North-West University

Doctoral Thesis

Automatic Recognition of Code-Switched Speech in Sepedi

Author:
Thipe Isaaih Modipa
(22047689)

Supervisor:
Prof. Marelie H. Davel

A thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

in the

School of Information Technology

in the

Faculty of Economic Sciences and IT

at the

North-West University (Vaal Triangle campus)

April 2016

Acknowledgements

I would like to express my appreciation and gratitude to the Human Language Technologies (HLT) Research Group for allowing me to be part of this group and help me achieve this milestone. I would also like to thank everyone in the group who assisted me directly or indirectly during my studies.

My gratitude also goes to my family and friends for their loyal support and pushing me during strenuous situations. To my boys, Tumedi and Katlego, thank you for keeping up with me when I was spending most of my evenings away from home.

Finally, I would forever be indebted to Prof. Marelie Davel, my supervisor, for her constant guidance, encouragement, support, patience, and helping me to reach greater heights.

Abstract

Code switching (CS) is a natural phenomenon that is often observed in multilingual speakers. These speakers use words, phrases or sentences from foreign languages and embed them in sentences in the primary language. Automatic speech recognition (ASR) systems find code-switched speech difficult to process, and ASR performance is known to degrade in CS environments.

We study the Sepedi/English CS phenomenon in the context of Sepedi ASR. Using experimentation, data collection and quantitative data analysis, we analyse techniques that can be used to effectively model code-switched speech in resource-scarce environments. The focus is on techniques that modify the pronunciation dictionary, in order to improve recognition accuracy.

For this purpose, three new speech resources are designed, collected and curated: (1) the Radio Broadcast Corpus contains real examples of code-switching as observed during radio broadcasts; (2) the Sepedi Prompted Code-Switched (SPCS) Corpus is based on true code switching prompts, with each individual prompt recorded by multiple speakers in order to capture pronunciation variability occurring in code-switched speech; and (3) the National Center for Human Language Technology (NCHLT) Sepedi-English code-switched subset (NSECSS) corpus does not contain naturally occurring code-switched speech, but rather English as spoken by Sepedi speakers. The latter corpus is particularly useful as its recording conditions and format match two related corpora: English produced by English speakers and Sepedi produced by Sepedi speakers. As part of corpus development, resource collection and analysis tools were developed and evaluated.

Utilising these corpora, the implications of code-switched speech for ASR systems were evaluated. Various approaches to pronunciation modelling of code-switched speech were investigated and a novel method for pronunciation prediction developed. This new variant selection approach to modelling code-switched speech requires a two-step process: after grapheme-to-phoneme prediction of foreign words, phoneme-to-phoneme prediction (mapping the foreign phonemes to in-language phonemes) does not only take phoneme identity into account, but also graphemic context. A practical implementation of such an algorithm performed well during recognition experiments, both as a single approach and in combination with other existing approaches. The best overall results were obtained when multiple variants were generated per CS word, and variant-selection included in this process. Even though specifically applied to the Sepedi/English task, the methods themselves are language-independent.

In addition, the methods, frequency of and reasons for code switching observed among Sepedi speakers were studied using corpus analysis. Among other results, it was found that the prevalence of code switching within naturally occurring Sepedi speech was much higher than initially anticipated, making this a task well worth studying.

Keywords: Code switching, automatic speech recognition, code-switched speech, grapheme-to-phoneme, phoneme-to-phoneme, pronunciation dictionary, pronunciation prediction, Sepedi

Contents

List of Tables

List of Figures

Abbreviations

ASR	**A**utomatic **S**peech **R**ecognition
CART	**C**lassification and **R**egression **T**rees
CMD	**C**ontinuous **M**ixture **D**ensity
CMN	**C**epstral **M**ean **N**ormalisation
CMVN	**C**epstral **M**ean and **V**ariance **N**ormalisation
CS	**C**ode **S**witching
CV	**C**onsonant **V**owel
DAC	**D**epartment of **A**rts and **C**ulture
DEC	**D**ynamically **E**xpanding **C**ontext
DP	**D**ynamic **P**rogramming
DTW	**D**ynamic **T**ime **W**arping
G2P	**G**rapheme-to-**P**honeme
HLT	**H**uman **L**anguage **T**echnologies
HMM	**H**idden **M**arkov **M**odel
HTK	**H**idden **M**arkov **M**odel **T**oolkit
IPA	**I**nternational **P**honetic **A**lphabet
LBD	**L**anguage **B**oundary **D**etection
LID	**L**anguage **I**dentification
LVCSR	**L**arge **V**ocabulary **C**ontinuous **S**peech **R**ecognition
MFCC	**M**el **F**requency **C**epstral **C**oefficient
MLF	**M**aster **L**abel **F**ile
NCHLT	**N**ational **C**enter for **H**uman **L**anguage **T**echnology
NSECSS	**NCHLT S**epedi-**E**nglish **C**ode-**S**witched **S**ubset
OOL	**O**ut **O**f **L**anguage
OOV	**O**ut **O**f **V**ocabulary

P2P	**P**hone-to-**P**hone
PDP	**P**hone-based **D**ynamic **P**rogramming
PER	**P**hone **E**rror **R**ate
POS	**P**art **O**f **S**peech
RMA	**R**esource **M**anagement **A**gency
SAMPA	**S**peech **A**ssessment **M**ethods **P**honetic **A**lphabet
SDS	**S**poken **D**ialogue **S**ystem
SPCS	**S**epedi **P**rompted **C**ode-**S**witched
WER	**W**ord **E**rror **R**ate

Chapter 1

Introduction

Spoken dialogue systems (SDSs) are automated systems that use voice as input and output when interacting with a user. These systems rely on speech technologies such as automatic speech recognition (ASR) and speech synthesis. SDSs are important tools for information service provision over the telephone, and are increasingly being developed for under-resourced languages in developing countries such as South Africa.

Amongst other things, the development of ASR systems relies on the accurate modelling of word pronunciation, typically using pronunciation dictionaries to map a word to its standard (or canonical) pronunciation [2]. Context-dependent phonetic effects are usually not modelled explicitly in the pronunciation dictionaries of speech recognition systems, as the statistical acoustic models are trained to take context-dependent effects into account.

One of the challenges encountered when developing a pronunciation dictionary in multilingual environments relates to the extent to which code switching occurs. Speakers naturally embed words or phrases from other languages. For example, even when constrained to a spoken dialogue, many speakers of South African languages use English numbers, dates and times. In addition, many place names have pronunciations that are clearly linked to other languages spoken in the vicinity.

There is a need to model the pronunciation of embedded words to advance the development of SDSs. Multilingual speech can be modelled at different levels, by developing multilingual acoustic models, multilingual language models, and/or multilingual pronunciation dictionaries. The focus of this thesis is on the development of multilingual pronunciation dictionaries, even though other aspects are also touched on.

1.1 Problem statement

The use of code-switched speech amongst multilingual speakers is a challenge for ASR systems, as code switching introduces additional variability with regard to both word usage and pronunciation. This results in increased recognition errors.

There are few resources (if any) available to model this type of speech for under-resourced languages such as Sepedi. There are also limited language-independent guidelines, which are directly applicable to the Sepedi task, for modelling code switching in resource-scarce environments.

Appropriate techniques to model code-switched speech have not yet been developed for speech recognition in any of the Sotho-Tswana languages.

1.2 Research questions

In this thesis, the following research questions are addressed, within the context of Sepedi ASR:

- What are the implications of code-switched speech for ASR systems?
- What are appropriate acoustic and pronunciation modelling approaches for code-switched speech in resource-scarce environments?
- What are the mechanisms and prevalence of code-switched speech in Sepedi?
- Which engineering techniques can be used to develop optimised ASR systems, capable of recognising code-switched speech in Sepedi?
- Which general tools and techniques can be used for the analysis and modelling of code-switched speech in resource-scarce environments?

1.3 Analysis and modelling of code-switched speech

We aim to determine, through experimentation, data collection and quantitative data analysis, which multilingual speech recognition techniques can be adapted to effectively model code-switched speech in resource-scarce environments. Specifically we develop tools and techniques to analyse and better model code-switched speech within ASR systems.

The development of a typical ASR system does not take into account the modelling of foreign speech. However, South African Bantu languages tend to contain a considerable amount of speech from foreign languages. We first develop a baseline Sepedi ASR system, and optimise this for later pronunciation modelling analysis. We start with the limited available transcriptions of speech available to investigate code switching in Sepedi. We soon find that there are no applicable speech resources such as a code-switched database for Sotho-Tswana languages.

Once the need for specialised speech corpora is established, we collect and annotate these. A first corpus is collected to determine the types and prevalence of code switching based on corpus analysis. It is also used to analyse the effect of speaker profile on type and frequency of code switching. A second corpus is collected to capture the pronunciation variability that occurs in code-switched speech. For the analysis of both corpora, we develop basic tools and resources to identify text-based code switching in Sepedi.

We use the Sepedi corpora for acoustic modelling of code-switched speech. We consider multilingual acoustic modelling which includes analysing various phoneme mapping strategies and pronunciation variant modelling strategies. The most promising approaches are then selected and refined, and dictionaries developed that can be evaluated for use within ASR systems.

ASR systems are developed by considering the various acoustic modelling techniques, using different (existing and new) speech corpora. These systems are evaluated with a purpose-built code-switched speech corpus. This allows us to both analyse performance, and to develop guidelines for modelling code-switched speech in under-resourced environments.

1.4 Thesis overview

In this thesis we aim to develop a robust ASR system to improve the recognition accuracy of code-switched speech. This is achieved by (a) developing appropriate acoustic Sepedi-English code-switched corpora, and (b) predicting phoneme labels to model pronunciation of English words, which can then be incorporated into the standard pronunciation dictionary.

The thesis is structured as follows:

- In Chapter 2 we discuss background information about the development of the code-switched speech as well as modelling techniques for the development of robust ASR systems.

- In Chapter 3 we discuss key methods used in this work, including the statistical models used to develop an ASR system as well as approaches for the construction of phoneme sets for code-switched speech.
- In Chapter 4 we develop an initial baseline ASR system and evaluate the recognition accuracy of code-switched speech. This provides the basis for evaluating improvements that are subsequently implemented.
- In Chapter 5 we discuss the design and collection of three specialised acoustic code-switched corpora. These corpora are used in different ways to analyse Sepedi-English code-switched speech.
- In Chapter 6 we analyse the Radio Broadcast corpus to determine the methods, frequency of and reasons for code switching to occur among Sepedi speakers.
- In Chapter 7 we consider context-independent approaches to modelling code-switched speech and analyse the performance of ASR systems developed using such an approach.
- In Chapter 8 we predict phoneme substitutions using non-acoustic features to model the pronunciation of English words to improve the recognition accuracy of an ASR system.
- In Chapter 9 we summarise the most significant contributions made during this study and discuss future work.

1.5 Conclusion

This chapter highlighted the need for obtaining a better understanding of code-switched speech in under-resourced languages such as Sepedi. The specific task addressed in this study was sketched and an overview provided of the work to follow. In the next chapter, background to the task is presented.

Chapter 2

Background

2.1 Introduction

Speech recognition used in applications such as voice search and utilising large vocabularies should be able to recognise naturally constructed phrases. Naturally constructed phrases sometimes contain code-switched speech. In code-switched speech, the spoken utterances consist of more than one language.

Multilingual speakers naturally embed words or phrases from a secondary language within their primary language. For example, even when constrained to a spoken dialogue, many speakers of South African languages use English numbers, dates and times. Also, many place names have pronunciations that are clearly linked to other languages spoken in the vicinity. Modelling such information becomes an important aspect of voice search and SDSs. At the same time, code switching can be unpredictable. As stated in [3]:

"How does it happen, for example, that among bilinguals, the ancestral language will be used on one occasion and English on another, and that on certain occasions bilinguals will alternate, without apparent cause, from one language to another?"

Code-switched speech poses a challenge to monolingual automatic speech recognition (ASR) systems. (Monolingual recognisers are trained to recognise speech in one language only.) For these recognisers, foreign words are often ignored and regarded as out-of-vocabulary (OOV) words [4]. Different approaches to recognising code-switched speech have been studied for some world language pairs such as Mandarin/Taiwanese, Cantonese/English, and English/Spanish [5, 6]. For these languages, huge data corpora are available, and modelling techniques have been developed based on these vast data resources.

This study will investigate the appropriate approach for automatic speech recognition of code-switched speech for the Sotho-Tswana languages, by focussing on Sepedi. Since these languages have significantly fewer resources available, the engineering challenges are different, and much less prior research is available.

2.2 Code switching

Code switching (CS) is defined as the use of words or phrases during conversations that originate from more than one language [3, 7]. Multilingual individuals mix words or perform code switching in their speech and, such utterances can appear in two different categories: *inter-sentential code switching* and *intra-sentential code switching* [8]. Inter-sentential code switching occurs when the language of an utterance changes from one sentence to the next, while intra-sentential code switching occurs when the language of an utterance changes within a sentence.

Based on the situation in which code switching occurs, in [3], Nilep defines two additional types of code switching:

1. situational code switching is where a transition in social setting is represented by linguistic change;
2. conversational (metaphorical) code switching occurs during conversations (within utterances).

Sepedi language has experienced linguistic borrowing just like any other language in the world as a result of contact between local languages, African languages, as well as European languages (English and Afrikaans) to end up with borrowed words [9]. Some of these words are borrowed from other languages (mostly European) due to some of the Sepedi words not expressing meaning explicitly [9]. Whether a word is considered a borrowed word or an example of code switching typically depends on the way in which it is used and pronounced. It also depends on the extent to which it has been assimilated into the other language. However, a borrowed word can be pronounced according to the rules of either the primary or the foreign language [4]. This pronunciation provides a continuum within which the boundaries between borrowed words and code-mixing are not always clear.

While some borrowed words, such as *radio* for *seyalemoya*, have a Sepedi indigenous version, other words do not. For example, the word *domain* is written in Sepedi as *domeine* and has no other Sepedi counterpart. Where both words do exist, a borrowed

word sometimes is preferred over its indigenous counterpart. For example, *Janaware* is mostly used instead of *Pherekgong*, which refers to 'January' in Sepedi.

Code switching is as prevalent amongst Sepedi bilinguals as it is among other language speakers. In [9], social motivations for code switching are mentioned, such as (a) to convey the meaning of a term, (b) to use terms not existing in the matrix language, (c) to emphasise a point. During conversations when code switching is used, lexical variations take place. The occurrence of code switching influences the morphological structure of the embedded words where, for instance, the English verb stems are used in the Sepedi language with the suffix *-a* appended or used as is.

In other studies, different language pairs were studied, such as isiXhosa or isiZulu and English [10–12]. In [10], de Klerk studied the code switching behaviour amongst Xhosa speakers to determine the conversational patterns. These patterns were used to determine if interlocutors are indeed bilingual and can converse without difficulties with both languages. It was found that there were more occurrences of borrowed words than code switching. However, these words become borrowed words through code switching. For example, nouns such as *ubuntu* (humanity) and *tsotsi* (street thug) have been borrowed and incorporated in Black South African English and do not constitute code switching [10]. Borrowed words play a major role in Southern Bantu languages, such as isiXhosa [10]. It also shows that Xhosa speakers convert some English words from English to Xhosa by appending a prefix such as *i* in *i-language* or *u* in *u-five* [10, 13].

In [11], Ndwe et al. established that during code switching, English is the language that is embedded within South African indigenous languages. Also, native speakers of these matrix languages prefer their home languages during conversations, however, they use English for things such as numbers, dates, and prices.

Factors that affect code switching were identified in several studies such as [3, 12, 14–17]. Social factors seem to influence bilingual individuals to engage in code-switched speech for different reasons. Code switching appears more often in young people than in older people [3]. One of the reasons young people in South Africa engage in code switching is that they have many language contacts. For example, they use their native language when interacting with friends or family members at home and use English in schools as a medium of instruction.

Some work has previously been done to study code switching both in linguistics and speech recognition. From a speech recognition perspective, in [12], the aim of the study was to determine the effectiveness of identifying embedded utterances in code-switched utterances. It was found to be easier to identify embedded utterances in code-switched speech than in monolingual speech. The development of multilingual systems requires

the collection of speech corpora relevant for the information system. In [14], a multilingual information system was developed to render service and was expected to recognise code-switched speech.

From a linguistic perspective, in [15, 16], the objective of the studies was to determine the factors that contribute to code switching and whether code switching has any effect in the classroom on the performance of the students. Code switching seems to be a way to enforce learning in class. Furthermore, similar factors were also studied in [17] but in this case it was from a written form rather than verbal.

2.3 Acoustic modelling of code-switched speech

The accurate recognition of code-switched utterances remains a challenge for ASR systems. For ASR systems, language boundary and language identification information can be added to recognise code-switched utterances [5].

Several approaches for the recognition of code-switched speech have been proposed in the literature [7, 18]. The two most popular approaches are:

- Multipass recognition. This approach relies on spoken language identification to segment speech into different language sections, and to then perform monolingual recognition on each separate speech segment. An example is the three layer multipass recognition as defined in [7]. The first layer is a language boundary detection (LBD) module, followed by a language identification (LID) module, and lastly, monolingual ASR. This multi-layer approach can cause a performance bottleneck since the success of the LBD will determine the success of the language identification. The success of the language identification, in turn, determines the success of monolingual ASR [7].
- One-pass recognition. Here multilingual information is embedded in the acoustic and language models, and in its extreme, results in multilingual systems capable of recognising different languages [7].

In the next sections, we briefly review monolingual and multilingual ASR systems, before discussing multilingual pronunciation dictionaries and, specifically, language-to-language phoneme mappings.

2.3.1 Monolingual systems

The development of speech technology systems has advanced for some languages in the world. Not all world languages receive attention for the development of speech technology systems. There are over 7 000 living natural languages in the world as found in the Ethnologue website[1], and for some languages resources are still very scarce. (This includes the Southern Bantu languages [19].)

Monolingual recognisers are trained to recognise utterances containing one language. Monolingual speech recognisers for resource-scarce languages often have poor recognition accuracies. Some of the disadvantages of monolingual ASR systems arise from the scarcity of data. These disadvantages could be overcome by adapting monolingual recognisers to data from other languages by using various approaches including bootstrapping, pooling and model adaptation [20–22].

Monolingual speech recognition systems do not model code-switched speech effectively if it contains words from different languages. Such words would normally be treated as out-of-language (OOL) words. However, OOL words can carry valuable information such as dates, times and destinations when used in spoken dialog systems queries [4]. The pronunciation of a secondary language word within a primary language has challenges that are not addressed by the monolingual acoustic models. The inclusion of OOL words can improve the recognition accuracy of automatic speech recognition systems thereby reducing the word error rate (WER).

Some approaches have been suggested for building pronunciation of OOL words when dealing with code-switched speech. The pronunciation of the secondary language word can be modified to match the primary language, or the pronunciation of the secondary language word can be retained in the primary language [4]. The use of secondary language pronunciation within a primary language can provide gains in the modelling of code-switched speech.

2.3.2 Multilingual systems

Multilingual speech recognition system can be achieved by multiple monolingual or multilingual systems. To improve the performance of monolingual ASR, language identification consisting of language boundary and language identity information, needs to be implemented [5]. Contrary to monolingual systems, multilingual acoustic models can be used to recognise more than one language and minimise overhead to manage multiple systems. Such systems can share language resources from under-developed languages

[1] http://www.ethnologue.com/

to cut cost, time, and the use of linguistic expertise. Multilingual speech recognition systems have the advantage of sharing acoustic and language models which result in good performance, and these results can be extended to language identification [8]. The development of multilingual automatic speech recognition systems has been successful for the past few years. Acoustic models for such systems were built by either using one language with adapted speech data from another language due to the scarcity of data in the primary language or by using non-native accent speakers to build speech recognisers [4].

We consider speech recognition systems that are based on an hidden Markov model (HMM) system architecture. Multilingual acoustic models are then developed using HMMs. These acoustic models share parameters between multiple languages. An HMM provides a framework to model a sequence of spectral vectors that vary over time [23]. The HMM framework can be modelled with continuous mixture density HMMs and semi-continuous HMMs [24–27].

Code switching speech recognition based on the two-pass system architecture, which consists of the automatic speech recognition and the re-scoring phases, has been proposed [28]. However, it uses two speech recognisers in parallel. Its drawback is that the performance of one speech recogniser affects the performance of the whole system. The performance of this approach, when compared to the use of LID, is lower.

2.4 Multilingual pronunciation dictionaries

One of the components of developing multilingual acoustic models is the creation of multilingual pronunciation dictionaries. With multilingual pronunciation dictionaries the aim is to develop acoustic models for multiple languages that will perform on par with language-dependent acoustic models [29].

Pronunciation dictionaries are typically extended using grapheme-to-phoneme rules, as discussed in Section 2.4.1. Either the rules from the matrix or embedded language can be used. If rules from the embedded language are used, a new phoneme set and/or language-to-language phoneme mappings are required. The phoneme set for multilingual dictionaries can be created by combining phonemes from multiple languages [30, 31]. The phoneme set for the multilingual dictionaries can also be mapped using linguistic feature-based mappings as described in Section 2.4.2. The acoustic data can also be used to determine the distance between two phone models by using data-driven mappings that include the phone clustering process discussed in Section 2.4.3.

2.4.1 Letter-to-sound rules

Letter-to-sound rules (also referred to as grapheme-to-phoneme (G2P) predictors) provide a relation between the graphemes of a word and its pronunciation. The relation is typically not one-to-one, and in languages such as English results in complex rules and many exceptions to predict the phoneme sequence for the pronunciation of a word [20].

Letter-to-sound rules are language-dependent and typically extracted from existing pronunciation dictionaries using data-driven algorithms. The predictions of pronunciation using letter-to-sound rules are discussed in various studies, including [20, 32–34]. The specific method used in this study is introduced in Section 3.6.

Many languages have very regular writing systems, and benefit from graphemic systems [35, 36]. That is, no phonemic pronunciation dictionary is used, and the graphemes in a word are directly used as acoustic units. English has a particularly complex writing system, and therefore – when the code-switched words are English – additional modelling is required.

2.4.2 Linguistic feature-based mappings

Phonetic experts have compiled a phonetic inventory that shows the similarity between speech sounds such as *International Phonetic Alphabet* (IPA) [37]. IPA symbols are used to represent the same sounds for different languages [20]. For multilingual speech recognition systems, many languages can share phonemes across languages with language-independent phonemes.

The main drawback of the approach is that the IPA based mapping does not take spectral features of the phonemes into consideration, due to language-dependent speaking properties that produce acoustic differences between the same IPA symbols of different languages [29].

2.4.3 Data driven mappings

A data-driven mapping uses a statistical approach that requires no a priori linguistic knowledge. Data-driven approaches used in code-switched speech to identify mappings include bottom-up clustering [29], distance measures [31] and posterior and bottleneck based approaches [38].

A bottom-up clustering algorithm measures the distance between two phone models to determine the similarities between the phonemes [29]. In data driven phoneme mapping,

distance measures are used, such as the Bhattacharyya distance measure, and the acoustic likelihood distance measure [31]. Another phoneme mapping strategy uses posterior and bottleneck features that contain complimentary information, combined using a neural network [38]. A data-driven approach that integrates acoustic and context-dependent cross-lingual articulatory features for phoneme set construction for code switching is described in [39]. Selected methods are described in more detail in Section 3.3.

2.5 Multilingual language models

Multilingual language models are developed within multilingual speech recognition systems to allow the sharing of text between different languages. Multilingual language models provide a framework to model different input languages, especially instances where there is a switch within utterances. This switching of language is prevalent in code-switched speech. The multilingual language model is defined as a statistical model that encapsulates the linguistic attributes of the speech from multiple languages [40].

Several methods can be used to combine text corpus from several languages to train a language model. The problem with combining data directly from various languages is that the n-gram probabilities are not evenly distributed. The linear interpolation method has been found to model data from many languages by assigning weights to the monolingual text data [40, 41]. Several language model toolkits are available to train multilingual language models [42–44]. In this study, n-gram based language modelling is used, as discussed in Section 3.8.

2.6 Related studies

Monolingual speech corpora are available in many languages, including Sepedi. Specifically, two important Sepedi ASR resources are the Lwazi [1] and NCHLT corpora [45], as described in more detail in Section 2.7. Unfortunately no code-switched corpora exist that have Sepedi as the matrix language. Various code-switched corpora exist for other language pairs, including Mandarin-English and Cantonese-English [46, 47]. We discuss the development and analysis of such corpora below.

2.6.1 Mandarin-English corpora

The Mandarin-English corpus is a Mandarin-English code-switched speech corpus that was collected using both interview and conversational settings [46]. The code-switched

speech consists of Mandarin as a matrix language and English as an embedded language, and the type of code switching considered was intra-sentential. The data was collected to study language boundary detection (LBD), language identification (LID), and multilingual large vocabulary continuous speech recognition (LVCSR).

No scripted prompts were used to generate spontaneous code-switched speech. The speech was generated from two settings, 'conversation' and 'interview'. A close talk microphone was used to record the interview and conversational speech in a quiet location. The corpus was transcribed using the ELAN annotation tool. The transcriptions include word transcriptions as well as the language boundary labels.

Even though code switching was expected to be spontaneous, it was interesting to learn that the questions from the interviewer influenced the amount of code switching observed. Furthermore, intra-sentential code switching was higher in the interviewed setting than the conversational setting. It was also found that spontaneous speech provided a challenge to sentence boundary annotators, as speakers often do not speak full sentences. The other observation made was that the number of occurrences of embedded single words was higher than the number of occurrences of embedded multiple words within the code-switched utterances.

In [48], another Mandarin-English code switching corpus was developed that consists of conversational speech, project meetings, student interviews, and text data from on-line news. The on-line news was used to collect data automatically. The research objective was to collect data that could be used to study the rules followed in code switching and also to train acoustic and language models. Both inter- and intra-sentential code switching were considered.

The interview-based corpus was collected using a microphone in a quiet environment. The participants were both Chinese and English speakers. The collected speech was then transcribed manually by Chinese and English speakers. Additional text data containing intra-sentential code switching was automatically collected from the web. (This data did not form part of the audio speech data.)

2.6.2 Cantonese-English corpora

In [47], a code-switched speech corpus, the Cantonese-English corpus, was developed to study LBD algorithms and evaluate code-switched speech recognisers. As the pronunciation of words in code-switched speech is expected to be different from monolingual speech, monolingual speech was deemed necessary but not sufficient to measure baseline

performance. The Cantonese-English corpus was developed to satisfy the need for large amounts of code-switched data, as was required to evaluate code-switched speech.

Cantonese has spoken and written forms that are different. To collect enough spoken code-mixing data, newsgroups and on-line diaries were sourced. The participants were required to read code-mixed prompts, with corrections allowed. These participants were able to read both English and Cantonese (bilinguals). The data was collected using a microphone in a quiet environment.

The Cantonese-English corpus annotation provided both orthographic and phonemic transcriptions. This corpus was verified manually by trained assistants as well as phonetic experts.

2.6.3 Chinese-English corpora

The Chinese-English code switching speech database (CECOS) is a corpus that was collected from native Chinese speakers who are non-native English speakers to do research on Chinese-English code switching ASR [49]. Two approaches were used to develop the CECOS corpus:

1. websites were used to create a text database; and
2. Chinese-English code-switched sentences were created from a machine translation system by replacing Chinese words with frequently used English words. The speech was collected from 77 Taiwanese speakers and the duration of the corpus is 12.1 hours.

The first and second methods that were used to collect the text data had nouns mostly, the preferred type of words for code switching. In this case, it was clear that nouns were the most frequently used code-switched words of the Taiwanese. This database is therefore mostly suitable for research on named-entity recognition.

2.7 Existing Sepedi ASR corpora

In this section, we introduce available Sepedi speech resources, namely the Lwazi and NCHLT Sepedi ASR corpora. The Lwazi ASR corpus was developed as part of the Lwazi project. Its aim was to collect annotated speech corpora for 11 South African languages [1]. The corpus contains speech data with approximately 200 speakers per

language. Each speaker contributed read and elicited speech recorded over a telephone channel.

The 2013 Sepedi NCHLT corpus [45] consists of prompted speech in Sepedi, but also includes some English speech. (The latter was generated from general English text and is not an example of actual code switching. Code switching events were not annotated.) This is a broadband corpus, collected using a smart-phone. The NCHLT corpus is discussed in more detail in Section 5.2.1, where it is first used.

At the start of this study, only the Lwazi corpus was available. During the course of this study, the NCHLT corpus was developed[2].

2.8 Conclusion

This chapter discussed background relevant to code switching and provided several techniques to model code-switched speech in ASR systems. Our focus is on multilingual speech recognition techniques, and specifically multilingual pronunciation dictionaries. Related studies on the development of code-switched speech corpora were provided. In the next chapter, the main technical methods used in the remainder of this study are discussed.

[2]The NCHLT corpus was developed by a larger team that included the author. It was not specifically collected as part of this study into code-switched speech.

Chapter 3

Methods

3.1 Introduction

This chapter presents the main modelling and analysis methods used in subsequent work. In Section 3.2, we discuss the HMM approach to speech recognition. Existing approaches used to map the embedded language phoneme set to a matrix language phoneme set follow in Section 3.3. The pronunciation assessment approach used in this thesis (Goodness of Pronunciation) is described in Section 3.4 and the classifier used in this study to measure the predictability of the phoneme labels, given specific attributes, is discussed in Section 3.5. This classifier can be used to learn rules to generate pronunciation for Sepedi code-switched speech using non-acoustic features, which is also possible using letter-to-sound rules, as discussed in Section 3.6. In Section 3.7 we describe a phone-based dynamic programming approach, which is used to measure differences between phoneme strings. It is followed by a discussion of statistical n-gram language models, used to predict word sequences.

3.2 Hidden Markov models

The HMM is a statistical technique that is popular for modelling speech signals by characterising the observed data samples of a discrete-time series [24].

3.2.1 Definition of hidden Markov model

An HMM is a statistical model that is relevant for dynamic stochastic sequences, which changes states based on the statistical properties of different piece-wise processes [25].

The HMM permits modelling of a succession of perceptions as a piece-wise stationary procedure [26]. HMMs are typically be used to model phonemes for under-resourced languages [50]

A HMM is defined as [24]:

- O = $\{o_1, o_2, ..., o_M\}$ - An output observation alphabet.
- $\Omega = \{1, 2, ..., N\}$ - A set of states.
- $\mathbf{A} = \{a_{ij}\}$ - A transition probability matrix.
- $\mathbf{B} = \{b_i(k)\}$ - An output probability matrix.
- $\pi = \{\pi_i\}$ - An initial state distribution.

Given this formulation, well-studied algorithms exist for training acoustic models from speech data, estimating the likelihood of speech given a specific model, and finding the best state sequence through an HMM, given a specific speech sequence. The HMM framework can be configured to model speech units using the continuous mixture density HMM and semi-continuous HMMs [24–27].

3.2.2 Continuous mixture density HMMs

The Continuous Mixture Density HMM (CMDHMM) selects the optimal Gaussians by predicting the probability density of a specific state [27]. The output probability density function $b_i(k)$ is a product of the multivariate Gaussian mixture density function in a CMDHMM [24]:

$$b_i(k) = \sum_{j=1}^{N} c_{ij} N(k, \mu_{ij}, \Sigma_{ij}) = \sum_{j=1}^{N} c_{ij} b_{ij}(k) \tag{3.1}$$

where $N(k, \mu_{ij}, \Sigma_{ij})$ or $b_{ij}(k)$ is a Gaussian density function; the mean vector is given by μ_{ij} and the covariance by Σ_{ij} for state i. The variable N indicates the number of mixture components, and the weight is given by c_{ij} for the kth mixture component with the condition:

$$\sum_{j=1}^{N} c_{ij} = 1 \tag{3.2}$$

3.2.3 Semi-continuous HMMs

The semi-continuous HMM expects that the mixture density functions are combined over every one of the models to create a set of shared kernels with the output probability distribution given by equation 3.3 [24]:

$$b_i(k) = \sum_{j=1}^{N} b_i(j) f(k|o_j) = \sum_{j=1}^{N} b_i(j) N(k, \mu_j, \Sigma_j) \tag{3.3}$$

where:

- o_j is the kth code word,
- $b_i(j)$ is the output probability distribution,
- $f(k|o_j)$ is the continuous probability density function for code words o_j, and
- $N(k, \mu_j, \Sigma_j)$ is a Gaussian density function with number of mixtures N.

3.3 Phoneme set construction

We studied the development of combined, multilingual systems, and specifically the process to construct a multilingual phoneme set. The most popular approaches to the development of the phoneme set and phoneme-to-phoneme mappings, are as follows:

- Combining the phoneme sets from multiple languages [30].
- Mapping the embedded phoneme set to the matrix phoneme set using IPA features directly [51].
- Mapping highly confusable phonemes from the embedded to the matrix language based on a confusion matrix obtained from an existing ASR system.
- Merging language-dependent phoneme sets using hierarchical phoneme clustering algorithms and acoustic distance measures [30].
- Mapping phonemes between source and target sequences using probabilistic phoneme mapping. [52]

3.3.1 Multilingual phoneme set

Every language is constructed from a sequence of sounds called phonemes. Phonemes describe the smallest unit of sound that can be used to differentiate between two words. When developing a phoneme set for a system where two or more languages are mixed, some phonemes will be shared among languages, and others will be unique per language.

When multilingual acoustic models are developed, a multilingual phoneme set can be constructed by pooling together phoneme sets from multiple languages [30, 31]. For English and Sepedi (given the specific phoneme sets used, see Section 4.2.2), 14 phonemes are shared between the languages. The remaining 31 and 29 phonemes are distinct for English and Sepedi languages, respectively. For this language pair, we will have a system with a combination of 45 and 43 phonemes from both languages making 88 phonemes in total. However, since there are phonemes that can be shared between the languages, 14 phonemes can be removed from the list.

The main disadvantage of both these approaches is that the size of the multilingual phoneme set will increase as the number of languages increases. The other problem is that the acoustic parameters are not shared between the languages [30, 31].

3.3.2 IPA feature-based mapping

IPA features categorise sounds dependent upon the phonetic characterisation of the individual speech sounds [51]. The entrenched IPA categorises sounds taking into account the learning of phonetic characterisation of speech sounds. The construction of the pronunciation dictionaries would then be to use the IPA features to find a good mapping from the embedded language phonemes to the matrix language phonemes.

