e	57	28	67	12	41	22	0	26	71	12	108	0	370	3	0	51	31	292	1	6
J	1	0	2	0	0	0	0	0	10	0	1	0	0	0	0	0	0	0	0	0
u~	15	53	14	10	7	82	0	0	28	21	6	0	0	0	16	0	0	0	0	0
v	26	3	13	0	2	4	0	0	3	0	3	0	31	2	0	9	3	7	0	1
s	10	8	12	24	56	5	12	4	23	11	11	0	12	2	0	21	20	25	11	5
b	9	5	5	2	2	10	0	19	8	21	5	0	233	0	0	11	2	44	0	3
i~	30	145	39	58	63	148	0	45	44	13	18	0	0	0	50	0	0	7	144	1
z	33	0	17	1	0	1	5	0	0	0	11	0	1	0	1	1	0	22	0	1
w	353	17	215	29	54	89	27	30	50	21	220	21	251	29	7	85	47	113	55	45
l~	1	25	0	24	64	120	0	32	10	4	5	0	0	0	14	4	85	0	18	1
r	63	130	71	180	43	244	0	61	120	13	56	0	0	0	23	70	134	13	21	0
w~	3	2	2	1	0	3	0	0	4	8	3	0	0	0	0	0	0	0	8	0
@	14	50	40	26	36	17	29	59	25	22	20	0	290	2	27	9	2	7	8	10
L	3	0	7	0	0	3	0	0	1	0	3	0	1	0	0	0	0	0	0	0
f	15	7	0	2	2	7	0	4	11	2	0	0	85	0	0	5	5	37	0	3
i	95	382	71	368	119	203	130	241	221	40	233	53	225	46	69	226	189	115	84	22
6	443	157	405	149	74	226	176	88	321	133	328	2	593	42	46	149	304	142	136	55
n	7	1	7	3	5	1	0	1	15	3	8	0	15	0	10	1	3	4	0	0
O	42	38	26	36	17	65	1	25	42	64	60	61	272	45	29	56	51	56	44	30
m	8	13	9	2	0	25	0	1	6	0	6	0	20	35	0	42	11	7	0	0
o~	24	118	29	19	11	137	0	13	97	31	21	0	0	0	21	0	0	0	73	3
l	18	0	7	0	7	5	0	35	3	8	7	0	2	0	2	0	18	0	5	0
p	7	26	4	14	0	22	7	0	36	0	9	0	456	0	0	7	0	56	2	0
6~	13	164	23	25	37	340	0	0	80	30	14	0	0	0	3	0	1	0	45	0
R	38	15	14	29	11	11	0	6	25	9	27	0	0	0	24	4	10	37	17	0
o	27	53	42	48	18	104	15	45	25	81	41	49	301	14	6	22	104	11	26	11

Table F.17: Table with the number of diphone occurrences, starting with vowel, between words, considering vocalic reduction



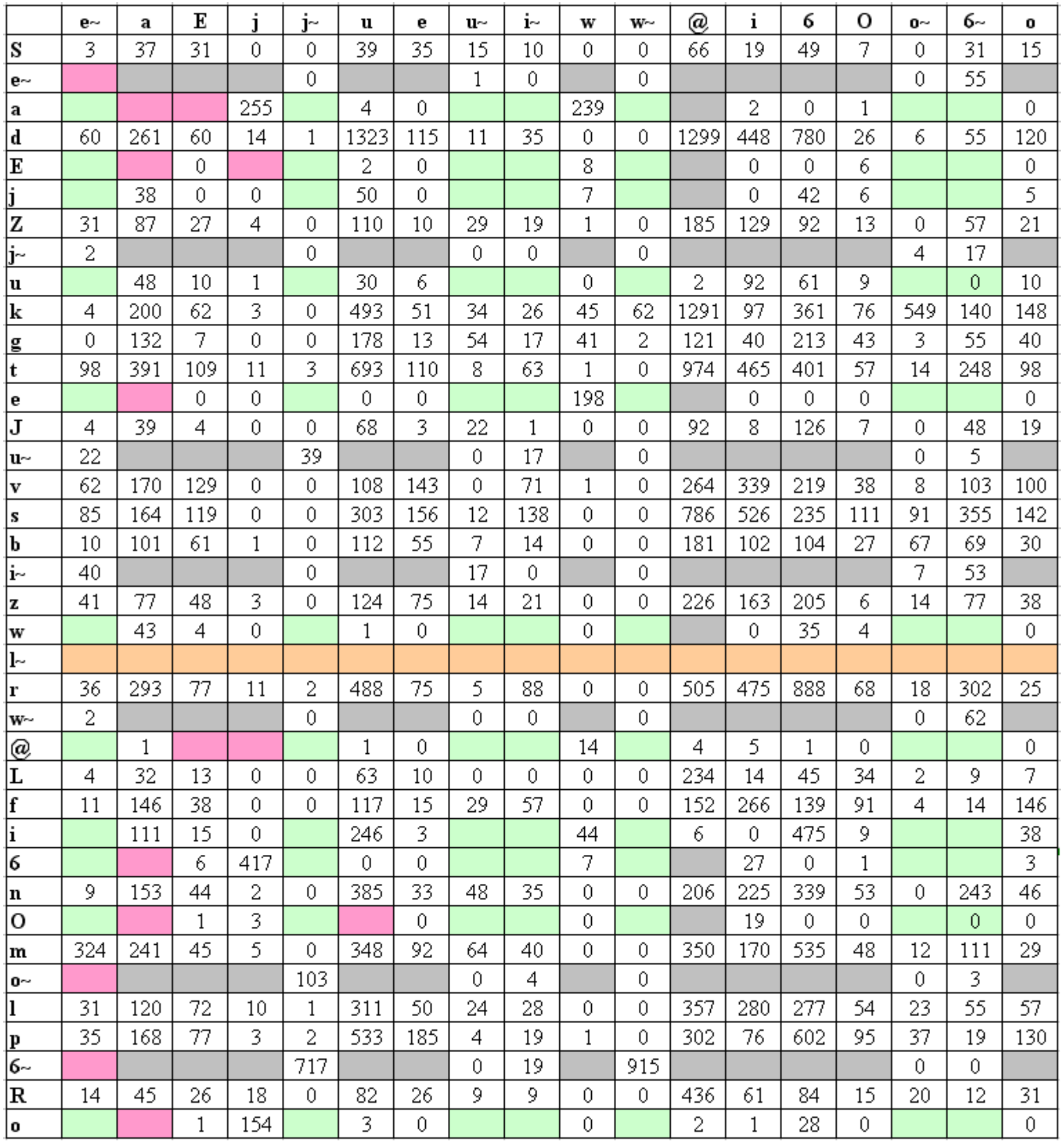
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	1	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	1	0
e~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	2	10	6	0	0	4	3	11	9	0	0	0	9	15	5	3	5	5
d	9	25	6	5	0	22	9	22	10	0	0	0	25	42	8	4	8	10
E	4	14	6	1	0	37	6	23	16	1	0	0	15	23	8	3	3	5
j	4	3	0	1	0	13	4	4	12	0	0	0	9	17	3	4	5	2
Z	1	0	3	0	0	5	1	3	6	0	0	0	2	4	2	2	1	1
j~	8	16	4	0	0	25	4	12	8	0	0	0	25	38	7	3	10	6
u	19	23	11	7	0	35	4	8	31	0	0	0	49	96	26	3	16	23
k	11	17	51	0	0	76	30	4	14	1	0	0	19	101	15	2	11	14
g	3	3	5	0	0	4	1	2	2	0	0	0	3	5	1	0	1	1
t	5	15	5	4	0	30	4	6	8	0	0	0	35	43	1	1	10	7
e	3	4	3	1	0	7	2	4	8	0	0	0	5	2	3	1	4	2
J	1	0	1	0	0	3	1	3	4	0	0	0	3	2	1	1	0	1
u~	9	11	8	1	0	13	12	7	6	0	0	0	14	17	10	5	6	7
v	1	3	1	0	0	5	0	5	3	0	0	0	5	4	2	1	3	2
s	5	17	8	3	0	18	7	8	11	0	0	0	14	27	5	1	3	10
b	0	0	0	0	0	3	0	0	3	0	0	0	0	1	0	1	0	1
i~	5	10	12	1	0	16	6	9	12	0	0	0	10	9	7	4	5	8
z	32	67	27	15	0	56	10	13	46	0	0	2	204	179	29	10	52	30
w	11	58	36	3	0	79	14	19	16	2	0	0	115	100	8	5	34	18
l~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
r	17	49	12	3	0	184	10	31	15	0	1	0	37	152	9	3	14	10
w~	11	27	20	4	1	51	3	11	10	0	0	0	34	68	4	1	15	15
@	2	1	3	0	0	3	4	2	2	0	0	0	3	1	2	3	1	1
L	2	2	1	0	0	17	1	7	8	0	0	0	8	21	3	3	1	2
f	1	1	1	0	0	2	1	0	2	0	0	0	3	1	1	1	1	0
i	12	14	18	0	0	44	9	12	25	0	0	0	26	81	10	5	11	8
6	27	60	45	5	0	129	20	51	51	1	0	0	148	219	27	14	42	20
n	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0
O	6	5	4	0	0	9	2	5	9	0	0	0	2	4	3	2	3	5
m	2	2	3	1	0	7	2	1	4	0	0	0	5	10	3	1	1	2
o~	7	7	5	0	0	47	7	15	2	0	0	0	10	40	6	2	3	6
l	4	12	15	2	0	19	4	7	16	1	0	0	32	26	10	3	9	8
p	1	1	1	0	0	1	1	3	1	0	0	0	1	1	1	1	1	1
6~	0	2	0	0	0	8	3	2	3	0	0	0	3	2	3	0	1	2
R	1	0	0	0	0	1	2	2	2	0	0	0	0	2	1	1	1	2
o	8	10	5	0	0	31	3	12	9	0	0	0	9	28	6	2	9	6

Table F.18: Table with the number of diphone occurrences, starting with consonant, between words, considering vocalic reduction