3.3.3 Confusion matrix based phoneme mapping

A confusion matrix is a data-driven approach to measuring the distance between two phonemes. The confusion matrix is computed by applying a speech recogniser of the source language to a target language acoustic data, where the target language acoustic data has been converted into the phoneme units of the target language [53].

Given the confusion matrix B(X,Y), the likeness or distance $d(x_i, y_j)$ between target language phoneme y_j and the source language phoneme x_i is given directly by:

$$d(x_i, y_j) = B_{i,j} \tag{3.4}$$

where $B_{i,j}$ is the (i, j) entry in the confusion matrix. Then $B_{i,j} \in [0, 1]$ for $i = 1..X, j = 1..Y$.

3.3.4 Hierarchical phone clustering based mapping

Hierarchical phone clustering based mapping uses acoustic distance measures to compute the distance between Gaussian distributions obtained from each phone model; such distance measures include the Kullback-Liebler measure, Bhattacharyya distance metric, Mahalanobis measure or a simple Euclidean measure [54]. The hierarchical phone clustering based mapping is based on the distance that is measured from the statistical similarity calculation obtained from the recognition models [30].

3.3.5 Probabilistic phone mapping

A probabilistic phone mapping method is used to automatically establish a mapping between phoneme sequences using a maximum likelihood criterion [52]. This technique can function admirably with limited training data since there is a moderate number of parameters in the mapping model. The major drawback with the probabilistic phone mapping method is the use of 1-best decoding results that serve as input parameters for mapping. As such, this method is not suitable for large vocabulary jobs which take into consideration the decoded results that are influenced by both the acoustic and language models [55].

A probabilistic phone mapping is a model that is used to map phonemes between to sequences given the source sequence A and the target sequence B, where the model parameters are given by $P(a \mid b)$ where a is a phoneme in the phoneme sequence A, and b a phoneme in the phoneme sequence B [52]. The results of the phoneme recogniser and the pronunciations modelled in the system are used to estimate the probabilistic phone mapping model. Note that this model can be context-sensitive.

3.4 Goodness-of-pronunciation score

The Goodness-of-pronunciation (GOP) score was initially developed by Witt and Young in the context of phone-level pronunciation assessment [56]. It is defined as the duration-normalised log of the posterior probability which determines that a speaker has spoken a certain phone given the acoustic data. It is approximated by the difference in log likelihood of the target and best matching phone, divided by the number of frames in

the segment, that is:

$$GOP(q) = \left| log \frac{(p(x|q)}{(p(x|q')} \right| /NF(x) \tag{3.5}$$

where q is the target phone, x the observed data, $NF(x)$ the number of frames observed and q' the model that matches the observed data best. In practice, the log likelihood scores are obtained directly from the ASR system, a HMM-based one in our case, and q' identified during a free phone decode.

GOP was developed for phone-level analysis. In [57] a word-level version of GOP is defined, with two variants – frame-based and phone-based – depending on how duration normalisation is applied. In the same study, it was found that triphones provide more accurate results than monophones for word-level analysis. (Monophones are more typical of GOP scores used for phone-level pronunciation assessment).

3.5 Classification process

In pattern recognition, a classification process is used for the analysis of data by training a classifier [58]. This process requires a decision class label to be assigned to a set of attributes (also referred to as *features*). In [59], examples of good algorithms for classification are c4.5, k-Means, Support Vector Machine, Apriori, Expectation-Maximization, PageRank, AdaBoost, k-nearest neighbour, Naive Bayes, and Classification and Regression Trees. In this section we focus on the Naive Bayes classifier. The Naive Bayes classifier provides a quick and straightforward way to perform classification. It is accurate despite the assumption that the feature values are independent of each other.

Naive Bayesian is a predictive classifier that is based on Bayes theorem that estimates parameters from training data [60]. The features used to construct the classifier are considered independent, hence the term *naive*. The Bayes' theorem is given as:

$$P(X|Y) = \frac{P(X)P(Y|X)}{P(Y)} \tag{3.6}$$

There are model validation techniques used to assess the accuracy of the classifiers such as hold-out, cross-validation, leaving one out, and many others. Cross-validation is normally referred to as n-fold cross-validation, and obtains an average score by splitting data into n equal subsets. During training $n-1$ subsets are used for training and the remaining set is left for testing. The overall accuracy is obtained from the average of the n accuracy observations.

3.6 Rules generation

The development of pronunciation dictionaries is a costly and labour intensive task. Several algorithms have been used to generate rule sets to predict pronunciation for new words in monolingual pronunciation dictionaries such as Joint Sequence Models [34] and the Default & Refine algorithm [32].

The letter-to-sound (also known as grapheme-to-phoneme) rules map a sub-word pattern or a grapheme to a word [32]. The rule set generated here can range from simple to complex depending on the whether the orthography of the language is regular as in Sotho-Tswana languages or irregular as in English.

The choice of a specific algorithm depends on how it performs for the type of language in question. It can also be evaluated on the accuracy of the algorithm with respect to the number of rules required compared to other algorithms. The time it takes to generate the rules as well as the memory usage can also serve as important factors.

In this work we use the Default & Refine algorithm as it generates readable rewrite rules, which are useful when trying to understand and model the transformation process. This algorithm has performed well for the Bantu languages for which the pronunciation dictionaries were developed.

3.7 Phone-based dynamic programming (PDP) scores

Phone-based Dynamic Programming (PDP) scoring [57] is a technique that can be used to score the similarity between speech patterns. PDP aligns a recognised and force-aligned phone string using dynamic programming (DP) and calculates the alignment cost using a trained scoring matrix. The matrix is typically trained on the same data that is being evaluated. The standard PDP implementation includes various smoothing options. (See [57] for detail.)

This is similar to the DP approach described in [24]. Each pair of speech patterns (S, T) expressed as a pair of sequences of feature vectors with M and N samples per pattern, respectively:

$$S = \{s_1, s_2, s_3..., s_M\}, T = \{t_1, t_2, t_3..., t_N\} \tag{3.7}$$

The distance between these two speech patterns with minimal change can be calculated using the equation [24]:

$$D(a, b) = min_k[D(a-1, k) + d(k, b)] \tag{3.8}$$

where a and b any elements within the speech patterns S and T, and $D(s_i, t_j)$ an appropriate sample-specific distance measure.

These approaches can be used effectively for corpus validation as well as the scoring of speaker pronunciations.

3.8 *n*-gram language modelling

In this section we define the *n*-gram language model and discuss why it is important. We also discuss some language model toolkits that are used within the speech recognition systems.

3.8.1 Definition of language model

A statistical *n*-gram model is a probabilistic model that predicts the next item in a sequence. It is effective in improving the speech recognition accuracies and has simplicity as its important feature [61]. Typically, the language model would predict the word sequence by assigning a probability to each word sequence [44]. The joint probability can be used to measure the probability of the word sequence $P(W)$:

$$P(W) = P(w_1, w_2, ..., w_n) \tag{3.9}$$

Using the Chain rule, we can compute the joint probability in equation 3.9 as follows:

$$P(w_1, w_2, ..., w_n) = \prod_i P(w_i | w_1, w_2, ..., w_{(i-1)}) \tag{3.10}$$

These probabilities can be estimated on a current word, in the case of a unigram model. For an *n*-gram language model, the probability can be conditioned on the previous $n-1$ words using the Markov assumption. To estimate n-gram probabilities, a large number of text corpora is required.

The parameters of the language model can be evaluated after training to measure how good the language model is. The perplexity is the evaluation metric used to measure the language models to determine how well they predict the sequence of unseen words. The perplexity is given by the equation:

$$PP(W) = 2^{H(w)} \tag{3.11}$$

where the function $H(W)$ is given by:

$$H(W) = \frac{1}{n} logP(W_1^n) \tag{3.12}$$

3.8.2 Language model toolkits

Language model toolkits are used to train statistical n-gram language models. There are currently several language model toolkits in use for speech recognition. These language models include the SRI Language Modeling (SRILM) toolkit, MIT Language Modeling (MITLM) toolkit, and the CMU-Cambridge Statistical Language Modeling (CMUSLM) toolkit. The SRILM is a toolkit implemented with C++ class libraries, executable tools and helper and wrapper scripts to develop statistical language models that are used for speech recognition. This toolkit is freely available for research purposes [62]. The MITLM toolkit was implemented to improve parameter optimization for the modified Kneser-Ney smoothing technique for the productive estimation of statistical n-gram language models [43]. The CMU-Cambridge Statistical Language Modeling toolkit is another toolkit for the development and evaluation of statistical language models and is limited to bigram and trigram language models [42].

In this study we used the SRILM toolkit to train a basic statistical language model. For smoothing or discounting technique, we selected the modified Kneser-Ney smoothing technique for its efficiency. We used language model interpolation, as it is considered an effective approach to combining information from different text corpora during language modelling [23].

3.9 Conclusion

The main modelling and analysis methods used in later chapters were introduced here. In the next chapter we develop a baseline speech recognition system, taking initial steps to model code-switched speech.

Chapter 4

Baseline Sepedi ASR

4.1 Introduction

In this chapter, we develop an initial Sepedi ASR system as a baseline. We use data collected from Sepedi speakers consisting mainly of Sepedi phrases, but also including examples of words in other languages as spoken by Sepedi speakers. We use a standard HMM-based approach to build the ASR system and implement simple existing strategies to model the pronunciation of foreign words for the baseline system.

Our aim is to develop an initial Sepedi recognition system to evaluate later improvements. The baseline has a standard pronunciation dictionary for Sepedi and foreign words and uses a straightforward approach to map English phonemes to Sepedi phonemes. Our overall focus is on evaluating techniques that improve the pronunciation dictionaries for code-switched speech. To build a credible baseline, we experiment with the complex consonants that are plentiful in Sepedi. We split these into separate phonemes, modelling each consonant as a sequence of sounds, and evaluate the effect.

As discussed in Section 3.3 of Chapter 3, the development of speech recognition systems typically requires carefully crafted resources such as speech corpora and pronunciation dictionaries, with pronunciation dictionaries of foreign language words a specific challenge. The creation of these pronunciation dictionaries is labour intensive. Our overall goal is to minimise the amount of human effort involved, and also minimise potential human error introduced during the development process.

The chapter is structured as follows: in Section 4.2 we describe the Sepedi ASR system development process, including the resources used. In Section 4.3 we describe an optimisation of the ASR system by splitting complex consonants. The results of evaluating the initial systems are discussed in Section 4.4.

4.2 Developing an initial Sepedi recogniser

In this section, we describe the data used to train the recogniser (Section 4.2.1), the pronunciation dictionary used in our experiments (Section 4.2.2) and the development process of the baseline system (Section 4.2.3).

4.2.1 Data

The baseline system was developed using the Lwazi Sepedi ASR corpus [63]. As introduced in Section 2.7, the corpus contains speech data from each of the eleven official languages of South Africa. Approximately 200 speakers per language (2,200 speakers in total) contributed read and elicited speech, recorded over a telephone channel. Each speaker produced approximately 30 utterances; 16 of these were randomly selected from a phonetically balanced corpus and the remainder consisted of short words and phrases.

While small, the corpus is ideal for a baseline for two reasons: Firstly, this corpus contains Sepedi utterances as spoken by Sepedi native speakers, as well as utterances containing English words. The English words are relevant in that they will help us analyse appropriate strategies to model the pronunciations of code-switched speech. Secondly, as earlier recognition results using the Lwazi ASR corpus are available [1, 19], this allows us to evaluate the correctness of our baseline system

TABLE 4.1: Lwazi Sepedi ASR corpus.

	Number of unique phonemes	Number of speakers	Total number of utterances	Number of unique words
Lwazi Sepedi	45	190	5 640	3 314

Table 4.1 provides statistics related to the Lwazi Sepedi ASR corpus: the number of phonemes in the phoneme set, the number of speakers, the total number of utterances and the number of unique words. Out of 190 speakers, 92 are males and 94 are females, with four speakers of unknown gender.

The Lwazi Sepedi ASR corpus is a useful resource for the initial analysis of the pronunciation of English by Sepedi speakers. At the time of this analysis, the Lwazi corpus was the only Sepedi/English ASR corpus available. This analysis was also used to obtain corpus counts: to determine the extent of actual code switching events contained in the corpus.

4.2.2 Dictionary development

We used the Sepedi subset of the Lwazi ASR corpus and developed our pronunciation dictionary by extending the Lwazi Sepedi pronunciation dictionary [64]. The phoneme sets used by the Lwazi dictionaries are different for Sepedi and English, as shown in Tables A.1 and A.2. English has 45 phonemes, of which 14 are shared; 31 phonemes are unique. Sepedi has 43 phonemes, and 29 are unique.

Each of the Lwazi dictionaries is accompanied by a set of letter-to-sound prediction rules, which can be used to predict words not contained in the original dictionary. These letter-to-sound prediction rules were produced using the Default&Refine algorithm, which has been shown to learn well from small data sets when extracting small-scale rule sets [32].

We developed an extended version of the dictionary as follows:

- All words in the ASR transcriptions were categorised according to language origin (Sepedi, English or other) and type of word (general word or proper name). An initial language categorisation was performed automatically, using existing English and Sepedi word lists. The type of word categorisation was extracted directly from the transcriptions (where proper names are capitalised). This capitalisation resulted in six word lists, which were then reviewed and corrected manually.
- Pronunciations for Sepedi words (both general words and proper names) were automatically generated based on the Lwazi Sepedi letter-to-sound rules.
- Pronunciations for English and other words were similarly generated using the Lwazi English letter-to-sound rules. A mapping was then defined to map each English phoneme to its closest matching Sepedi phoneme. This mapping was based on linguistic knowledge. Where no close match could be found, and an English phoneme occurred sufficiently frequently in the corpus, the phoneme inventory was extended. The mappings were refined based on a number of iterations: Table 4.2 includes the final mappings selected.
- The problematic word lists (all proper names and general words that were neither of Sepedi nor of English origin) were reviewed manually, and pronunciation errors found were manually corrected.

Table 4.3 lists the number of words of each category contained in the final dictionary. ‘Other’ words were mostly of Afrikaans, isiZulu and Tshivenda origin, but were not individually categorised per language.

TABLE 4.2: Phoneme substitution choices for English words occurring in the Sepedi corpus.

Substitutions			
from	to	from	to
{	E	Oi	O i
3:	E	p	p_h
A:	a	Q	O
ai	a i	r\	r
au	a u	t	t_h
d	l`	T	f
D	l`	tS	tS_h
e@	e @	@u	O
g	k_>	u:	u
@i	@ i	u@	u
i	i	U	u
i@	i @	v	B
k	k_h	z	s
O	O	Z	d0_Z
Additions			
@		b	

TABLE 4.3: Number of words in Lwazi Sepedi corpus-based pronunciation dictionary.

	Sepedi	English	Other
General words	2751	132	41
Proper names	476	88	51

The resulting dictionary contained both English and Sepedi phonemes. This dictionary was extended with only two phonemes (i.e. *@* and *b*). These phonemes were found to occur most frequently in the Sepedi corpus. Even though proper names were a challenge for letter-to-sound prediction rules, no special consideration was made in predicting the pronunciation of English proper names.

4.2.3 ASR system development

The recogniser followed a standard HMM design, as introduced in Section 3.2. It was developed using the Hidden Markov Model Toolkit (HTK) [65]. HTK is widely used for training and testing of HMMs for research fields such as speech recognition and speech synthesis. During the course of this study, HTK was surpassed in popularity by the newer Kaldi toolkit [66]. For comparative purposes, HTK was used throughout this thesis.

Acoustic models consisted of cross-word tied-state triphones modelled using a 3-state continuous density HMM. A 6-mixture multivariate Gaussian modelled each HMM state

distribution with a diagonal co-variance matrix. The 39-dimensional feature vector consisted of 13 static Mel-Frequency Cepstral Coefficients (MFCCs) with 13 delta and 13 delta-delta coefficients appended. The final preprocessing step applied Cepstral Mean Normalisation (CMN) which calculates a per utterance bias and removes it. The different HMM state distributions were estimated by running multiple iterations of the Baum-Welch re-estimation algorithm. Once the triphone acoustic models were trained, a 40-class semi-tied transform was calculated to improve further acoustic model robustness.

4.3 Optimising the Sepedi recogniser

The Sepedi phoneme inventory – like that of most Sotho-Tswana languages – contains a large proportion of complex consonants. Large phoneme inventories typically produce less accurate ASR systems when training data is scarce. Possibly even more important for the current work, these complex clusters of sounds increase the complexity of both error analysis and cross-lingual comparisons.

The phoneme set from the Lwazi Sepedi pronunciation dictionary contains 15 complex consonants, of which 9 are affricates. We, therefore, investigate approaches to the acoustic modelling of Sepedi affricates for automatic speech recognition (ASR) systems. Specifically, we determine whether affricates are better represented as a single segment (a single complex cluster of consonants) or whether it is possible to model them as a sequence of segments constituted from individual consonants. Only one affricate (voice post-alveolar) *d_0z* was not considered for splitting due to its composition.

Selecting an appropriate phoneme inventory is one of the important decisions to be made when developing speech technology systems. The selected phoneme inventory is not necessarily an exact match to what is proposed by linguistic theory but is optimised for a particular technological application in mind. For example, context-dependent phonetic effects are typically not modelled explicitly in the pronunciation dictionaries of speech recognition systems, as the statistical acoustic models were trained to take context-dependent effects into account.

We approached this task by first reviewing linguistic arguments for and against the splitting of complex clusters, and evaluated the acoustic evidence in support of the two approaches using spectrograms. Based on this analysis and considering basic phoneme statistics, we selected what we considered to be the most appropriate modelling strategies, and compared the ASR accuracies obtained when implementing these.

4.3.1 Complex consonants

In phonological terminology (as used by [67] quoting [68]) a *simple segment* is a sound characterised by a single place and manner of articulation. For example in the English word 'task', both /s/ and /k/ are considered to be simple segments, with the /s/ an alveolar fricative, and the /k/ a velar plosive (using X-SAMPA notation, as in the rest of this chapter). By comparison, a *complex segment* is characterised by at least two articulatory features, with two or more oral tract constrictions occurring simultaneously. From a phonological perspective, there is, therefore, a distinction between a sequence of simple segments (compare /s k/ in 'task') and a complex segment (compare /tS/ occurring in 'church').

Comparable definitions exist in phonetics, with /s k/ referred to as a sequence, and /tS/ as a double articulation. The mapping from the phonological to the phonetic term is not direct, but phonologically or phonetically motivated. Additional terms describe the continuum between a simple and complex segment, a distinction we did not make for the purposes of this study.

Sepedi, Sesotho and Setswana all contain a large number of consonants, traditionally considered complex segments. These include the affricates: /tS_>/, /tS_h/, /ts_>/, /ts_h/, /kx/, /pS_>/, /pS_h/, /ps_>/, /ps_h/ and /d_0Z/ as well as the fricatives /p\s/, /p\S/ and /BZ/ [69]. (All fricatives are formed by forcing air through a narrow constriction, while affricates begin as plosives – consonants produced by stopping the airflow completely – but release as fricatives.)

In [67], Zerbian reviews the phonetic and phonological considerations concerning the segmental status of these sequences in Tswana. Some phonetic evidence points towards a single segment analysis, as '*a constriction and a narrow constriction at the same time naturally exclude each other*', while results from duration studies were found to be contradictory. From a phonotactics perspective, there is some value in viewing these as complex clusters, as that allows for a simpler syllable structure analysis. Phonotactic distribution patterns, though worth investigating further, do not point directly towards one outcome or the other. While Zerbian argues (in conclusion) that the stop sequences should be analysed as complex segments, she points out that the evidence for such an analysis is relatively weak.

4.3.2 Affricate splitting

We utilised the Lwazi TTS corpus, which contains carefully articulated audio obtained from a single speaker under studio conditions, as well as phonemic alignments of the

audio [70]. We found that many of the realisations of the traditionally considered complex consonants could be viewed as a sequence of sounds, though the components are visually more or less distinguishable in different realisations.

For illustration, we provide the spectrogram of /ts_>/ and /tS_h/, where the components are clearly visible (see Figure 4.1 and 4.2). The same phenomena were visible in the Lwazi ASR corpus (described in Section 4.2.1), which contained greater speaker diversity.

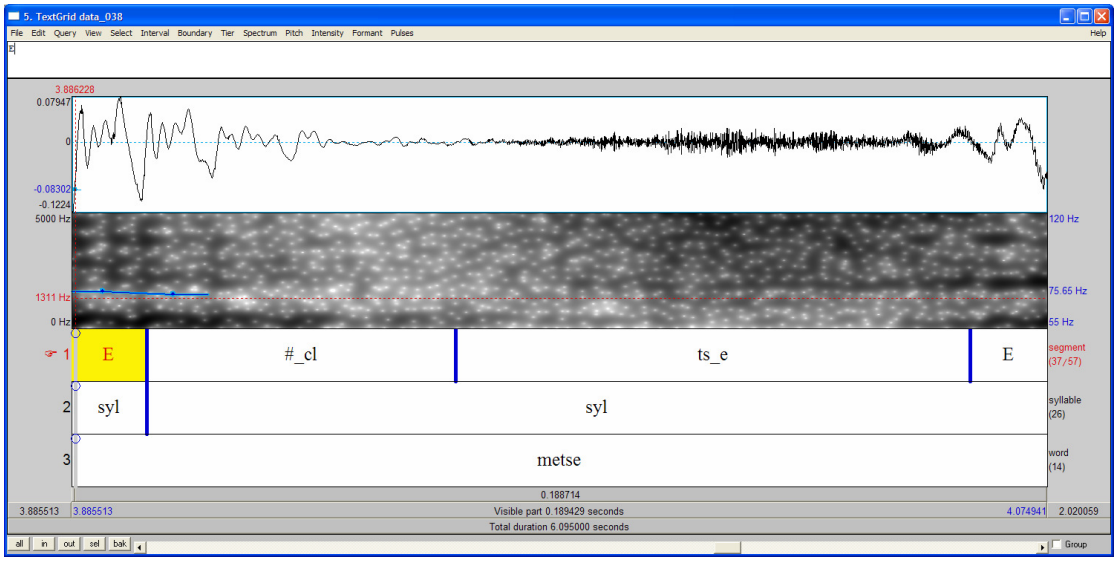

FIGURE 4.1: Spectrogram of /ts_>/

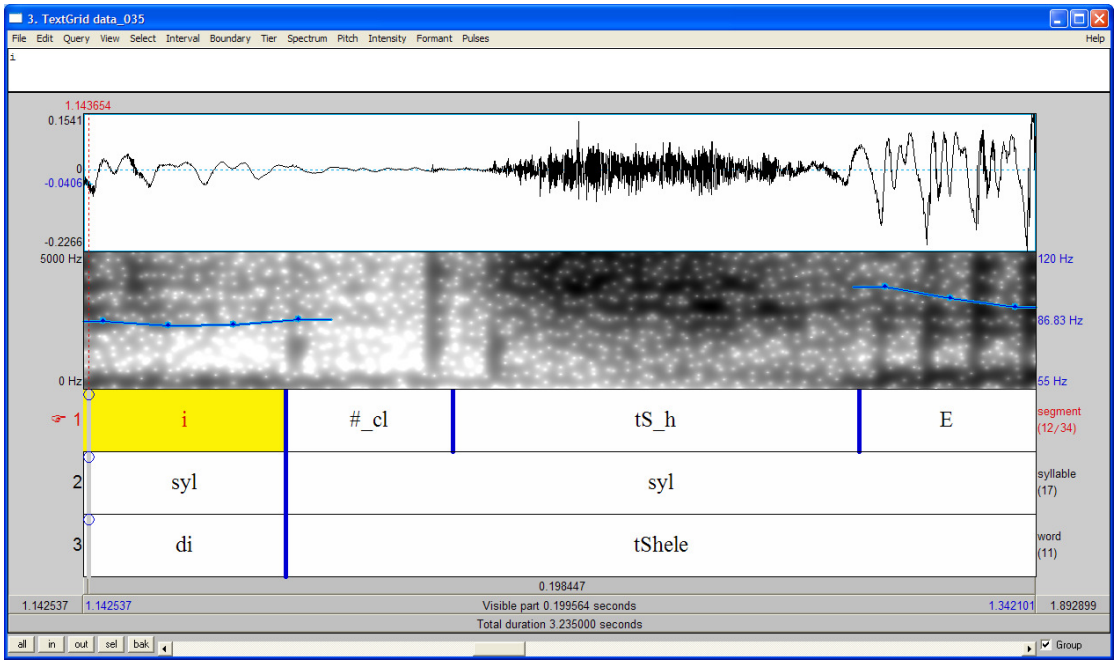

FIGURE 4.2: Spectrogram of /tS_h/

Using this information, we developed two modelling approaches for the splitting of the

Sepedi affricates: (1) splitting each complex consonant into two parts and, more controversially, (2) modelling each aspiration separately. Table 4.4 shows the possible substitutions using the first approach. This resulted in two new phonemes being created: /s_h/ and /S_h/. Table 4.5 contains alternative substitutions using the second approach. Note that other aspirated consonants such as /p_h/ and /tl_h/, also required separate aspiration modelling.

TABLE 4.4: Possible phoneme substitutions when splitting all unvoiced affricate and two fricative sequences.

Original	Substitutions	
ts_>	t_>	s
tS_>	t_>	S
ts_h	t_>	s_h
tS_h	t_>	S_h
ps_>	p_>	s
pS_>	p_>	S
ps_h	p_>	s_h
pS_h	p_>	S_h
p\s	f	s
p\S	f	S
kx	k_>	G
Additions		
s_h		
S_h		

4.3.3 System development

We trained the baseline ASR system using the HTK toolkit as described in Section 4.2.3. The performance of the system was measured using phone recognition.

Phone recognition was performed using a flat language model (all phones were considered equally likely at all times), and phone accuracy was measured. This phone recognition accuracy was a conservative measure: an accuracy of approximately 60% when performing flat phone recognition can translate into an accuracy of 90% when performing word recognition for a small ($<$ 100 word) vocabulary. Phone recognition provides a more robust measure than word recognition, which is heavily influenced by the recognition vocabulary. However, care should be taken when comparing the phone accuracy of systems utilising different phoneme sets, as the change in number of phonemes has a direct effect on reported accuracy (irrespective of whether the system is more accurate or not). We, therefore, created an *adjusted baseline accuracy* per experiment: for each phone to be substituted, we multiplied the number of correct occurrences, insertions, deletions and substitutions (of each phone individually) with the number of components in its

TABLE 4.5: Additional phoneme substitutions possible when modelling aspiration separately.

Original	Substitutions		
ts_>	t_>	s	-
tS_>	t_>	S	-
ts_h	t_>	s	h
tS_h	t_>	S	h
ps_>	p_>	s	-
pS_>	p_>	S	-
ps_h	p_>	s	h
pS_h	p_>	S	h
p\s	f	s	-
p\S	f	S	-
kx	k_>	G	-
k_h	k_>	h	-
p_h	p_>	h	-
t_h	t_>	h	-
tl_h	tl_>	h	-

substitution. While this did not change the accuracy dramatically, it provided a more directly comparable measure.

Accuracy was measured using 10-fold cross validation. The set of 190 speakers was divided into ten folds. Each training set consisted of nine folds (171 speakers) and the test set consisted of the remaining 19 speakers (per cross-validation run).

4.4 Results

The results previously obtained for Sepedi ASR were reported in [1]. We show the same results in Table 4.6. These results are used here to determine if the results we obtain are in the same ballpark.

TABLE 4.6: Phone recognition correctness and accuracy for Lwazi Sepedi corpus [1].

	%Corr	%Acc
Lwazi Sepedi	66.44	55.19

Some of the complex consonants considered were quite rare, as shown in Table 4.7. We ran a number of experiments, splitting a subset of phonemes in each experiment.

Specifically, we did the following:

- **Experiment ts**: We substituted /ts_>/, /ts_h/, /tS_>/ and /tS_h/ according to Table 4.4.

TABLE 4.7: Frequency counts of simple and complex consonants.

Simple		Complex	
Phoneme	Frequency	Phoneme	Frequency
h\	536	ps_>	0
f	2 211	pS_h	19
S	2 301	p\bS	23
t_>	3 037	p\bs	30
p_>	3 677	pS_>	70
G	5 441	d_Z	342
s	6 109	BZ	642
B	6 407	ps_h	765
k_>	10 066	ts_>	986
		ts_h	429
		tS_h	510
		kx	1 348
		tS_>	3 289

- **Experiment kx**: We substituted /p\s/, /p\S/ and /kx/ according to Table 4.4.
- **Experiment ps**: We substituted /ps_>/, /ps_h/, /pS_>/ and /pS_h/ according to Table 4.4.
- **Experiment asp**: We performed all the substitutions listed in Table 4.5.

The results of these experiments are shown in Table 4.8. The number of phonemes in each phoneme set, the phone accuracy and adjusted baseline accuracy (see Section 4.3.3) are provided. The standard error of the mean phone accuracy across the 10 folds is also listed as a measure of statistical significance. The last column shows the absolute improvement in phone accuracy, from the adjusted baseline to the new result.

TABLE 4.8: Phone recognition accuracy using various modelling approaches.

Approach	Number Phonemes	Adjusted baseline (%)	Phone accuracy (%)	$\pm\sigma_{10}$	Improvement (%)
baseline	45	57.45	57.45	0.81	-
exp ts	43	57.00	57.54	0.85	0.54
exp kx	42	56.93	57.62	0.86	0.69
exp ps	37	56.78	57.84	0.89	1.06
exp asp	32	**56.10**	**58.37**	0.84	2.27

From the results in Table 4.8, it is clear that the recognition accuracy did not degrade as the phonemes were reduced from 45 to 32. In fact, there was a small but significant improvement in recognition accuracy as the phonemes were reduced. The good results obtained using the (unconventional) aspiration modelling approach were most surprising.

4.5 Conclusion

In this chapter, we developed a baseline Sepedi ASR system using the Lwazi Sepedi ASR corpus. The pronunciation dictionary used to develop this system was implemented in two steps: first predicting the pronunciation of Sepedi and English words using letter-to-sound rules for each language; and then splitting complex consonants into a sequence of simple sounds. The English phonemes were also mapped to Sepedi phonemes.

The phoneme mapping approach resulted in the Sepedi phoneme set being extended with two additional English phonemes that appeared frequently in the Sepedi ASR corpus.

We investigated the effect of modelling complex consonants in Sepedi (mostly affricates, but also considering two fricatives) as sequences of sounds. We found that the simpler modelling strategy is not detrimental and that, in fact, recognition accuracy improves with a small but significant margin. Additional benefits of the simpler modelling approach include simpler dictionary development and transcription processes: both of which are labour intensive.

In Chapter 7, the English to Sepedi phoneme mapping strategy used here is further explored to find better modelling techniques for the English phonemes. The results obtained in this chapter also serve as a basis for evaluating different modelling techniques.

In this chapter, we analysed the implications of embedded English speech for a Sepedi recogniser using the Lwazi Sepedi corpus. We realised that a more appropriate corpus was required, specifically a corpus that includes true code switching events. This would allow more detailed analysis and modelling. The details of the data collection are discussed next in Chapter 5.

Chapter 5

Corpus development

5.1 Introduction

Sepedi can be regarded as one of the under-resourced languages. An extensive definition of under-resourced language is given in [50], which is defined as a language with lack of resources such as pronunciation dictionaries, acoustic data, etc. As no existing corpora were available to support the analysis of Sepedi-English code-switched speech, we developed three such corpora specifically for this purpose.

The first was developed by selecting relevant data from a more general corpus of Sepedi speech. The second was selected by observing true intra- and inter-sentential code switching events in radio broadcasts and the third by presenting custom-designed code switching prompts to first language Sepedi speakers.

The first, referred to as the National Centre for Human Language Technology (NCHLT) Sepedi-English Code-Switched Subset (NSECSS) corpus was developed by using data obtained from the 2013 Sepedi NCHLT speech corpus. This is a more general corpus of both Sepedi and English speech, spoken by Sepedi first language speakers. It aims to provide samples of the pronunciations of English utterances as spoken by Sepedi speakers.

The second corpus, the Radio Broadcast corpus, was created by recording and transcribing a number of Sepedi radio broadcasts. It contains natural examples of intra- and inter-sentential code switching and provides additional information about the factors involved when code switching occurs naturally.

The final corpus, the Sepedi Prompted Code-Switched (SPCS) corpus was developed using prompts derived from the Radio Broadcast corpus, and aimed to capture some of

the speaker-specific pronunciation variability that occurs when the same code-switched event is presented to different speakers.

In this chapter, we describe the development of the Sepedi-English code-switched corpora in detail and discuss the process used to develop and evaluate each corpus.

5.2 The NSECSS corpus

Different types of speech recognition corpora exist, one being a prompted speech corpus. During the development of a prompted speech corpus, carefully selected written material is provided to users who are then recorded, typically in a somewhat controlled environment. The creation of such a corpus involves four stages:

1. Designing the speech prompts necessary to be read by the participants.
2. Collecting actual speech data samples.
3. Annotating and tagging.
4. Evaluating the tagged acoustic data.

For the first corpus, we aimed to develop a resource containing English as spoken by Sepedi speakers (whether in a code-switched setting or not) by re-using data from an existing Sepedi prompted speech corpus. Specifically, we used data from the NCHLT corpus, discussed in more detail below.

5.2.1 The NCHLT corpus as source material

As introduced in Section 2.7, the NCHLT speech corpus is a broadband corpus sponsored by the Department of Arts and Culture (DAC) to promote the use of the official languages of South Africa. For each of the 11 official languages, 50 hours of speech corpus was collected from approximately 200 native speakers [71]. These corpora are made freely available to the public for use.

We used the 2013 Sepedi NCHLT speech corpus (one of the NCHLT corpora) as a source corpus for this work. The corpus was collected with reasonably balanced male and female participants. There was no special consideration for Sepedi dialects spoken by the participants. During the development of the prompts, the number of words per prompt was selected to optimise the coverage of the vocabulary.

The Sepedi NCHLT corpus was collected using a locally developed smart-phone based speech data collection tool, Woefzela [45]. This tool has two advantages over similar tools available: (a) no Internet connection is required during utterance collection, and (b) it performs necessary quality checks during recording, making efficient use of limited recording opportunities. Both these factors are important when collecting data in outlying rural areas in South Africa.