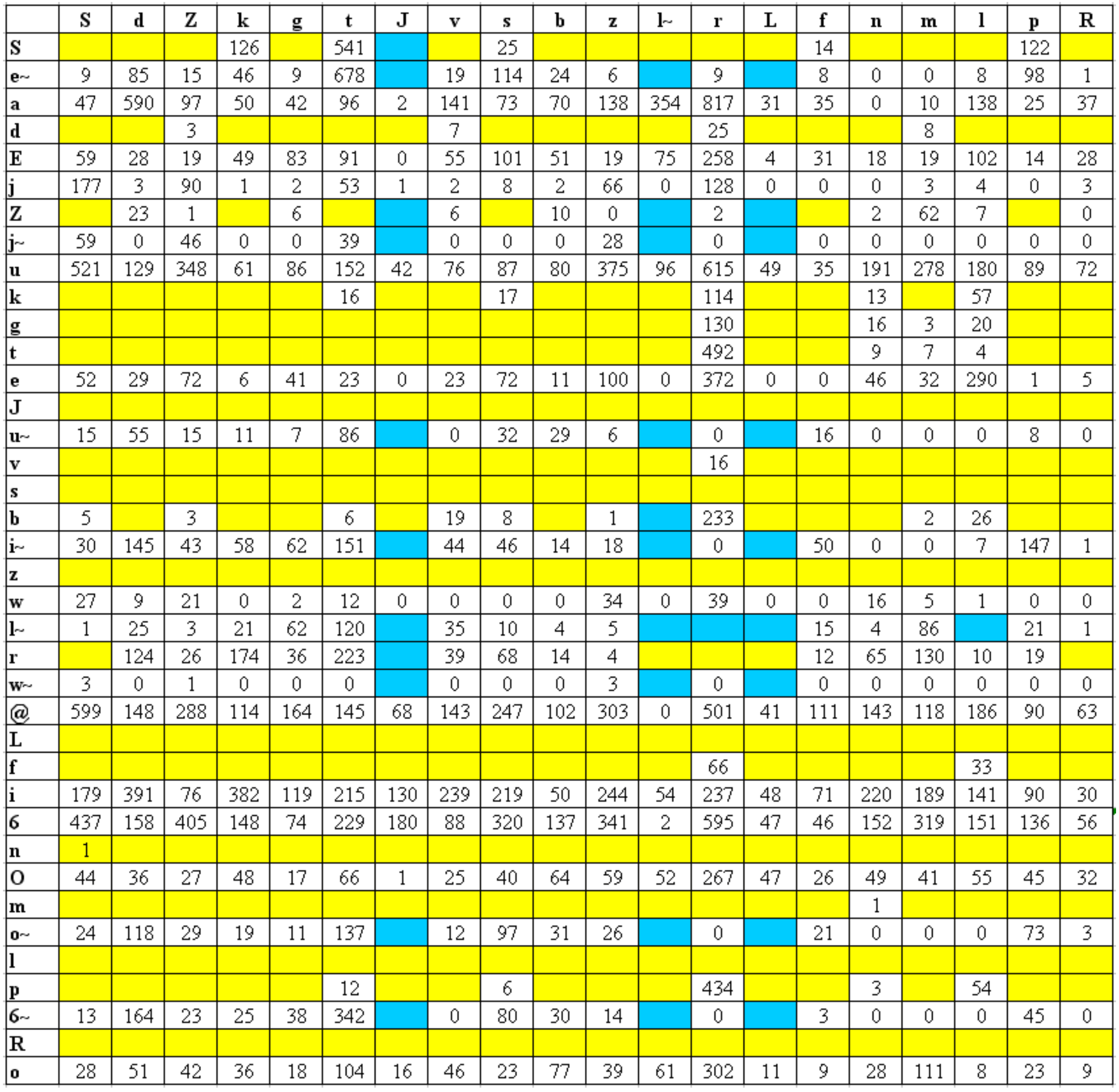
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	42	2	2	382	0	72	0	2	208	0	0	0	0	0	115	1	2	0	250	1
e~	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	10	16	3	25	3	13	0	9	19	6	3	0	0	6	12	16	27	17	25	9
d	17	73	18	116	16	52	0	19	66	21	5	0	0	4	36	28	53	22	52	20
E	7	20	3	14	3	10	0	8	17	8	0	0	0	2	8	7	15	2	14	4
j	7	27	0	27	3	7	0	1	18	4	0	0	0	14	9	12	8	4	16	5
Z	1	517	38	4	18	8	0	90	12	42	8	0	0	7	3	121	120	41	5	68
j~	10	52	8	130	8	34	0	23	51	15	2	0	0	8	23	27	23	9	43	11
u	45	121	43	330	33	64	0	36	106	43	6	0	0	2	53	39	90	35	142	64
k	33	79	17	59	9	47	0	90	169	18	1	0	0	21	46	67	39	36	92	28
g	2	9	3	5	3	6	0	5	7	4	4	0	0	4	5	1	4	1	6	2
t	6	70	4	44	5	23	0	7	36	8	3	0	0	3	19	19	22	4	52	5
e	1	2	1	6	1	2	0	1	4	1	0	0	0	3	3	2	1	7	3	2
J	1	4	3	3	2	4	0	4	4	4	3	0	0	3	4	2	2	3	3	3
u~	12	45	8	22	10	11	0	7	27	18	3	0	0	3	12	9	23	10	37	11
v	2	10	1	6	1	2	0	6	14	5	1	0	0	3	5	4	4	2	8	7
s	9	49	6	51	7	17	0	20	30	4	1	0	0	10	12	25	15	10	33	10
b	0	5	4	10	0	3	0	3	8	3	0	0	0	3	4	0	1	2	3	2
i~	2	12	2	9	1	2	0	3	5	1	1	0	0	2	4	6	7	2	7	6
z	4	20	0	11	3	7	0	5	10	8	4	0	0	4	6	7	5	2	12	7
w	25	305	22	266	11	43	0	49	108	20	1	0	0	18	60	88	75	16	227	31
l~	7	46	3	38	2	9	0	6	26	4	0	0	0	3	17	9	18	4	28	8
r	14	112	9	172	15	37	0	22	82	20	3	0	0	21	39	54	44	8	119	35
w~	20	128	2	86	3	33	0	30	61	13	2	0	0	6	28	40	35	8	70	21
@	2	10	2	8	3	2	0	2	11	2	3	0	0	7	4	3	6	4	9	3
L	2	25	4	11	3	9	0	4	7	8	2	0	0	2	16	6	7	5	8	9
f	1	4	4	3	2	1	0	2	2	2	2	0	0	1	5	2	1	1	2	1
i	11	71	5	43	9	32	0	32	52	20	4	0	0	6	23	24	33	13	45	21
6	55	331	35	409	23	95	0	107	200	75	4	0	0	7	143	127	196	66	291	74
n	1	1	0	1	0	0	0	0	1	0	0	0	0	0	0	2	1	0	1	2
O	3	5	0	1	3	1	0	4	5	1	0	0	0	1	4	1	2	1	8	3
m	2	17	1	6	1	3	0	3	9	5	1	0	0	2	8	6	4	3	12	5
o~	4	12	3	24	3	6	0	2	12	7	0	0	0	2	4	6	11	2	7	4
l	2	9	2	17	3	4	0	12	9	2	2	0	0	2	5	2	7	4	9	6
p	1	1	1	3	1	2	0	5	2	5	1	0	0	2	3	2	5	1	3	4
6~	0	3	0	2	1	1	0	3	0	0	1	0	0	1	1	1	1	2	4	2
R	1	2	1	5	1	1	0	0	3	1	1	0	0	3	1	2	1	2	1	1
o	2	17	4	22	5	7	0	3	26	1	1	0	0	2	4	17	14	4	14	4

Table F.19: Table with the number of diphone occurrences, starting with vowel, within the same word, following the rules, color highlighted



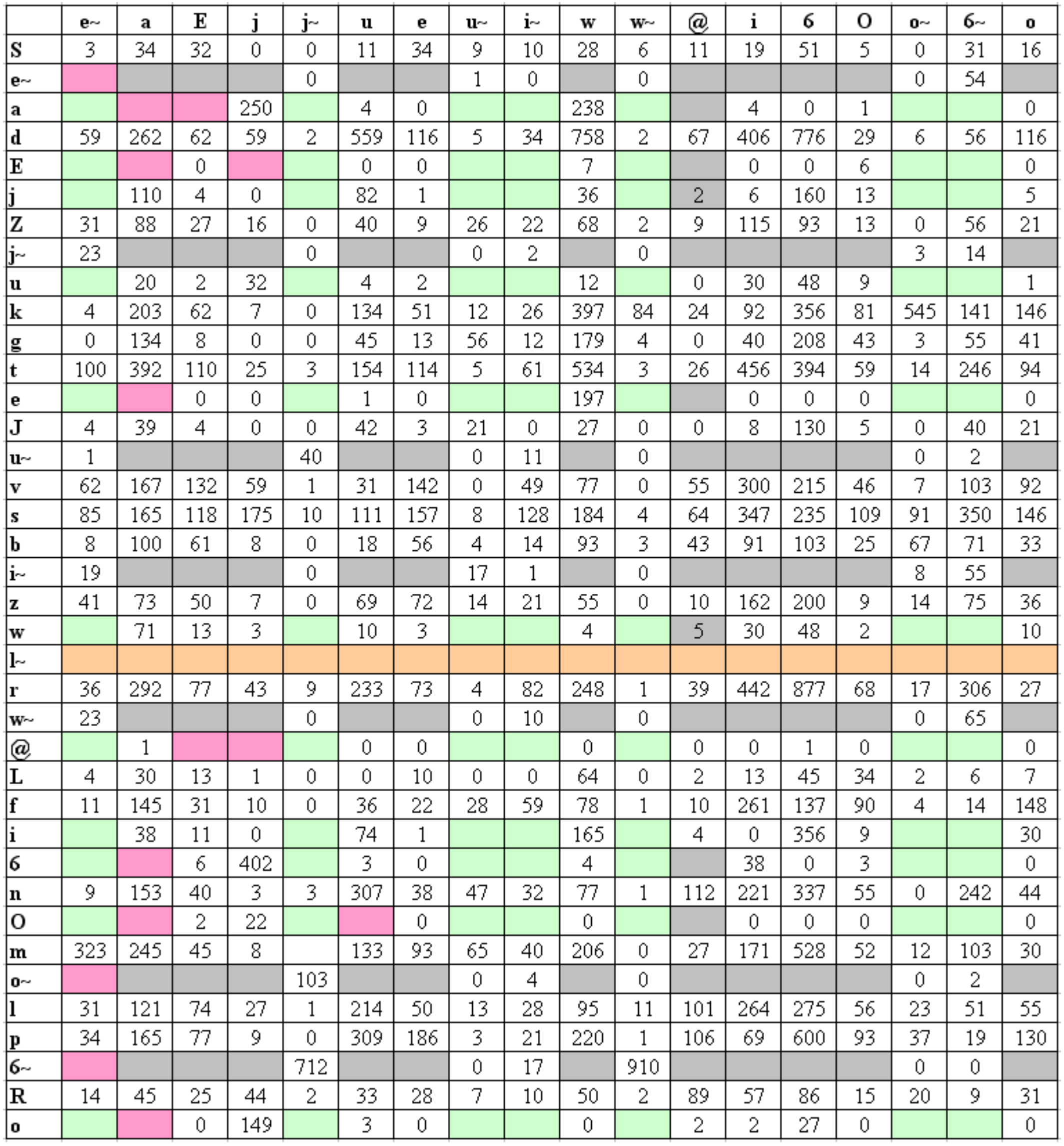
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	3	37	31	0	0	39	35	15	10	0	0	66	19	49	7	0	31	15
e~	🟥	▨	▨	▨	0	▨	▨	1	0	▨	0	▨	▨	▨	▨	0	55	▨
a	🟩	🟥	🟥	255	🟩	4	0	🟩	🟩	239	🟩	▨	2	0	1	🟩	🟩	0
d	60	261	60	14	1	1323	115	11	35	0	0	1299	448	780	26	6	55	120
E	🟩	🟥	0	🟥	🟩	2	0	🟩	🟩	8	🟩	▨	0	0	6	🟩	🟩	0
j	🟩	38	0	0	🟩	50	0	🟩	🟩	7	🟩	▨	0	42	6	🟩	🟩	5
Z	31	87	27	4	0	110	10	29	19	1	0	185	129	92	13	0	57	21
j~	2	▨	▨	▨	0	▨	▨	0	0	▨	0	▨	▨	▨	▨	4	17	▨
u	🟩	48	10	1	🟩	30	6	🟩	🟩	0	🟩	2	92	61	9	🟩	0	10
k	4	200	62	3	0	493	51	34	26	45	62	1291	97	361	76	549	140	148
g	0	132	7	0	0	178	13	54	17	41	2	121	40	213	43	3	55	40
t	98	391	109	11	3	693	110	8	63	1	0	974	465	401	57	14	248	98
e	🟩	🟥	0	0	🟩	0	0	🟩	🟩	198	🟩	▨	0	0	0	🟩	🟩	0
J	4	39	4	0	0	68	3	22	1	0	0	92	8	126	7	0	48	19
u~	22	▨	▨	▨	39	▨	▨	0	17	▨	0	▨	▨	▨	▨	0	5	▨
v	62	170	129	0	0	108	143	0	71	1	0	264	339	219	38	8	103	100
s	85	164	119	0	0	303	156	12	138	0	0	786	526	235	111	91	355	142
b	10	101	61	1	0	112	55	7	14	0	0	181	102	104	27	67	69	30
i~	40	▨	▨	▨	0	▨	▨	17	0	▨	0	▨	▨	▨	▨	7	53	▨
z	41	77	48	3	0	124	75	14	21	0	0	226	163	205	6	14	77	38
w	🟩	43	4	0	🟩	1	0	🟩	🟩	0	🟩	▨	0	35	4	🟩	🟩	0
l~	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧
r	36	293	77	11	2	488	75	5	88	0	0	505	475	888	68	18	302	25
w~	2	▨	▨	▨	0	▨	▨	0	0	▨	0	▨	▨	▨	▨	0	62	▨
@	🟩	1	🟥	🟥	🟩	1	0	🟩	🟩	14	🟩	4	5	1	0	🟩	🟩	0
L	4	32	13	0	0	63	10	0	0	0	0	234	14	45	34	2	9	7
f	11	146	38	0	0	117	15	29	57	0	0	152	266	139	91	4	14	146
i	🟩	111	15	0	🟩	246	3	🟩	🟩	44	🟩	6	0	475	9	🟩	🟩	38
6	🟩	🟥	6	417	🟩	0	0	🟩	🟩	7	🟩	▨	27	0	1	🟩	🟩	3
n	9	153	44	2	0	385	33	48	35	0	0	206	225	339	53	0	243	46
O	🟩	🟥	1	3	🟩	🟥	0	🟩	🟩	0	🟩	▨	19	0	0	🟩	0	0
m	324	241	45	5	0	348	92	64	40	0	0	350	170	535	48	12	111	29
o~	🟥	▨	▨	▨	103	▨	▨	0	4	▨	0	▨	▨	▨	▨	0	3	▨
l	31	120	72	10	1	311	50	24	28	0	0	357	280	277	54	23	55	57
p	35	168	77	3	2	533	185	4	19	1	0	302	76	602	95	37	19	130
6~	🟥	▨	▨	▨	717	▨	▨	0	19	▨	915	▨	▨	▨	▨	0	0	▨
R	14	45	26	18	0	82	26	9	9	0	0	436	61	84	15	20	12	31
o	🟩	🟥	1	154	🟩	3	0	🟩	🟩	0	🟩	2	1	28	0	🟩	🟩	0

Table F.20: Table with the number of diphone occurrences, starting with consonant, within the same word, following the rules, color highlighted



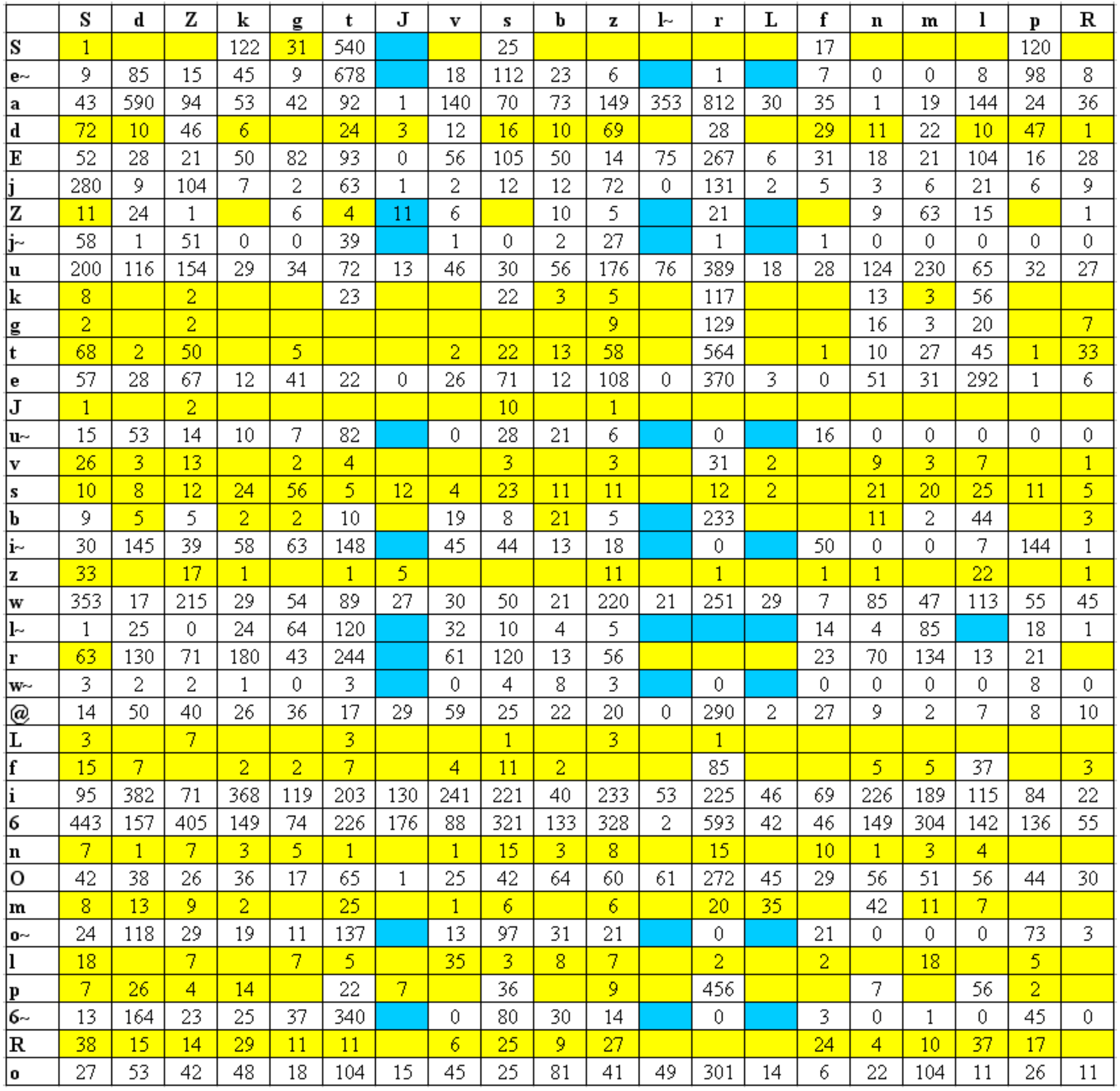
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S				126		541			25						14				122	
e~	9	85	15	46	9	678		19	114	24	6		9		8	0	0	8	98	1
a	47	590	97	50	42	96	2	141	73	70	138	354	817	31	35	0	10	138	25	37
d			3					7					25				8			
E	59	28	19	49	83	91	0	55	101	51	19	75	258	4	31	18	19	102	14	28
j	177	3	90	1	2	53	1	2	8	2	66	0	128	0	0	0	3	4	0	3
Z		23	1		6			6		10	0		2			2	62	7		0
j~	59	0	46	0	0	39		0	0	0	28		0		0	0	0	0	0	0
u	521	129	348	61	86	152	42	76	87	80	375	96	615	49	35	191	278	180	89	72
k						16			17				114			13		57		
g													130			16	3	20		
t													492			9	7	4		
e	52	29	72	6	41	23	0	23	72	11	100	0	372	0	0	46	32	290	1	5
J																				
u~	15	55	15	11	7	86		0	32	29	6		0		16	0	0	0	8	0
v													16							
s																				
b	5		3			6		19	8		1		233				2	26		
i~	30	145	43	58	62	151		44	46	14	18		0		50	0	0	7	147	1
z																				
w	27	9	21	0	2	12	0	0	0	0	34	0	39	0	0	16	5	1	0	0
l~	1	25	3	21	62	120		35	10	4	5				15	4	86		21	1
r		124	26	174	36	223		39	68	14	4				12	65	130	10	19	
w~	3	0	1	0	0	0		0	0	0	3		0		0	0	0	0	0	0
@	599	148	288	114	164	145	68	143	247	102	303	0	501	41	111	143	118	186	90	63
L																				
f													66					33		
i	179	391	76	382	119	215	130	239	219	50	244	54	237	48	71	220	189	141	90	30
6	437	158	405	148	74	229	180	88	320	137	341	2	595	47	46	152	319	151	136	56
n	1																			
O	44	36	27	48	17	66	1	25	40	64	59	52	267	47	26	49	41	55	45	32
m																1				
o~	24	118	29	19	11	137		12	97	31	26		0		21	0	0	0	73	3
l																				
p						12			6				434			3		54		
6~	13	164	23	25	38	342		0	80	30	14		0		3	0	0	0	45	0
R																				
o	28	51	42	36	18	104	16	46	23	77	39	61	302	11	9	28	111	8	23	9

Table F.21: Table with the number of diphone occurrences, starting with vowel, within the same word, considering vocalic reduction, color highlighted