The Sepedi NCHLT corpus consists of prompted speech, mostly in Sepedi, but also includes some English speech (generated from English text) as produced by Sepedi first language speakers. The corpus consists of 11 196 unique word tokens (with a total of 294 081 word tokens) produced by 210 speakers. The total corpus duration is approximately 56 hours [71].

The NCHLT corpus was verified to ensure that the transcriptions were true representations of the audio utterances. The verification process, described in [72], makes use of phone-based confidence scoring to identify either poor quality audio or poor transcriptions. Affected utterances were removed. Manual validation of a subset of the resulting corpus predicted a word-based accuracy of between 71.1% and 88.2% for the languages evaluated (isiNdebele and Afrikaans).

One unanticipated effect had to be compensated for when creating the subset corpus. As the same team did the data collection for the Sepedi NCHLT corpus and the SPCS corpus, this resulted in some unexpected speaker and content overlap. (Speakers and content from the SPCS corpus were used to supplement the Sepedi NCHLT corpus.) All these utterances and speakers were, as a result, removed prior to using the Sepedi NCHLT corpus as source material. Only 206 speakers were retained, resulting in the nchlt_206 corpus, that we use from here onwards. This corpus consists of all Sepedi and English data and excludes any possible overlap with the SPCS corpus. From the nchlt_206 corpus, we created two subsets: the nchlt_206_sep consisting only of pure Sepedi utterances and the nchlt_206_eng corpus that includes, at least, one English word in each utterance. For later use, we also partitioned the Sepedi NCHLT corpus into a train, development and test set, as shown in Table 5.1.

TABLE 5.1: The distribution of male and female speakers, and the duration of the train, test, and development sets of the Sepedi NCHLT corpus.

	Male	Female	Duration (hrs)
Train	95	85	49.0
Test	10	10	5.4
Dev	3	3	1.5

5.2.2 Data collection

Data from the Sepedi NCHLT corpus [45] was used as source material. As mentioned in Section 5.2.1, this is a broadband speech corpus with each speaker producing on average approximately 460 usable utterances. From this existing corpus, a subset of English words spoken by Sepedi speakers was selected, as described 5.2.2.1.

Event selection consisted of two steps: (1) selecting appropriate text samples and (2) then identifying the corresponding audio.

5.2.2.1 Selecting text samples

Existing Sepedi and English word lists [64] were used to perform initial language identification on the Sepedi NCHLT corpus. Unknown words were categorised manually. The number of utterances containing Sepedi or English words is listed in Table 5.2. Most utterances were Sepedi only, but multi-word (sentences consisting of more than one word) English sentences were also relatively common. A relatively large set of utterances contained partial words and acronyms (indicated as 'Other').

Table 5.2: Number of Sepedi, English, mixed, single, and other utterances in Sepedi NCHLT corpus.

Sepedi multi-word	Mixed	English multi-word	English single words	Other
92 520	630	6 115	20	1 643

From the available associated transcriptions of the Sepedi NCHLT corpus, a subset was selected for the code switching corpus. To ensure the corpus is useful in the automated analysis two types of events were selected:

1. English words contained within Sepedi utterances: the top 50 most frequent words that were produced by at least ten different speakers were selected. These words were prepared for manual classification.

2. Sepedi words with known pronunciations: again, the top 50 most frequent words that were produced by at least ten different speakers were selected. These were extracted as reference values, but not manually classified.

5.2.2.2 Identifying corresponding audio

In order to identify the audio matching the selected text portions, a Sepedi recogniser (Sepedi ASR system) was built. The recogniser followed a standard HMM design.

Acoustic models consisted of cross-word tied-state triphones modelled using a 3-state continuous density HMM. Each HMM state distribution was modelled by a 6-mixture multivariate Gaussian with a diagonal covariance matrix. The 39-dimensional feature vector consisted of 13 static Mel-Frequency Cepstral Coefficients (MFCCs) with 13 delta and 13 delta-delta coefficients appended. Cepstral Mean Normalization (CMN) and a 40-class semi-tied transform were applied. Sepedi only material was selected from the Sepedi NCHLT corpus to train the recogniser. The corpus was partitioned into a training and testing set. The number of speakers was 93 and 20 for the train and test set, respectively. There were no English utterances in the sets.

As a dictionary is required to build the Sepedi recogniser, and an analysis of the pronunciations of English words is the focus of the study, no assumptions were allowed about how the English words should be pronounced. For this reason, no English audio samples were included in the training corpus. The dictionary was developed using the Lwazi letter-to-sound rule set [32, 64] and contains only standard word pronunciations as found in the Sepedi corpus. The initial pronunciations were predicted using the same Sepedi letter-to-sound rules as discussed in Section 4.2.2. We further split the affricates and that reduced the number of Sepedi phonemes from 45 to 32 (see Section 4.4 for details).

As the current version of the Sepedi NCHLT corpus does not have time alignments, forced alignment was used to generate approximate alignments that matched the transcriptions with the audio. During forced alignment, a garbage model was used to absorb (filter out) non-matching audio from the utterances. The garbage model consisted of a 3-state global HMM (trained on all the training data) and a silence model combined, with free transitions allowed internally to the garbage model [72]. These time alignments were used to identify the location of target audio segments and to extract them automatically from the corpus. This process resulted in a set of words, each associated with multiple audio clips as produced by various speakers, but not yet manually reviewed.

5.2.3 Verification

Four participants were recruited to assist with the review process. The participants had different language backgrounds. One participant was a native speaker of English (eng1), and three were native speakers of Sepedi (sep1, sep2 and sep3) who speak English. Different participants performed different sub-tasks, as detailed below.

The primary task of the participants was to match the audio to the transcription and to determine if there are no mismatches between the audio and the transcriptions. In addition, we were also interested in understanding whether the orthographic or audio errors were because of human error and wished to capture a number of additional annotations.

The participants were as a result asked to perform the following tasks, with each task performed in isolation:

1. Language verification: verifying whether the language identity of a given word was correct.
2. Word categorisation: classifying whether or not a word was a proper noun.
3. Utterance/audio matching: determining whether the text and audio match at the utterance level.

The first two tasks used a simple spreadsheet system. During language verification, participants were requested to validate previously assigned language categories; while in the word categorisation, participants were requested to classify whether a word was a proper noun or not. In both tasks, the participants could also give comments if there was anything they wanted to point out about a particular word, e.g. indicate if the word was misspelled, an acronym or abbreviation. The same set of words was given to all the participants. For the last task, the Praat environment [73] was used to present tasks to the participants. For each item to be evaluated, the text and corresponding audio were presented to the participants, who could then select from the available options (see below). Participants were allowed to replay an audio segment as often as required.

In the utterance/audio matching task, participants were presented with text and audio versions of whole utterances and were required to select from three options: *match, no match* and *not sure*. In this case, *match* indicated an exact match. In cases where there were partial matches, participants were advised to choose *no match*. It was only when the participant could not make a choice that the *not sure* option was used.

A total of 940 English words and 424 Sepedi words were validated in text format. The participants also categorised the same number of English and Sepedi words. 378 utterance-level audio samples were reviewed and validated.

5.2.3.1 Results: Transcription verification

Table 5.3 shows the verification results for both the English and Sepedi words. (The first participant did not participate in the Sepedi verification task.) The 'Blank' category

refers to words for which the participants did not provide a verdict. The same table shows the number of words for which the four participants were in agreement for each of the verification categories. The disagree rows in the table correspond to words for which one or more of the participants did not agree with the others. The values in the table indicate that the participants' verdicts were the same for the majority of the valid words.

It is observed that the participants did not agree on 65 English words. It is interesting that 93% of the participants agreed on the validity of English words. The disagreement observed was due to two main reasons:

- Words that are proper names are not easy to validate as valid English words.
- The orthography of certain words is the same in both the target and foreign language, for example, *be*.

We observed 79% agreement on the validity of Sepedi words. There were a number of borrowed words that contributed to higher disagreement among participants. The results in the table confirm that the majority of the words in the respective word lists were valid within-language words. However, they also indicate that the participants did not agree with all their verdicts.

TABLE 5.3: Verification of English and Sepedi words; Agreement of English speaker on verification of English words; Agreement of Sepedi speakers on verification of Sepedi and English words.

	Lang	Valid	Invalid	Blank	Total
eng1	English	917	22	1	940
	-	-	-	-	-
sep1	English	926	14	0	940
	Sepedi	400	24	0	424
sep2	English	934	4	2	940
	Sepedi	400	22	2	424
sep3	English	901	39	0	940
	Sepedi	333	91	0	424
agree	English	874	1	0	875
disagree		65	0	0	65
agree	Sepedi	326	10	0	336
disagree		87	1	0	88

5.2.3.2 Results: Utterance matching

Audio verification was performed per utterance (full sentence). Table 5.4 shows the verification of English utterances, agreement among the participants, and their level

of agreement. The participants were in agreement on the verification of 283 out of 378 utterances. There was no consensus on the remaining utterances on whether they match the transcriptions or not. The *not sure* option is used when the participant cannot decide between the *match or do not match* options. Participant disagreement is surprisingly high for a task that was expected to be fairly straightforward. An analysis of results shows that in most instances, the transcription did match the audio, but that either the beginning or end of the audio was cut: participants differed with regard to the leniency with which they accepted these utterances.

TABLE 5.4: Verification of English utterances, agreement and disagreement between participants.

	Match	Do not match	Not sure	Total
eng1	338	40	0	378
sep1	306	68	4	378
sep2	338	40	0	378
sep3	296	81	1	378
Agree	266	17	0	283
Disagree	77	0	0	77

In an initial experiment (not described here), an additional verification of words in isolation was performed: each text phrase was auto-aligned to its matching audio, and the per-word audio snippets presented to the participants. This task turned out to be difficult for the participants to perform, resulting in very low inter-subject agreement. For this reason, phrase-level rather than word-level verification was used during corpus construction and evaluation.

The final corpus generated from the verification process has 266 utterances out of a total of 378. These were the utterances that all the participants agreed on during the verification process.

5.3 The Radio Broadcast corpus

The second consideration was to develop a speech corpus from true examples of code switching. The objective was to understand the frequency and the mechanisms of code switching.

The radio broadcast corpus is a corpus that was collected by recording the speakers on the radio during a broadcast. We were interested in natural code-switched speech where speakers were not influenced when and how to switch between languages. We only transcribed portions that contained code-switched speech and these were used further for

analysis. The data was collected from the DJs [1] (radio presenters), radio/studio guests, news readers, callers (people calling into the studio during discussions), and actors (from short dramas). The transcribed data was validated by Sepedi first language speakers.

5.3.1 Data collection

A number of programmes that are broadcast between 7 am and 4 pm were selected to be recorded. These included a general breakfast show, youth and current affairs programmes and an afternoon show. The recorded audio files were reviewed, and orthographic transcriptions created manually. Once transcribed, the prompt development material was divided into three portions, namely, code-switched, Sepedi and the other portion (containing music and adverts). For the code-switched portion, the start and end times of the utterances or phrases that contained code-switched words were captured. In many instances, it was not easy to determine sentence boundaries (as is typical of conversational speech).

In such cases, sentence boundaries were estimated based on naturally occurring phrases within the range of sentence lengths, as observed in the set of more clearly delineated sentences. The start and end times of Sepedi portions that did not contain code-switched words were also marked, but the corresponding transcriptions were not created. Music and adverts were marked as other and were not considered for analysis in this study. The transcriptions corresponding to the code-switched speech sections were used to compile a list of phrases. These phrases were subsequently used as prompts to collect an acoustic database of code-switched speech discussed in Section 5.4.

5.3.2 Verification

First language speakers of Sepedi validated the transcriptions as well as the word lists that were extracted from the transcriptions of the radio broadcasts. The word lists were classified as English and semi-transformed. This term is used to refer to words that are evidently of English origin, not part of existing Sepedi vocabulary and transformed from the original English so that they are no longer the exact English words, e.g., *diwheelchair*. The duration of the sections of speech tagged as instances of code switching was calculated to quantify the frequency of code switching.

To analyse the mechanisms of code switching, a word list was created from the transcriptions of the code-switched data. A number of labels were assigned to each word in the word list:

[1] Disc jockey (DJ) is a radio presenter who plays music to radio audience

1. English words with and without a Sepedi alternative.
2. Semi-transformed words.
3. Part of speech per code-switched word.
4. Phrases containing single and multi-word examples of code switching.

To be able to determine some of the reasons why code switching occurs, events were identified where English words were used in conjunction with Sepedi words. Frequency counts were then compiled for events where English was used for emphasis and events where English was used because a Sepedi alternative was not known.

5.3.3 Prompt preparation

The prompts consisting of both Sepedi and English words were prepared for acoustic data collection. There were 450 prompts that were created to be recorded by each speaker. Each prompt consisted of 5 words, since Sepedi is a disjunctive language.

5.4 The SPCS corpus

The SPCS corpus is a Sepedi code-switched corpus consisting of audio and transcription files collected using prompts generated from natural code switching examples. This corpus was collected from native Sepedi speakers who can speak English as a secondary language. We describe the process followed to develop the corpus that includes its design (Section 5.4.1), data collection (Section 5.4.2), and corpus evaluation (Section 5.4.3).

5.4.1 Design

Natural examples of code-switched speech were required. It is impossible to collect samples of all possible types of code-switched speech, and there was a need to select an environment where we could understand the prevalence of code switching as well as the factors involved when code switching is used. The development of the SPCS corpus consisted of the following main steps:

1. Radio broadcast recording.
2. Code-switched event annotation.
3. Prompt preparation.

4. Prompted speech collection.

5. Further annotations.

Steps 1 - 3 were discussed in Section 5.3, and in this section we discuss the prompted speech collection, specifically, how the SPCS corpus was collected and evaluated.

5.4.2 Data collection

The prompts that were derived from code-switched transcriptions of radio recordings were used to collect the SPCS corpus. The data was collected using Woefzela. Every recorded prompt had to undergo quality checks during the recording phase. In cases where a recording did not meet all the required criteria, an additional prompt was loaded for the participant to record. This corpus was collected from native speakers of Sepedi, who can also speak English. The prompts consisted of both Sepedi and English text and audio samples that were obtained from conversational speech. Only adult conversations were considered, and the level of literacy was not measured. Twenty speakers (12 males, 8 females) each read approximately 450 utterances, resulting in 10 hours of prompted speech.

5.4.3 Verification

We describe the process of verifying the SPCS corpus. This process entails the removal of the utterances that may hurt the system during training or testing. Such utterances would include background noise, low volume, non-speech sounds (e.g. pressing a button), and blank or empty utterances. There are two main steps that are followed to evaluate the corpus. Firstly, we train acoustic models using the same data intended to be evaluated. Then, using a phone-based dynamic programming (PDP) scoring approach [57], the corpus is evaluated by aligning the reference string with the decoded string and their scores measured. Utterances with PDP scores less than the defined threshold are then discarded. The scores observed for all utterances are shown in Figure 5.1).

5.4.3.1 Acoustic model development

The acoustic models were trained by following standard HMM design as discussed in Section 5.2.2.2. The pronunciation dictionary was developed using Sepedi G2P rules for all the words (English and Sepedi). The English words that had erroneous pronunciations were manually corrected. A 4-fold cross-validation approach was used, and the

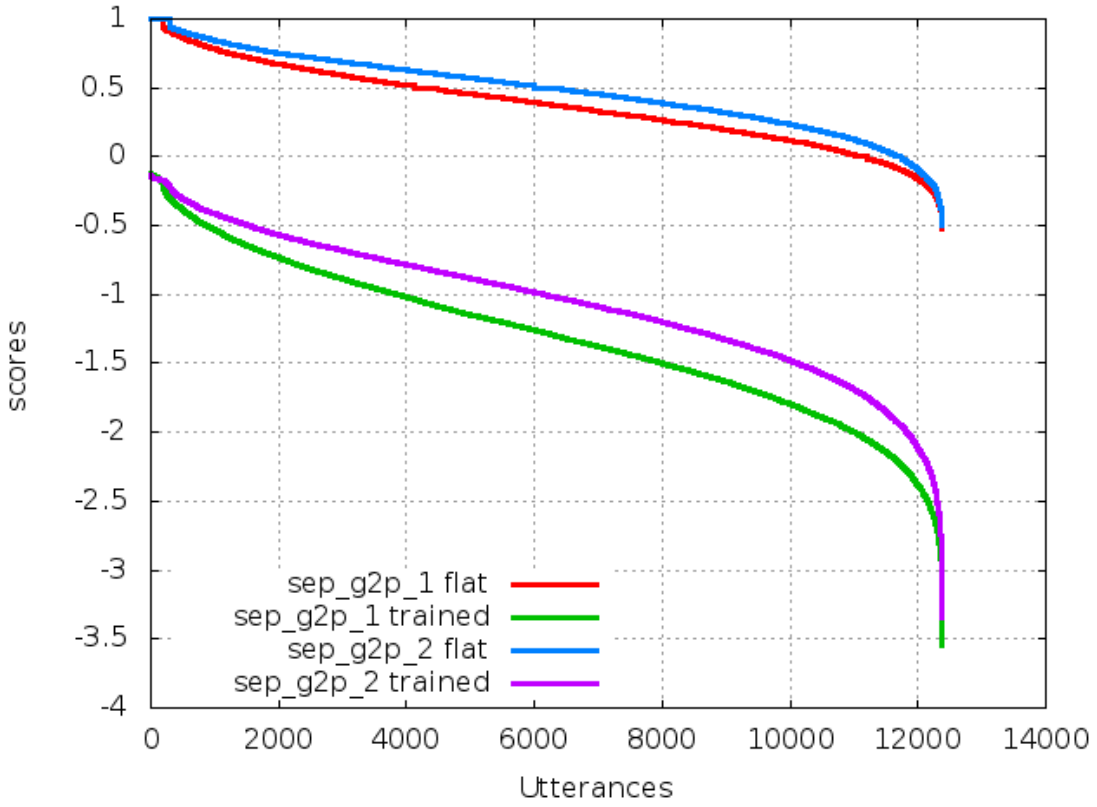

FIGURE 5.1: PDP scores using the sep_g2p_1 and sep_g2p_2 dictionaries with either a flat or trained scoring matrix.

results obtained are shown in Table 5.5. The results show phone accuracies obtained before the evaluation of the corpus, as well as the results obtained after clean up with 10K, 11K, and 12K corpus sizes. The difference in recognition accuracies between the whole corpus (65.11%) and the 12K size corpus (64.86%) was tiny due to the number of utterances removed.

TABLE 5.5: Phone accuracies for SPCS corpus before evaluation and after clean up at 10K, 11K, and 12K corpus size.

Corpus size	*spcs_manual* (%)	*spcs_manual_clean* (%)
10K files	–	63.40
11K files	–	64.27
12K files	–	64.86
All files	65.11	–

5.4.3.2 Manual verification

To determine the effectiveness of this approach, a subset of consecutive utterances was selected at various thresholds and manually evaluated. Instances of clipping, incorrectly

produced audio and empty utterances were all considered errors. As long as an utterance was clearly audible, low volume was not considered an error. In Table 5.6 we provide a summary of the percentage of "good" utterances observed at the different data points. Problems were only encountered from the 8K data point onwards. At this point, most of the utterances contained correctly produced audio with clipping, low volume and background noise affecting the PDP scores. At lower scores, more actual errors were observed. Only the top 11K utterances were retained in the final corpus.

Table 5.6: The percentage of good utterances at different data points.

Data points	No. of good utterances (%)
0 - 20	100
2,000 - 2,020	100
4,000 - 4,020	100
6,000 - 6,020	100
8,000 - 8,020	90
10,000 - 10,020	95
11,000 - 11,020	55
12,000 - 12,020	35
12,364 - 12,384	10

5.5 Corpus composition

In Table 5.7 we show the SPCS and NSECSS corpus composition. The SPCS is the larger of the two corpora with 2 183 utterances compared to the 261 utterances of the NSECSS. While SPCS is a corpus that contains naturally code-switched speech, the NSECCS contains English words that are not from typical code-switched speech.

Table 5.7: The SPCS and NSECSS corpus composition.

	No. of speakers	Duration (hrs)	No of Prompts	No of Utterances	Unique words
SPCS	20	10	450	2 183	784
NSECSS	95	0.14	157	261	223

In Table 5.8, we show the word distribution of the two corpora. The NSECSS contains English words only, and the SPCS contains English, Sepedi, and semi-modified words (i.e. words that are of English origin, but transformed to conform to Sepedi consonant-vowel structure).

In Table 5.9, we show the Radio Broadcast corpus duration per speaker category. The DJ speaker category has the largest number of utterances with the duration of 0.68 hours since they are the dominant speakers on the radio.

TABLE 5.8: The SPCS and NSECSS corpus word distribution.

	English words	Sepedi words	Semi-modified words
SPCS	346	384	58
NSECSS	223	–	–

TABLE 5.9: The Radio Broadcast corpus duration per speaker category.

Speakers	Duration (hrs)	Number of utterances
DJ	0.68	456
Guest	0.29	204
Reader	0.12	73
Caller	0.02	19
Actors	0.03	19

In Fig. 5.2 and Fig. 5.3, we present the structure of the SPCS corpus and the NSECSS corpus. All utterances are labelled as <*speaker id*>_<*gender*>_<*age*>_<*sentence id*>, where <*speaker id*> is the unique code to identify each speaker utterance. <*gender*> and <*age*> are the information about the gender and the age of the speakers, respectively. Lastly, the <*sentence id*> identifies the specific utterances. For example, a typical utterance would be *001_Female_23_1*. (See Fig. 5.2 and Fig. 5.3)

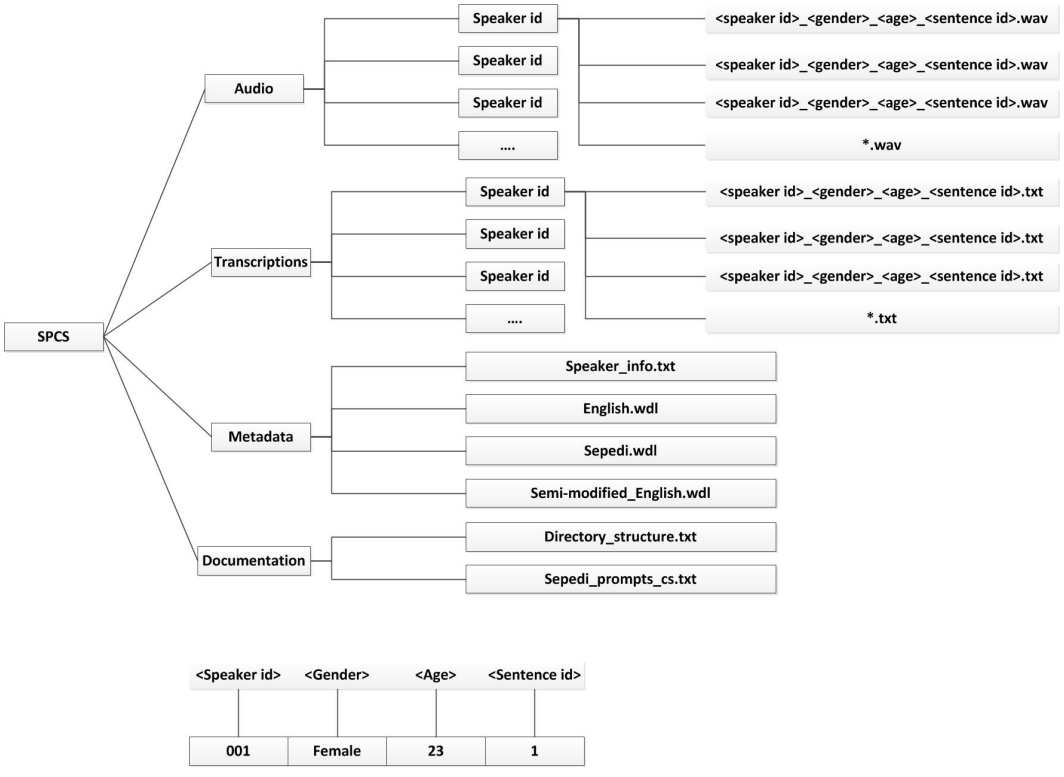

FIGURE 5.2: The structure of the SPCS corpus.

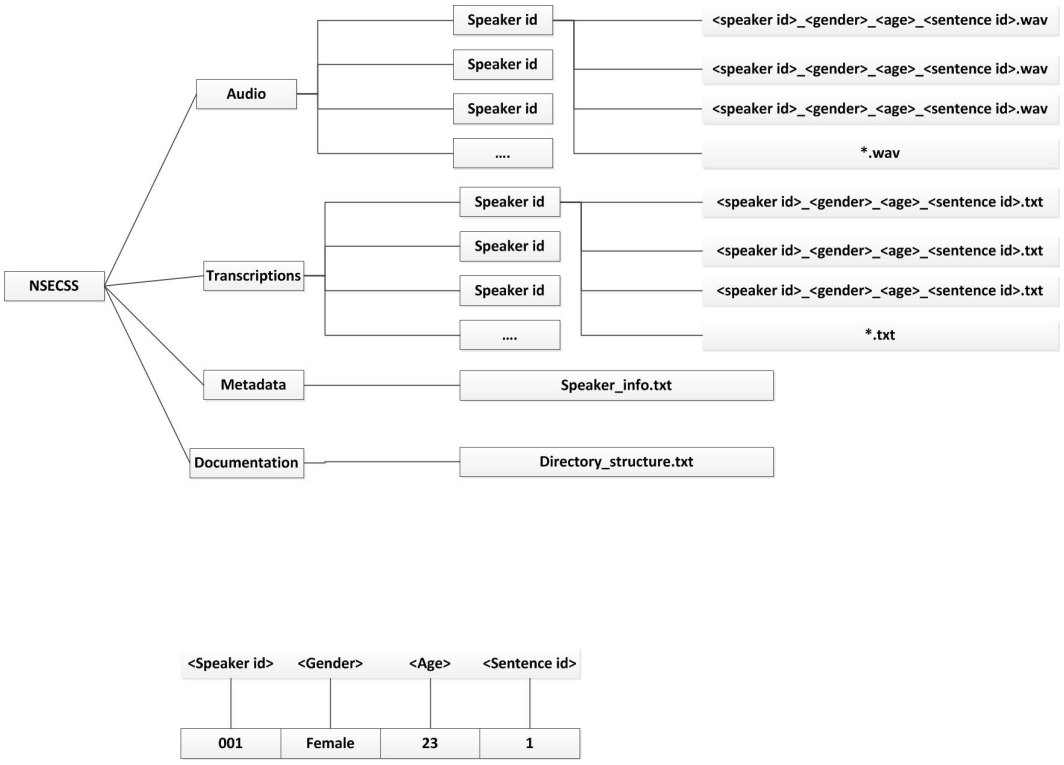

FIGURE 5.3: The structure of the NSECSS corpus.

5.6 Conclusion

In this chapter, three new corpora – the National Centre for Human Language Technology Sepedi-English code-switched subset (NSECSS) corpus, the Radio Broadcast corpus and the Sepedi Prompted Code-Switched (SPCS) corpus – were introduced. The design, development process and verification of the corpora were discussed. The NSECSS and SPCS corpora are freely distributed under an open content license[2].

The NSECSS corpus was developed to provide samples of Sepedi-English code-switched pronunciations. The development of the NSECSS corpus involved validating Sepedi and English text, as well as the audio utterances. The participants were in agreement with text and utterance segments. We also observed challenges such as errors in alignment, difficulties to correctly identify abbreviations and acronyms. A total of 266 valid utterances were generated from NSECSS. These utterances will make it possible to analyse the acoustic factors that are involved when Sepedi speakers produce English speech, but does not address the factors involved when code switching occurs naturally.

The SPCS corpus used the Radio Broadcast corpus as prompt development material. A high percentage of code switching was observed, and these events were used to create

[2]http://rma.nwu.ac.za/

the prompts to collect the SPCS corpus. We were able to develop the SPCS corpus with 11K utterances, and carefully evaluated the corpus collection process. This corpus will be used as a tool to develop and test code-switched acoustic models in Chapters 7 and 8. The Radio Broadcast corpus itself provides information about the factors involved when code switching occurs naturally, as analysed further in the next chapter.

Chapter 6

Methods and frequency of code switching

6.1 Introduction

In this chapter, we aim to determine the extent to which code switching occurs in Sepedi. We also determine if there are patterns for the use of code switching that can assist in modelling this phenomenon for ASR purposes. We do this by analysing the Sepedi Radio Broadcast corpus and focus on the methods and frequency of code switching.

The interchanging of languages by multilingual speakers is a common and interesting phenomenon. In some instances, these speakers switch between languages unconsciously [74]. As a result, code switching does not often follow a predictable switching mechanism. It is not easy to tell in advance what type of code switching is going to occur for a particular speaker. Even speakers who are fluent in their mother tongue find a need to switch to foreign languages.

Code switching is speaker dependent [75], so the methods, as well as the mechanisms used, differ significantly. There are some methods that multilingual speakers employ to switch between the matrix language and the embedded language. (As introduced in Chapter 2, the 'matrix language' is the primary language of the utterance, and the 'embedded language' is the second language embedded in the primary language [76].) These methods include the modification of the foreign language to conform to the matrix language. In other cases, the foreign language can be used as is.

The prevalence of code switching among Sepedi speakers is a continuing phenomenon. In [14], Roux et al. mentions that African language speakers (including Sepedi) will

continue to use English words such as numbers, time, and place names in their conversations. In [12], the same sentiments are corroborated about African language speakers using English words for numbers, dates and money. This prevalence of code switching poses a challenge for future ASR systems. It is, therefore, important to pay particular attention to such code switching events when developing ASR systems by modelling code-switched speech explicitly.

In Section 6.2, we outline the approach used to analyse the Radio Broadcast corpus. In Section 6.3 we discuss the methods of code switching observed in the corpus, and in Section 6.4 the frequency of code switching events observed. In Section 6.5 we describe the reasons why code switching occurs among specific multilingual speakers, both as predicted by other researchers, and as found in this corpus.

6.2 Analysis overview

We performed an analysis of the Radio Broadcast corpus transcriptions to evaluate the code switching rate. During the development of the code switching corpus (as discussed in Section 5.3.2), the following tasks were performed with the assistance of a linguist/Sepedi native speaker. The methods applied when code switching was used, as well as the reasons for code switching to occur, were analysed by manually performing the following tasks:

1. Tagging the English words/phrases during the manual transcription process in the code-switched utterances.
2. Creating a list of unique English words from the tagged English words/phrases.
3. Measuring the frequency of occurrence of the English words.
4. Classifying the English words as either 'pure' (exactly as used in English) or modified.
5. Classifying the English words as those with a Sepedi alternative or not.
6. Identifying the parts of speech (POS) of the English words.

In the following sections, the results of this analysis are presented per topic.

6.3 Methods of code switching

Code switching occurs naturally among multilingual speakers, and this section analyses the different ways in which code switching occurs. It has been observed that often, during code switching, speakers tend to either insert or delete vowels or consonants to reproduce a syllable structure comparable to their native language [2]. This process affects the pronunciation of the embedded words and blurs the distinction between code-switched words, and words of foreign origin that have been incorporated into the matrix language.

TABLE 6.1: Number of pure and modified English words in the Radio Broadcast corpus.

	#Unique	#Total
Pure	922	2528
Modified	96	128
No numbers	890	2033
Total	1018	2653

Table 6.1 shows the number of pure and modified English words observed in the corpus. There were 1 018 unique code-switched words observed in the corpus. Of these, 922 were English words that speakers used without any modification. There were also instances where English words were modified to conform to the Sepedi consonant-vowel (CV) structure, mostly by adding vowels to the end of the words.

Typically, the modification of the English words involves appending a prefix or suffix to the word. This modification results in a combination of morphemes from two languages. In Sepedi, most verb stems end with a suffix *-a* [9]. So the amendment of English verbs to Sepedi typically involves a change in the morphological structure of the English word to conform to the Sepedi structure. This change of morphological structure was observed in the Radio Broadcast corpus with verbs modified in this way, for example, *access* became *accessa*. Modified English nouns also undergo a similar process by combining morphemes of two languages to indicate the plural form. Most noun stems use a prefix di- to indicate plural form. For instance, the modified word would be *diwheelchair* where the *di* morpheme is from Sepedi with *wheelchair* an English word.

TABLE 6.2: Phenomena observed where embedded English words were modified.

Example	English word	Sepedi morpheme	Frequency
diimport	import	di-	66.67%
blocka	block	-a	12.50%
commente	comment	-e	9.38%
provincing	province	-ing	2.08%

Frequently, English words were appended with suffixes such as *-e, -a, -ing* and the prefix *di-* (see Table 6.2). The suffixes *-e, -a, -ing* and the prefix *di-* were, respectively, observed in 9.38%, 12.50%, 2.08%, and 66.67% unique words in cases of code switching that involved modified words. The remaining 9.38% were formed using more rarely occurring mechanisms, for example, the suffixes *-ong, -ile, -eng* and *-i*.

Table 6.3 shows the occurrence of the different parts of speech of English words observed. The words that were categorised as 'other' include the POS with the least counts, for example, the preposition, conjunction, interjection, pronoun and article. The nouns included both regular and proper nouns. In the noun category, proper nouns (with people and place names) occurred 20.4%, with the remaining 79.6% being regular nouns. It is evident that the most code-switched words observed were nouns.

TABLE 6.3: Part of speech of embedded English words.

POS	Frequency	Example
Noun	789	america, group, earthworm, parsley, province
Adjective	89	national, natural, preventative, integrated, immaculate
Verb	72	see, wonder, think, talk, seal, say, pulled
Adverb	33	almost, any, been, before, better, certain
Other	35	hey, okay, of, on, out, in, for

In summary, there were 90% of pure English words in the corpus compared to modified English words. This suggests that speakers use English words when applying code switching. The numbers, which included telephone numbers, contributed about 12% to the Radio Broadcast corpus. The common modification of the English words took place when the prefix di- was applied to the plural form. The most common POS used in code switching is a noun, with an occurrence of 77% in this corpus.