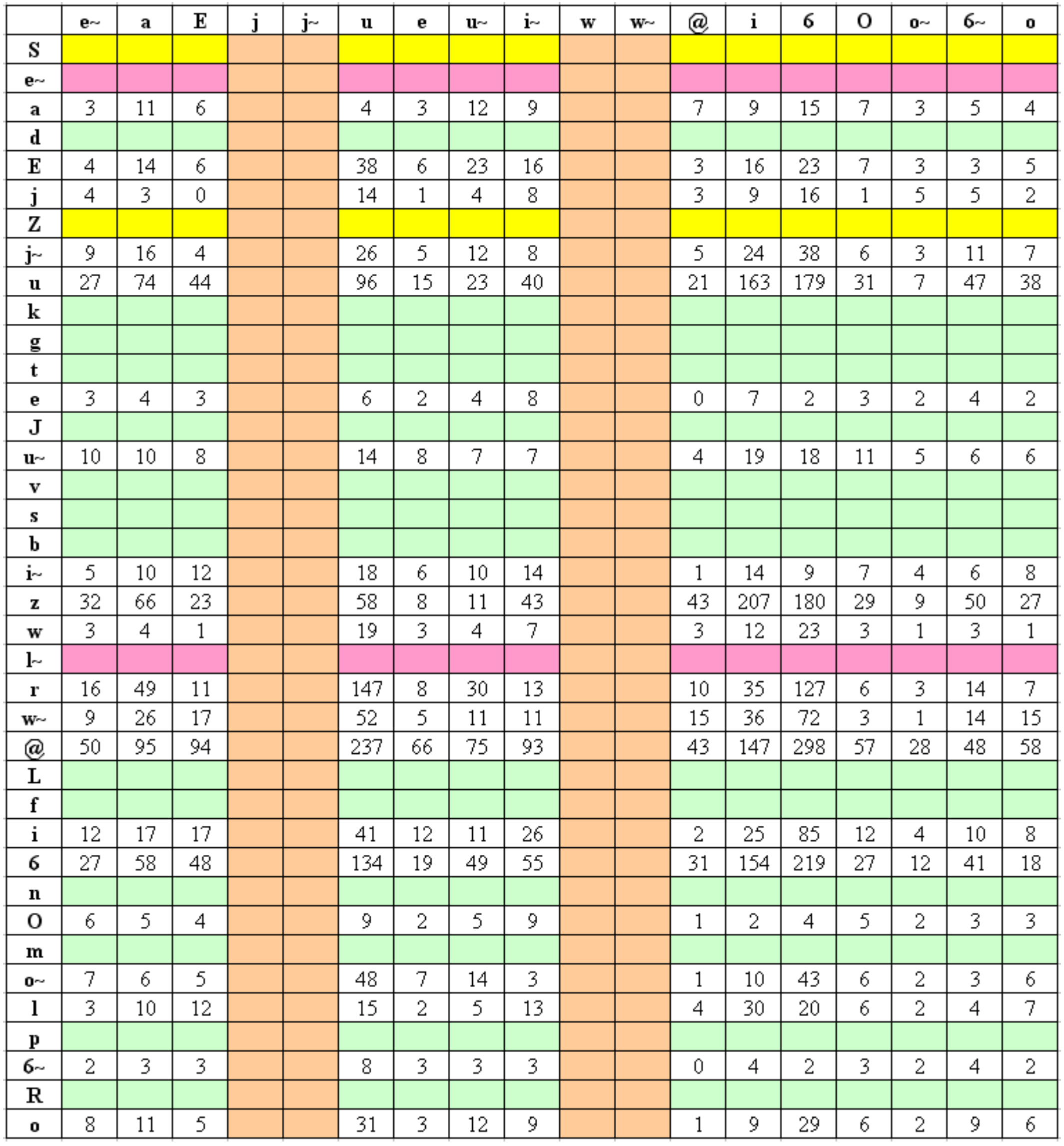
	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	3	34	32	0	0	11	34	9	10	28	6	11	19	51	5	0	31	16
e~	🟥	▨	▨	▨	0	▨	▨	1	0	▨	0	▨	▨	▨	▨	0	54	▨
a	🟩	🟥	🟥	250	🟩	4	0	🟩	🟩	238	🟩	▨	4	0	1	🟩	🟩	0
d	59	262	62	59	2	559	116	5	34	758	2	67	406	776	29	6	56	116
E	🟩	🟥	0	🟥	🟩	0	0	🟩	🟩	7	🟩	▨	0	0	6	🟩	🟩	0
j	🟩	110	4	0	🟩	82	1	🟩	🟩	36	🟩	2	6	160	13	🟩	🟩	5
Z	31	88	27	16	0	40	9	26	22	68	2	9	115	93	13	0	56	21
j~	23	▨	▨	▨	0	▨	▨	0	2	▨	0	▨	▨	▨	▨	3	14	▨
u	🟩	20	2	32	🟩	4	2	🟩	🟩	12	🟩	0	30	48	9	🟩	🟩	1
k	4	203	62	7	0	134	51	12	26	397	84	24	92	356	81	545	141	146
g	0	134	8	0	0	45	13	56	12	179	4	0	40	208	43	3	55	41
t	100	392	110	25	3	154	114	5	61	534	3	26	456	394	59	14	246	94
e	🟩	🟥	0	0	🟩	1	0	🟩	🟩	197	🟩	▨	0	0	0	🟩	🟩	0
J	4	39	4	0	0	42	3	21	0	27	0	0	8	130	5	0	40	21
u~	1	▨	▨	▨	40	▨	▨	0	11	▨	0	▨	▨	▨	▨	0	2	▨
v	62	167	132	59	1	31	142	0	49	77	0	55	300	215	46	7	103	92
s	85	165	118	175	10	111	157	8	128	184	4	64	347	235	109	91	350	146
b	8	100	61	8	0	18	56	4	14	93	3	43	91	103	25	67	71	33
i~	19	▨	▨	▨	0	▨	▨	17	1	▨	0	▨	▨	▨	▨	8	55	▨
z	41	73	50	7	0	69	72	14	21	55	0	10	162	200	9	14	75	36
w	🟩	71	13	3	🟩	10	3	🟩	🟩	4	🟩	5	30	48	2	🟩	🟩	10
l~	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧
r	36	292	77	43	9	233	73	4	82	248	1	39	442	877	68	17	306	27
w~	23	▨	▨	▨	0	▨	▨	0	10	▨	0	▨	▨	▨	▨	0	65	▨
@	🟩	1	🟥	🟥	🟩	0	0	🟩	🟩	0	🟩	0	0	1	0	🟩	🟩	0
L	4	30	13	1	0	0	10	0	0	64	0	2	13	45	34	2	6	7
f	11	145	31	10	0	36	22	28	59	78	1	10	261	137	90	4	14	148
i	🟩	38	11	0	🟩	74	1	🟩	🟩	165	🟩	4	0	356	9	🟩	🟩	30
6	🟩	🟥	6	402	🟩	3	0	🟩	🟩	4	🟩	▨	38	0	3	🟩	🟩	0
n	9	153	40	3	3	307	38	47	32	77	1	112	221	337	55	0	242	44
O	🟩	🟥	2	22	🟩	🟥	0	🟩	🟩	0	🟩	▨	0	0	0	🟩	🟩	0
m	323	245	45	8		133	93	65	40	206	0	27	171	528	52	12	103	30
o~	🟥	▨	▨	▨	103	▨	▨	0	4	▨	0	▨	▨	▨	▨	0	2	▨
l	31	121	74	27	1	214	50	13	28	95	11	101	264	275	56	23	51	55
p	34	165	77	9	0	309	186	3	21	220	1	106	69	600	93	37	19	130
6~	🟥	▨	▨	▨	712	▨	▨	0	17	▨	910	▨	▨	▨	▨	0	0	▨
R	14	45	25	44	2	33	28	7	10	50	2	89	57	86	15	20	9	31
o	🟩	🟥	0	149	🟩	3	0	🟩	🟩	0	🟩	2	2	27	0	🟩	🟩	0

Table F.22: Table with the number of diphone occurrences, starting with consonant, within the same word, considering vocalic reduction, color highlighted



	S	**d**	**Z**	**k**	**g**	**t**	**J**	**v**	**s**	**b**	**z**	**l~**	**r**	**L**	**f**	**n**	**m**	**l**	**p**	**R**
S	1			122	31	540			25						17				120	
e~	9	85	15	45	9	678		18	112	23	6		1		7	0	0	8	98	8
a	43	590	94	53	42	92	1	140	70	73	149	353	812	30	35	1	19	144	24	36
d	72	10	46	6		24	3	12	16	10	69		28		29	11	22	10	47	1
E	52	28	21	50	82	93	0	56	105	50	14	75	267	6	31	18	21	104	16	28
j	280	9	104	7	2	63	1	2	12	12	72	0	131	2	5	3	6	21	6	9
Z	11	24	1		6	4	11	6		10	5		21			9	63	15		1
j~	58	1	51	0	0	39		1	0	2	27		1		1	0	0	0	0	0
u	200	116	154	29	34	72	13	46	30	56	176	76	389	18	28	124	230	65	32	27
k	8		2			23			22	3	5		117			13	3	56		
g	2		2								9		129			16	3	20		7
t	68	2	50		5			2	22	13	58		564		1	10	27	45	1	33
e	57	28	67	12	41	22	0	26	71	12	108	0	370	3	0	51	31	292	1	6
J	1		2						10		1									
u~	15	53	14	10	7	82		0	28	21	6		0		16	0	0	0	0	0
v	26	3	13		2	4			3		3		31	2		9	3	7		1
s	10	8	12	24	56	5	12	4	23	11	11		12	2		21	20	25	11	5
b	9	5	5	2	2	10		19	8	21	5		233			11	2	44		3
i~	30	145	39	58	63	148		45	44	13	18		0		50	0	0	7	144	1
z	33		17	1		1	5				11		1		1	1		22		1
w	353	17	215	29	54	89	27	30	50	21	220	21	251	29	7	85	47	113	55	45
l~	1	25	0	24	64	120		32	10	4	5				14	4	85		18	1
r	63	130	71	180	43	244		61	120	13	56				23	70	134	13	21	
w~	3	2	2	1	0	3		0	4	8	3		0		0	0	0	0	8	0
@	14	50	40	26	36	17	29	59	25	22	20	0	290	2	27	9	2	7	8	10
L	3		7			3			1		3		1							
f	15	7		2	2	7		4	11	2			85			5	5	37		3
i	95	382	71	368	119	203	130	241	221	40	233	53	225	46	69	226	189	115	84	22
6	443	157	405	149	74	226	176	88	321	133	328	2	593	42	46	149	304	142	136	55
n	7	1	7	3	5	1		1	15	3	8		15		10	1	3	4		
O	42	38	26	36	17	65	1	25	42	64	60	61	272	45	29	56	51	56	44	30
m	8	13	9	2		25		1	6		6		20	35		42	11	7		
o~	24	118	29	19	11	137		13	97	31	21		0		21	0	0	0	73	3
l	18		7		7	5		35	3	8	7		2		2		18		5	
p	7	26	4	14		22	7		36		9		456			7		56	2	
6~	13	164	23	25	37	340		0	80	30	14		0		3	0	1	0	45	0
R	38	15	14	29	11	11		6	25	9	27				24	4	10	37	17	
o	27	53	42	48	18	104	15	45	25	81	41	49	301	14	6	22	104	11	26	11

Table F.23: Table with the number of diphone occurrences, starting with vowel, between words, following the rules, color highlighted

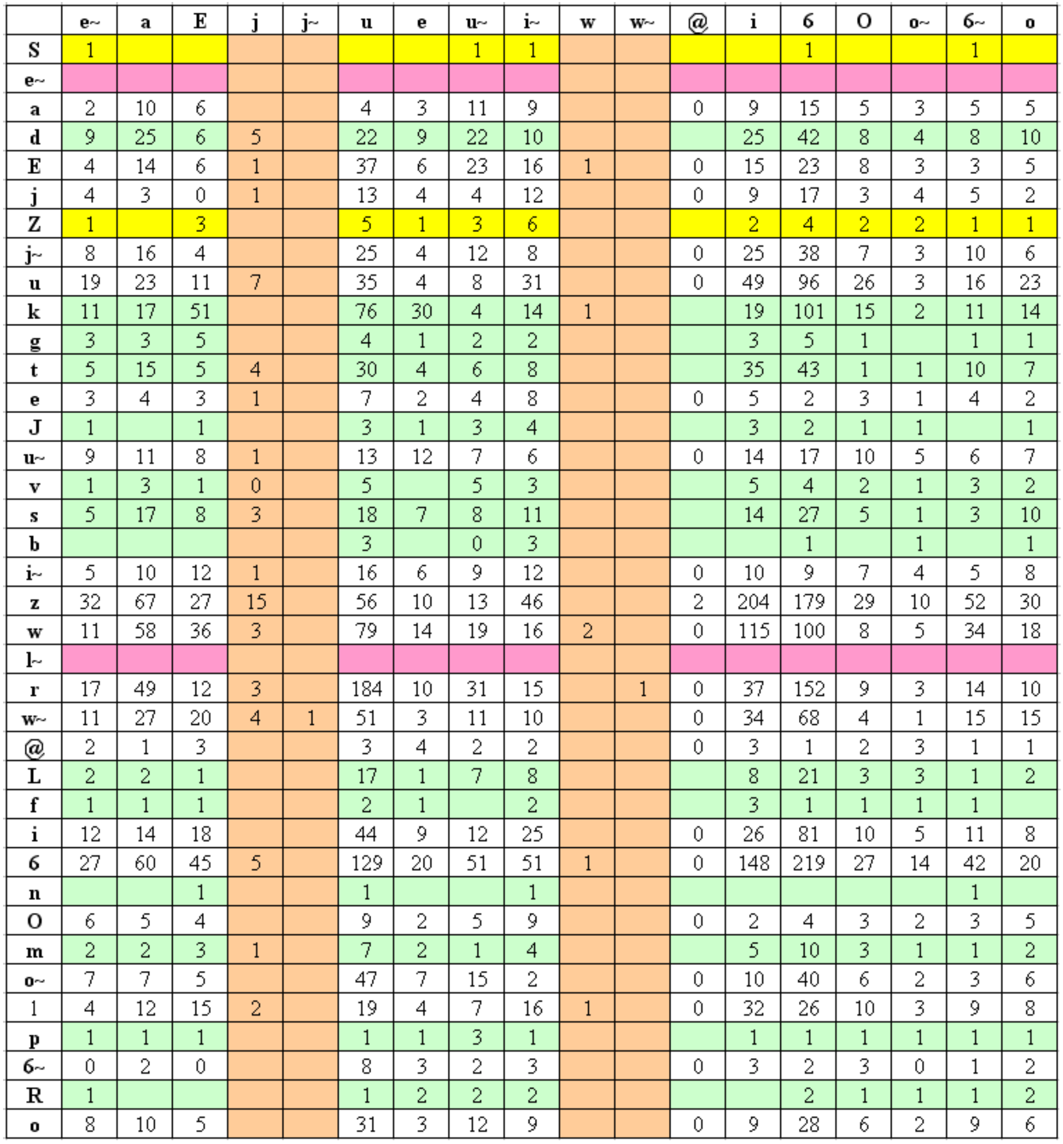


	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S																		
e~																		
a	3	11	6			4	3	12	9			7	9	15	7	3	5	4
d																		
E	4	14	6			38	6	23	16			3	16	23	7	3	3	5
j	4	3	0			14	1	4	8			3	9	16	1	5	5	2
Z																		
j~	9	16	4			26	5	12	8			5	24	38	6	3	11	7
u	27	74	44			96	15	23	40			21	163	179	31	7	47	38
k																		
g																		
t																		
e	3	4	3			6	2	4	8			0	7	2	3	2	4	2
J																		
u~	10	10	8			14	8	7	7			4	19	18	11	5	6	6
v																		
s																		
b																		
i~	5	10	12			18	6	10	14			1	14	9	7	4	6	8
z	32	66	23			58	8	11	43			43	207	180	29	9	50	27
w	3	4	1			19	3	4	7			3	12	23	3	1	3	1
l~																		
r	16	49	11			147	8	30	13			10	35	127	6	3	14	7
w~	9	26	17			52	5	11	11			15	36	72	3	1	14	15
@	50	95	94			237	66	75	93			43	147	298	57	28	48	58
L																		
f																		
i	12	17	17			41	12	11	26			2	25	85	12	4	10	8
6	27	58	48			134	19	49	55			31	154	219	27	12	41	18
n																		
O	6	5	4			9	2	5	9			1	2	4	5	2	3	3
m																		
o~	7	6	5			48	7	14	3			1	10	43	6	2	3	6
l	3	10	12			15	2	5	13			4	30	20	6	2	4	7
p																		
6~	2	3	3			8	3	3	3			0	4	2	3	2	4	2
R																		
o	8	11	5			31	3	12	9			1	9	29	6	2	9	6

Table F.24: Table with the number of diphone occurrences, starting with consonant, between words, following the rules, color highlighted

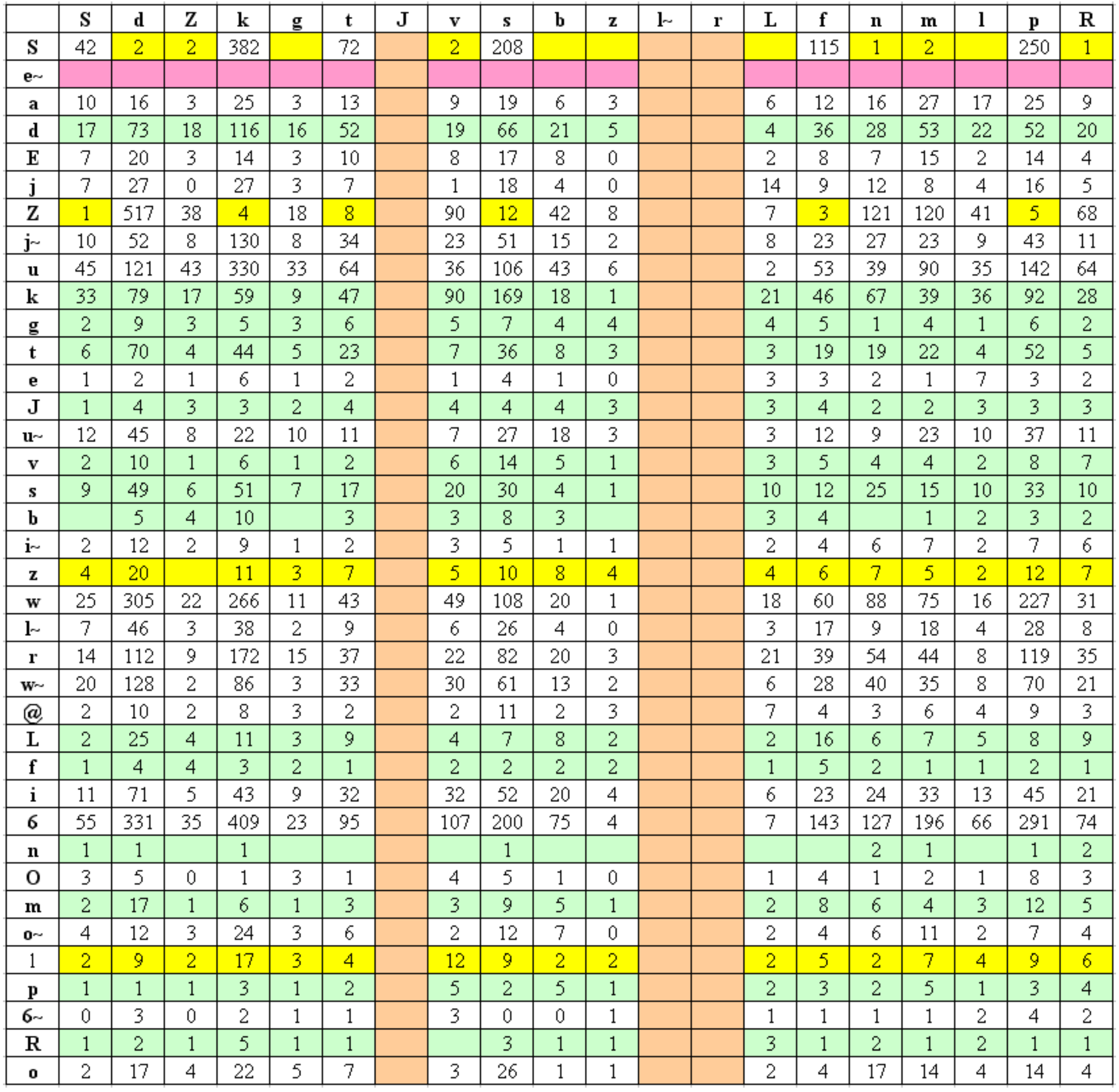
	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	14			388		71			208						115				253	
e~																				
a	4	16	4	25	3	14		10	19	6	3			6	12	17	28	17	25	10
d																				
E	4	21	3	15	3	10		7	17	8	0			2	8	7	16	2	14	5
j	4	26	0	24	3	6		1	13	3	0			14	8	12	7	3	14	5
Z		513	32		17			87		42	7			6		121	121	40		63
j~	5	52	8	131	8	34		23	52	15	2			8	23	28	23	9	44	11
u	44	405	59	554	40	95		78	188	60	7			9	100	111	150	43	331	86
k																				
g																				
t																				
e	1	2	1	6	1	2		1	4	1	0			3	3	2	1	7	3	2
J																				
u~	8	45	8	22	10	12		7	27	18	3			3	12	9	24	11	37	11
v																				
s																				
b																				
i~	2	12	2	9	2	2		3	5	2	2			5	4	7	7	2	6	7
z																				
w	3	24	4	46	4	12		9	27	4	0			9	13	17	14	7	45	8
l~	3	48	3	40	2	9		6	26	4	0			3	17	9	17	4	28	8
r	6	106	6	166	15	34		17	81	17	3			19	32	50	41	4	106	32
w~	6	128	2	90	3	32		30	61	13	2			6	28	40	35	8	73	21
@	53	403	83	369	63	197		199	401	102	35			79	188	183	181	107	317	124
L																				
f																				
i	8	73	5	44	8	32		33	57	20	3			3	24	24	34	14	47	20
6	25	329	34	414	23	94		106	202	75	4			7	145	126	199	67	292	76
n																				
O	1	4	1	1	3	1		4	5	1	1			1	4	1	2	1	8	3
m																				
o~	3	12	3	24	3	7		2	12	7	0			2	4	6	11	3	7	4
l																				
p																				
6~	1	4	1	2	1	1		3	0	0	1			1	1	1	1	2	4	2
R																				
o	1	19	4	21	5	7		3	26	1	1			2	4	17	14	4	14	4