6.4 Frequency of code switching

In this section, we aim to understand the prevalence of code switching in Sepedi when spoken naturally by native Sepedi speakers. To analyse the frequency of code switching, transcriptions were generated from 10 hours of audio. This Radio Broadcast corpus contained 3.6 hours of content, and the remainder was non-content such as music and advertisements. In the content portion, the code-switched portion (as described in Chapter 5, Section 5.3) was 30.91%, and the remaining part constituted Sepedi. The duration of the utterances with code-switched words was 31%. The speakers used, on average, 3.3

occurrences of embedded words per utterance (with the mean length of the utterances being 15.4). Of the top 10 most frequently occurring English words, numbers constituted 60%.

In Figure 6.1, we show the number of English words per utterance, ranging between 1 and 17. The red bars indicate the frequency of English words when we consider numbers as single concept. The green bars indicate the frequency of English words when numbers are included as individual words. There were 249 utterances that contained a single English word. The utterances with close to 17 English words occurred when the speaker mentioned numbers multiple times (with few occurrences of English generic words)

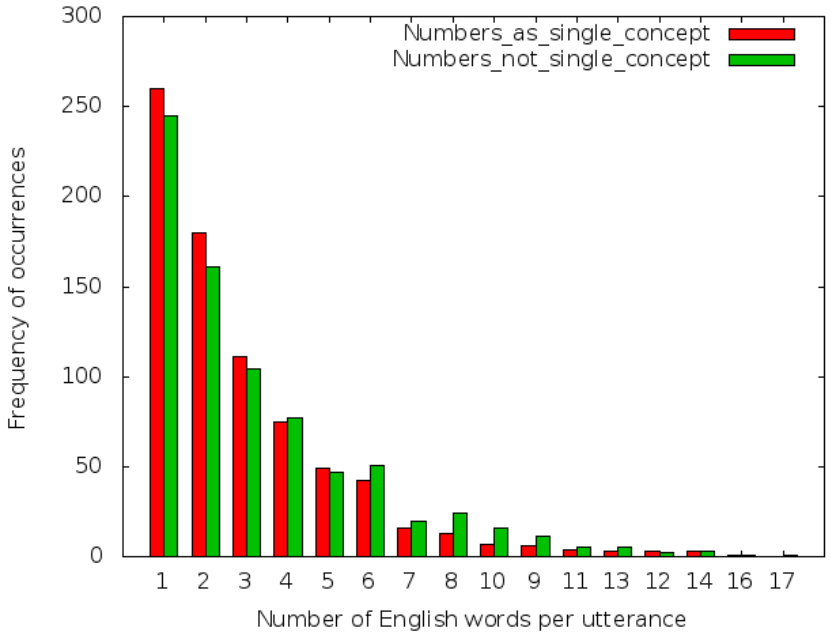

FIGURE 6.1: The number of English words per utterance in Radio Broadcast corpus.

The participants or speakers who formed part of the Radio Broadcast corpus were divided into five categories, namely: DJ, Reader, Actor, Caller, and Guest. The DJ was a radio station employee who broadcast programmes for the station using Sepedi. For each of the categories, multiple speakers were recorded, for example, there were eight individual speakers for the DJ category. The Reader was also an employee whose job was to read news on an hourly basis. The Actor was a person who reads a script to portray a character in a drama or short story. Callers were regular radio station listeners who took part in discussions on the radio programmes. At times, the radio station would invite an individual to take part in discussing life issues by visiting the radio station, and such a person is referred to here as the Guest.

We measured the number of times a single speaker used English words in a sentence as a factor of the number words in the same code-switched utterance, and averaged this

value over all utterances by that speaker. We refer to this measure as the *CS sentence ratio*. We also measured the number of times the speaker used code-switched utterances as a factor of the total utterances of the same speaker category. This ratio is referred to here as *CS overall ratio*.

In Table 6.4, we show the results of this CS sentence ratio for various speaker categories (Guest, Reader, DJ, Caller, and Actor) from the Radio Broadcasts. We used the data available from the Radio Broadcasts and, therefore, did not measure the level of proficiency of English among the speakers. Nor did we obtain any speaker-specific background concerning language exposure. The results show that the DJs applied code switching 22.25% of the times for every sentence they uttered. The average number of English words per code-switched utterance for the DJs was found to be 3.5 words. The studio guests used code switching approximately 17% of the time, the second highest ratio of English to Sepedi word usage. It was interesting that the DJs used code switching more often when they were expected to cater for an audience not necessarily fluent in English. It was also interesting to find that the top ten most frequently occurring English words uttered by the DJs included *numbers* (including the studio telephone numbers), *FM, Audrey, hello, the, of, and* and *doctor*.

The Reader and Guest had the highest CS overall ratio of 38.02% and 38.71% respectively. This ratio was greater than what was observed when analysed per sentence, and the DJ produced a very similar ratio of 36.07%. For the Reader, the majority of code switching events related to the use of words such as *of, board, South, Johannesburg, Mathews, Moodley* and *council*. The Guest category's frequently occurring code-switched words were *and, hemia, HIV, of, then, twenty, SASSA* and *the*.

Table 6.4: CS overall ratio and CS sentence ratio per speaker category.

	Guest(%)	Reader(%)	DJ(%)	Caller(%)	Actor(%)
CS overall ratio	38.71	38.02	36.07	21.48	4.10
CS sentence ratio	17.78	12.17	22.25	17.22	13.17

The most unexpected result of this work was the high frequency of code switching that was observed. When we analysed the frequency of code switching among different speaker categories, we found that DJs use English words most frequently. However, the type of words used were numbers as mentioned earlier and they included telephone numbers that were repetitively mentioned throughout the programmes, followed by the words *hello* and *FM*, which are usually used to answer telephone calls.

In the Reader and Guest categories, we found that the most frequently occurring code-switched words were different. Proper names featured more often in their utterances.

Even though the CS utterance ratio overall is the same for both the Reader and Guest categories, their code-switched words were not necessarily the same.

6.5 Reasons for code switching

From a linguistic perspective, certain factors influence speakers to code switch at a particular point in time. In [15], the factors that affect code switching were identified during a study of bilingual students who were learning English at university. Note that the code switching that was taking place among the participants was from English (second language) to home language. These factors were:

1. A shortage of words in the English language as compared to their home language.
2. Lack of knowledge of a word in English.
3. To fill the gap in speaking.
4. Students find it easier to talk in their languages.
5. To avoid misunderstanding between the speakers.
6. To exclude other people from the conversation, i.e. privacy.
7. To add emphasis to a point.

Other scholars, in [16], also conducted a study on school children and found the following motivations for code switching:

- Equivalence: using a word in another (usually native) language when the speaker does not know it in the target language;
- Floor-holding: the use of native language to fill the gap while searching for the target language word;
- Metalanguage: using native language to make a comment or evaluation or saying something about an activity;
- Reiteration: repetition of messages in both the native and target language for emphasis or clarification;
- Group membership: conversational switching understood within a group;
- Conflict control: to minimise conflict or confusion; and

- Alignment and dis-alignment: to shift or gain control of a conversation that is going astray.

The list of hypotheses that were compiled by linguists as motivations for code switching in [9] is shown below. However, in the study, no particular investigation was conducted concerning motivations for code switching.

1. The speaker may use words in other languages while trying to retrieve words in the target language.
2. There could be a shortage of words in the target language.
3. The speaker excluding others from the conversation.
4. Code switching shows emotions.
5. For emphasis.

We observed that code switching often occurred where the concept being discussed did not exist in the vocabulary of the matrix language. In Table 6.5 we show the unique number of English words observed during the code switching analysis that does not have a Sepedi alternative. It must be noted that speakers still used code switching even for words with Sepedi alternatives. Time and age can be said in English by speakers for clarity or emphasis. There were 18 instances where speakers used both English and Sepedi words and phrases for emphasis by first uttering the Sepedi word/phrase, then repeating it in English.

TABLE 6.5: Number of unique English words in the Radio Broadcast corpus with and without Sepedi alternatives.

	# unique words
Has Sepedi alternative	484
No Sepedi alternative	534

Interestingly, we found that there were no Sepedi alternatives for over 50% of the English words observed, which predicts that many of these words would be incorporated into Sepedi over time. For about 10% of English words found, these words were already semi-transformed into Sepedi words through the addition or transformation of syllables. Furthermore, the speaker preference to use words from other languages (depending on the situation/environment they are in) would contribute to the use of code switching.

In Table 6.6, the reasons for code switching observed in this corpus, and some of their corresponding examples are shown. We evaluate observations from this corpus with observations from studies in other language pairs.

Multilingual speakers are exposed to many languages both professionally and socially. Some of these speakers may be proficient in more than one of these languages. In studies such as in [15] and [16], code switching that was taking place was from the secondary language (English) to the mother tongue. This code switching was different from that observed in the Sepedi language where speakers switch from native language (Sepedi) to secondary language (English). It was interesting to note that Sepedi speakers would have had similar reasons as mentioned by the other scholars to switch between languages, yet switched in different directions.

The reasons given in Table 6.6 suggest that we are likely to see more secondary language terms embedded in the matrix language by the native speakers. In [77], research found that the accent of the English words change and becomes Cantonese-accented in code-mixing speech. We expect similar challenges to the ASR system when English words are embedded within Sepedi utterances.

6.6 Conclusion

In this chapter, we discussed the methods, frequency and motivation for code switching that took place among native Sepedi speakers, as determined through the analysis of the Radio Broadcast corpus.

The main findings in the corpus studied are:

- Code switching is a very frequently occurring phenomenon. The rate of code switching (CS ratio overall) among different speaker categories was found to be 36.02% for DJ, 4.10% for Actor, 38.02% for Reader, 21.48% for Caller and 38.71% for Guest categories.
- 90% of the English words were used without modification.
- Of the modified English words, the prefix *di-* appeared most frequently. A small set of affixes (*-e, -a, -i, -ing, -eng, -ong, -ile*) were also observed.
- The most frequently used POS for embedded English words was the noun.
- The most observed code-switched words were numbers.
- Numbers were often repeated in both English and Sepedi (first using one language, then switching to the other).
- DJs seemed to be applying code switching more than other categories. However, the English words used mostly were numbers. The Reader and Guest used proper names more often.

- The reasons for code switching include: no Sepedi alternative, emphasis, filling a gap in speaking, easier to use a secondary language, and avoidance of misunderstandings.

In this chapter, we analysed the phenomenon of code switching by using the Radio Broadcast corpus developed in Chapter 5. In Section 5.3.1 of the same chapter, we also described the creation of the SPCS corpus: a collection of prompted speech based on events observed in the Radio Broadcast corpus. In the next chapter, we use the new corpus for experimentation by analysing possible phoneme mappings using different strategies.

Table 6.6: Examples observed that demonstrate the reasons for code switching.

Reasons	Example	Outcome from study
Alignment and disalignment [16]	–	Not analysed
Avoid misunderstanding [15]	"Tšeo tše pedi tša tšona e le go tša kgapeletšo, ke gore *two compulsory subjects*" (Two of which are compulsory, meaning two compulsory subjects)	Found
Conflict control [16]	–	Not analysed
Emphasis or Clarity [9] [15] [17]	"Setšokotša legano re bolela ka *mouth wash*" (mouth wash, we are talking about mouth wash)	Found
Equivalence [16]	–	Not analysed
Fill gap in speaking [15] [17]	"Seo se direga *usually* ge motho a godile, e *common* go bontate" (That usually happens to older people, common among males)	Found
Floor-holding [9] [16]	"*Somehow* šetše ke thoma go nwa bjala bjale" (Somehow I started drinking alcohol)	Found
Group membership [16] [17]	–	Not analysed
Lack of register [17]	–	Not analysed
Metalanguage [16]		
No alternative (shortage of words) [9] [15] [17]	"Ngwana a ka ba a nale *hydroseal* goba *hemia*" (The child might have hydroseal or hemia)	Found
Pragmatic reasons [17]	–	Not analysed
Privacy [9] [15]	–	Not analysed
Reiteration [16]	"T͡hyroid ga botse Lebo ke gland ka sekgowa re e bitša *gland* e mo molaleng mo" (Thyroid, actually Lebo, is a gland in English, we call it gland and it is located on the neck)	Found
Revert to English for numbers and dates/-time [14] [12]	"Nteletšeng go *zero one five two nine seven*" (Call me on zero one five two nine seven)	Found
Semantic significance [17]	–	Not analysed
Showing emotions [9]	–	Not analysed
To address a different audience [17]	–	Not analysed
To attract attention [17]	–	Not analysed

Chapter 7

Context-independent acoustic modelling of code-switched speech

7.1 Introduction

In this chapter we discuss an approach to map English phonemes to Sepedi phonemes for the development of a robust ASR system for code-switched speech. The development of the ASR system for code-switched speech requires a robust phoneme set construction to improve recognition accuracy [39]. We perform an analysis of the implications of code-switched speech on the performance of the ASR system and then analyse possible phoneme mappings using different context-independent approaches. Specifically, we determine the implications of different phoneme mapping techniques for code-switched speech when considering individual phonemes (rather than phonemes in context), training and testing using different types of data.

The analysis in Chapter 4 used a very small corpus: the only corpus that was available at the time. During the course of this study, the Sepedi NCHLT corpus (introduced in Section 2.7) became available. The Sepedi NCHLT corpus, as well as the SPCS corpus, developed subsequently and introduced in Chapter 5, are used for the analysis here.

In Section 7.2 we discuss the data used to train, develop and test the acoustic models. The language model data is described in Section 7.3, as well as the evaluation of the language models. In Section 7.4 we develop a baseline system using the new corpora, applying the techniques from Chapter 4. The main phoneme mapping analysis is performed in Section 7.5, and results discussed in Section 7.6.

7.2 Data

The NCHLT-206 Sepedi corpus was used for the development of the acoustic model for the analysis of the code-switched speech. The NCHLT-206 corpus is a subset of Sepedi NCHLT clean v1.0 (official RMA version) corpus. This corpus has an official training set of 202 speakers and a test set of eight speakers. There were four SPCS speakers in the official RMA version, who were removed to form the NCHLT-206 corpus. This corpus was partitioned into train (NCHLT-206-trn), development (NCHLT-206-dev), and testing/evaluation (NCHLT-206-eval) sets. The resulting NCHLT-206-trn has about 180 speakers with 51,545 utterances as shown in Table 7.1.

TABLE 7.1: The number of speakers, utterances and duration of the Sepedi NCHLT and SPCS corpora for train, test, and development sets.

	# of speakers	# of utterances	Duration (hrs)
NCHLT-206-trn	180	51 545	49.20
NCHLT-206-dev	6	1 489	1.30
NCHLT-206-eval	20	5 668	5.22
NCHLT-nso-only-dev	6	1 427	1.26
NCHLT-nso-only-eval	20	5 414	5.06
SPCS-dev	4	2 183	1.57
SPCS-eval	16	10 203	8.51

7.2.1 Evaluation data

We created three data sets to evaluate the performance of the ASR system and understand the effect of each test/evaluation set. The features of these data sets differ significantly from each other, and help us develop robust modelling techniques. The three evaluation sets created are shown in Table 7.1:

- The NCHLT-206-eval set is the newly created evaluation portion of NCHLT-206. It has 20 speakers with 5,668 utterances.
- The SPCS-eval set is the evaluation portion of the SPCS corpus. It has 16 speakers with 10,203 utterances.
- The last evaluation set is the NCHLT-nso-only-eval. It is the NCHLT-206 Sepedi corpus with Sepedi utterances only (identified using word lists). It has 20 speakers with 5,414 utterances.

7.2.2 Development data

Three development sets were created to fine-tune the parameters of the corresponding evaluation set. The details of the development sets are shown in Table 7.1:

- The NCHLT-206-dev is the newly created development portion of NCHLT-206. It has six speakers with 1,489 utterances.
- The SPCS-dev is the development portion of the SPCS corpus. It has four speakers with 2,183 utterances.
- The NCHLT-nso-only-dev is the portion of the NCHLT-206 Sepedi corpus with Sepedi utterances only (identified using word lists). It has six speakers with 1427 utterances.

7.3 Language models

We built the language models using two text corpora, namely, NCHLT and SPCS text corpora. The NCHLT text corpus is a collection of long sentences obtained from the South African government website. It is a clean Sepedi text v1.6 (from CText) with approximately two million words. (These are not the NCHLT corpus prompts/transcriptions themselves).

The SPCS text corpus is a collection of text transcriptions obtained from Radio Broadcasts. The corresponding code-switched parts of the corpus were transcribed. This superset includes the source text from which the prompts for the SPCS speech corpus were constructed, but not the exact corpus transcriptions.

7.3.1 Language model training

The first language models (*nchlt_lm*) were trained using the NCHLT text corpora for both bigram and trigram language models. The second (*spcs_lm*) and third (*spcs_nchlt_lm*) language models were trained as follows:

1. Radio Broadcast transcriptions were used to train a code-switched based language model, also referred to as SPCS language model.
2. NCHLT text corpus was used to train the Sepedi based language model (this corpus has a few examples of English words).

3. The language models in (1) and (2) were interpolated to generate the final language model combining the features from both text corpora.

The *nchlt_lm* bigram and trigram language models were trained using the SRILM toolkit. The vocabulary size of the language models was 61 960 words. These words included both Sepedi and English words as well as other (unknown) words. The SPCS bigram and trigram language models were also trained using the same SRILM toolkit. The vocabulary size of the language models was 3 340 words. The *spcs_nchlt_lm* bigram and trigram language models were interpolated by determining the optimal weight according to the test set. The optimal weight is obtained by optimising the perplexity of each language model using the test set.

The interpolation weights were determined by using the SRILM command on the test set transcriptions. The weights obtained for the NCHLT/SPCS corpora were, respectively, 0.00174/0.9982 and 0.00614/0.9938 for bigram and trigram language models. The SPCS-eval set matches the SPCS text corpus much better than the NCHLT text corpus. The two text corpora were used wholly for training and a different set (SPCS-eval) was used to test the language models.

7.3.2 Language model testing

The evaluation data (from the SPCS corpus) contain real instances of code switching. To obtain an appropriate language model, we would like to use matching data. For this purpose, we use the original text transcriptions of the Radio Broadcast corpus. These match the SPCS prompts to an extent: as it was used in the creation of the SPCS corpus, it matches the style of the test set. As this text corpus is quite small, we add text from a larger corpus to train a larger language model. The pronunciation dictionary used contained all the words in the test set as well as the words occurring in the language model.

The bigram and trigram language models were tested with the test transcriptions of the SPCS-eval set. The number of sentences observed was 10203, with 45034 words and 23 OOV words. The perplexity was found to be 46.317. In Table 7.2 we provided details of the NCHLT, SPCS and interpolated NCHLT_SPCS text corpora bigram and trigram language models. The perplexity and the OOV rates of the language models, are provided.

TABLE 7.2: The NCHLT SPCS, and interpolated NCHLT_SPCS text corpora bigram and trigram language models.

	nchlt_lm		spcs_lm		spcs_nchlt_lm	
LM order	2	3	2	3	2	3
Interpolated weight	–	–	–	–	0.00174/0.9982	0.00614/0.9938
Perplexity	415.066	393.018	50.3499	43.0477	54.4027	46.3178
# of sentences	10 203	10 203	10 203	10203	10203	10 203
# of tokens	45 034	45 034	45 034	45 034	45 034	45 034
# of types	784	784	784	784	784	784
OOV tokens	6 663	6 663	199	199	784	784
OOV types	213	213	8	8	23	23

7.4 Baseline system development

A typical ASR system consists of an acoustic model, pronunciation dictionary and a language model. For code-switched speech, the ASR system would consist of bilingual pronunciation dictionary and bilingual language model [78]. We built a code-switched language model using two corpora and applied linear interpolation as described in [78]. This language model was trained using the SRI Language Modeling (SRILM) Toolkit [62].

7.4.1 ASR system and related resources

Acoustic data: The NCHLT-206-trn training set was used to develop the acoustic models. The details about this training set are discussed in Section 7.2.

ASR system: The acoustic models were trained using the HMM-based ASR system using HTK toolkit with parameters as described in Section 4.2.3.

Pronunciation dictionary: In Section 4.2.2, the effect of different approaches to the pronunciation modelling of foreign words in a Sepedi ASR was investigated. One of the observations made was that better results were obtained when the pronunciation of English words was predicted using a direct, simple approach – Sepedi letter-to-sound rules. We used this pronunciation modelling approach for the baseline system developed here.

Language models: The *nchlt_lm* bigram and trigram language models together with *spcs_nchlt_lm* bigram and trigram language models were used to evaluate the recognition accuracy of the ASR system using the SPCS-eval test set. The details of the language models are discussed in Section 7.3.

7.4.2 Results

In Table 7.3, we show the word recognition accuracies for the acoustic models trained with the NCHLT corpus and evaluated with the SPCS data set. The performance of the system improved when the language model was trained with the data set that matched the evaluation set even when the acoustic models were trained with the mismatching data. The best word recognition accuracy obtained in this experiment was 45.24%.

When the matching data was used to train the language model, the performance of the system improved significantly. We obtained an increase of 18.35% and 23.05% for bigram and trigram language models, respectively. These results served as the baseline for the context-independent analysis in Section 7.5.

TABLE 7.3: Word recognition accuracy (*Acc*), using SPCS-eval as the evaluation set. The language model (*LM*), language model order (*LM order*), language model weight (*LMW*) and interpolated language model weight (*InterW*) are shown.

Eval set	LM	LM order	Acc	LMW	InterW (nchlt/spcs)
SPCSeval	nchlt_lm	2	25.56	18	–
SPCSeval	nchlt_lm	3	22.19	10	–
SPCSeval	spcs_nchlt_lm	2	43.91	23	0.00174/0.9982
SPCSeval	spcs_nchlt_lm	3	45.24	22	0.00614/0.9938

The interpolated trigram language model obtained better word recognition accuracy when tested with the code-switched speech as expected. When there is a mismatch between the real-world data and the acoustic models, the performance of the system is low. It is evident that when the language model does not match the test data the ASR system relies more on the acoustic models. For the bigram language model, the language model weight changed from 18 to 23, while for the trigram, it shifted from 10 to 22.

7.5 Context-independent analysis

The aim of the context-independent analysis is to understand the implications of Sepedi-English code-switched speech in an ASR system when considering phonemes in isolation.

In Section 7.4, we already obtained results by considering specific acoustic models (training corpora), pronunciation dictionaries and language models. From these results, the best recognition accuracy was obtained under the following conditions:

- Sepedi NCHLT speech corpus with Sepedi and English utterances was used to train the acoustic models.

- For language model, we used interpolated trigram language models using SPCS and NCHLT text corpora.
- The pronunciation dictionary was developed using Sepedi phonemes only and predicted with Sepedi G2P rules.

We now investigate the effect of improving the pronunciation dictionary by mapping phonemes (all phonemes of the same type, in the same way). Improvement is evaluated by performing word recognition when training on the Sepedi NCHLT corpus. We measured the phone error rates (PER) and the word error rates (WER) using three different data sets, namely a mismatching data set (SPCS), matching data set (NCHLT), and Sepedi only subset of the NCHLT.

Some of this analysis is a repeat of the work done in Section 4.2. There are, however, different resources used compared to the previous experiment. In the previous work the following conditions held:

1. English pronunciation dictionaries that were not manually verified
2. Sepedi Lwazi corpus was used.
3. Only phone error rates were considered.
4. No language model was used but flat phone-loop grammar.

However, the rest of the approach remained the same. In addition, we also examined the WER, which was not used previously. Compared to Section 4.2, the new resources under consideration included:

1. Sepedi NCHLT 2013 ASR speech corpus;
2. Manually verified English pronunciation dictionaries (NCHLT and SPCS);
3. Interpolated language models.

7.5.1 Data

The NCHLT-206 Sepedi corpus was used for the development of the acoustic model. We specifically used the training set NCHLT-206-trn as described in Section 7.2. The details of this training data are shown in Table 7.1. The development and the evaluation sets were also used to fine-tune the parameters and test the system, respectively.

7.5.2 Dictionaries

The pronunciation dictionary for Bantu languages has been developed for native or monolingual speech [64]. When code switching is introduced, different variations of speech pronunciation come with challenges. Native language speakers apply articulatory features of the native language to the secondary or foreign language.

We developed a set of pronunciation dictionaries for the Sepedi NCHLT corpus and the SPCS corpus. The Lwazi G2P rule sets were used to predict the pronunciations of the words in the corpora [64]. The Sepedi word pronunciations were predicted with the Lwazi Sepedi G2P rules. Since the focus of our work is on the pronunciations of the English words, the pronunciations of the English words were predicted in four different ways:

(a) We used the Lwazi Sepedi G2P rules to predict the pronunciations of the English words for the two corpora. The resulting dictionaries are referred to here as *nchlt_nso* and *spcs_nso* for the words in the NCHLT and SPCS corpora, respectively. The English characters that do not exist in the Sepedi vocabulary were mapped to Sepedi phonemes by extending the Lwazi Sepedi G2P rules as follows:

(i) c => k_h

(ii) v => B

(iii) q => k_h

(iv) x => k_> s

(v) z => s

The left-hand side shows the English characters and the right-hand side shows the corresponding Sepedi phoneme used.

(b) We used the Lwazi English G2P rules and manually verified the predicted pronunciations of the English words. The predictions of the pronunciations resulted in a dictionary with an extended (containing both English and Sepedi) phoneme set for both the NCHLT and SPCS corpus. The generated dictionaries are referred to here as *nchlt_nso_eng* and *spcs_nso_eng*.

(c) We used Lwazi English G2P rules with manually verified predictions for the English words, and the English phonemes were mapped to the Sepedi phonemes using the closest linguistic IPA feature mapping. The resulting dictionary contained Sepedi only phones. These dictionaries are called *nchlt_nso_ipa* and *spcs_nso_ipa*.

(d) We used Lwazi English G2P rules with manually verified predictions for the English words, and English phones were mapped to Sepedi phones using a confusion matrix. These dictionaries are referred to as *nchlt_nso_mat* and *spcs_nso_mat*.

The English words in (a) above were not manually reviewed. All the English word pronunciations in (b) were manually verified. In (c) and (d), we manually corrected the English pronunciations before mapping to Sepedi phonemes using the linguistic IPA mapping and the confusion matrix based phoneme mapping. The Sepedi pronunciations were not manually verified. Sepedi is a regular language, and the G2P rules achieve much better accuracy. For that reason, no manual verification was deemed necessary. English is not a regular language, and we are interested in modelling English phonemes. For that reason, there was a need to manually verify both the words and the pronunciations before modelling.

7.5.2.1 Linguistic IPA mapping

All the mappings have been reviewed using the international phonetic alphabet (IPA) symbol features [69]. The English phoneme and Sepedi phoneme with similar features were considered. In instances where similar features cannot be determined, most probable phonemes were mapped, e.g. *@* and *E*, *3:* and *E*, *A:* and *a*. The English phoneme *@* has two closest candidates, namely the Sepedi phonemes *a* or *E*. For this experiment, we chose to replace the phoneme *@* with Sepedi phoneme *E*. We also have two cases where the mapped phonemes are from different classes but it made sense to map them. These mappings were *Z* to *d_0Z* and *r* to *r*. The other controversial choice relates to the mapping of the diphthongs. The diphthongs were mapped as separate monophones. The mapping of English phonemes to Sepedi phonemes using linguistic (IPA) information is shown in Appendix A, Table A.5.

7.5.2.2 Confusion matrix mapping

In this section we describe the process followed to map the English phonemes to Sepedi phonemes using the confusion matrix. Since we are interested in code-switched speech, we decided to use the SPCS corpus to generate the confusion matrix.

First, the confusion matrix was obtained as follows:

1. Freely decoded phone-level labels were generated from the Sepedi system (Sepedi only system), with Sepedi phonemes only using the test set (see below).

2. The test set was aligned using a dictionary that contains the extended phoneme set (i.e. Sepedi and English phonemes).

3. Iterative dynamic programming was used to obtain an accurate confusion matrix at phone-level.

4. For every English phoneme, the Sepedi phoneme with the highest confusability was selected from the confusion matrix.

In Table 7.4 we show the mapping of English phonemes to Sepedi phonemes using the confusion matrix generated from the full SPCS corpus as test set. The confusion matrix was generated from two phone-based Master Label Files (MLFs) generated with two different dictionaries (one with Sepedi only phonemes, and the other with the extended English phoneme set). The dictionaries used were the *spcs_nso* and *spcs_nso_eng* dictionaries.

TABLE 7.4: The mapping of English to Sepedi phonemes using confusion matrix.

English phoneme	Mapped Sepedi phoneme	English phoneme	Mapped Sepedi phoneme	English phoneme	Mapped Sepedi phoneme
@	E	ai	a	p	p_>
@i	E	au	O	r\	r
@u	O	b	p_>	t	t_>
A:	a	d	t_>	tS	S
D	t_>	e	E	u@	O
3:	E	e@	E	u:	O
O:	O	g	k_>	v	B
Q	O	i@	E	z	s
T	s	i:	i	{	a
Z	S	k	k_>		

7.5.3 Language models and system description

Language models: The *spcs_nchlt_lm* bigram and trigram language models were used to evaluate the recognition accuracy of the ASR system using the SPCS-eval test set. The details of the language models are discussed in Section 7.3.

ASR system: The acoustic models were trained using the HMM-based ASR system using the HTK toolkit with parameters as described in Section 4.2.3.

7.5.4 Experimental setup

In this section we outline the process followed to train the acoustic models and the language models. We also discuss the data set used to evaluate the performances of the systems.

1. The five sets of acoustic models were trained using the NCHLT-206-trn train data as follows:
   (a) The acoustic models, referred to here as *nchlt_nso_am*, are trained using the *nchlt_nso* dictionary with both Sepedi and English words predicted with Sepedi G2P rules;
   (b) The acoustic models, referred to as *nchlt_nso_eng_am*, are trained using the *nchlt_nso_eng* dictionary with English words predicted with language specific letter-to-sound rules and manually corrected. So, for English words, English G2P rules are applied to predict their pronunciation and sent to a language practitioner for manual verification. For Sepedi words, Sepedi G2P rules are applied;
   (c) The acoustic models, referred to as *nchlt_nso_ipa_am*, are trained using the *nchlt_nso_ipa* dictionary with English words predicted with language specific letter-to-sound rules. So, for English words, English G2P rules are applied to predict their pronunciation and manually verified. For Sepedi words, Sepedi G2P rules are applied. The English phonemes are mapped to Sepedi phonemes using linguistic IPA mapping;
   (d) The acoustic models, referred to as *nchlt_nso_mat_am*, are trained using the *nchlt_nso_mat* dictionary with English phonemes mapped to Sepedi phonemes using a confusion matrix.
   (e) The acoustic models, referred to as *nchlt_nso_ipa_var_am*, are trained using the *nchlt_nso* and *nchlt_nso_ipa* pronunciation dictionaries.
2. The *spcs_nchlt_lm* bigram and trigram language models were trained as described in Section 7.3.
3. To evaluate the recognition accuracy of the systems, both word and phone recognition accuracy were measured. The phone recognition was performed using flat (where every phone has an equal probability) grammar (phone loop). The word recognition was performed using the bigram and trigram LM trained using NCHLT and SPCS text corpora. The OOV rate and the perplexity of the development set (transcriptions), as well as the evaluation set (transcriptions), were measured to

determine the extent of mismatch between the LM training corpus and the test data.

4. The acoustic models were tested with the SPCS-eval and the parameters fine tuned with SPCS-dev: The five acoustic models (i) *nchlt_nso_am*, (ii) *nchlt_nso_eng_am*, (iii) *nchlt_nso_ipa_am*, (iv) *nchlt_nso_mat_am* and (v) *nchlt_nso_ipa_var_am*, were evaluated with the SPCS-eval set. The decoding parameters were optimised using the development (spcs-dev) set of the SPCS corpus and the optimal parameters for the evaluation (spcs-eval) set.

5. The five acoustic models (i) *nchlt_nso_am*, (ii) *nchlt_nso_eng_am,* (iii) *nchlt_nso_ipa_am*, (iv) *nchlt_nso_mat_am* and (v) *nchlt_nso_ipa_var_am*, were also evaluated with the Sepedi NCHLT corpus containing Sepedi utterances only. The decoding parameters were optimised using the development (nchlt-nso-dev) set of the subset of NCHLT corpus and the optimal parameters for the evaluation (nchlt-nso-eval) set.

6. Using the matching data set, the five acoustic models (i) *nchlt_nso_am*, (ii) *nchlt_nso_eng_am*, (iii) *nchlt_nso_ipa_am*, (iv) *nchlt_nso_mat_am* and (v) *nchlt_nso_ipa_var_am*, were also evaluated with the NCHLT corpus. The decoding parameters were optimised using the development (nchlt-206-dev) set of the Sepedi NCHLT corpus and the optimal parameters for the evaluation (nchlt-206-eval) set.

7.5.5 Results

The phone recognition results obtained using the flat grammar with different evaluation sets where each phone has an equal probability of occurrence are shown in Table 7.5. We generated the results using the five ASR systems and the three evaluation sets mentioned in Section 7.5.4. The Insertion Penalty (IP) was fine-tuned using the development sets to optimise the parameters.

We test all systems using Sepedi only (nchlt-nso-only-eval), Sepedi and English (nchlt-206-eval), and code-switched speech (spcs-eval) evaluation sets. We show the results for phone recognition accuracy when the systems were developed as follows:

(a) the acoustic models trained with a combination of English and Sepedi phonemes (nchlt_nso_eng_am).

(b) the acoustic models trained with Sepedi only phonemes (*nchlt_nso_am*).

(c) the acoustic models trained with Sepedi phonemes and where the English phonemes were mapped to Sepedi phonemes using linguistic IPA mapping (*nchlt_nso_ipa_am*).

(d) the acoustic models trained with Sepedi phonemes and where the English phonemes were mapped to Sepedi phonemes using a confusion matrix (*nchlt_nso_mat_am*) were tested.

(e) the acoustic models trained with dictionaries using techniques described in (b) and (c) above as variants (*nchlt_nso_ipa_var_am*) were also tested.

In Table 7.5 we show the phone recognition accuracy for the three evaluation sets using the five acoustic models described above. Only the flat loop grammar was utilised in this experimental setup. The best phone recognition accuracy is shown in bold. The code-switched speech gave the best phone recognition accuracy when evaluated on the *nchlt_nso_ipa_var_am* acoustic models. As for the nchlt-206-eval and nchlt-nso-only-eval test sets, the best phone recognition results were obtained with the *nchlt_nso_am* acoustic models. When the English phonemes were used during the training of the acoustic models, we got the worst accuracy with the three evaluation sets.

The experiments suggest that the English phonemes have an adverse effect on the Sepedi data, possibly because there was not enough data to estimate the English phone models properly. None of these modelling techniques could achieve the best results for all three evaluation data sets together. However, the introduction of variants does provide better results for code-switched speech.

TABLE 7.5: Phone recognition accuracy for different evaluation sets, obtained using different acoustic model/dictionary combinations with a flat phone-loop grammar.

	Data set evaluated		
Dictionary/Acoustic model	spcs-eval	nchlt-206-eval	nchlt-nso-only-eval
nchlt_nso_eng_am	40.81	73.51	70.19
nchlt_nso_am	51.95	**82.16**	**82.85**
nchlt_nso_ipa_am	52.53	72.70	74.56
nchlt_nso_mat_am	53.13	76.39	75.11
nchlt_nso_ipa_var_am	**53.65**	79.50	81.71

In Table 7.6, the results show that for the bigram language model, the nchlt-nso-only-eval and nchlt-206-eval perform better with the acoustic models trained using *nchlt_nso_eng_am*. The spcs-eval still improves with using Sepedi G2P rules as variants and various approaches to mapped English phonemes. The *nchlt_nso_ipa_am* seems to have an adverse impact on the recognition of Sepedi words, and thereby results in a slight word recognition accuracy deterioration.

TABLE 7.6: Word recognition accuracy for different evaluation sets, obtained using different acoustic model/dictionary combinations with interpolated bigram language model.

	Data set evaluated		
Dictionary/Acoustic model	spcs-eval	nchlt-206-eval	nchlt-nso-only-eval
nchlt_nso_eng_am	38.10	**69.49**	**71.00**
nchlt_nso_am	43.91	68.78	70.09
nchlt_nso_mat_am	47.66	65.87	67.45
nchlt_nso_ipa_am	50.04	66.47	68.23
nchlt_nso_ipa_var_am	**55.98**	68.95	70.23

TABLE 7.7: Word recognition accuracy for different evaluation sets, obtained using different acoustic model/dictionary combinations with interpolated trigram language model.

	Data set evaluated		
Dictionary/Acoustic model	spcs-eval	nchlt-206-eval	nchlt-nso-only-eval
nchlt_nso_eng_am	39.51	72.88	74.55
nchlt_nso_am	45.24	**73.34**	**74.68**
nchlt_nso_mat_am	49.31	70.37	72.77
nchlt_nso_ipa_am	51.49	71.27	72.57
nchlt_nso_ipa_var_am	**58.02**	73.26	74.49

7.6 Discussion

Different modelling techniques were analysed for the pronunciation modelling of code-switched speech. In Section 7.6.1 we first compare the effect of the different modelling techniques on code-switched speech. We then consider the effect of the approaches on Sepedi-only data (that does not include code-switching) in Section 7.6.2. Finally, we perform a word-based error analysis in Section 7.6.3 to understand the recognition results for English words better.

7.6.1 Comparison of modelling techniques

We introduced five modelling techniques for code-switched speech. Figures 7.1, 7.2, and 7.3 show a graphical representation of the word recognition accuracy results for bigram and trigram language models with histograms for the evaluation sets (spcs-eval, nchlt-206-eval, and nchlt-nso-only).

In Figure 7.1 we show the performance of the acoustic models when presented with the code-switched (mismatched) data using bigram and trigram language models. We evaluated the effect of developing a pronunciation dictionary where each English word was associated with the English phoneme set (and foreign phonemes retained). With this approach (using *nchlt_nso_eng_am* acoustic models), the spcs-eval achieves the worst

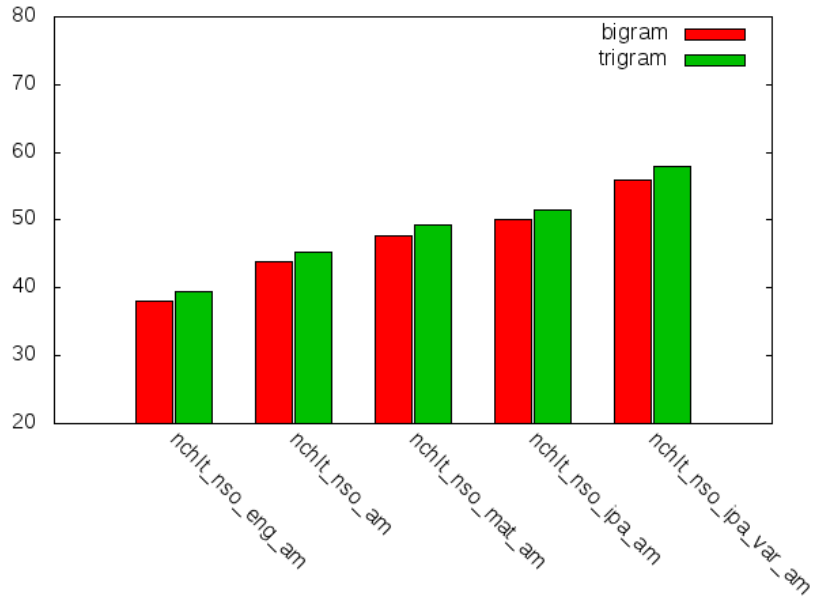

FIGURE 7.1: Word recognition accuracy for SPCS evaluation set with bigram and trigram language model.

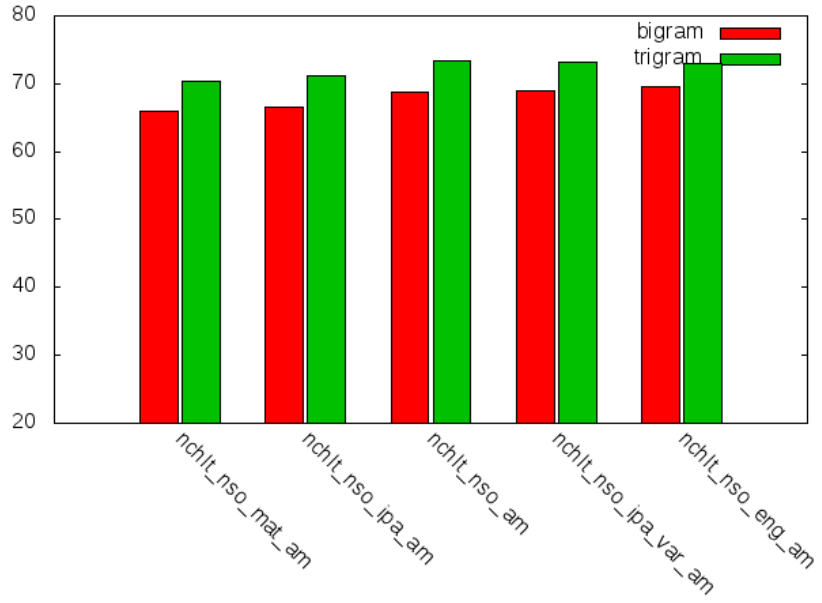

FIGURE 7.2: Word recognition accuracy for the NCHLT evaluation set with bigram and trigram language model.

results with a word recognition accuracy of 38.10% for bigram language model and accuracy of 39.51% for trigram language model.

The best recognition accuracy for mismatched data (spcs-eval) of 55.98% (bigram) and 58.02% (trigram) are obtained when the *nchlt_nso_ipa_var_am* acoustic models are used.

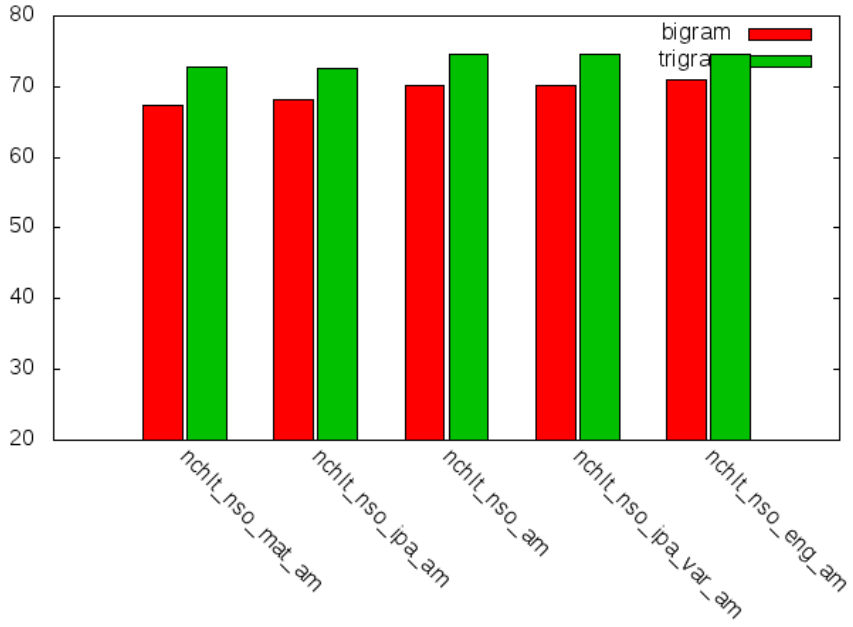

FIGURE 7.3: Word recognition accuracy for the NCHLT Sepedi only evaluation set with bigram and trigram language model.

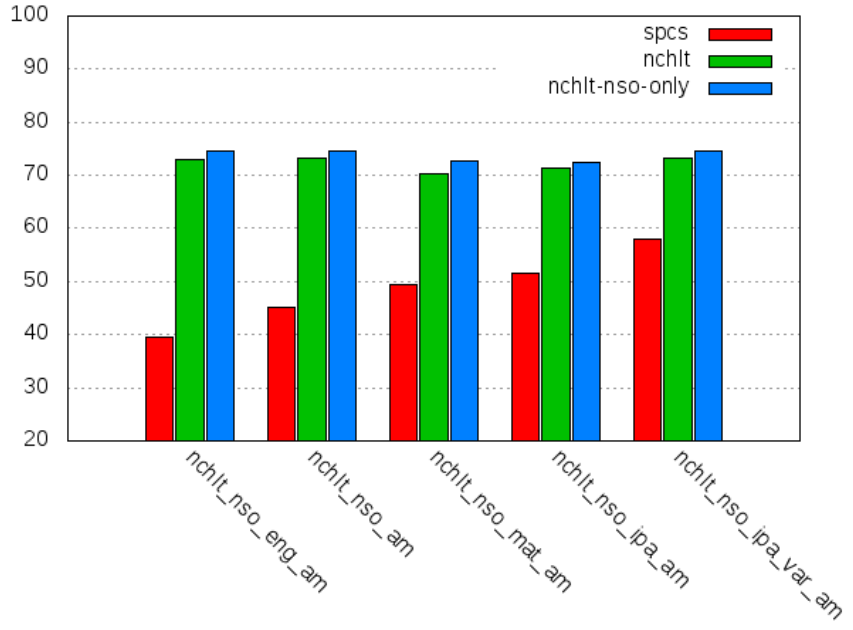

FIGURE 7.4: Word recognition accuracy for the SPCS, NCHLT and NCHLT Sepedi only evaluation sets with trigram language model.

Among the context-independent modelling techniques that were evaluated, this modelling technique was the best for code-switched speech. This suggests that better recognition accuracy for code-switched speech are obtained when variants are introduced.

When we compared the frequency of the occurrence of the English words specifically

(rather than the whole code-switched utterance) in the reference MLF file with the frequency of the recognition of the English words in the decoded MLF, there was a difference in the frequency of their recognition. With the *nchlt_nso_eng_am* acoustic models, most English words were not even recognised once. However, with the *nchlt_nso_ipa_var_am* there were many more English words that were recognised at least once. This result is analysed further in Section 7.6.3 but is reflected in the difference in word recognition accuracy, with the absolute difference between the above two models being 17.88%. The same trend was also observed in the trigram language models with an absolute difference of 18.51%, the highest so far (see Figure 7.1).

In this experiment, it was shown that the use of Sepedi phoneme for the English words improves the word recognition accuracy. When the Sepedi G2P rules were used to predict the pronunciation of the English words, an improvement was obtained. This improvement was obtained even though the predicted pronunciations were not manually verified.

When the data-driven technique was introduced, where the English phonemes were mapped to Sepedi phonemes using the confusion matrix, there was also an improvement in the recognition accuracy. This approach achieved an absolute increase of 3.75% and 4.07% with bigram and trigram language models, respectively, when compared to the manually verified predicted English pronunciations. Since the trained acoustic models used to generate the confusion matrix had split affricates, the mapped phonemes, in this case, did not include affricates and aspirated phonemes.

The linguistic IPA mapping technique was also evaluated to determine how well it can recognise code-switched speech. It was interesting to note another level of improvement in the recognition accuracy of the mismatching data. While this approach was suitable for code-switched speech, it had detrimental effects on the matching data (Sepedi). For this reason, we added the benefits of using Sepedi G2P rules for Sepedi data and the linguistic mapping of English phonemes to Sepedi phonemes by generating English pronunciation variants. This approach improved the accuracy for code-switched speech further.

7.6.2 Effect of modelling techniques on Sepedi-only speech

When the acoustic models were trained with the pronunciation dictionary containing English words with English phonemes and Sepedi words with Sepedi phonemes we got the best accuracy for nchlt-206-eval and nchlt-nso-only-eval with 69.49% and 71.00%, respectively, using the bigram language model. The other modelling techniques, by focusing on the code-switched speech, deteriorated the word recognition accuracy of the

Sepedi data. The lowest word recognition accuracy was obtained when the mapping of the English phonemes to Sepedi was applied using the confusion matrix with nchlt-206-eval and nchlt-nso-only-eval evaluation sets. When we compared these two techniques, we found an absolute difference of 3.67% and 3.55% for the bigram language model.

Not surprisingly, given the relatively small number of English words in the NCHLT data, the best word recognition accuracy when evaluating the acoustic models with these two data sets was obtained with a modelling technique where the pronunciation of Sepedi and English words was predicted with Sepedi G2P rules. The absolute difference between this technique and the lowest observed accuracy is 2.97% and 1.91%, respectively. These differences were small but significant, and needed to be taken into consideration when we modelled code-switched speech. The aim was to retain good word recognition accuracy for Sepedi words while trying to get a better modelling technique for English words. However, the best modelling approach for code-switched speech has caused harm to our Sepedi speech recognition. (The degradation in Sepedi recognition is small compared to the win in English recognition, but still noticeable.)

7.6.3 Word-based error analysis

In this section we analyse the recognition results for the English words occurring within the code-switched utterances. We consider the role of word length and evaluate the recognition accuracy of individual words.

We determine whether the length of the English words had an effect on the recognition performance by measuring the number of times the correctly decoded word occurred as a percentage of the number of times it occurred in the reference MLF file, referred to here onwards as the mean decoded ratio (*mean_dr*). In Figure 7.5 we show the percentage *mean_dr* of the English words for the five acoustic models using the bigram language model.

In Figure 7.6, we measured the percentage *mean_dr* of the English words using the trigram language model. Word length did not seem to have a significant effect. The discrepancies observed for longer words are more probably due to the fact that the number of long words is small, introducing noise.

We analyse the English words that were not even recognised once by different acoustic models. We compare two acoustic models of mis-recognised English words with the same test set (spcs-eval) using *nchlt_nso_eng_am* and *nchlt_nso_ipa_am*. In Table 7.8 we show the number of unique words that were not recognised by these two acoustic models on the spcs-eval test set. The *nchlt_nso_ipa_am* had the least number of mis-recognised words, 9

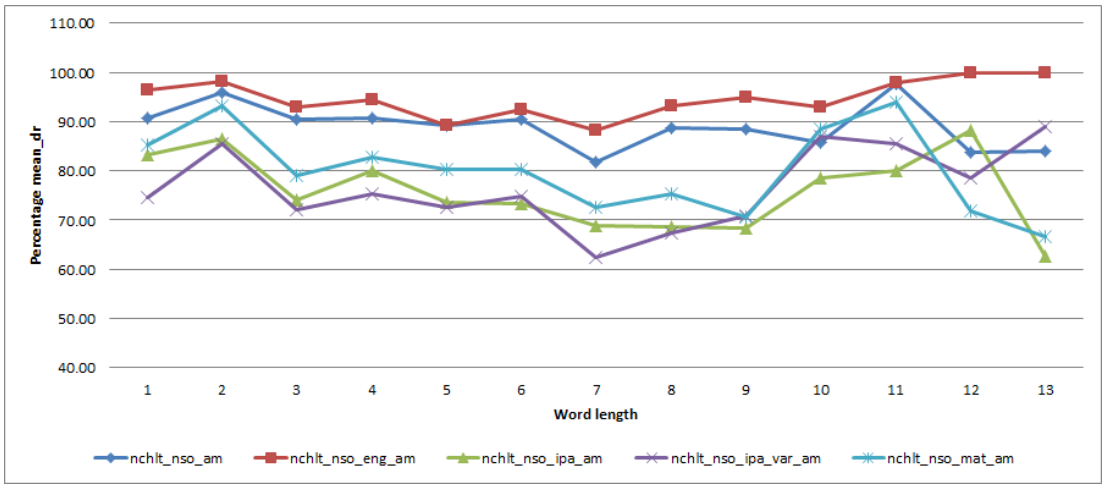

FIGURE 7.5: The percentage mean_dr of the English words lengths using bigram language model.

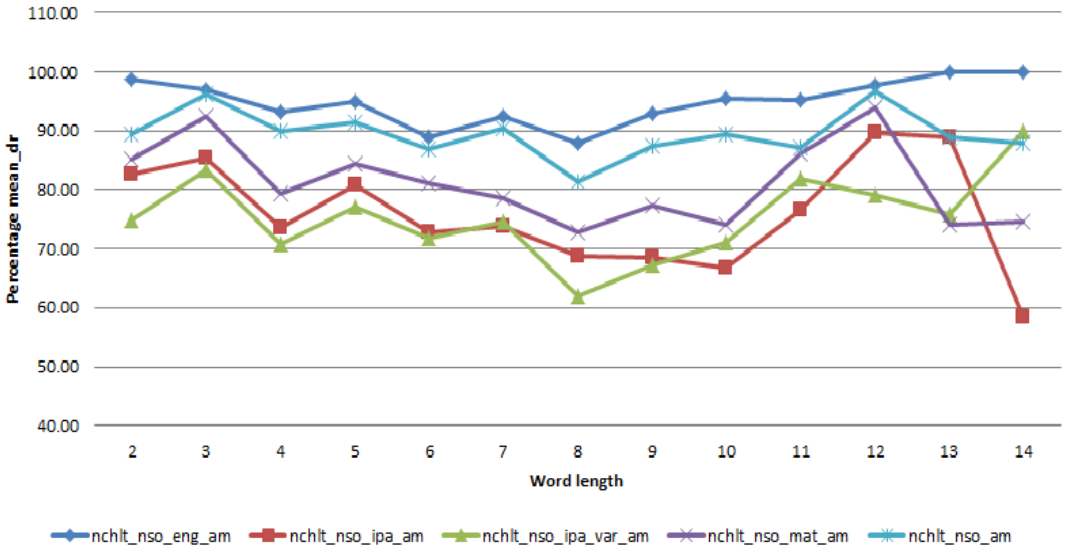

FIGURE 7.6: The percentage mean_dr of the English words lengths using trigram language model

and 6, for bigram and trigram language models, respectively. The *nchlt_nso_eng_am* had the most mis-recognised English words, 130 and 133, using bigram and trigram language models, respectively.

TABLE 7.8: The number of English words that were not recognised at all for *nchlt_nso_eng_am* and *nchlt_nso_ipa_am* acoustic models using bigram and trigram language models.

Acoustic models	Bigram	Trigram
nchlt_nso_eng_am	130	133
nchlt_nso_ipa_am	9	6

In Table 7.9, we also show the number of the English words that were not even recognised once using the *nchlt_nso_ipa_var_am* and *nchlt_nso_ipa_am* acoustic models evaluated with the spcs-eval test set with both bigram and trigram language models. The performance of these two acoustic models was not that different.

TABLE 7.9: The number of English words that were not recognised at all for two acoustic models using bigram and trigram language models evaluated with spcs-eval test set.

Acoustic Models	Bigram	Trigram
nchlt_nso_ipa_var_am	21	22
nchlt_nso_ipa_am	23	17

Since the *nchlt_nso_ipa_var_am* acoustic models were trained with a dictionary (for English variants) that was used to train the *nchlt_nso_am* acoustic models, it was important to compare them again with the English words that were completely mis-recognised. The results of this comparison are shown in Table 7.10.

TABLE 7.10: The number of English words that were not recognised at all for two acoustic models using bigram and trigram language models evaluated with spcs-eval test set.

Acoustic models	Bigram	Trigram
nchlt_nso_ipa_var_am	12	15
nchlt_nso_am	111	104

The *nchlt_nso_am* acoustic models had over 100 English words that were not recognised compared to the words observed from the *nchlt_nso_ipa_am* acoustic models. These results show that their performance was very different and the introduction of these variants had a positive effect on word recognition accuracy.

7.7 Conclusion

In this chapter, we used a context-independent approach to analyse the implications of code-switched speech on the performance of ASR systems. It was clear that recognition accuracy drops with the introduction of code-switched data.

To improve the performance of the ASR system, four modelling techniques were evaluated:

- Develop pronunciation dictionary using Sepedi G2P rules for all words (English and Sepedi);

- Develop pronunciation dictionary using Sepedi G2P rules for Sepedi words and manually corrected predicted English G2P rules pronunciations for English words;
- Develop pronunciation dictionary using Sepedi G2P rules for Sepedi words and manually corrected predicted English G2P rules pronunciations for English words and map English phonemes to Sepedi using linguistic IPA mapping;
- Develop pronunciation dictionary using Sepedi G2P rules for Sepedi words and English G2P rules for English words and map English phonemes to Sepedi using confusion matrix (data-driven) mapping;
- Develop pronunciation dictionary using Sepedi G2P rules for all words (English and Sepedi) and add English words with mapped English phoneme to Sepedi using linguistic IPA mapping as variants.

The prediction of the pronunciation dictionary using the last modelling technique achieved the best results. In the next chapter, we consider a more sophisticated approach to determine and analyse phoneme mappings.

Chapter 8

Context-dependent acoustic modelling of code-switched speech

8.1 Introduction

In the previous chapter, phoneme mappings applicable to Sepedi/English code-switched speech were developed while considering phonemes in isolation. That is, the same phoneme mapping was used for a phoneme, irrespective of the context in which it was found. In this chapter, we investigate whether better mappings are possible, if the context is taken into account.

We apply a technique to automatically tag an acoustic corpus to predict phoneme labels for code-switched phonemes. These phoneme labels are then analysed to determine the features – including context – that are useful to predict pronunciations. To investigate whether this is a feasible approach, we first need to establish how predictable such phoneme substitutions really are. Specifically, we would like to determine whether phoneme substitutions can be predicted based on features such as phoneme context, word orthography or even speaker characteristics. Our focus starts with vowels, as they exhibit significantly more variability in pronunciation than consonants, and we start specifically with schwa, as its pronunciation tends to be the most unpredictable of all English vowels found in code-switched speech.

In Section 8.2, we describe techniques for predicting the phoneme substitutions suitable to map English phonemes to Sepedi phonemes. The final mappings are developed in Section 8.3 based on the analysis of the predicted phoneme labels. Pronunciation dictionaries are created in Section 8.4 to analyse the effect of the phoneme mapping techniques, and these evaluated using ASR systems (Section 8.5). In Section 8.6, we

extend the analysis by reviewing rewrite rules extracted from the predicted labels. The different techniques for modelling code-switched speech are discussed in Section 8.7.

8.2 Phoneme substitution prediction

We describe a technique for predicting the phoneme substitutions that are expected to occur during code switching, using non-acoustic features only. Within the context of Sepedi/English code switching, we analyse the different realisations of the English schwa first. A code-switched speech corpus is used as input and phoneme substitutions are identified by auto-tagging this corpus based on *acoustic* characteristics. We first evaluate the accuracy of the auto-tagging process, before determining the predictability of the auto-tagged corpus, using *non-acoustic* features. We first analyse only the schwa phoneme. Once the schwa phoneme is analysed, we repeat the process for all other English phonemes that do not naturally occur in the Sepedi phoneme set.

8.2.1 Embedded language pronunciations

In code-switched speech, pronunciations are typically produced in one of two ways. The true embedded language pronunciation is produced or at least approximated fairly closely, or the target phoneme string is substituted for a counterpart from the matrix language. As there is always a possibility of producing the true pronunciation, modelling both these possibilities requires variants to be introduced to the pronunciation lexicon. Both an 'embedded language' and 'matrix language' pronunciations for each code-switched word are therefore required. This substitution can either happen at the G2P level (producing an embedded or matrix language phoneme string) or at the phone-to-phone (P2P) level (producing the embedded language phoneme string but then mapping these to the matrix language). In this analysis we focus on the P2P substitutions that can be expected. Examples of these pronunciation variants are shown in Table 8.1, using SAMPA notation[1].

Table 8.1: Examples of embedded and matrix language pronunciations

word	embedded language (English)	matrix language (Sepedi)
pressure	/p r\E S @ r\/	/ p_h r E S **a** /
fifteen	/ f @ f t i: n /	/ f **i** f t_>i n /

[1]The 'Speech Assessment Methods Phonetic Alphabet' is a standard computer-readable notation for phoneme descriptions. See [79].

Our goal is to determine which features influence phoneme substitutions when they do occur. Specifically, we would like to predict which phoneme substitutions can be expected in the matrix pronunciation. Given the two examples above, we, therefore, would like to predict /@/ $\rightarrow$ /*a*/ in one case, and /@/ $\rightarrow$ /*i*/ in the other.

8.2.2 Selecting candidate mappings

We obtain mapping candidates from the same confusion matrix described in Section 7.5.2.2 (previously used to identify a single best match). This time, we flag all phonemes that are confused with the target phoneme in more than 20% of the target phoneme occurrences.

Table 8.2 lists the frequency of occurrence of the English vowels in the SPCS corpus. For each vowel, we identify the mapping candidates and provide, in brackets, the number of times a target phoneme to mapping candidate pair was observed, per candidate, in the confusion matrix.

TABLE 8.2: Phoneme mapping candidates obtained from confusion matrix. For each English vowel, the number of times it was observed in the SPCS corpus is provided. For each phoneme-candidate pair, the number of times that the confusion was observed in the data is provided in brackets.

Phoneme	Counts (SPCS)	Candidates
@	10 445	a (4448), E (2534) i (1165), O (1156) u(78)
i:	711	i (389), E (205)
A:	749	a (635), E (51)
{	2 479	a (1775), E (536)
u:	1 065	u (434), O (216)
Q	1 811	O (1208), a (429)
O:	1 333	O (1009), a (283)
E:	991	E (663), a (196)

Table 8.3 lists the frequency of occurrence of the English consonants in the SPCS corpus. For each consonant, we identify the mapping candidates and provide, in brackets, the number of times a target phoneme to mapping candidate pair was observed, per candidate, in the confusion matrix. However, we do not model the consonants that are straight forward (i.e. have a single candidate phoneme). We choose the most frequent Sepedi phonemes that are confusable from the confusion matrix.

TABLE 8.3: Phoneme mapping candidates obtained from confusion matrix. For each English consonant, the number of times it was observed in the SPCS corpus is provided. For each phoneme-candidate pair, the number of times that the confusion was observed in the data is provided in brackets.

Phoneme	Counts (SPCS)	Candidate(s)
t	6 856	t_h (2 549), t_>(1 760)
r‘	5 131	r (2 840)
k	4 200	k_h (1 933), k_>(1 086)
d	3 825	l‘ (1 499), t_>(392)
p	3 098	p_h (1 183), p_>(1 068)
b	1 960	B (787), p_>(607)
z	1 702	s (1 144)
v	1 325	f (366), B (307)
g	1 024	k_>(321), G (123)
tS	405	tS_h (125), S (45), tS_>(36)
T	345	t_h (57), t_>(38), f (32)
Z	184	S (68), d_0Z (21)

8.2.3 Schwa analysis

As the schwa is by far the most problematic of the English phonemes, we first evaluate only this one phoneme in isolation. We first identify which phoneme substitutions occur by auto-tagging a speech corpus based on acoustic characteristics in Section 8.2.3.1. To determine the accuracy of the auto-tagger, we manually create a small labelled test set and evaluate the accuracy of the auto-tags against the manual labels in Section 8.2.3.3. The accuracy of the GOP and variant-selection based auto-taggers is analysed in Section 8.2.3.4. The number of observations, as well as the frequency of the predicted labels for the schwa phone, are discussed in Section 8.2.3.6. Once the auto-tagging process has been verified, we then tag a much larger corpus and determine the predictability of this auto-tagged corpus, using non-acoustic features in Section 8.2.3.7.

8.2.3.1 Auto-tagging

The SPCS corpus is first partitioned into a training and test set, and the training set is used to develop a standard HMM-based ASR system, as described in Section **??**. During training, a pronunciation lexicon is used that retains all the English phonemes (see Table A.7). Every word in this lexicon has a single variant: Sepedi words are modelled using Sepedi pronunciation models and English words using English pronunciation models. Sepedi pronunciations are obtained directly from G2P models (default-and-refine models [32] trained on the NCHLT *in-lang* dictionaries [71]). English words are first predicted using English G2P models [80], then manually reviewed. (As English words

form the focus of the analysis, accurate pronunciations are expected to be important for the rest of the study.)

Once the ASR system has been trained, the same training data is re-aligned, using different options, each replacing the English phoneme in the pronunciation lexicon with a different Sepedi phoneme, e.g. */a/, /E/, /i/, /O/ or /u/.* Given the data, the resulting alignments then produce timing information, as well as the likelihood of each vowel. These alignments are referred to as '*spcs_@*', '*spcs_a*', '*spcs_E*', etc. depending on which phone was used during alignment.

Time alignments were manually verified as accurate before proceeding. In this case, we determined if the boundaries (start and end times) of the phones matched the audio. This process included the following steps:

1. Create textgrids for the selected samples
2. Listen to the words in question with focus on the start and end times
3. Listen to the English schwa focusing on start and end times
4. Label the phone as a match or not

Accurate alignments were only obtained when training and aligning on the same corpus. Note that the lexicons are only changed during alignment and no re-training is performed. This ensures that all the data being studied are combined in the current phone-model, and not incorporated into any of the other phone-models.

For pronunciation assessment, we use the Goodness of Pronunciation (GOP) score, as introduced in Section 3.4. Once the likelihoods have been obtained, word-level GOP scores are extracted for each alignment option. Word-level GOP scores are used, as timing information otherwise introduces unnecessary variability. Word beginning and end times stay fairly consistent, while phone beginning and end times may differ significantly, especially if there is a mismatch between the vowel actually produced and the alignment candidate. For the same reason, frame-based (rather than phone-based) GOP scores are extracted, as defined in [57]. Word-level GOP, using triphones and frame-level normalisation, forms the basis for the analysis performed here.

8.2.3.2 Alternative implementation: variant-selection

Another approach to predict phoneme substitution, referred to as the *variant-selection* approach, is discussed. We considered this approach to model more than one phoneme

per word and also to determine how it performs against the GOP approach. In this approach, for every phoneme candidate we generate pronunciation variants. The pronunciation only contains Sepedi phonemes; all English phonemes are replaced by candidates. We use the HTK tool to perform forced alignment for the ASR system to select the most probable Sepedi phoneme string (i.e. the best pronunciation would be selected from the dictionary). These alignments are then used to determine the phoneme tags for the English phonemes.

8.2.3.3 Manual tagging

To evaluate the effectiveness of the auto-tagging process, we created a manually labelled verification set. The set consisted of a list of English words where the schwa phoneme occurs. These words also had the associated utterance id, the grapheme string, as well as the word pronunciation. Only the grapheme string associated with the schwa phoneme was provided.

Two subjects were asked to listen to the words in question independently and to manually label each schwa with the most probable phoneme produced. Subjects were encouraged to only select 'schwa' if a true schwa was produced, and to otherwise select the closest vowel candidate. Both subjects were bilingual English/Afrikaans speakers, with subject B being a trained linguist with exposure to Northern Sotho.

We evaluated the agreement between the two subjects to determine the validity of the manual labels. The subjects were presented with the same set of samples for evaluation. They were allowed to listen to the audio samples many times to make their verdicts.

Any labels that were not true examples of code-switched schwas were removed from the evaluation set, for example, where 'pressure' is only produced as / p r\E S / without articulating the last phoneme at all. The number of remaining labels and inter-subject agreement measured using these labels, is shown in Table 8.4. While inter-subject agreement is medium to fair at 70.0%, a review of the data shows that disagreement is mainly due to schwa boundaries being drawn in different places by the two subjects. If labels that are not considered to be very close to schwa by either of the subjects are the only ones considered, the inter-subject agreement increases to 85.7%.

Table 8.4: Inter-subject agreement during manual tagging.

	#observations	#agreement	%agreement
All labels	50	35	70.00
Schwa labels excluded	35	30	85.71

8.2.3.4 Accuracy of the auto-tagger

As the auto-tagger is restricted to always select a possible alternative pronunciation (excluding schwa), it makes sense to only evaluate tagging accuracy against non-schwa labels. For the sake of completeness, results on both the full set and the non-schwa set are included in Table 8.5.

TABLE 8.5: Accuracy of the GOP auto-tagger when measured against different manually labelled test sets.

	#observations	#agreement	%agreement
All labels: subject A	53	23	43.40
All labels: subject B	53	28	52.83
Schwa excluded: subject A	38	23	60.53
Schwa excluded: subject B	33	28	84.85
Schwa excluded: overall	71	51	71.83

From this analysis, it is clear that the auto-tagging process produces usable results, but agrees more strongly with subject B. While overall agreement is only at 71.8%, agreement between the auto-tagging process and subject B reaches 84.9%.