Table F.25: Table with the number of diphone occurrences, starting with vowel, between words, considering vocalic reduction, color highlighted



	e~	a	E	j	j~	u	e	u~	i~	w	w~	@	i	6	O	o~	6~	o
S	1							1	1					1			1	
e~																		
a	2	10	6			4	3	11	9			0	9	15	5	3	5	5
d	9	25	6	5		22	9	22	10				25	42	8	4	8	10
E	4	14	6	1		37	6	23	16	1		0	15	23	8	3	3	5
j	4	3	0	1		13	4	4	12			0	9	17	3	4	5	2
Z	1		3			5	1	3	6				2	4	2	2	1	1
j~	8	16	4			25	4	12	8			0	25	38	7	3	10	6
u	19	23	11	7		35	4	8	31			0	49	96	26	3	16	23
k	11	17	51			76	30	4	14	1			19	101	15	2	11	14
g	3	3	5			4	1	2	2				3	5	1		1	1
t	5	15	5	4		30	4	6	8				35	43	1	1	10	7
e	3	4	3	1		7	2	4	8			0	5	2	3	1	4	2
J	1		1			3	1	3	4				3	2	1	1		1
u~	9	11	8	1		13	12	7	6			0	14	17	10	5	6	7
v	1	3	1	0		5		5	3				5	4	2	1	3	2
s	5	17	8	3		18	7	8	11				14	27	5	1	3	10
b						3		0	3					1		1		1
i~	5	10	12	1		16	6	9	12			0	10	9	7	4	5	8
z	32	67	27	15		56	10	13	46			2	204	179	29	10	52	30
w	11	58	36	3		79	14	19	16	2		0	115	100	8	5	34	18
l~																		
r	17	49	12	3		184	10	31	15		1	0	37	152	9	3	14	10
w~	11	27	20	4	1	51	3	11	10			0	34	68	4	1	15	15
@	2	1	3			3	4	2	2			0	3	1	2	3	1	1
L	2	2	1			17	1	7	8				8	21	3	3	1	2
f	1	1	1			2	1		2				3	1	1	1	1	
i	12	14	18			44	9	12	25			0	26	81	10	5	11	8
6	27	60	45	5		129	20	51	51	1		0	148	219	27	14	42	20
n			1			1			1								1	
O	6	5	4			9	2	5	9			0	2	4	3	2	3	5
m	2	2	3	1		7	2	1	4				5	10	3	1	1	2
o~	7	7	5			47	7	15	2			0	10	40	6	2	3	6
1	4	12	15	2		19	4	7	16	1		0	32	26	10	3	9	8
p	1	1	1			1	1	3	1				1	1	1	1	1	1
6~	0	2	0			8	3	2	3			0	3	2	3	0	1	2
R	1					1	2	2	2					2	1	1	1	2
o	8	10	5			31	3	12	9			0	9	28	6	2	9	6

Table F.26: Table with the number of diphone occurrences, starting with consonant, between words, considering vocalic reduction, color highlighted



	S	d	Z	k	g	t	J	v	s	b	z	l~	r	L	f	n	m	l	p	R
S	42	2	2	382		72		2	208						115	1	2		250	1
e~																				
a	10	16	3	25	3	13		9	19	6	3			6	12	16	27	17	25	9
d	17	73	18	116	16	52		19	66	21	5			4	36	28	53	22	52	20
E	7	20	3	14	3	10		8	17	8	0			2	8	7	15	2	14	4
j	7	27	0	27	3	7		1	18	4	0			14	9	12	8	4	16	5
Z	1	517	38	4	18	8		90	12	42	8			7	3	121	120	41	5	68
j~	10	52	8	130	8	34		23	51	15	2			8	23	27	23	9	43	11
u	45	121	43	330	33	64		36	106	43	6			2	53	39	90	35	142	64
k	33	79	17	59	9	47		90	169	18	1			21	46	67	39	36	92	28
g	2	9	3	5	3	6		5	7	4	4			4	5	1	4	1	6	2
t	6	70	4	44	5	23		7	36	8	3			3	19	19	22	4	52	5
e	1	2	1	6	1	2		1	4	1	0			3	3	2	1	7	3	2
J	1	4	3	3	2	4		4	4	4	3			3	4	2	2	3	3	3
u~	12	45	8	22	10	11		7	27	18	3			3	12	9	23	10	37	11
v	2	10	1	6	1	2		6	14	5	1			3	5	4	4	2	8	7
s	9	49	6	51	7	17		20	30	4	1			10	12	25	15	10	33	10
b		5	4	10		3		3	8	3				3	4		1	2	3	2
i~	2	12	2	9	1	2		3	5	1	1			2	4	6	7	2	7	6
z	4	20		11	3	7		5	10	8	4			4	6	7	5	2	12	7
w	25	305	22	266	11	43		49	108	20	1			18	60	88	75	16	227	31
l~	7	46	3	38	2	9		6	26	4	0			3	17	9	18	4	28	8
r	14	112	9	172	15	37		22	82	20	3			21	39	54	44	8	119	35
w~	20	128	2	86	3	33		30	61	13	2			6	28	40	35	8	70	21
@	2	10	2	8	3	2		2	11	2	3			7	4	3	6	4	9	3
L	2	25	4	11	3	9		4	7	8	2			2	16	6	7	5	8	9
f	1	4	4	3	2	1		2	2	2	2			1	5	2	1	1	2	1
i	11	71	5	43	9	32		32	52	20	4			6	23	24	33	13	45	21
6	55	331	35	409	23	95		107	200	75	4			7	143	127	196	66	291	74
n	1	1		1					1							2	1		1	2
O	3	5	0	1	3	1		4	5	1	0			1	4	1	2	1	8	3
m	2	17	1	6	1	3		3	9	5	1			2	8	6	4	3	12	5
o~	4	12	3	24	3	6		2	12	7	0			2	4	6	11	2	7	4
1	2	9	2	17	3	4		12	9	2	2			2	5	2	7	4	9	6
p	1	1	1	3	1	2		5	2	5	1			2	3	2	5	1	3	4
6~	0	3	0	2	1	1		3	0	0	1			1	1	1	1	2	4	2
R	1	2	1	5	1	1			3	1	1			3	1	2	1	2	1	1
o	2	17	4	22	5	7		3	26	1	1			2	4	17	14	4	14	4

Appendix G

Table with Portuguese and German Phonemes Comparison

Table G.1: Portuguese and German phonemes comparison, in SAMPA.

	Portuguese			**German**		
	Symbol	**Word**	**Transcription**	**Symbol**	**Word**	**Transcription**
Plosives (6 Plosives)						
	p	pai	p a j	p	Pein	p aI n
	b	barco	“b a r k u	b	Bein	b aI n
	t	tenho	“t e J u	t	Teich	t aI C
	d	doce	“d o s @	d	Deich	d aI C
	k	com	k o∼	k	Kunst	k U n s t
	g	grande	“g r 6 n d @	g	Gunst	g U n s t
Affricates (0 in EP and 4 in Ger)						
	- -	- -	- -	pf	Pfahl	pf a: l
	- -	- -	- -	ts	Zahl	ts a: l
	- -	- -	- -	tS	deutsch	d OY tS
	- -	- -	- -	dZ	Dschungel	“dZ u N 6l
Fricatives (6 in EP and 10 in Ger)						
	f	falo	“f a l u	f	fast	f a s t
	v	verde	“v e r d @	v	was	v a s
	- -	- -	- -	- -	- -	- -
	- -	- -	- -	- -	- -	- -
	s	céu	s E w	s	Tasse	“t a s @
	z	casa	“k a z 6	z	Hase	“h a: z @
	S	chapéu	S 6“ p E w	S	waschen	“v a S 6 n
	Z	jóia	“Z O j 6	Z	Genie	Z e“ n i:
	- -	- -	- -	C	sicher	"zIC6
	- -	- -	- -	j	Jahr	ja:6
	- -	- -	- -	x	Buch	b u: x
	- -	- -	- -	h	Hand	h a n t
Nasals (3 in EP and 3 in Ger)						
	m	mar	m a r	m	mein	m aI n
	n	nada	“n a d 6	n	nein	n aI n
	- -	- -	- -	N	Ding	d I N
	J	vinho	“v i J u	- -	- -	- -
Liquids (4 in EP and 2 in Ger)						
	l	lanche	“l 6 n S @	l	Leim	l aI m
	l∼	sal	s a l∼	- -	- -	- -
	L	trabalho	t r 6“ b a L u	- -	- -	- -
	r	caro	“k a r u	- -	- -	- -
	R	rua	“R u a	R	Reim	R aI m
Glides (4 in EP and 0 in Ger)						
	w	pau	p a w	- -	- -	- -
	j	pai	p a j	- -	- -	- -
	w∼	pão	p 6∼ w∼	- -	- -	- -
	j∼	mãe	m 6∼ j∼	- -	- -	- -
Checked (Short) Vowels (8 in EP and 8 in Ger)						
	i	fita	“f i t 6	I	Sitz	z I ts
	e	dedo	“d e d w	–	–	–
	E	pé	p E	E	Gesetz	g@"z E ts
	a	pá	p a	a	Satz	z a ts
	6	maré	m 6 “r E	6	besser	"bEs 6
	O	pó	p O	O	Trotz	tr O ts
	o	toda	“t o d 6	–	–	–
	u	mulher	m u “L E r	U	Schutz	S U ts
	–	–	–	Y	hübsch	h Y pS
	–	–	–	9	plötzlich	"pl 9 tslIC
Schwa Vowel (4 Schwa Vowel)						
	@	peruca	p @ “r u k 6	@	bitte	"bIt @
Nasal Vowels (5 in EP and 0 in Ger)						
	i∼	sim	s i∼	–	–	–
	e∼	menta	“m e∼ t 6	–	–	–
	6∼	manto	“m 6∼ t w	–	–	–
	o∼	bom	b o∼	–	–	–
	u∼	mundial	m u∼ d j “a l∼	–	–	–
Free (Long) Vowels (0 in EP and 8 in Ger)						
	–	–	–	i:	Lied	l i: t
	–	–	–	e:	Beet	b e: t
	–	–	–	E:	spät	Sp E: t
	–	–	–	a:	Tat	t a: t
	–	–	–	o:	rot	r o: t
	–	–	–	u:	Blut	bl u: t
	–	–	–	y:	süß	z y: s
	–	–	–	2:	blöd	bl 2: t

Bibliography

[Acero, 1999] A. Acero, *Formant analysis and synthesis using hidden Markov models*,Proc. Eurospeech'99, Budapest, Hungary, pp. 1047-1050, 1999.