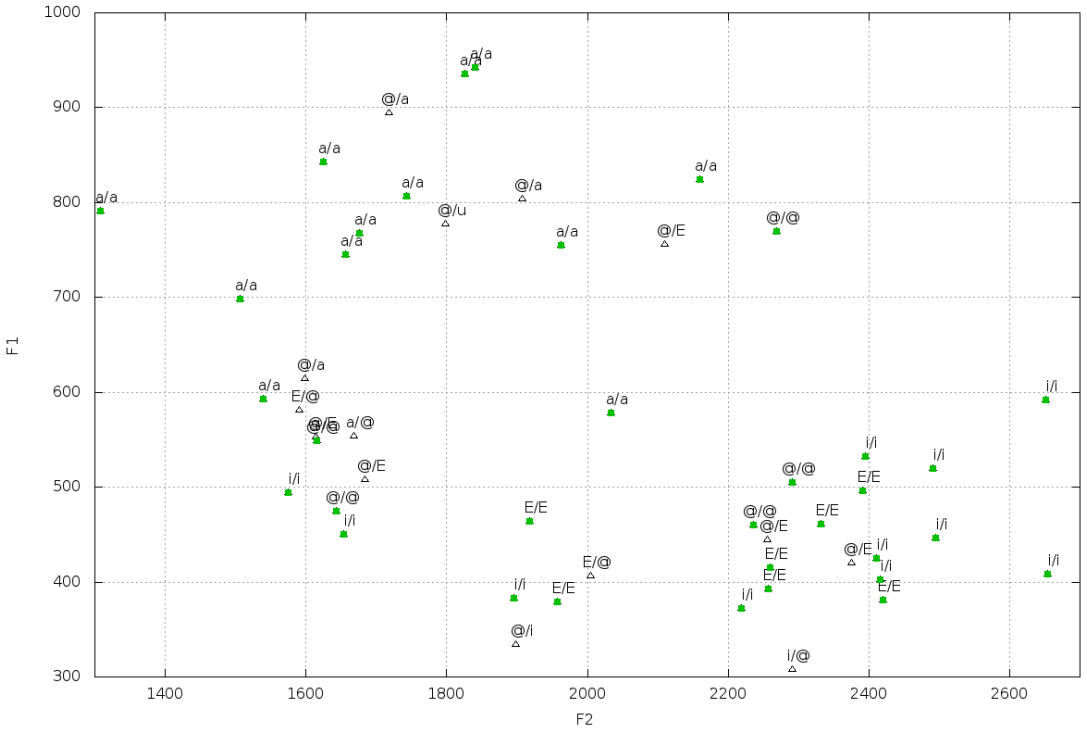

FIGURE 8.1: F1/F2 positions of labels. Each A/B legend displays the tag provided by subject A and B, respectively.

To understand the discrepancies between the two subjects and the auto-tagger better, we analysed the formant frequencies. The formants are resonance frequencies of the

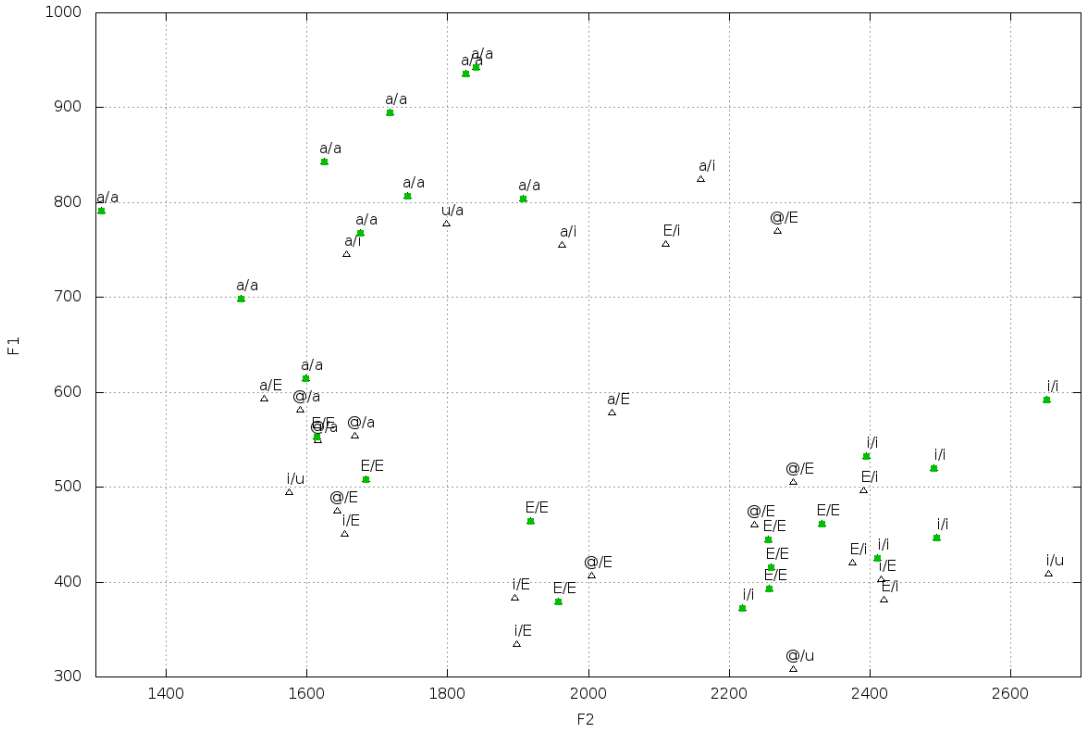

FIGURE 8.2: F1/F2 positions of labels. Each B/T legend displays the tag provided by subject B and the GOP auto-tagger, respectively.

vocal tract [81]. The frequencies of the formants are normally used to perform vowel classification. In this case, we only considered two formant frequencies, F1 and F2.

The formant figure was created as shown in Fig. 8.1 and Fig. 8.2. The first and second formants (F1 and F2) were extracted for each sample using Praat [73], and each sample plotted in the F1/F2 space. In Fig. 8.1, each sample is labelled with the tags from subject A and B. In Fig. 8.2, the sample is labelled with the tags of subject B and the auto-tagger. From the figures, it can be seen that most discrepancies among the three tags occur on the boundaries between classes. Tags in agreement are shown in green in both figures.

We then evaluate the performance of the auto-tagger implemented using the GOP approach and when implemented using the variant-selection approach. We measure the accuracy of the two approaches against the manually labelled set. In Table 8.6, we show the number of observations, number of agreement and percentage agreement for all labels in subject B with and without schwa labels. When the variant selection approach is compared to subject B, the percentage agreement is 56.60%. This percentage agreement is higher by absolute 3.77% compared to the GOP approach. When we measure the percentage agreement between the GOP and variant selection approaches on a full set, we get 93.35%. This suggests that the two approaches are equally good with the variant selection being a little bit better.

TABLE 8.6: Accuracy of the variant-selection auto-tagger when measured against manually labelled test set and the GOP auto-tagger.

	#observations	#agreement	%agreement
All labels: subject B	53	30	56.60
Schwa excluded: subject B	33	30	90.91
All labels: GOP auto-tagger	53	46	86.79
Full set	962	898	93.35

In Table 8.7, we compare the number of times a specific label was selected by either the GOP or variant-selection approaches using words with single vowel occurrence. The results were comparable between the two approaches. This is not unexpected, as the variant-selection approach was specifically created to better deal with words that contained multiple schwas.

TABLE 8.7: Comparing performance of GOP and variant-selection approaches using words with single vowel occurrence.

	Alignment # words	GOP # words
a	8	7
E	8	8
i	10	11
O	3	3
u	1	1

8.2.3.5 Tag analysis

The data is first split into two parts to simplify analysis: where the schwa phoneme appears once in a word and where it appears multiple times. Since it is expected that one schwa may be realised in different ways in the same word, the single-schwa words (where the schwa phoneme appears once in the corresponding phoneme string of the word) are analysed first. Also, we consider two modelling approaches: the GOP and the variant-selection approaches. We begin with the GOP approach.

Two main results are obtained from the tagging process, for each schwa observed:

- The most likely vowel candidate (apart from schwa).
- Whether the phoneme matches the broad schwa category best, or is better matched to one of the Sepedi vowels.

Single-schwa analysis

The tags were evaluated to determine how the observed vowel distribution is affected by non-acoustic factors. Specifically, the following non-acoustic factors were considered: speaker identity, word identity, and grapheme identity.

In Fig. 8.3 the vowel distribution is displayed per speaker. This figure shows that speaker identity plays a minor role in determining which vowel is produced: this observation is different when compared with the word identity shown in Fig. 8.4. This would immediately exclude other speaker-specific factors such as gender or age. Speaker-specific features are therefore not considered further.

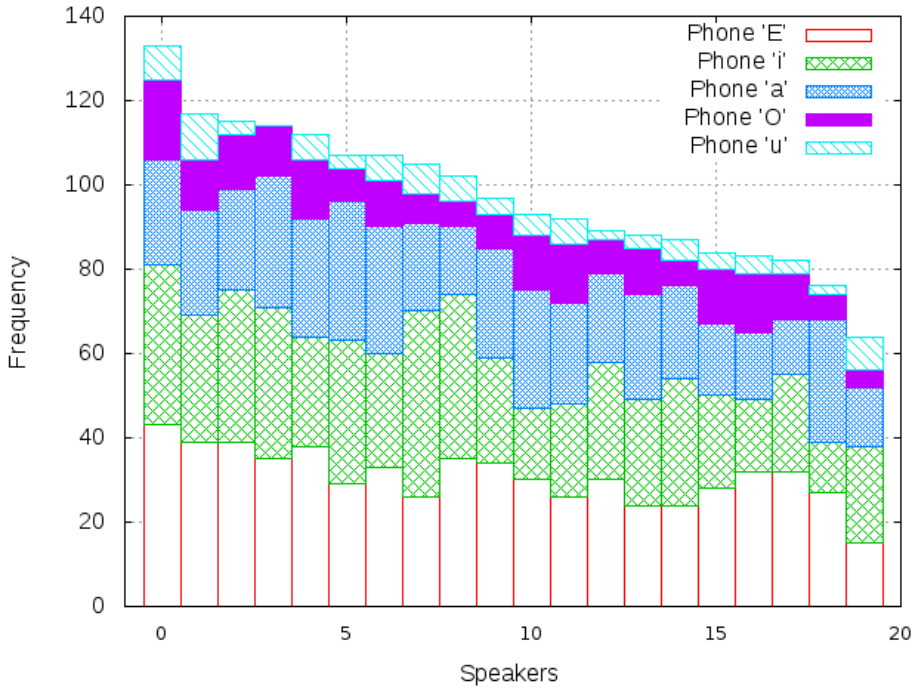

FIGURE 8.3: The number of times each vowel was observed per speaker using GOP approach for single-schwa words.

When analysing the tags per target word (all examples of the English words produced during code switching are considered together), a clearer pattern emerges. The number of times a specific vowel is observed per word is shown in Figure 8.4.

Based on the large role the word orthography plays, we next consider the graphemic string that produced a specific vowel (for example, *-a-, -i-, -io-* or *-ure-*). Note that the true prediction for each of these strings, in the specific context used, is 'schwa'.

In Fig. 8.5 we display the number of times each vowel is observed per unique grapheme string, for the single-schwa subset. We see that the graphemes *-ure-, -a-, -ia-, -ou-* and *-ur-* were mostly (but not exclusively) realised as phoneme */a/* during code switching. There are three graphemes *-e-, -io-, -er-* which were mostly realised as the Sepedi

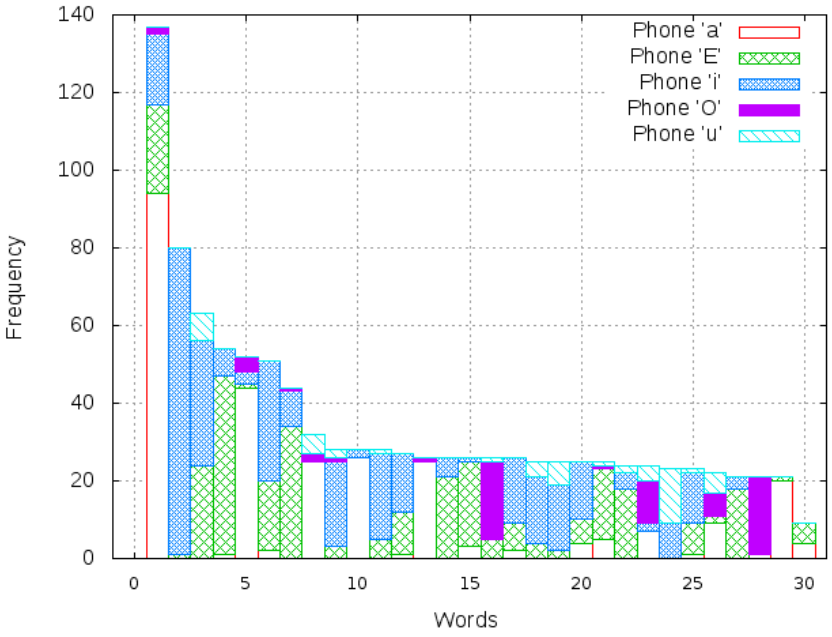

FIGURE 8.4: The number of times each vowel was observed per unique word using GOP for single-schwa words.

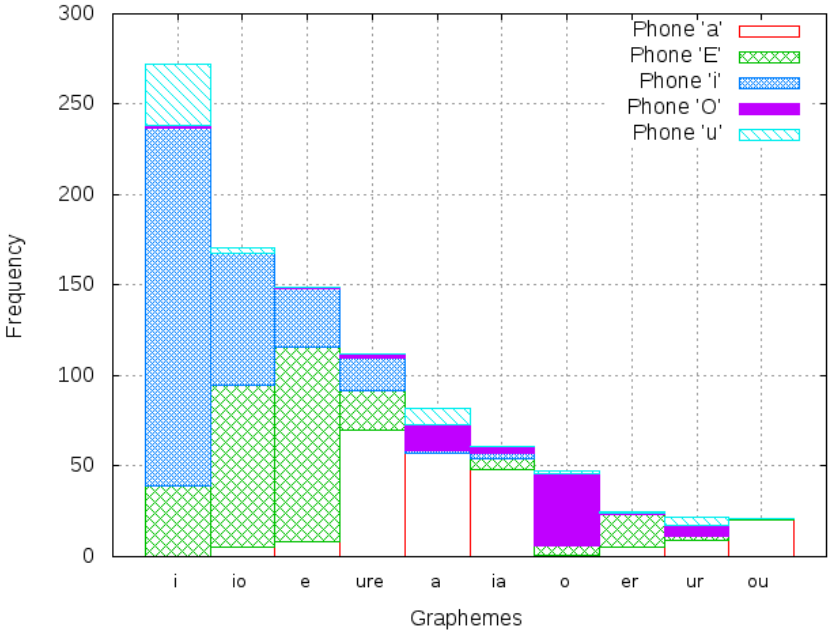

FIGURE 8.5: The number of times each vowel was observed per unique grapheme string using GOP approach for single-schwa words.

phoneme */E/*. There was very little confusion with the graphemes *-o-* and *-i-* as they were almost always realised as phonemes */O/ and /i/*. The most unpredictable grapheme string was *-io-*, which was realised as either */i/* or */E/*, two phonemes that overlap on the vowel chart in Fig. 8.1.

The variant-selection approach was primarily developed to deal with words that contained multiple schwas within a single word. However, since it is a bit more accurate, the single-schwa analysis was repeated. This produced similar results, as shown in Appendix D.

Multi-schwa analysis

We also analysed the number of times each vowel was observed when the schwa appeared multiple times in a word using the GOP approach. Here, all schwas in a word are forced to have the same identity. (Per word, all occurrences of schwa are mapped to the same phoneme.) The results are shown in Figure 8.6. In this figure, we considered words with a multiple schwa occurrences only. In Figure 8.4 the same behaviour was observed as it was with a single schwa per word.

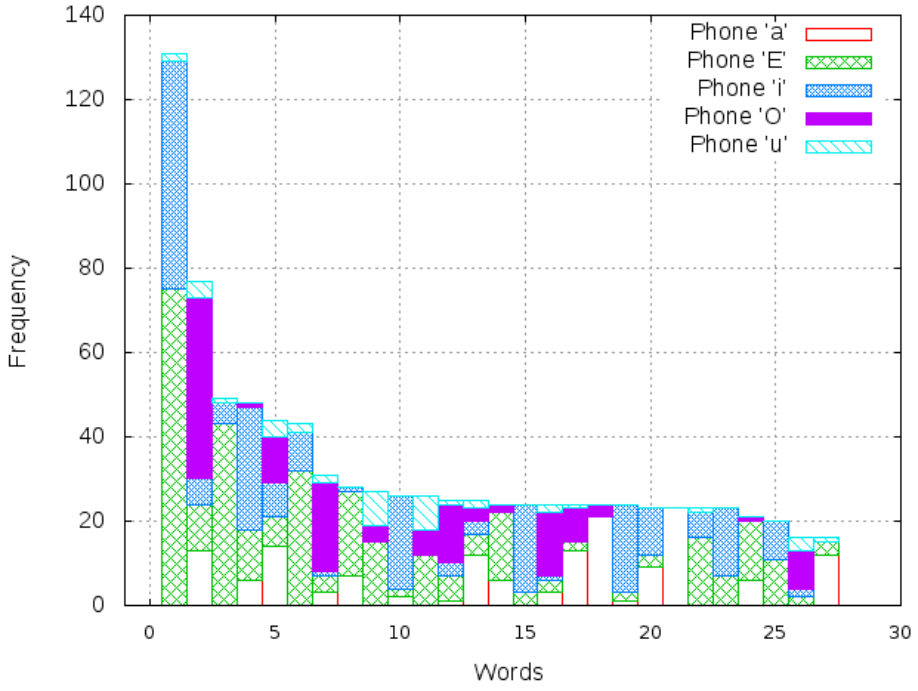

FIGURE 8.6: The number of times each vowel was observed per unique word using GOP approach for multiple-schwa words.

We again analysed the number of times each vowel was observed when the schwa appeared multiple times in a word using the variant-selection approach. In Figure 8.7 we compare the prediction of phoneme labels for words with multiple occurrences of schwa. We measure the number of times each vowel was observed per unique word using the

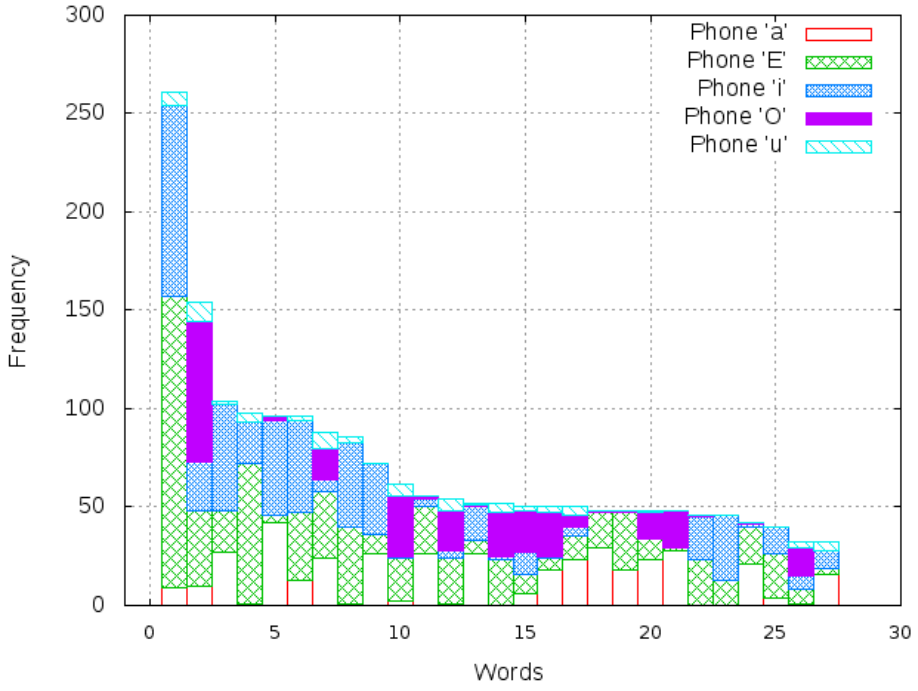

FIGURE 8.7: The number of times each vowel was observed per unique word using variant-selection approach for multiple-schwa words.

variant-selection approach. There were ten out of 27 words whose schwa phoneme was realised as the Sepedi phoneme *E*. The five words produced a Sepedi phoneme */O/*. The phoneme labels */a/, /i/* were predicted with the same number of words (six).

In Figures 8.8 we compare the prediction of the phoneme labels using the same approach (that is, variant-selection) for words containing multiple occurrences of schwa. We measure the number of times each vowel was observed per unique grapheme string. With words containing multiple occurrences of schwa, the graphemes *io, e, u* were realised as phoneme */E/*. The other graphemes *a,er, ia, or* were realised as phoneme */a/*. The grapheme *i* and *o* were not confusable at all and realised as phoneme */i/* and */o/*, respectively.

8.2.3.6 Tag distribution

In total, 1 947 observations of schwa were auto-tagged, using the process described in Section 8.2.3.1. The vowel /E/ was tagged most frequently, and the vowel /u/ least. The number of tags associated with each vowel is shown in Fig. 8.9.

Phoneme label */u/* was the least predicted, which suggest that it can be left out.

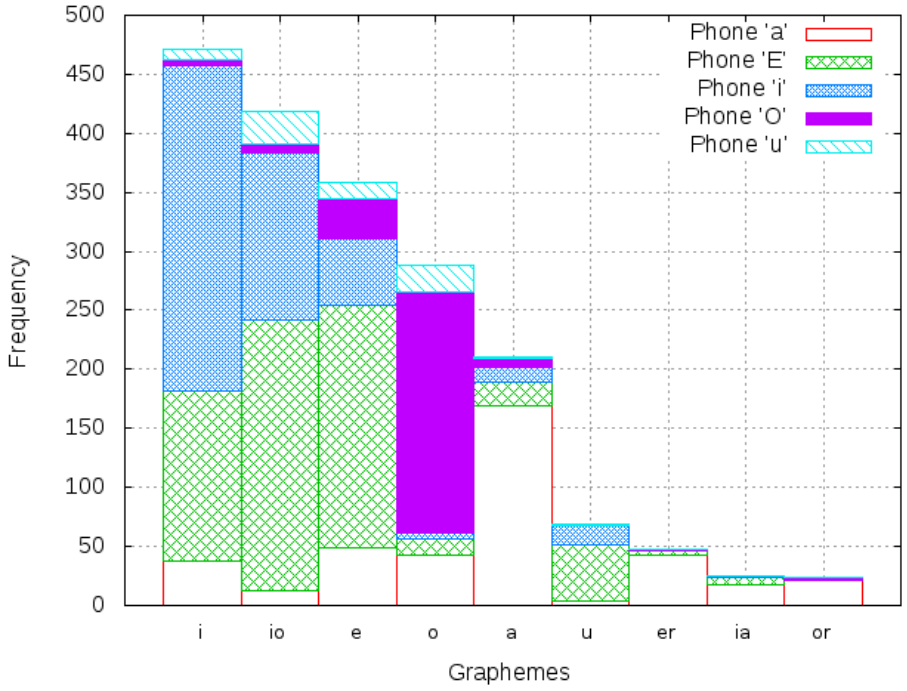

FIGURE 8.8: The number of times each vowel was observed per unique grapheme string using variant-selection approach for multiple-schwa words.

8.2.3.7 Classification process

Once the entire corpus has been labelled (using the auto-tagging procedure described in Section 8.2.3.1), we review the results and identify possible features that influence vowel prediction. The features under consideration are (a) utterance id, (b) source word, (c) grapheme string, (d) word pronunciation, and (e) speaker id. We use Naive Bayes classification to obtain an indication of whether these features are applicable.

From the above analysis, we select triphone and grapheme string as input for a simple Naive Bayes (NB) classifier to obtain an initial indication of predictability. Using 10-fold cross-validation, we train models using only these non-acoustic features, and evaluate using the ten test partitions. Results are shown in Table 8.8 obtained from the GOP auto-tagger for single vowel occurrence per word. The rows show the phoneme label tags and the columns are the label predictions from the NB classifier. We show the agreement level on the diagonal, both in terms of counts and percentage accuracy. An overall classification accuracy of 67.36% is achieved.

We see that the Sepedi phoneme labels /E/ and /i/ can be predicted fairly easily, although they are mutually highly confusable. On the other hand, the phoneme tag /u/ is never hypothesised. It also has the lowest occurrence and predictability when analysed from a speaker-based, word-based, as well as grapheme-based perspective. It therefore

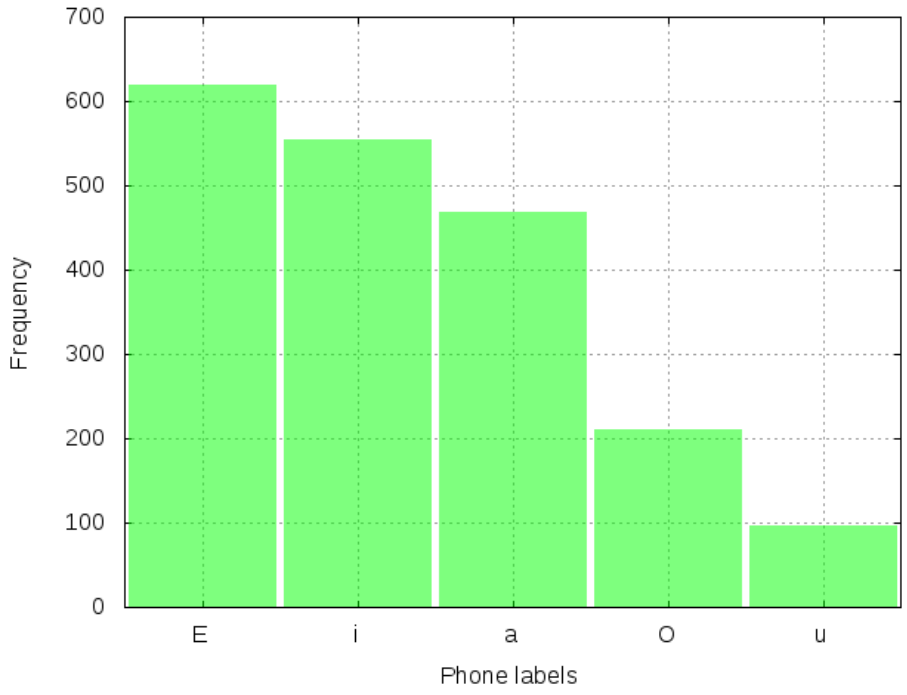

FIGURE 8.9: Vowel distribution in the GOP auto-tagged SPCS corpus.

TABLE 8.8: Confusion matrix when performing 10-fold cross-validation with non-acoustic features only using the GOP approach for single vowel occurrence per word.

	E	O	a	i	u
E	**214(73.5)**	6	29	41	1
O	2	**47 (68.1)**	15	1	4
a	18	13	**189 (84.8)**	0	3
i	105	1	22	**198 (60.7)**	0
u	4	10	5	34	**0 (0)**

seems better to not predict /u/ at all, but rather select the second-most probable candidate. Alternatively, it is better not to introduce a variant for words where /u/ is predicted as the most probable realisation. Which of these two strategies is better, will require further experimentation. The grapheme string seems to be the main predictor of the matrix pronunciation of code-switched speech.

A NB classifier is used to train the models to predict phoneme labels using triphone and grapheme string using variant-selection approach for single vowel occurrence per word. The results are shown in Table 8.9. An overall classification accuracy of 67.57% was achieved. As it was observed with the GOP approach, the Sepedi phoneme labels /E/ and /i/ are highly confusable. Also, the phoneme tag /u/ was the least hypothesised label.

TABLE 8.9: Confusion matrix when performing 10-fold cross-validation with non-acoustic features only using the variant-selection approach for single vowel occurrence per word.

	E	O	a	i	u
E	**205 (73.21)**	5	32	38	1
O	3	**53 (79.10)**	6	1	4
a	18	8	**194 (76.68)**	23	10
i	103	0	18	**191 (61.22)**	0
u	3	5	2	32	**7 (14.29)**

The NB classifier is again used to predict phoneme labels tags, this time, when there are multiple vowels per word using the variant-selection approach. The results are shown in Table 8.10. We consistently see the confusability of the Sepedi phonemes /E/ and /i/. The phoneme tag /u/ is not hypothesised in this case. An overall classification accuracy of 68.99% is achieved.

TABLE 8.10: Confusion matrix when performing 10-fold cross-validation with non-acoustic features only using the variant-selection approach for multiple vowel occurrences per word.

	E	O	a	i	u
E	**503 (74.74)**	14	30	126	0
O	18	**231 (87.50)**	15	0	0
a	45	44	**270 (68.35)**	36	0
i	173	5	15	**315 (62.01)**	0
u	39	25	2	6	**(0) 0**

8.2.4 Vowel analysis

After analysing the performances of the GOP and variant-selection approaches using the schwa phoneme, we decided to proceed with the variant-selection approach which performed better and can model more than one vowel per word. In Figures 8.10 and 8.11, we show the number of times each vowel was observed per unique grapheme string occurring once in a word using the variant-selection approach for phoneme /A:/ and /3:/. For the English phoneme /A:/, all the graphemes (*a, al, ar, are*) were mostly observed as Sepedi phoneme /a/, with limited samples tagged as /E/. The graphemes *er, ur, ir, or, ear* for the English phoneme /3:/ were tagged as either /a/ or /E/, but for most, the number of times the secondary (/E/) phoneme was tagged was relatively low. One exception is *ur*, which was clearly realised as both /a/ and /E/.

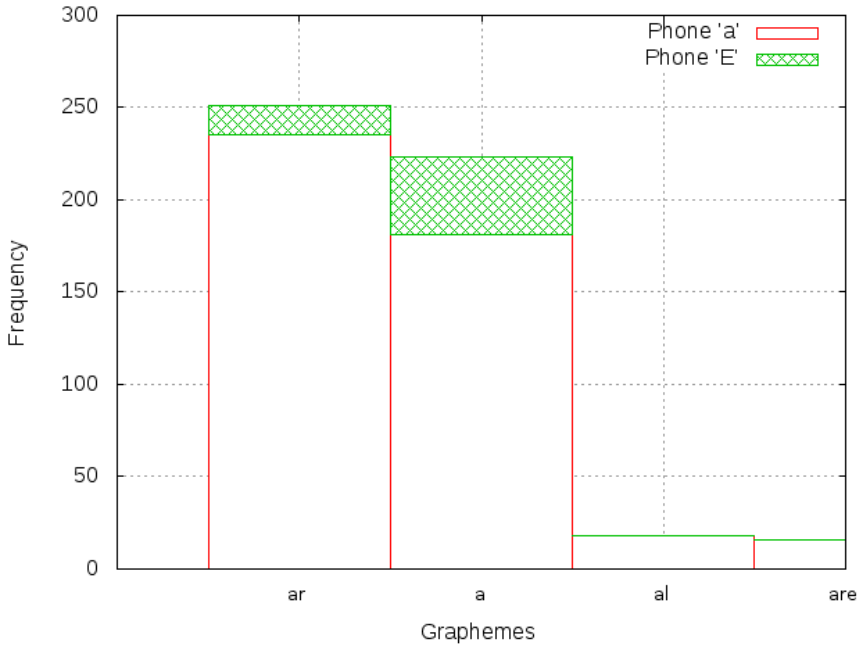

FIGURE 8.10: The number of times each vowel was observed per unique grapheme string occurring once in a word using variant-selection approach (*A:*).

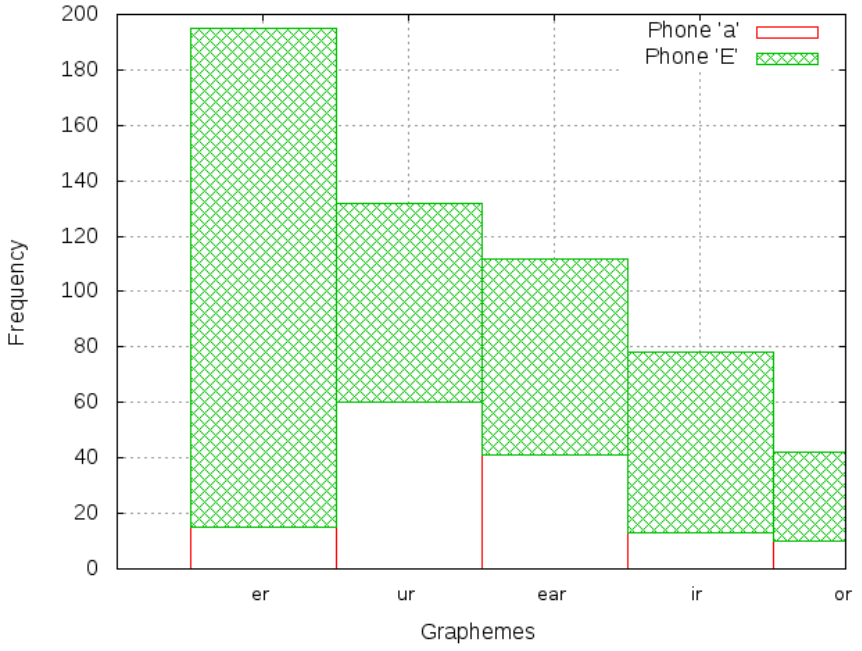

FIGURE 8.11: The number of times each vowel was observed per unique grapheme string occurring once in a word using variant-selection approach (*3:*).

In this section we analyse the vowel substitution prediction using the variant-selection approach. In Table 8.11, we show vowel prediction which is based on grapheme strings. The schwa phone and diphthongs are mapped based on the grapheme string representation. These English phones are mapped to more than one Sepedi phone. The multiple phone labels of the English phones generated from this analysis suggest that during code switching pronunciation variations are generated.

TABLE 8.11: Grapheme-based vowel substitution prediction: graphemes influence results.

Phone	Grapheme	Phone tag
ai	i, igh, ye	a
ai	i, igh, ye	E
ai	i, igh, ye	i
ai	y	a
@	ur, ou, a, ure, ia	a
@	io, e, er	E
@	o	O
@	i	i
i@	e,io	j
i@	iu	E
i@	e, ia	i
u@	eu, ua	u
u@	ou	O
@i	a, ay, eigh	E
@i	a, ai, ay	i

In Table 8.12 we show the vowel substitution prediction for the phonemes which directly map from English phonemes to Sepedi phonemes. Apart from schwa, all the English vowels are mapped to a single phoneme except for one phoneme /{/ which has two labels.

8.2.5 Consonant analysis

The consonants were also modelled and the respective phoneme labels predicted. In Table 8.13 we show the phoneme substitution prediction. The top part shows the single predicted phoneme labels. The bottom part has multiple predicted phoneme labels.