[ActiveTcl, visited in 2010] http://www.activestate.com/Products/ActiveTcl/, last visit on 12/08/2010.

[Amaral et al., 1999] R. Amaral, P. Carvalho, D. Caseiro, I. Trancoso and L. Oliveira, *Anotação fonética automática de corpora de fala transcritos ortograficamente*, Proc. of PROPOR'99, Evora, Portugal, 1999.

[ANSI s3.2-1989, 1989] *Method for Measuring the Intelligibility of Speech over Communication Systems (ANSI s3.2-1989- A revision of ANSI s3.2-1960)*, American Standard Association, 1989.

[Badino, Barolo & Quazza, 2004] L. Badino, C. Barolo and S. Quazza, *A general approach to TTS reading of mixed-language texts*, Proc. ICSLP'04, Jeju, Korea, 2004 .

[Barbosa et al., 2003] F. Barbosa, G. Pinto, F. Resende, C. Gonçalves, R. Monserrat and M. Rosa, *Grapheme-Phone Transcription Algorithm for a Brazilian Portuguese TTS*, Proc. of PROPOR'03, Faro, Portugal, 2003.

[Barros & Moebius, 2009] M. J. Barros and B. Möbius, *Speech Corpus designed for Context based European Portuguese TTS*, Proc. of LTC'09, Poznan, Poland, 2009.

[Barros & Weiss, 2006] M. J. Barros and C. Weiss, *Maximum Entropy Motivated Grapheme-to-Phoneme, Stress and Syllable Boundary Prediction for Portuguese Text-to-Speech*, Proc. of the IV Biennial Workshop on Speech Technology, Saragoza, Spain, 2006.

[Barros et al., 2001] D. Braga, D. Freitas, M. J. Barros, J. P. Teixeira and V. Latsch, *Backclose Nonsyllabic Vowel [u] Behavior in European Portuguese: Reduction or Supression*, Proc. of ICSP'01, Taejon, South Korea, 2001.

[Barros et al., 2005] M. J. Barros, R. Maia, K. Tokuda, D. Freitas and F. G. Resende, *HMM-based European Portuguese Speech Synthesis*, Proc. Interspeech'05, Lisbon, Portugal, 2005.

[Barros, 2002] M. J. Barros, *A Comparative Study and Techniques of Signal Generation to Speech Synthesis*, Master thesis, Faculty of Engineering University of Porto, Portugal, 2002.

[Berger, consulted in 2010] http://www.cs.cmu.edu/afs/cs/user/aberger/www/html/tutorial/tutorial.h last visit on 12/08/2010.

[Berger, Pietra & Pietra, 1996] A. Berger, S. Pietra, V. Pietra, *A Maximum Entropy Approach to Natural Language Processing*, Computational Linguistics 22(1), 1996.

[Black & Lenzo, 2004] A. Black and K. Lenzo, *Multilingual Text-to-Speech Synthesis*, Proc. ICASSP'04, Montreal, Canada, 2004.

[Black, Lenzo & Pagel, 1998] A. Black, K. Lenzo and V. Pagel, *Issues in Building General Letter to Sound Rules*, Proc. of the 3rd International Workshop on Speech Synthesis, Blue Mountains, Australia, 1998.

[Black, Zen & Tokuda, 2007] A. Black, H. Zen and K. Tokuda, *Statistical Parametric Speech Synthesis*, Proc. ICASSP'07, Honolulu, Hawaii, USA, 2007.

[Blevins, 1995] J. Blevins, *The syllable in phonological theory*, In Goldsmith, John A.(ed.), The handbook of phonological theory. Oxford: Blackwell, 1995.

[Blizzard, 2009] http://www.synsig.org/index.php/Blizzard_Challenge_2009, last visit on 12/08/2010.

[Braga & Coelho, 2008] D. Braga, L. Coelho, *Automatic Word Stress Marker for Portuguese TTS*, Proc. of the V Biennial Workshop on Speech Technology, Bilbao, Spain, 2008.

[Braga, 2008] D. Braga, *Algoritmos de Processamento da Linguagem Natural para Sistemas de Converso Texto-fala em Portugus*, PhD dissertation, Faculty of Philology, University of Coruña, Spain, 2008.

[Braga, Coelho & Resende, 2007] D. Braga, L. Coelho and F.G. Resende, *Homograph Ambiguity Resolution in Front-End Design for Portuguese TTS Systems*, Proc. of Interspeech'07, Antwerp, Belgium, 2007.

[Braga, Freitas & Ferrreira, 2003] D. Braga, D. Freitas and H. Ferreira, *Processamento Linguístico Aplicado à Síntese da Fala*, Proc. of 3rd CLME, Maputo, Mozambique, 2003.

[Brill, 1995] E. Brill , *Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging*, Computational linguistics 21(4), pp. 543-565, 1995.

[Campbell, 1998] N. Campbell, *Foreign-Language Speech Synthesis*, Proc. of the 3rd ESCA/COCOSDA Intern. Workshop on Speech Synthesis, Jenolan Caves, Australia, 1998.

[Caseiro et al., 2002] D. Caseiro, I. Trancoso, L. Oliveira and C. Viana, *Grapheme-to-Phone using Finite-State Transducers*, Proc. of IEEE Workshop on Speech Synthesis, Santa Monica, California, 2002.

[Chen, 2003] S. Chen, *Conditional and joint models for Grapheme-to-phoneme conversion*, Proc. Eurospeech'03, Geneva, Switzerland, 2003.

[Childers, 1999] D. Childers, *Speech Processing and Synthesis Toolboxes*, John Wiley & Sons, Inc., 1999.

[CRI, visited in 2010] Código de Redacção Interinstitucional, http://publications.europa.eu/code/pt/pt-4100200pt.htm, last visit on 12/08/2010.

[Damper et al., 1998] R. Damper, Y. Marchand, M. Adamson and K. Gustafson, *A comparison of letter-to-sound conversion techniques for English text-to-speech synthesis*, Proc. of the Institute of Acoustics 20 (6), 1998.

[Davies & Elder, 2005] A. Davies and C. Elder, *The Handbook of Applied Linguistics*, Wiley-Blackwell Publishing, 2005.

[Donovan & Woodland, 1995a] R. Donovan and P. Woodland, *Improvements in an HMM-based speech synthesiser*, Proc. Eurospeech'95, pp. 573-576 vol. 1, Madrid, Spain, 1995.

[Donovan & Woodland, 1995b] R. Donovan and P. Woodland, *Automatic speech synthesiser parameter estimation using HMMs*, Proc. ICASSP'95, Detroit, MI, USA, pp. 640-643 vol. 1, 1995.

[Donovan, 1996] R. Donovan, *Trainable Speech Synthesis*, PhD thesis, Cambridge Univ., Eng. Dept., 1996.

[Dutoit, 1997] T. Dutoit, *An Introduction to Text-to-Speech Synthesis*, Kluwer Academic Publishers, vol. 3, 1997.

[Falaschi, Giustiniani, & Verola, 1989] A. Falaschi, M. Giustiniani and M. Verola, *A hidden Markov model approach to speech synthesis*, Proc. Eurospeech'89, pp. 2187-2190, 1989.

[Farges & Clements, 1986] E. Farges and M. Clements, *Hidden Markov Models Applied to Very Low Bit Rate Speech Coding*, Proc. ICASSP'86, pp. 433-436, Tokyo, 1986.

[Farges & Clements, 1988] E. Farges and M. Clements, *An analysis-synthesis hidden Markov model of speech*, Proc. ICASSP'88, pp. 323-326, 1988.

[Fukada et al., 1992] T. Fukada, K. Tokuda, T. Kobayashi and S. Imai, *An adaptative algorithm for Mel-cepstral analysis of speech*, Proc. ICASSP'92, pp. 137-140 vol. 1, 1992.

[Fukada et al., 1994] T. Fukada, Y. Komori, T. Aso and Y. Ohora, *A study of pitch pattern generation using HMM-based statistical information*, Proc. ICSLP'94, pp. 723-726, 1994.

[Furui, 1986] S. Furui, *Speaker-Independent Isolated Word Recognition Using Dynamic Features of Speech Spectrum*, IEEE Trans. Acoust., Speech, Signal Processing, pp. 52-59 vol. ASSP-34, 1986.

[Giustiniani & Pierucci, 1991] M. Giustiniani and P. Pierucci, *Phonetic ergodic HMM for speech synthesis*, Proc. Eurospeech'91, pp. 349-352, 1991.

[Gouveia, Teixeira & Freitas, 2000] P. Gouveia, J. Teixeira and D. Freitas, *Divisão Silábica Automática do Texto Escrito e Falado*, Proc. of PROPOR'00, S. Paulo, Brazil, 2000.

[Hermansky & Sharma, 1998] H. Hermansky and S. Sharma, *TRAPS - Classifiers of Temporal Patterns*, Proc. ICSLP'98, Sydney, Autralia, 1998.

[HTK, visited in 2010] http://htk.eng.cam.ac.uk/, last visit on 12/08/2010.

[HTS, visited in 2010] http://hts.sp.nitech.ac.jp/, last visit on 12/08/2010.

[Huang et al., 1996] X. Huang, A. Acero, J. Adcock, H. Hon, J. Goldsmith, J. Liu and M. Plumpe, *Whistler: A Trainable Text-to-Speech System*, Proc. ICSLP'96, Philadelphia, USA, 1996.

[Huang, Acero & Hon, 2001] X. Huang, A. Acero and H. Hon, *Spoken Language Processing, A guide to theory, Algorithm and System Development*, Prentice Hall, ISBN 0-13-022616-5, 2001.

[Hunt & Black, 1996] A. Hunt and A. Black, *Unit selection in a concatenative speech synthesis system using a large speech database*, Proc. ICSLP'96, pp. 373-376, 1996.

[IPA, visited in 2010] The International Phonetic Association, http://www.langsci.ucl.ac.uk/ipa/index.html, last visit on 12/08/2010.

[IRB, last visit in 2010] http://dms.irb.hr/tutorial/tut_mod_eval_1.php, last visit on 12/08/2010.

[Latorre, Iwano & Furui, 2005a] J. Latorre, K. Iwano and S. Furui, *Cross-language Synthesis with a Polyglot Synthesizer*, Proc. Interspeech'05, Lisbon, Portugal, 1477–1480, 2005.