Some phonemes have a direct mapping to one Sepedi candidate phoneme. Others have multiple mappings with more than one Sepedi candidate phoneme. Most English fricatives are mapped to single Sepedi candidate phoneme. On the other hand, the tops are mapped to two Sepedi candidate phonemes.

TABLE 8.12: Grapheme-based vowel substitution prediction: graphemes do not influence results for these vowels.

Phoneme	Phoneme tag
{	a, E
O:	O
e	E
u:	u
i:	i
au	a
e@	E
@u	O
A:	a
3:	E
Q	O

TABLE 8.13: Consonant substitution prediction

Phoneme	Phoneme tag
r	r
z	s
g	k_>
D	l'
T	t_>
Z	S
tS	tS_h
v	B,f
k	k_>, k_h
p	p_>, p_h
t	t_>, t_h
b	B, p_>
d	l', t_>

8.3 English-Sepedi phoneme mappings

Here we aim to develop mappings based on the analysis of the predicted phoneme labels to determine the actual phonemes suitable to improve the recognition of code-switched speech. All the English phonemes were modelled using the variant-selection approach and we considered those with more than one Sepedi candidate phoneme.

In Appendix A, Table A.6, we show the final mappings obtained using the variant-selection approach. The English schwa phoneme has been found to be most heavily influenced by the grapheme string. For this reason, we created a phoneme mapping by including grapheme string for every occurrence of the schwa phoneme. These mappings

are used to create different pronunciation dictionaries in Section 8.4, which are then evaluated in Section 8.5.

8.4 Pronunciation dictionary

The SPCS baseline pronunciation dictionary was developed as discussed in Section 7.5.2. In this section we discuss the development of the pronunciation dictionaries that are based on the learning from the tag analysis. These will be used to evaluate recognition accuracy of the ASR system for code-switched speech.

The two pronunciation dictionaries are generated as follows:

1. The pronunciation dictionary referred to here as *dict_var_selection* was developed by using the mapping we learned in Section 8.3. We used the Lwazi Sepedi G2P rules to predict the pronunciation of the Sepedi words in the SPCS corpus. The pronunciation of the English words was developed by starting from manually fixed pronunciations, and mapping all the English phonemes to Sepedi phonemes using the phoneme labels generated in Section 8.2. The final mappings are shown in Appendix A, Table A.6. The resulting dictionary contains multiple variants with Sepedi phonemes only as some phonemes have multiple mappings.

2. The pronunciation dictionary referred to as *dict_comb_g2p_var_selection* was created by using two approaches. It was developed with pronunciations generated with Sepedi G2P rules for both Sepedi and English words. The pronunciations from *dict_var_selection* dictionary generated in (1) for the English words were added to this dictionary as variants. This dictionary also contains multiple variants with Sepedi phonemes only.

3. The pronunciation dictionary, referred to as *dict_comb_g2p_ipa_var_selection*, was created from three approaches. The pronunciations of the English words were generated by mapping the English phonemes to Sepedi phonemes using IPA features. These pronunciations were added to the *dict_var_selection* and *dict_comb_g2p_var_selection* as additional variants.

Note that the tag analysis was only performed on the SPCS *training* data. (See section 8.2.3.1.) The tags of the test set (SPCS-eval) were not included in the analysis used to generate the pronunciation dictionaries.

8.5 Recognition evaluation of code-switched speech

In this section we describe the process followed to train the acoustic models. The data and the language model used are also mentioned. The three acoustic models are trained as follows:

1. Training data: The NCHLT-206-trn training set was used to develop the acoustic models. The details about this training set are discussed in Section 7.2.
2. Evaluation set: The evaluation set used in this analysis is the SPCS-eval set as described in Section 7.2.1.
3. Language model: The interpolated trigram language model trained using NCHLT and SPCS text corpora as described in Section 7.3 was used. Word recognition was performed.
4. ASR system: A HMM-based ASR system was developed using the HTK toolkit. The system parameters are as discussed in Section **??**.
5. Acoustic models: The process followed to train the acoustic models is outlined below. These models are trained using the NCHLT-206-trn train data with dictionaries described in Section 8.4.
   (a) The acoustic models are trained using the *dict_var_selection* pronunciation dictionary.
   (b) The acoustic models are trained using the *dict_comb_g2p_var_selection* pronunciation dictionary.
   (c) The acoustic models are trained using the *dict_comb_g2p_ipa_var_selection* pronunciation dictionary.

8.5.1 Results

The word recognition accuracy for different acoustic models and dictionary combinations are shown in Table 8.14. The results were generated by training three systems, using the three dictionaries mentioned in Section 8.4. Three matching sets of acoustic models were created: *nchlt_nso_var_selection* (using *dict_var_selection*), *nchlt_nso_g2p_var_selection*, (using *dict_comb_g2p_var_selection*) and *nchlt_nso_g2p_ipa_var_selection* (using *dict_comb_g2p_ipa_var_selection*).

In this section, we determine whether the results obtained in Section 7.5.5, Table 7.7 can be improved further. The word recognition accuracy obtained when the acoustic

models *nchlt_nso_ipa_var_am* were tested on the spcs-eval test set was 58.02%. (See Table 7.7.) In Table 8.14 we show the word recognition accuracy for acoustic models trained with single approaches and a combination of approaches. The single approach is the process of predicting the pronunciation of foreign language words using techniques such as G2P rules or IPA mapping. The combination of the approaches then combines the pronunciations from different techniques as variants.

When we generated the *dict_var_selection* pronunciation dictionary using variant selection, the recognition accuracy obtained was 58.12%. These results were better by 6.6% absolute when compared to the previous best single approach (*nchlt_nso_ipa_am*). With the combined approaches, the results for *nchlt_nso_g2p_ipa_var_selection* were better by 3.7% compared to *nchlt_nso_ipa_var_am*.

TABLE 8.14: Word recognition accuracy for acoustic models and dictionary combinations for single and combined approaches.

Single approaches		Combined approaches	
Dictionary/ Acoustic models	spcs-eval	Dictionary/ Acoustic models	spcs-eval
nchlt_nso_eng_am	39.51	nchlt_nso_ipa_var_am	58.02
nchlt_nso_am	45.24	nchlt_nso_g2p_var_selection	58.83
nchlt_nso_mat_am	49.31	nchlt_nso_g2p_ipa_var_selection	61.68
nchlt_nso_ipa_am	51.49		
nchlt_nso_var_selection	58.12		

8.5.2 Modelling technique summary

When evaluating the performance of different approaches to model code-switched speech, it became apparent that one approach on its own is not enough. The pronunciation variations encountered during code switching require different techniques to model. Combining the pronunciations generated with variant-selection with both the matrix language G2P and IPA mapping approaches yielded better results, and achieved recognition accuracy of 61.68%. Interestingly, before the variant-selection approach was available, a recognition accuracy of 58.02% was achieved when combining earlier methods. The variant-selection approach on its own achieves 58.12%, which is not significantly different.

The variant-selection pronunciation modelling approach requires a speech corpus to obtain pronunciation modelling rules. Once these have been obtained, they can be applied to other corpora. (The audio analysis need not be repeated.) The process to obtain the rules can be summarised as follows:

1. Using known word lists, or other text-based LID techniques, split the corpus vocabulary into matrix and embedded word lists.

2. Develop a pronunciation dictionary (*matrix_dict*) by predicting the pronunciation of all words with the matrix language G2P rules.

3. Predict the pronunciations of the foreign language with the foreign language G2P rules ($foreign\_dict$). It might be important to manually verify pronunciations, especially if the foreign language is English.

4. Obtain mapping candidates:

   - Select portions of the corpus that only contain matrix language speech.
   - Train acoustic models using only the selected portions and *matrix_dict*.
   - Recognise all data using the acoustic models and dictionary just developed, and *matrix_dict* as recognition dictionary.
   - Create a new reference dictionary that replaces the pronunciations of all foreign words in *matrix_dict* with the pronunciations in $foreign\_dict$. (Combining these two dictionaries in this way then still results in only a single pronunciation per word.)
   - Score the recognition and reference dictionary against one another using PDP scoring.
   - From the PDP confusion matrix, determine candidate mappings.

5. Develop a new set of acoustic models (*mixed_am*) using the acoustic training data and the same combined *matrix_dict* and $foreign\_dict$ pronunciation dictionary used previously.

6. For each foreign phoneme individually:

   - Create pronunciation variants using the mapping candidates.
   - Force align the possible candidates using the training data and the *mixed_am* acoustic models.
   - Extract phoneme labels per word, per utterance from the alignments. (Each word will, therefore, be separately tagged for every occurrence in an utterance.)
   - Perform G2P alignment to identify the relevant grapheme strings. These may be longer than a single grapheme, including additional graphemic context.
   - Using the grapheme strings, determine the most probable phoneme labels for the foreign language phoneme.

- Measure the frequency counts of the phoneme labels with respect to the grapheme string associated with the foreign language phoneme.

7. Generate phoneme mapping rules based on the frequency counts for all phoneme and grapheme string pairs. Include one or more options per pair.

8. Generate a pronunciation dictionary from the phoneme mapping rules. This will then only contain phonemes from the matrix language phoneme set.

In addition, once the variant-selection dictionary has been created, the following variants may be introduced:

1. Add pronunciations generated with the G2P rules as variants to the final dictionary above.

2. Add pronunciations generated with the IPA approach as variants to the dictionary above

8.5.3 Frequency of misrecognised words

We measure the frequency of the number of times a word was recognised by counting the number of occurrence in the reference file in relation to the number of times it was recognised in the decoded file. This frequency is compared between the *nchlt_nso_var_selection*, *nchlt_nso_g2p_var_selection*, and *nchlt_nso_g2p_ipa_var_selection* acoustic models. The overall performance of the *nchlt_nso_g2p_ipa_var_selection* acoustic models is better and they only perform poorly with some words as shown in Figure 8.12.

8.6 Additional G2P analysis

From work to this point, there is strong evidence that the grapheme string is an important indicator of the P2P mapping that can be expected. Foreign word pronunciation prediction is often considered to consist of a two-step process: foreign G2P, followed by foreign-to-native P2P. We conjecture that it is more useful to consider the second step to rely on both phoneme and grapheme identity, that is, grapheme-based P2P (G+P2P). It should be noted that the initial G2P process is still required. Just, instead of G2P then P2P, we propose G2P then G+P2P. In this section, we demonstrate such a process.

In this section we analyse the G2P rules obtained from the phoneme labels generated with the variant-selection approach to predict phoneme labels for code-switched data. In Section 8.6.1 we describe the input data that was used with the G2P predictions. In

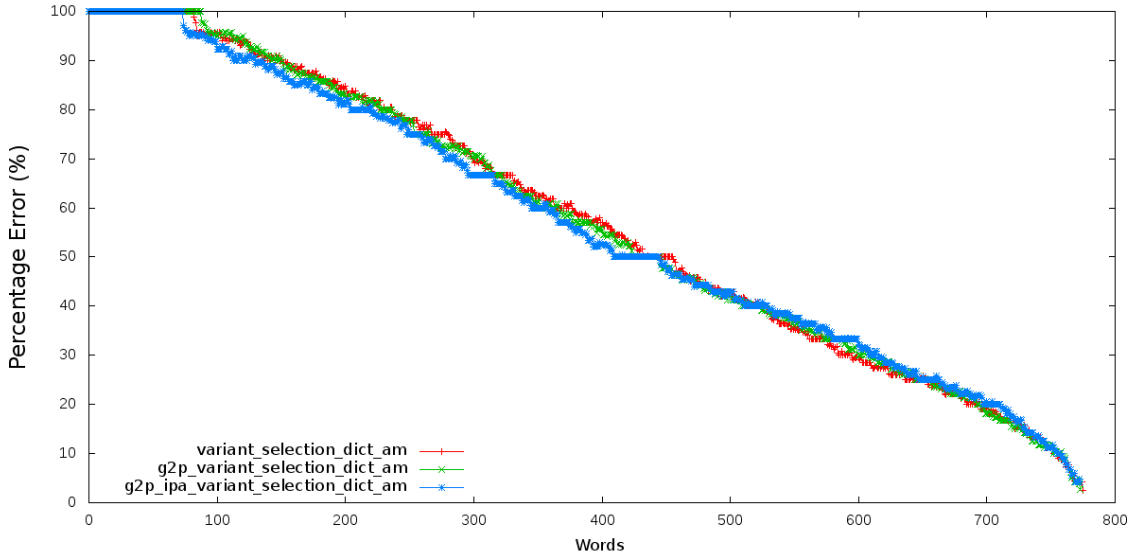

FIGURE 8.12: The percentage error of misrecognised words with trigram language model

Section 8.6.2 we discuss the process followed to generate the results for analysis. The results obtained are presented in Section 8.6.3.

8.6.1 Data

The results of the entire corpus that has been labelled for single and multi words separately using variant-selection based auto-tagging procedure described in Section 8.2.3.1 is used as input to the D&R algorithm to extract the rewrite rules. This data consists of the English word, the English pronunciation phoneme sequence, the G2P alignment, and per phoneme to be analysed, the phoneme labels predicted by the auto-tagger (one per actual audio sample of the word). The process therefore only starts once English pronunciations have been generated, and per phoneme, all predicted occurrences of that phoneme have been auto-tagged with a Sepedi observed phoneme.

8.6.2 G+P2P process

A dictionary is created with all the possible variants observed in the tagged data. From the data, we select only those variants that occur more than a certain percentage of the time using a specific threshold. This threshold can be increased or decreased, and allows us to ignore idiosyncratic pronunciations or recognition errors. The threshold was set at 10%, 20% and 40%. We then create a pseudo-phoneme by combining the possible phoneme variants [82]. For example, phonemes could be /a/, /E/, or /a_E/ to indicate that both /a/ and /E/ can be expected.

Each English (predicted) phoneme is considered in isolation. When modelling each phoneme, only pronunciations containing that specific phoneme are selected to form the dictionary. All words are now replaced with 'word patterns' that identify the grapheme aligned to the target phoneme, and maps this to the auto-tagged phoneme. Examples of such word patterns are shown in Table 8.15.

TABLE 8.15: Examples of word patterns and their corresponding auto-tags.

word pattern	auto-tag	#tagged
-a-fforde	a	26
ag-e-ncy	E	34
ag-e-ncy	i	10
chall-e-nge	E	22
chall-e-nge	a	3
chall-e-nge	i	1
econ-o-my	E	5
econ-o-my	O	20

These word patterns are then used to extract the rewrite rules describing the phoneme in question using the D&R algorithm. More specifically:

- we identify those patterns that give rise to the target phoneme where each pattern consists of a focal grapheme, surrounded by a left and right context. For example, *-a-n* or *-a-#*, where # indicates a word boundary.
- we extract rewrite rules from these patterns using D&R. For example:

```
-a- # =>  O_a      #auditiona#   #drama#   #socialisa#
-a- n =>  E_a       #johannesburg#
```

The example of the rewrite rules above shows the grapheme patterns *-a-n* which give rise to a phoneme label /O_a/, followed by words from which the pattern was observed. The phoneme label /O_a/ shows either the phoneme /O/ or /a/ can be hypothesised.

The results obtained from the D&R algorithm needed to be refined. For the schwa analysis, all /u/ patterns were excluded since it did not have much effect. It was also evident from the NB classifier that the /u/ label was not hypothesised. Since the number of samples is limited, the G2P had miss-alignments. For this analysis these were corrected by introducing grapheme clusters prior to alignment. These clusters were treated as single graphemes. For example: *kk*, *ll*, and *io*.

8.6.3 G2P results

We show the results of the rewrite rules for one example in the given listings below. These listings show the rules extracted for phoneme { with the threshold of 10%, 20% and 40%. The symbol # is used to indicate word boundaries. The rules become more complex as the threshold is decreased. At high thresholds rules become very straightforward, e.g. at threshold of 40%.

With limited data, the extracted rewrite rules were able to successfully extract pattern to predict specific phoneme labels. The example below shows that the pattern *l-a-n* and *pl-a-* can do well in predicting phoneme labels /E/ and /a/, respectively.

```
l-a-n   =>      E       #l-a-ndline#
pl-a-   =>      a       #pl-a-nete#   #pl-a-stic#
```

However, there are instances where the extracted patterns predicted interesting variants. This also could be contributed by the number of examples available in the corpus. In this instance, the pattern *-o-* could predict both the /E/ and /a/ phonemes.

```
-o-             =>      E_a      w-o-rkers
-ear-s  =>      E_a       y-ear-s
```

The rules show that with more data and high thresholds the rules become more straight forward. Also, it is clear that the grapheme strings have positive influence on phoneme to phoneme mappings.

Phoneme { with 10% threshold

```
-a-             =>      E_a     #b-a-g#   #ch-a-llenge#   #fl-a-t#   #g-a-s#
    #gr-a-duata#   #gr-a-duation#   #j-a-nuary#   #joh-a-nnesburg#   #municip-a
    -lity#   #r-a-nd#   #r-a-nds#   #re-a-lity#   #s-a-lad#   #st-a-ndard#   #
    st-a-ndford#   #tr-a-ffic#
 -a-    =>      a       #-a-ccessa#   #-a-damant#   #-a-dvocate#   #-a-
    griculture#   #-a-lbum#   #-a-ngle#
c-a-    =>      a       #ac-a-demy#   #c-a-tegory#
r-a-c   =>      E       #tr-a-ck#   #tr-a-cksuit#
b-a-n   =>      a       #b-a-nka#   #b-a-nkeng#
-a-m    =>      E_a     #-a-m#   #c-a-mera#   #l-a-mb#   #p-a-mphlets#
l-a-n   =>      E       #l-a-ndline#
pl-a-   =>      a       #pl-a-nete#   #pl-a-stic#
-a-x    =>      E       #t-a-xi#
-a-sh   =>      E       #f-a-shion#
d-a-    =>      E       #d-a-ddy#
n-a-    =>      E       #n-a-tional#
```

```
m-a-    =>      E       #autom-a-tic#
c-a-n   =>      E       #c-a-n#
 -a-s   =>      E_a     #-a-s#
p-a-n   =>      a       #p-a-nga#
-a-mp   =>      a       #c-a-mpaign#
l-a-n   =>      E_a     #pl-a-n#
l-a-nn  =>      E       #pl-a-nning#
 -a-nd  =>      E_a     #-a-nd#
-a-lar  =>      a       #s-a-lary#
```

Phoneme { with 20% threshold

```
-a-             =>      a       #-a-ccessa#   #-a-damant#   #-a-dvocate#   #-a-
    griculture#   #-a-lbum#   #-a-m#   #-a-ngle#   #b-a-nka#   #b-a-nkeng#   #
    ch-a-llenge   #municip-a-lity#   #p-a-mphlets#   #p-a-nga#   #s-a-lad#   #s
    -a-lary#   #st-a-ndard#
r-a-    =>      E_a     #gr-a-duata#   #gr-a-duation#   #r-a-nd#   #tr-a-ffic#
l-a-    =>      E       #l-a-mb#   #l-a-ndline#
-a-s    =>      E_a     #-a-s#   #g-a-s#
r-a-c   =>      E       #tr-a-ck#   #tr-a-cksuit#
-a-t    =>      E       #autom-a-tic#   #fl-a-t#   #n-a-tional#
pl-a-   =>      a       #pl-a-nete#   #pl-a-stic#
-a-x    =>      E       #t-a-xi#
-a-sh   =>      E       #f-a-shion#
e-a-    =>      E_a     #re-a-lity#
d-a-    =>      E       #d-a-ddy#
j-a-    =>      E_a     #j-a-nuary#
c-a-    =>      a       #ac-a-demy#   #c-a-mpaign#   #c-a-tegory#
b-a-g   =>      E       #b-a-g#
h-a-n   =>      E_a     #joh-a-nnesburg#
c-a-n   =>      E       #c-a-n#
-a-me   =>      E_a     #c-a-mera#
l-a-nn  =>      E       #pl-a-nning#
 -a-nd  =>      E       #-a-nd#
l-a-n   =>      E_a     #pl-a-n#
-a-nds  =>      E       #r-a-nds#
-a-ndf  =>      E_a     #st-a-ndford#
```

Phoneme { with 40% threshold

```
-a-             =>      a       #-a-ccessa#   #-a-damant#   #-a-dvocate#   #-a-
    griculture#   #-a-lbum#   #-a-m#   #ac-a-demy#   #c-a-mpaign#   #ch-a-
    llenge#   #gr-a-duata#   #gr-a-duation#   #municip-a-lity#   #p-a-mphlets#
      #pl-a-stic#   #s-a-lad#   #s-a-lary#   #tr-a-ffic#
```

```
-a-n    =>      E       #-a-nd#   #c-a-n#   #j-a-nuary#   #joh-a-nnesburg#   #l
    -a-ndline#   #pl-a-n#   #pl-a-nning#   #r-a-nd#   #r-a-nds#
-a-t    =>      E       #autom-a-tic#   #fl-a-t#   #n-a-tional#
b-a-    =>      a       #b-a-nka#   #b-a-nkeng#
r-a-c   =>      E       #tr-a-ck#   #tr-a-cksuit#
-a-ng   =>      a       #-a-ngle#   #p-a-nga#
-a-x    =>      E       #t-a-xi#
-a-sh   =>      E       #f-a-shion#
g-a-    =>      E       #g-a-s#
d-a-    =>      E       #d-a-ddy#
e-a-    =>      E_a     #re-a-lity#
 -a-s   =>      E_a     #-a-s#
b-a-g   =>      E       #b-a-g#
l-a-m   =>      E       #l-a-mb#
t-a-n   =>      E_a     #st-a-ndford#
c-a-t   =>      a       #c-a-tegory#
-a-me   =>      E_a     #c-a-mera#
-a-ne   =>      a       #pl-a-nete#
-a-nda  =>      a       #st-a-ndard#
```

To apply these rules the following process would then be followed:

1. The English words are predicted with standard G2P rules.
2. Apply English phoneme-specific G2P rules per phoneme to be remapped.

There are factors that still need to be addressed before exploring this avenue further, which are that (a) we have observed some examples of vowel harmony, where words are more likely to be realised as for example, /m i n i s t r i/ or /m E n E s t r i/, than other combinations. This outcome requires further study; (b) The actual tagged corpus is small, and therefore quite noisy. Grapheme-to-phoneme misalignment is to be better addressed in a way that requires less human intervention. Human intervention is labour intensive and prone to errors. While the G2P approach is not immediately applicable as a modelling approach, the rules generated (as included in Appendix C) are themselves of interest for further analysis.

8.7 Discussion

In this chapter, two interlinked processes were demonstrated: (1) a technique for auto-tagging an acoustic corpus, and (2) an analysis of the auto-tags to determine useful

non-acoustic features for pronunciation prediction. These could make it possible to generate a lexicon for code-switched speech, prior to having encountered acoustic data related to the specific vocabulary to be modelled.

We found that the vowel-tagging process is quite difficult for humans to do, showing a medium to fair inter-subject agreement if subjects are allowed to select the embedded phoneme itself (schwa in this case) as an option. When samples where speakers were deemed to have produced a substitute from the matrix language only were included, inter-subject agreement increased to 85.7%. The auto-tagger agreed more strongly with one of the subjects, achieving a matching accuracy of 60.5% agreement with one and 84.9% with the other.

The non-acoustic features found to be most useful were triphone and grapheme strings. A simple Naive Bayes classifier achieved an average of 67.6% classification accuracy by using these features when evaluated using 10-fold cross-validation and the variant-selection approach on single-schwa words. The classification accuracy achieved for words with multiple occurrences of schwa was 69.0%.

From the schwa phoneme analysis, we conclude that the best matrix pronunciations for the English schwa phoneme are *E, i, O* and *a*, with specific substitutions occurring based on the larger grapheme string. A one-to-one vowel substitution is only possible if the grapheme representation is either *i* or *o*. Pronunciation dictionaries can now be developed for predicting the matrix pronunciations of English words in code-switched Sepedi speech by using the trained classifier when using only the vocabulary (no acoustic data) as input.

We determined that the phoneme to phoneme mapping was not enough. The grapheme strings have influence on the mapping on the phonemes. Some phonemes were mapped to more than one candidate phonemes while others were mapping to single candidate phonemes. During the vowel analysis, we found that most vowels, including some diphthongs but excluding /{/ have a one-to-one phoneme mapping. During the consonant analysis, we found that fricatives have a one-to-one phoneme substitution. The stops, however, have mapping of the English phoneme to more than one Sepedi candidate phonemes.

The best single modelling technique evaluated, was the one where the phoneme mappings were predicted with the variant-selection approach. The best recognition accuracy was obtained when the pronunciations predicted with variant-selection approached were combined with both the G2P predictions as well as the IPA-based phoneme mapping approach. The obtained recognition accuracy was 61.7%.

8.8 Conclusion

In this chapter, we demonstrated the use of a new technique for generating pronunciations for code-switched data. This approach maps the pronunciations from the embedded language (English here) to the matrix language (Sepedi) using information not only about the English phoneme identity, but also from the graphemes in context. Further analysis suggests that, with sufficient data, accurate *phoneme-specific* G2P rules can be extracted in this manner.

In this way, it can be shown that pronunciation variants can be created for pronunciation dictionaries without involving acoustic data. This approach becomes a cost saving mechanism for pronunciation modelling for ASR systems in resource-scarce environments. In Chapter 9, we give a summary and discuss the implications of our findings.

Chapter 9

Conclusion

9.1 Introduction

Dealing with code-switched speech in ASR systems is an interesting but challenging task. This chapter discusses the milestones achieved in understanding and addressing some of these challenges. We discuss the contribution made in this thesis (Section 9.2), the significance of this contribution (Section 9.3), as well as future work (Section 9.4).

9.2 Summary of contribution

This thesis analysed the phenomenon of code switching in Sepedi and investigated approaches to model code-switched speech within ASR systems. It demonstrated that a lexicon-based approach using non-acoustic features could improve the modelling of code-switched speech. The main contributions made in this study can be summarised as follows:

- Development of new speech resources tailored for the studying of code-switched speech in a language pair not previously studied. These include:

    1. Development and analysis of a new speech resource containing English speech as spoken by Sepedi speakers. The NCHLT Sepedi-English code-switched subset (NSECSS) corpus does not contain naturally occurring code-switched speech, but rather English as spoken by Sepedi speakers. (Section 5.2.) This NSECSS corpus is particularly useful as its recording conditions and format match two related NCHLT corpora: English produced by English speakers and Sepedi produced by Sepedi speakers.

2. Design and development of a speech corpus from true examples of code switching (the Radio Broadcast Corpus, in Section 5.3). This corpus allowed better insight into the mechanisms, reasons and frequency of code switching.
3. Design and development of a Sepedi code-switched corpus consisting of audio and transcription files. The Sepedi Prompted Code-Switched (SPCS) Corpus was based on true code switching prompts, with each prompt recorded by multiple speakers to capture pronunciation variability. (Section 5.4.)

As part of corpus development, resource collection and analysis tools were developed and evaluated.

- Investigating the methods of code switching, the frequency of code switching and the reasons for code switching observed among Sepedi speakers, using corpus analysis. Among other results, it was found that the prevalence of code switching within naturally occurring Sepedi speech was much higher than initially anticipated.

- Development of an initial Sepedi recognition system capable of recognising code-switched speech. Optimisation of this recogniser, amongst others by splitting complex consonants, thereby reducing the size of the Sepedi phoneme set and simplifying analysis. (Section 4.3). Interestingly, modelling aspiration as a separate unit, further simplified later analysis, without any detrimental effects on measured system performance.

- Analysis of different lexicon-based approaches to modelling code-switched speech, and the implications of these approaches for ASR system performance (Chapter 7). Mainly focussing on context-independent mappings, best results were obtained when creating two variants for each word identified as English: one using Sepedi G2P prediction, and the other English mapped to Sepedi using IPA. This straightforward approach outperformed other data-driven approaches evaluated.

- Development of a novel pronunciation modelling approach that can be used to predict phoneme substitutions in code-switched speech, in Chapter 8. This process is fully automated (requiring an initial code-switched speech corpus but no manual linguistic input). Once the predictors are trained, no additional acoustic information is required when modelling new vocabulary.

- The optimal phone mapping approach that considers both the graphemic context as well as the identity of the phone. The identity of the phone provides a standard sound. The same phone can be realised differently given the graphemic context.

Some of the above findings have already been published or accepted for publication [83–87]. The speech recognition accuracies obtained in this study improved significantly

after applying the optimal approaches to model code-switched speech. However, these results do not match the monolingual speech recognition accuracies.

9.3 Significance of contribution

There is a high frequency of code switching in naturally occurring Sepedi speech, and when this is not explicitly modelled in ASR systems, performance is poor. At the same time, the code switching speech phenomenon is not well studied for Bantu languages, and very little information is available on this topic from a speech technology perspective. Techniques that only require a corpus as input, rather than extensive linguistic information, would be ideal.

Prior to this study, no speech corpora were available to study code switching in any of the Sotho-Tswana languages. Both the corpora themselves, as well as the techniques used to develop them are therefore of future use. Corpus development in resource-scarce environments is often unexpectedly time- and resource-intensive, and new corpora can therefore be of value to many additional researchers.

Techniques developed in this study are essentially language-independent, and can therefore be of use to other languages with limited resources. As many of the world's minority languages coincide with other languages – and therefore also experience code switching – these techniques should be applicable well beyond Sepedi.

9.4 Future work

There are three main directions for future work emanating from this thesis. The first one concerns the design and development of additional code-switched speech corpora. There are few, if any, readily available code-switched corpora for most language pairs. The second one involves investigating the extent to which the newly proposed techniques are applicable to other languages. The third relates to further refinement of the current approach.

Using the techniques and tools developed in this study, the process to design, collect and annotate code-switched corpora can now be made very efficient. It would, therefore, be useful to extend the collection of code-switched speech corpora to additional language pairs. As such a collection will focus on true examples of code switching, this would also allow a more detailed understanding of the code switching phenomenon.

Using newly developed or existing corpora in additional language pairs, it would be possible to determine if the results obtained here are corpus-specific or whether these findings indeed generalise across languages.

Additional refinements to the current approach can include:

- We would like to determine the implications of modelling various categories of code-switched speech, such as standard English, semi-transformed English and Sepedi loan words and would like to consider ways in which these can be modelled separately.
- In general, we would like to confirm the implications of newer modelling techniques (such as deep neural networks) on the trends observed here.
- It is expected that using the auto-tagging process to generate training data to train alternative G2P models will be a productive area for further research. These G2P models would then be used to predict the pronunciations of code-switched words.
- Further analysis of the use of a more sophisticated classifier to predict phoneme labels would be desirable. The proposed mappings would be tested by conducting empirical ASR experiments to determine the effect of such mappings.

Finally, as part of this study, it became clear that language practitioners have difficulty in identifying individual phonemes accurately, especially when language mixing occurs and phonemes may be realised in non-standard ways. We have therefore started experimenting with tools to support this process, and would like to extend this in future work.

Appendix A

Phone mappings

TABLE A.1: The English and Sepedi phone sets in X-SAMPA notation: vowels.

English phones	Sepedi phones
i	i
i:	–
–	u
u:	–
E	E
3:	–
–	O
O:	–
a	a
A:	–
@	–
{	–
Q	–
@i	–
ai	–
Oi	–
@u	–
au	–
i@	–
u@	–
e@	–

TABLE A.2: The English and Sepedi phone sets in X-SAMPA notation: consonants.

English phones	Sepedi phones
tS	–
–	tS_>
–	tS_h
–	ts_>
–	ts_h
–	kx
–	ps_>
–	ps_h
–	pS_h
d_0Z	d_0Z
f	f
v	–
T	–
D	–
s	s
z	–
S	S
Z	–
x	–
–	h
h\	h\
–	K
–	G
–	B
–	p\s
–	p\S
–	BZ
p	–
–	p_h
–	p_>
b	–
t	–
–	t_h
–	t_>
d	–
k	–
–	k_h
–	k_>
g	–
–	tl_>
–	tl_h
m	m
n	n
–	J
N	N
–	m_j
–	r
l	l
r\	–
j	j
w	w
–	l'

TABLE A.3: The mapping of English to Sepedi phones using confusion matrix

English phone	Mapped Sepedi phone
@	E
@i	E
@u	O
A:	a
D	t_>
3:	E
O:	O
Q	O
T	s
Z	S
ai	a
au	O
b	p_>
d	t_>
e	E
e@	E
g	k_>
i@	E
i:	i
k	k_>
p	p_>
r	r
t	t_>
tS	S
u@	O
u:	O
v	B
z	s
{	a

Table A.4: Linguistic IPA mapping of English to Sepedi phones

English phone	Mapped Sepedi phone	Feature that's different: Eng value	Feature that's different: Sep value
@	E (E, O and a all options)	central vowel	Unrounded, mid-low, front
{	E		
3:	E	rounded, central, duration	Unrounded, front
A:	a	low-back, duration	low-front
ai	a i	diphthong	two monophones, same identities
au	a u	diphthong	two monophones, same identities
b	p_>	voiced	unvoiced, ejective
d	t_>	voiced	unvoiced, ejective
D	B	dental	bilabial
e@	E E	diphthong	@ close to front vowel, more likely to be produced as E
g	k_>	voiced	unvoiced, ejective
@i	E i	diphthong	@ close to front vowel, more likely to be produced as E
i:	i	duration	
i@	i E	diphthong	@ close to front vowel, more likely to be produced as E
k	k_h	aspirated	
O:	O	duration	
Oi	O i	diphthong	two monophones, same identities
p	p_h	aspirated	
Q	O	low	mid-low
r	r	approximant	trill
t	t_h	aspirated	
T	f	dental	labiodental
tS	tS_h	aspirated	
@u	O u	diphthong	@ close to back vowel, more likely to be produced as O
u:	u	duration	
u@	u O	diphthong	@ close to back vowel, more likely to be produced as O
U	u	near-high, near-back	high, back
v	B	labiodental	bilabial
z	s	voiced	voiceless
Z	d_0Z	fricative	affricate

Table A.5: The mapping of English to Sepedi phones using IPA

English phone	Mapped Sepedi phone
@i	a i
@u	a u
A:	a
D	B
3:	E
O:	O
Q	O
T	f
Z	d_0Z
ai	a i
au	a u
b	p_>
d	t_>
e	E
e@	E a
g	k_>
i@	i a
i:	i
k	k_>h
p	p_>h
r	r
t	t_>h
tS	t_>S h
u@	u a
u:	u
v	B
z	s
{	a

TABLE A.6: The mapping of English phones to Sepedi phones using variant-selection approach

English Phone	Grapheme	Sepedi Phone
ai	i, igh, ye	a
ai	i, igh, ye	E
ai	i, igh, ye	i
ai	y	a
@	ur, ou, a, ure, ia	a
@	io, e, er	E
@	o	O
@	i	i
i@	e,io	j
i@	iu	E
i@	e, ia	i
u@	eu, ua	u
u@	ou	O
@i	a, ay, eigh	E
@i	a, ai, ay	i
{	–	a, E
O:	–	O
e	–	E
u:	–	u
i:	–	i
au	–	a
e@	–	E
@u	–	O
A:	–	a
3:	–	E
Q	–	O
r\	–	r
z	–	s
g	–	k_>
D	–	l’
T	–	t_>
Z	–	S
tS	–	tS_h
v	–	B,f
k	–	k_>, k_h
p	–	p_>, p_h
t	–	t_>, t_h
b	–	B, p_>
d	–	l’, t_>

Table A.7: The English phones

English phone
O:
e
u:
i:
r\
v
k
p
t
z
g
b
d
D
T
Z
tS
ai
au
@
e@
i@
u@
@i
@u
A:
3:
Q
{

Appendix B

Dictionary Validation

Firstly, the English words were manually verified to determine if they were generic words, proper names, or acronyms. The purpose of this process was to determine the complexity of predicting the pronunciations of these words. Secondly, the predicted pronunciations were also manually verified to determine whether the word pronunciation pairs were valid, with the aim of developing correct pronunciation dictionaries.