[Latorre, Iwano & Furui, 2005b] J. Latorre, K. Iwano and S. Furui, *Speaker Adaptable Multilingual Synthesis*, Proc. Symposium on Large-Scale Knowledge Resources(LKR2005), Tokyo, Japan, 235–238, 2005.

[Ljolje & Fallside, 1986] A. Ljolje and F. Fallside, *Synthesis of natural sounding pitch contours in isolated utterances using hidden Markov models*, IEEE Trans. Acoust., Speech, Signal Processing, pp. 1074-1080 vol. ASSP-34, 1986.

[Maia et al., 2003] R. Maia, H. Zen, K. Tokuda, T. Kitamura and F. G. Resende, *Towards the development of a Brazilian Portuguese Text-to-Speech System Based on HMM*, Proc. Interspeech'03, Geneva, Switzerland, 2003.

[Maia et al., 2007] R. Maia, T. Toda, H. Zen, Y. Nankaku and K. Tokuda, *An Excitation Model for HMM-Based Speech Synthesis Based on Residual Modeling*, Proc. of SSW6, Bonn, Germany, 2007.

[Masuko, 1996] T. Masuko, K. Tokuda, T. Kobayashi and S. Imai, *HMM-based speech synthesis with various voice characteristics*, Proc. ASA and ASJ 3rd Joint Meeting, Honolulu, USA, pp. 1043-1046, 1996.

[Masuko, 2002] T. Masuko, *Multi-space probability distribution HMM*, PhD thesis, Tokyo Institute of Technology, 2002.

[Mateus & Andrade, 2000] M. Mateus and E. Andrade, *Phonology of Portuguese*, Oxford University Press, 2000.

[Meinedo, Neto & Almeida, 1999] H. Meinedo, J. Neto and L. Almeida, *Syllable onset detection applied to the Portuguese language*, Proc. of Eurospeech'99, Budapest, Hungary, 1999.

[Mermelstein, 1973] P. Mermelstein, *Articulatory model for the study of speech production*, Journal of the Acoustic Society of America 53, pp. 1070-1082, 1973.

[Milner & Shao, 2002] B. Milner and X. Shao, *Speech Reconstruction from Mel-Frequency Cepstral Coefficients using a Source-filter Model*, Proc. of ICSLP'02, Denver, USA, 2002.

[Moore, consulted in 2010] A. Moore, *Hidden Markov Models*, at http://www.cs.cmu.edu/ awm/tutorials, last visit on 12/08/2010.

[Morfolimpiadas, visited in 2010] http://www.linguateca.pt/Morfolimpiadas/, last visit on 12/08/2010.

[Oliveira, 1996] L.C. Oliveira, *Síntese de Fala a Partir de Texto*, PhD dissertation, University of Lisbon, Portugal, 1996.

[Oliveira, Moutinho & Teixeira, 2005a] C. Oliveira, L. Moutinho and A. Teixeira, *On European Portuguese Automatic Syllabification*, Proc. of the III Conference on Experimental Phonetics, Santiago de Compostela, Spain, 2005.

[Oliveira, Moutinho & Teixeira, 2005b] C. Oliveira, L.C. Moutinho, A. Teixeira, *On European Portuguese Automatic Syllabification*, Proc. of Interspeech'05, Lisbon, Portugal, 2005.

[Oliveira, Viana & Trancoso, 1991] L.C. Oliveira, M.C. Viana and I. Trancoso, *DIXI - Portuguese Text-to-Speech System*, Proc. of EUROSPEECH'91, Genoa, Italy, 1991.

[Oliveira, Viana & Trancoso, 1992] L. Oliveira, M. Viana and I. Trancoso, *A Rule-Based Text-to-Speech System for Portuguese*, Proc. ICASSP'92, San Francisco, USA, 1992.

[Patent 5230037] http://www.freepatentsonline.com/5230037.html, last visit on 12/08/2010.

[Peterson et al., 1958] G. Peterson, W. Wang and E. Sivertsen, *Segmentation techniques in speech synthesis*, Journal of the Acoustical Societ Yof America, 30, pp. 743-746, 1958.

[Polyakova & Bonafonte, 2006] T. Polyakova and A. Bonafonte, *Learning from Errors in Grapheme-to-Phoneme Conversion*, Proc. Interspeech'06, Pittsburgh, USA, 2006.

[PT4100100, visited in 2010] http://publications.europa.eu/code/pt/pt-4100100pt.htm, last visit on 12/08/2010.

[Rabiner, 1989] L. Rabiner, *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, Proc. IEEE, pp. 257-287 vol. 77 no 2, 1989.

[Ratnarparkhi, 1998] A. Ratnarparkhi, *Maximum Entropy Models for Natural Language Ambiguity Resolution*, PhD Dissertation, University of Pennsylvania, 1998.

[Realspeak, visited in 2010] http://www.nuance.com/realspeak/, last visit on 12/08/2010.

[SAMPA, visited in 2010] SAMPA-Speech Assessment Methods Phonetic Alphabet, http://www.phon.ucl.ac.uk/home/sampa, last visit on 12/08/2010.

[Sharman, 1994] R. Sharman, *Concatenative Speech Synthesis Using Sub-phoneme Segments*, Proc. Institute of Acoustics, pp. 367-374 Vol. 16, 1994.

[Snack, visited in 2010] http://www.speech.kth.se/snack/, last visit on 12/08/2010.

[SPTK, visited in 2010] http://sp-tk.sourceforge.net/, last visit on 12/08/2010.

[Syrdal, Bennett & Greenspan, 1994] A. Syrdal, R. Bennett and S. Greenspan, *Applied Speech Technology*, CRC Press, 1994.

[Taylor, 2005] P. Taylor, *Hidden Markov Models for grapheme-to-phoneme Conversion*, Proc. Interspeech'05, Lisbon, Portugal, 2005.

[Teixeira & Freitas, 1998] J.P. Teixeira and D. Freitas, *MULTIVOX- Conversor Texto-Fala para Português*, Proc. of PROPOR'98, Porto Alegre, RS, Brazil, 1998.

[Teixeira et al., 1998] J. Teixeira, D. Freitas, P. Gouveia, G. Olaszy and G. Németh, *MULTIVOX Conversor Texto Fala Para Portugus*, Proc. of III Encontro Para o Processamento Computacional da Língua Portuguesa Escrita e Falada, Porto Alegre, Brasil, 1998.

[Teixeira et al., 2001] J. Teixeira, D. Freitas, D. Braga, M. J. Barros and V. Latsch, *Phonetic Events from the Labeling the European Portuguese Database for Speech Synthesis, FEUP/IPB-DB*, Proc. Eurospeech'01, Aalborg, Denmark, September 2001.

[Teixeira, 1998] J. Teixeira, *Conversor Texto-Fala para o Português Desenvolvimentos / Ferramentas*, Proc. of 1as jornadas do CEFAT, Bragança, Portugal, 1998.

[Teixeira, 2004] J. Teixeira, *A Prosody Model to TTS Systems*, PhD dissertation, Faculty of Engineering of University of Porto, Portugal, 2004.

[Teixeira, Trancoso & Serralheiro, 1996] C. Teixeira, I. Trancoso and A. Serralheiro, *Accent Identification*, Proc. of ICSLP'96, Philadelphia, USA, 1996.

[Tokuda et al., 1995] K. Tokuda, T. Masuko, T. Yamada, T. Kobayashi and S. Imai, *An algorithm for speech parameter generation from continuous mixture HMMs with dynamic features*, Proc. Eurospeech'95, Madrid, Spain, pp. 757-760, 1995.

[Tokuda et al., 2000a] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi and T. Kitamura, *Speech parameter generation algorithms for HMM-based speech synthesis*, Proc. ICASSP'00, Istanbul, Turkey, pp. 1315-1318 vol. 3, 2000.

[Tokuda et al., 2000b] K. Tokuda et al., *Reference Manual for Speech Signal Processing Toolkit Ver. 2.0*, Nagoya Institute of Technology, revision of June 2000.

[Tokuda et al., 2002] K. Tokuda, T. Masuko, N. Miyazaki and T. Kobayashi, *Multi-space probability distribution HMM*, IEICE Trans. Information and Systems, pp. 455-464, no. 3 vol. E85-D, March 2002.

[Tokuda, consulted in 2009] K. Tokuda, *HMM-Based Speech Synthesis toward Human-like Talking Machines*, Seminar on HMM-Based Speech Synthesis, Nagoya Institute of Technology, Japan.

[Tokuda, Kobayashi, & Imai, 1995] K. Tokuda, T. Kobayashi and S. Imai, *Speech parameter generation from HMM using dynamic features*, Proc. ICASSP'95, Detroit, MI, USA, pp. 660-663 vol. 1, 1995.

[Tokuda, Zen & Black, 2002] K. Tokuda, H. Zen, A. Black, *An HMM-based speech synthesis system applied to English*, Proc. IEEE Speech Synthesis Workshop, Santa Monica, California, 2002.

[Traber, 1995] C. Traber, *SVOX: The Implementation of a Text-to-Speech System for German*, PhD thesis, Swiss Federal Institute of Technology, Zurich, 1995.

[Traber et al., 1999] C. Traber, K. Huber, K. Nedir, B. Pfister, E. Keller and B. Zellner, *From Multilingual to Polyglot Speech Synthesis*, Proc. of Eurospeech'99, Budapest, Hungary, 835-838, 1999.

[Trancoso et al., 1994] I. Trancoso, M. Viana, F. Silva, G. Marques and L. Oliveira, *Rule-Based vs. Neural Network Based Approaches to Letter-to-Phone Conversion for Portuguese Common and Proper Names*, Proc. of ICSLP'94, Yokohama, Japan, 1994.

[Tychtl & Psutka, 2000] Z. Tychtl and J. Psutka, *Pitch Synchronous Residual excited Speech Reconstruction on the MFCC*, Proc. of EUSIPCO'2000, Tampere, Finland, pp. 761-764, vol. II, 2000.

[Weiss et al., 2005] C. Weiss, R. Maia, K. Tokuda and W. Hess, *Low Resource HMM-based Speech Synthesis applied to German*, Proc. ESSP'05 - 16th Conf. on Electronic Speech Signal Processing, Prague, Czech Republic, 2005.

[Wikipedia, visited in 2010] http://en.wikipedia.org/wiki/List_of_languages_by_number _of_native_speakers, last visit on 12/08/2010.

[Wikipedia, visited in 2011] http://en.wikipedia.org/wiki/Syllable_rime, last visit on 15/04/2011.

[Yamagishi, 2006] J. Yamagishi, *Average-Voice-Based Speech Synthesis*, PhD thesis, Tokyo Institute of Technology, March 2006.

[Yoshimura, 1999] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi and T. Kitamura, *Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis*, Proc. Eurospeech'99, Budapest, Hungary, pp. 2347-2350, 1999.

[Young et al., 2001] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, V. Valtchev and P. Woodland, *The HTK Book, for HTK version 3.1*, Cambridge University Engineering Department, revision of December 2001.