The validation process of the words was as follows:

1. The list of word pronunciation pairs was provided in a spreadsheet to the respondent (professional linguist).
2. The task was to read a word, then determine if it was valid or not. If valid, the word would be marked valid. Otherwise, the word would be marked invalid and the respondent would indicate whether the word was an abbreviation, acronym, spelled out word, proper name, foreign word, spelling error, partial word, or not sure what the token was.

The validation process of the pronunciations was as follows:

1. Using the same spreadsheet above and given the word-pronunciation pair, the pronunciation of every word would be evaluated.
2. The respondent was asked to use the standard pronunciation of each word (rather than including additional variants) unless needed.
3. Only one pronunciation was provided, as there was no need for multiple pronunciations at that stage

4. The pronunciations were marked if valid. If not, correct pronunciation was provided.

5. A not sure option was given if the pronunciation of the words could not be determined.

Validation results:
The results of the validation process for the English words occurring in the NCHLT and SPCS corpus are shown in Table B.1. We also show the results of the validation of the root (English words used to derive the semi-modified English words) words of the semi-English words.

TABLE B.1: The categorisation results of English words in Sepedi NCHLT and SPCS corpus.

Word category	NCHLT	SPCS	SPCS (semi-English)
Generic	422	329	27
Proper name	83	10	1
Proper noun	2	–	–
Foreign word	17	1	–
Spelled out	30	3	–
Spelling error	4	3	–
Acronym	2	–	–
Abbreviation	1	–	–

In Table B.2, we show the results of the validation of the English words pronunciations in the NCHLT and SPCS corpus. As in the previous table, the pronunciations of the root words of the SPCS semi-English words are also verified.

TABLE B.2: The manual correction results of English words pronunciation in the Sepedi NCHLT and SPCS corpus.

Corpus	Lookup	Predicted	Valid	Corrected
NCHLT	332	229	451	110
SPCS	273	73	274	72
SPCS (semi-English) root	28	0	22	6

Appendix C

Default & Refine rules analysis

Phone {

```
-a- => a        -a-ccessa   -a-damant   -a-dvocate   -a-griculture   -a-lbum
    -a-m  ac-a-demy   c-a-mpaign   ch-a-llenge   gr-a-duata   gr-a-duation
   mu0nicip-a-lity   p-a-mphlets   pl-a-stic   s-a-lad   s-a-lary   tr-a-ffic
-a-n => E       -a-nd   c-a-n   j-a-nuary   joh-a-nnesburg   l-a-ndline   pl-a
   -n   pl-a-nning   r-a-nd   r-a-nds
-a-t => E       autom-a-tic   fl-a-t   n-a-tional
b-a- => a       b-a-nka   b-a-nkeng
r-a-c =>       E       tr-a-ck   tr-a-cksuit
-a-ng =>       a       -a-ngle   p-a-nga
-a-x => E       t-a-x0i
-a-sh =>       E       f-a-shion
g-a- => E       g-a-s
d-a- => E       d-a-ddy
e-a- => E_a     re-a-lity
 -a-s =>       E_a     -a-s
b-a-g =>       E       b-a-g
l-a-m =>       E       l-a-mb
t-a-n =>       E_a     st-a-ndford
c-a-t =>       a       c-a-tegory
-a-me =>       E_a     c-a-mera
-a-ne =>       a       pl-a-nete
-a-nda =>      a       st-a-ndard
```

Phone A_c

```
-a- => a        -a-fter   -a-re   -a-rt   -a-rts   b-a-rcode   d-a-nsa   dep-a
   -rtment   dr-a-ma   gr-a-nt   gr-a-nts   h-a-lf   h-a-rmony   p-a-rkeng   p
   -a-rtying   rech-a-rga   sh-a-rp   tr-a-nsport   tr-a-nsporta   tr-a-
   nsportilego
```

Phone b

```
-b- =>  p_>      -b-ail   -b-allroom   -b-asis   -b-locka   -b-lockile   -b-ody
       -b-u0tternu0t   -b-umper   -b-ursary   cele-b-rities   contri-b-u0ta
    disa-b-ility   distur-b-a   exhi-b-ita   pu-b-lic   reha-b-ilitation   ro-b
    -ot
-b-an =>        B        -b-anka   -b-ankeng
-b-us =>        B        -b-us   -b-usy
-b-e => B_p_>    -b-etter   novem-b-er
s-b- => B        johannes-b-urg
t-b- => B        net-b-all
l-b- => B        al-b-um
a-b-l =>        B        mokonsta-b-le
e-b-o =>        B        face-b-ook
a-b-o =>        B_p_>    a-b-ortion
-b-oa =>        B        -b-oard
-b-es =>        p_>      -b-est
-b-ag =>        B        -b-ag
-b-ar =>        B_p_>    -b-arcode
-b-ito =>       B_p_>    exhi-b-itors
-b-itio =>      B        exhi-b-ition
-b-usi =>       B_p_>    -b-usiness
cem-b- =>       B        decem-b-er
pta-b- =>       B_p_>    corrupta-b-le
-e- =>  B_p_>    cub-e-s
```

Phone d

```
-d- =>  l_a      -d-aily   -d-ate   -d-ays   -d-ecember   -d-egree   -d-
    emocracy   -d-epartment   -d-ia0gnostic   -d-iesel   -d-octor   -d-ollara
     aca-d-emy   affor-d-e   king-d-om   presi-d-ente   ra-d-io0   resi-d-ents
      sha-d-es   stu0-d-io0   up-d-ata
-d-  => t_>      boar-d-   hea-d-   sala-d-
n-d- => t_>      an-d-   atten-d-ance   girlfrien-d-   in-d-usteri   lan-d-line
       ran-d-   ten-d-era
 a-d- =>        l_a_t_>  a-d-dress   a-d-ministration   a-d-vocate
-d-i => l_a      -d-isability   -d-iscount   -d-istrict   -d-istricting   -d-
    isturba   au0-d-itiona   au0-d-itione   in-d-icators   lea-d-ing   sta-d-
    ium
or-d-  =>       l_a_t_>  recor-d-   standfor-d-
-d-r => t_>      -d-rama   -d-rugs
o-d- => t_>      barco-d-e   bo-d-y
-d-s => t_>      awar-d-s   ran-d-s
```

```
l-d- => l_a      chil-d-
e-d- => t_>      intereste-d-   me-d-al   re-d-   scare-d-   u0nite-d-
a-d-a =>         t_>      a-d-amant
o-d-i =>         l_a_t_>  reco-d-ilwego
l-d-e =>         l_a_t_>  stakehol-d-ers
-d-o  =>         t_>      -d-o
-d-re =>         l_a_t_>  -d-ressing
-d-ow =>         t_>      -d-ownfall
on-d- =>         l_a      con-d-ition   secon-d-
r-d-er =>        t_>      or-d-er
r-d-en =>        l_a_t_>  awar-d-eng
a-d-e  =>        l_a_t_>  gra-d-e
 -d-an =>        l_a_t_>  -d-ansa
-d-rai =>        l_a      -d-rainwa
san-d- =>        l_a      thousan-d-
 co-d- =>        l_a      co-d-e
```

Phone E_c

```
-o- =>  E_a      w-o-rkers
-r- =>  E        ci-r-cle   ci-r-cleng   fu-r-nitura   gi-r-lfriend   je-r-sey
     johannesbu-r-g   netwo-r-keng   pe-r-sonally   pu-r-pose   se-r-vice   se
   -r-vices   unive-r-sities   unive-r-sity   ve-r-se
-u- =>  a        b-u-rsary   dist-u-rba
-ear- =>         E        l-ear-ners   l-ear-nership
-ear-s =>        E_a      y-ear-s
```

Phone g

```
-g- =>  k_>      -g-as   -g-irlfriend   -g-ospel   -g-rade   -g-raduation   -g-
   rant   -g-rants   -g-riller   cate-g-ory   de-g-ree   dia0-g-nostic   pan-g
   -a
-g-  => G        johannesbur-g-
-g-o  =>         G        recodilwe-g-o   transportile-g-o
a-g- => G_k_>    a-g-riculture   lea-g-ue
-g-s => G        dru-g-s
-g-l => G_k_>    an-g-le
a-g-  =>         G        ba-g-
-g-rou =>        G_k_>    -g-roup
 -g-ree =>       G_k_>    -g-reen
-g-raduata =>    G_k_>    -g-raduata
-x- =>  k_>      e-x-hibita   e-x-hibitors
-x-hibitio =>    G_k_>    e-x-hibition
```

Phone i_c

```
-e- =>  i        -e-asy   app-e-al   l-e-ading   l-e-ague   outr-e-ach   p-e-
    ace   p-e-ople   p-e-ter   r-e-charga   r-e-habilitation   t-e-am
-e-  => E        m-e-
-e-p => E_i      r-e-possesa
-i- =>  i        pol-i-ce
-i-ll =>        E        sk-i-lls
-ie- => i        d-ie-sel
-ee- => i        degr-ee-   eight-ee-n   fift-ee-n   gr-ee-n   ninet-ee-n
    sevent-ee-n   wh-ee-lchair
```

Phone k

```
-c- =>  k_>h     -c-oach   -c-ode   -c-olleges   -c-ommenta   -c-omments   -c-
    ommission   -c-ommu0nity   -c-ondition   -c-onfront   -c-onfu0sion   -c-
    ontribu0ta   -c-opa   -c-rate   -c-ru0tches   a-c-count   a-c-counteng
    bar-c-ode   cir-c-le   demo-c-racy   dis-c-ount   out-c-ome   s-c-hool
    sit-c-om
i-c- => k_>      au0tomati-c-   dia0gnosti-c-   mu0si-c-   plasti-c-   publi-c-
       ti-c-   traffi-c-
-c-k => k_>      blo-c-kile   kno-c-kile   lu-c-ky   stu-c-kile   ti-c-kets
    tra-c-k   tra-c-ksuit
e-c- => k_>_k_>h         e-c-onomy   re-c-odilwego   re-c-ord   se-c-ond
-c-t => k_>      distri-c-t   distri-c-ting   do-c-tor   infe-c-tion   proje-c-
    t
-c-u => k_>_k_>h         -c-ubes   -c-ulture
-c-orr =>       k_>_k_>h         -c-orruptable   -c-orruption
 -c-om =>       k_>_k_>h         -c-ompany   -c-ompanying
-c-a => k_>h     -c-all   -c-amera   -c-ampaign   -c-an   -c-ategory   a-c-
    ademy   advo-c-ate   indi-c-ators   s-c-ared
i-c-u =>        k_>h     agri-c-ulture
e-c-h =>        k_>      te-c-hnology
-c-ce =>        k_>      a-c-cessa
-c-ka =>        k_>_k_>h         blo-c-ka
-c-opy =>       k_>      -c-opy
-c-ket  =>      k_>h     ti-c-ket
-c-ommente =>   k_>      -c-ommente
-c-onferenc =>  k_>_k_>h         -c-onferenceng
-c-onference  =>        k_>h     -c-onference
-k- =>  k_>      -k-ilometer   -k-ingdom   ban-k-a   ban-k-eng   faceboo-k-
    mo-k-onstable   networ-k-eng   par-k-eng   sta-k-eholders
s-k- => k_>h     s-k-ills
o-k-a =>        k_>h     o-k-ay
-k-er =>        k_>h     wor-k-ers
```

```
-q- =>  k_>      -q-ualifaele   -q-uality   -q-uantum
-r- =>  k_>h     ci-r-cleng
-x- =>  k_>      si-x-0   ta-x-0i
-x-p => k_>_k_>h         e-x-port
-x-c => k_>h     e-x-cellence
```

Phone O_c

```
-a- =>  O        -a-uditiona   -a-uditione   -a-uto   -a-utomatic   c-a-ll
    downf-a-ll
-a-r => a        aw-a-rdeng   aw-a-rds
b-a- => a        b-a-llroom
 -a-ll =>       O_a      -a-ll
tb-a- =>        O_a      netb-a-ll
-o- =>  O        -o-rder   ab-o-rtion   b-o-ard   diimp-o-rt   p-o-rtrait
    pret-o-ria   rec-o-dilwego   rec-o-rd   rep-o-rt
-r- =>  O        affo-r-de   expo-r-t   fo-r-   fo-r-m   isuppo-r-ta   spo-r-t
      spo-r-ts   suppo-r-t   tou-r-namenteng   transpo-r-t   transpo-r-ta
    transpo-r-tilego
-r-th =>        O_a      no-r-th
 o-r- =>        O_a      o-r-
ifo-r- =>       O_a      unifo-r-m
```

Phone p

```
-o- =>  p_>      exp-o-rt
-p- =>  p_>h     -p-age   -p-amphlets   -p-anga   -p-arkeng   -p-artying   -p-
    eace   -p-ercent   -p-ersonally   -p-eter   -p-illar   -p-olio0   -p-olish
      -p-ortrait   -p-osition   -p-osts   -p-rofile   -p-roject   -p-rovince
    -p-ublic   a-p-peal   cam-p-aign   co-p-a   co-p-y   com-p-anying   de-p-
    artment   diim-p-ort   o-p-eration   o-p-portu0nities   re-p-ort   re-p-
    ossesa   sho-p-ping   soa-p-ie
s-p- => p_>      es-p-ecially   gos-p-el   res-p-onse   s-p-inach   s-p-ort   s
    -p-orts   s-p-rinter   trans-p-ort   trans-p-orta   trans-p-ortilego
-p-  => p_>      grou-p-   shar-p-   to-p-
-p-l => p_>      -p-lan   -p-lanete   -p-lastic   sim-p-le
u-p- => p_>      isu-p-porta   su-p-port   u-p-   u-p-data
-p-re =>        p_>      -p-residente   -p-retoria   im-p-ressa
-p-pl =>        p_>      a-p-plyelang   a-p-plyele
-p-t => p_>      corru-p-table   o-p-tometry
i-p- => p_>      learnershi-p-   mu0nici-p-ality
m-p-e =>        p_>      bum-p-er
-p-tio =>       p_>S_p_>h        corru-p-tion
-p-o  =>        p_>_p_>h         re-p-o
```

```
em-p- =>        p_>_p_>h         tem-p-orary
 -p-ri =>       p_>_p_>h         -p-rimary
-p-olo =>       p_>      -p-olo
 -p-ress =>     p_>_p_>h         -p-ressure
-p-lann =>      p_>_p_>h         -p-lanning
-p-olic =>      p_>_p_>h         -p-olice
-p-rofe =>      p_>_p_>h         -p-rofessor
-p-any  =>      p_>_p_>h         com-p-any
-p-rovins =>    p_>_p_>h         -p-rovinsing
```

Phone Q

```
-a- =>  a        qu-a-lifaele   qu-a-lity   qu-a-ntum
-o- =>  0        -o-f   -o-n   -o-peration   -o-pportunities   b-o-dy   bl-o-
    cka   bl-o-ckile   c-o-lleges   c-o-mmenta   c-o-mmente   c-o-mments   c-o-
    nference   c-o-nferenceng   c-o-ntribu0ta   c-o-py   dem-o-cracy   dia0gn-o
    -stic   ec-o-nomy   f-o-llow   g-o-spel   kil-o-meter   kn-o-ckile   p-o-
    lish   pr-o-ject   pr-o-vince   pr-o-vinsing   recodilweg-o-   resp-o-nse
     rob-o-t   sh-o-pping   sitc-o-m   t-o-p   techn-o-logy   transportileg-o-
d-o- => a        d-o-ctor   d-o-llara
n-o-n =>        a        n-o-n
l-o-n =>        a        l-o-ng
```

Phone t

```
-t- =>  t_>      adaman-t-   adminis-t-ration   advoca-t-e   ar-t-   commen-t-a
       commen-t-e   cons-t-ruction   cricke-t-e   da-t-e   dia0gnos-t-ic   dis-
    t-urba   disabili-t-y   eigh-t-   expor-t-   fes-t-ival   fes-t-ivaleng
    fla-t-   gradua-t-a   gran-t-   isuppor-t-a   minis-t-ry   mokons-t-able
    mu0nicipali-t-y   ne-t-ball   ne-t-workeng   ou-t-come   percen-t-   plas-t
    -ic   presiden-t-e   projec-t-   psychome-t-ric   quali-t-y   repor-t-
    righ-t-   roas-t-   robo-t-   s-t-adium   s-t-akeholders   s-t-andard   s-t
    -andford   s-t-ation   s-t-ations   s-t-u0dio0   s-t-uckile   s-t-yle   se-
    t-   si-t-com   spor-t-   sugges-t-a   sui-t-   suppor-t-   u0niversi-t-ies
       u0niversi-t-y
 -t- => t_>h     -t-echnology   -t-elevision   -t-emporary   -t-endera   -t-ic
      -t-iming   -t-ouris0m   -t-racer   -t-rack   -t-raffic   -t-raining   -t-
    welve   -t-wo
-t-s => t_>s     ar-t-s   commen-t-s   gran-t-s   ligh-t-s   pamphle-t-s   pos-
    t-s   residen-t-s   resul-t-s   spor-t-s
-t-o => t_>h     -t-op   au-t-o   doc-t-or   exhibi-t-ors   indica-t-ors   pho-
    t-o   pre-t-oria
-t-er =>        t_>h     cen-t-er   in-t-erview   kilome-t-er   ma-t-erial   pe
    -t-er   sprin-t-er
```

```
i-t-a =>        t_>h     exhibi-t-a   invi-t-a   rehabili-t-ation
oun-t- =>       t_>_t_>h         accoun-t-   accoun-t-eng   discoun-t-
-t-t => t_>h     a-t-tendance   be-t-ter
s-t-e =>        t_>      indus-t-eri   roas-t-er
r-t-y =>        t_>h     for-t-y   par-t-ying
ni-t- =>        t_>h     commu0ni-t-y   u0ni-t-ed
-t-ee =>        t_>_t_>h         eigh-t-een   nine-t-een
p-t- => t_>h     corrup-t-able
f-t- => t_>h     af-t-er   fif-t-y
ou-t-r =>       t_>_t_>h         ou-t-reach
n-t-y =>        t_>_t_>h         seven-t-y
u-t-e =>        t_>s     minu-t-es
f-t-ee =>       t_>      fif-t-een
 -t-a =>        t_>_t_>h         -t-ax0i
r-t-i =>        t_>h     air-t-ime
i-t-i =>        t_>_t_>h         sensi-t-ive
n-t-u =>        t_>h     quan-t-um
n-t-ee =>       t_>h     seven-t-een
-t-eg =>        t_>h     ca-t-egory
-t-ea =>        t_>_t_>h         -t-eam
-t-iz =>        t_>h     ci-t-izen
ra-t- =>        t_>_t_>h         cra-t-e
ri-t- =>        t_>_t_>h         celebri-t-ies
on-t- =>        t_>h     confron-t-
da-t-a =>       t_>h     upda-t-a
es-t-  =>       t_>_t_>h         bes-t-
ne-t-e =>       t_>_t_>h         plane-t-e
-t-atu =>       t_>h_t_>h        s-t-atus
-t-ick =>       t_>_t_>h         -t-icket
eali-t- =>      t_>h     reali-t-y
mpor-t- =>      t_>_t_>h         diimpor-t-
```

Phone T

```
-r- =>  f_t_>    no-r-th
-t- =>  t_>      -t-housand   wi-t-h
n-t- => fs       mon-t-hs
```

Phone tS

```
-a- =>  t_>S     co-a-ch   outre-a-ch
-c- =>  t_>S     -c-hallenge   -c-hangile   -c-hild   re-c-harga   wheel-c-hair
a-c- => S        spina-c-h
-s- =>  S        sugge-s-tion
```

```
-t- =>  t_>S     agricul-t-ure   cul-t-ure   furni-t-ura
-t-c => S_t_>S   cru-t-ches
```

Phone v

```
-e- =>  B        univ-e-rsities   univ-e-rsity
-f- =>  f        o-f-
-v- =>  B        -v-erse   ad-v-ocate   ele-v-en   lo-v-e   no-v-ember   pro-v-
    insing   se-v-enteen   se-v-enty   ser-v-ice
i-v- => f        festi-v-al   festi-v-aleng   sensiti-v-e
-v-ie =>        f        inter-v-iew
l-v- => f        twel-v-e
n-v- => f        in-v-ita
e-v-i =>        f        tele-v-ision
-v-el =>        B_f      le-v-el
-v-inc =>       B_f      pro-v-ince
-v-ices =>      B_f      ser-v-ices
se-v-en  =>     f        se-v-en
```

Phone Z

```
-g- =>  d_0Z     chan-g-ile
-s- =>  S        confu0-s-ion   televi-s-ion
```

Appendix D

single-schwa variant selection

A variant-selection approach which provides an alternative implementation to predict phone substitutions is now used, as discussed in Section 8.2.3.2. We consider the single-schwa words for analysis first. In Figure D.1 we show the number of times each vowel was observed per unique word using variant-selection approach. From the 30 words that were analysed with this approach, ten words were realised as the Sepedi phone */i/*.

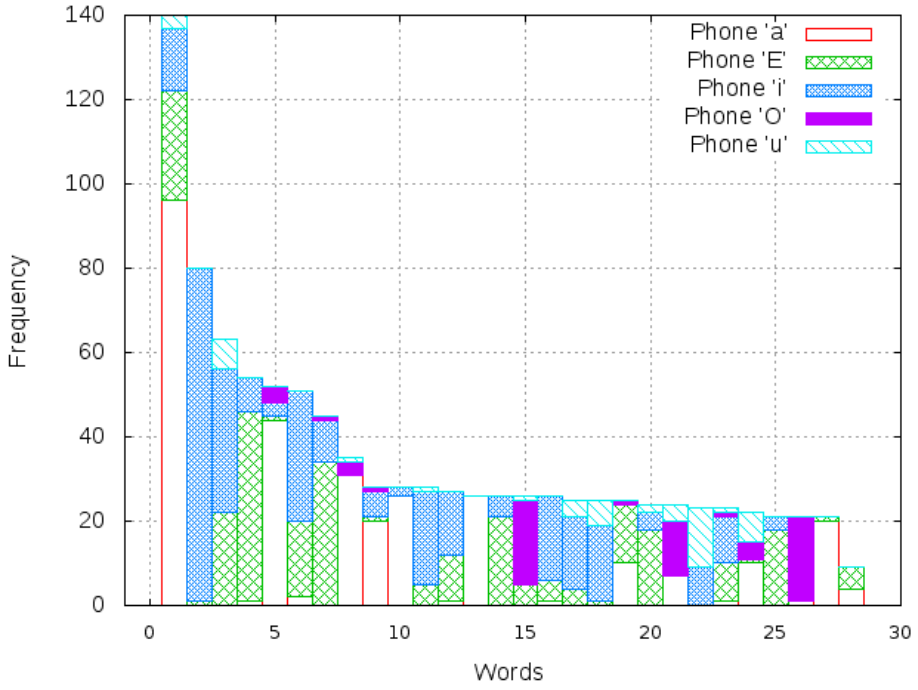

FIGURE D.1: The number of times each vowel was observed per unique word using variant-selection approach for single-schwa words.

In Figure D.2 we show the number of times each vowel was observed per unique grapheme string occurring once in a word using variant-selection approach. It is apparent that the grapheme strings *a, ure, ia, ou* were hypothesised as the Sepedi phone */a/*. As previously

observed, the Sepedi phones */E/* are very confusable. However, the grapheme *o* is always realised as the phone */O/* with little confusion.

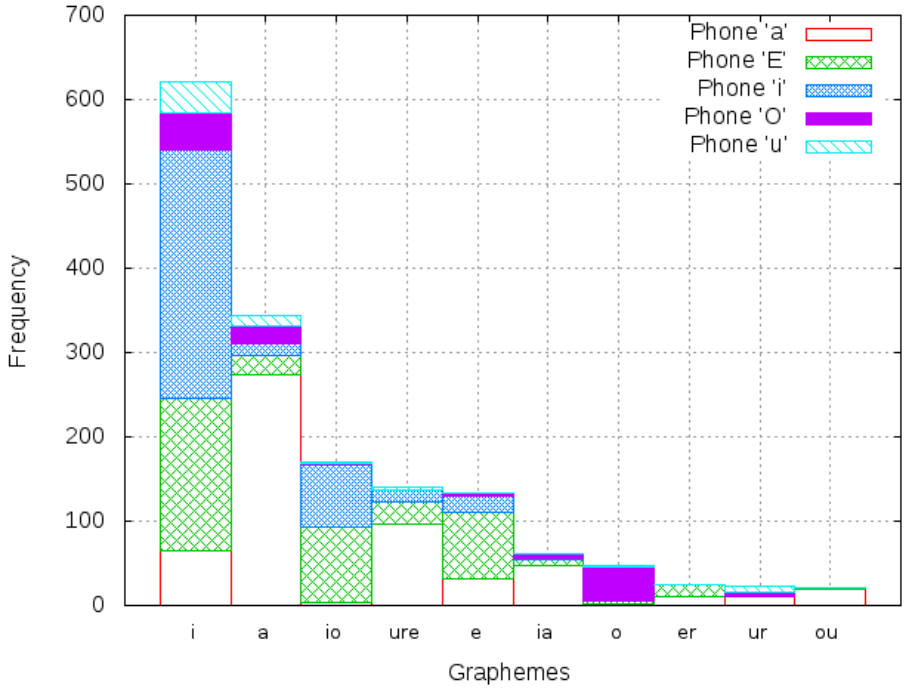

FIGURE D.2: The number of times each vowel was observed per unique grapheme string using variant-selection approach for single-schwa words.

Bibliography

[1] C. van Heerden, E. Barnard, and M. Davel, "Basic speech recognition for spoken dialogues," in *Proc. Interspeech*, Brighton, UK, Sept. 2009, pp. 3003–3006.

[2] S. Goronzy, *Robust adaptation to non-native accents in Automatic speech recognition*, Lecture Notes in Artificial Intelligence 2560. Springer-Verlag Berlin Heidelberg, 2002.

[3] C. Nilep, "Code Switching in Sociocultural Linguistics," *Colorado Research in Linguistics*, vol. 19, pp. 1–22, 2006.

[4] C. White, S. Khudanpur, and J. Baker, "An investigation of acoustic models for multilingual code switching," in *Proc. Interspeech*, 2008, pp. 2691–2694.

[5] D. Lyu and R. Lyu, "Language identification on code-switching utterances using multiple cues," in *Proc. Interspeech*, 2008, pp. 711–714.

[6] Joyce. Y. C. Chan, P. C. Ching, Tan Lee, and Houwei Cao, "Automatic speech recognition of Cantonese-English code-mixing utterances," in *Proc. Interspeech*, 2006, pp. 113 – 116.

[7] D. Lyu, R. Lyu, Y. Chiang, and C. Hsu, "Speech recognition on code-switching among the Chinese dialects," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006, pp. 1105–1108.

[8] F. Weng, H. Bratt, L. Neumeyer, and A. Stolcke, "A study of multilingual speech recognition," in *Proc. Eurospeech*, 1997, pp. 359–362.

[9] Mabule L. Mokwana, "The melting pot in Ga-Matlala Maserumule with special reference to the Bapedi culture, language and dialects," M.S. thesis, University of South Africa, South Africa, 2009.

[10] V. de Klerk, "Codeswitching, Borrowing and Mixing in a corpus of Xhosa English," *The International Journal of Bilingual Education and Bilingualism*, vol. 9, pp. 597–614, 2006.

[11] T. J. Ndwe, Etienne Barnard, and M. R. de Villiers, "Admixture practises in South African languages: Impact on speech-enabled technology design," in *Proc. IEEE IST-Africa*, Gaborone, Botswana, May 2011, pp. 1–8.

[12] Thomas Niesler and Febe de Wet, "The effect of code-mixing on accent identification accuracy," *Computer Speech and Language*, vol. 23, pp. 435–443, 2009.

[13] T. Niesler and D. Willett, "Language identification and multilingual speech recognition using discriminatively trained acoustic models," in *First ISCA ITRW on Multilingual Speech and Language Processing (MULTILING)*, 2006.

[14] J. C. Roux, E. C. Botha, and J. A. du Preez, "Developing a multilingual telephone based information system in african languages," in *Proc. 2nd International Conference on Language Resources and Evaluation (LREC)*, 2000, pp. 975–980.

[15] Krishna Bista, "Factors of code switching among bilingual English students in the University classroom: A survey," *English for specific purposes world*, vol. 9, pp. 1–19, 2010.

[16] John Eldridge, "Code-switching in a Turkish secondary school," *English Language Teaching (ELT) Journal*, vol. 50, pp. 303–311, 1996.

[17] Ahmad Abdel Tawwab Sharaf Eldin, "Socio linguistic study of code switching of the Arabic language speakers on social networking," *International Journal of English Linguistics*, vol. 4, pp. 78–86, 2014.

[18] J. Y. C. Chan, P. C. Ching, Tan Lee, and H. M. Meng, "Detection of language boundary in code-switching utterances by bi-phone probabilities," in *International Symposium on Chinese Spoken Language Processing*, 2004, pp. 293 – 296.

[19] J. Badenhorst, C. van Heerden, M. Davel, and E. Barnard, "Collecting and evaluating speech recognition corpora for nine Southern Bantu languages," in *Proc. of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2009, pp. 1–8.

[20] T. Schultz and A. Waibel, "Language-independent and language adaptive acoustic modeling for speech recognition," *Speech Communication*, vol. 35, pp. 31–51, 2001.

[21] T. Niesler, "Language-dependent state clustering for multilingual acoustic modeling," *Speech Communication*, vol. 49, pp. 453–463, 2007.

[22] C. van Heerden, N. Kleynhans, E. Barnard, and M. Davel, "Pooling ASR data for closely related languages," in *Proc. International Workshop on Spoken Languages Technologies for Under-Resourced Languages (SLTU)*, 2010, pp. 17–23.

[23] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," *Foundations and Trends Signal Processing*, vol. 1, no. 3, pp. 195–304, 2008.

[24] Xuedong Huang, Alex Acero, and Hsiao-Wuen Hon, Eds., *Spoken Language Processing: A guide to theory, algorithm, and system development*, Prentice Hall PTR, New Jersey, 2001.

[25] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. Institute of Electrical and Electronics Engineers (IEEE)*, vol. 77, no. 2, pp. 257–285, 1989.

[26] A. Benouareth, A. Ennaji, and M. Sellami, "Semi-continuous HMMs with explicit state duration for unconstrained Arabic word modeling and recognition," *Pattern Recognition Letters*, vol. 29, no. 1, pp. 1742–1752, 2008.

[27] T. Vaich and A. Cohen, "Comparison of continuous-density and semi-continuous HMM in isolated words recognition systems," in *Proc. 6TH European Conference on speech communication and technology*, Budapest, Hungary, Sept 1999, pp. 1515–1518.

[28] Basem H. A. Ahmed and Tien-Ping Tan, "Automatic speech recognition of code switching speech using 1-best rescoring," in *Proc. International Conference on Asian Language Processing (IALP)*, 2012, pp. 137–140.

[29] J. Köhler, "Multilingual phone models for vocabulary-independent speech recognition tasks," *Speech Communication*, vol. 34, pp. 21–30, 2001.

[30] Chieng-Lin Huang and Chung-Hsien Wu, "Phone set generation based on acoustic and contextual analysis for multilingual speech recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, pp. 1017–1020.

[31] Shengmin Yu and Zhang Bo Xu, "Chinese-English bilingual phone modeling for cross-language speech recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2004, pp. 917–920.

[32] M. Davel and E. Barnard, "Pronunciation prediction with Default&Refine," *Computer Speech and Language*, vol. 22, pp. 374–393, 2008.

[33] K. Torkkola, "An efficient way to learn English grapheme-to-phoneme rules automatically," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Minneapolis, 1993, pp. 199–202.

[34] Maximilian Bisani and Hermann Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.

[35] Stephan Kanthak and Hermann Ney, "Context-dependent acoustic modeling using graphemes for large vocabulary speech recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Citeseer, 2002, vol. 2, pp. 845–848.

[36] Willem D. Basson and Marelie H. Davel, "Category-based phoneme-to-grapheme transliteration," in *Proc. Interspeech*, 2013, pp. 1956–1960.

[37] International Phonetic Association, *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet*, Cambridge University Press, 1999.

[38] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li, "Context-dependent phone mapping for LVCSR of under-resourced languages," in *Proc. Interspeech*, Lyon, France, Aug. 2013, pp. 500–504.

[39] Chung-Hsien Wu, Han-Ping Shen, and Yan-Ting Yang, "Chinese-English phone set construction for code-switching ASR using acoustic and DNN-Extracted articulatory features," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, pp. 858–862, 2014.

[40] Zhirong Wang, Umut Topkara, Tanja Schultz, and Alex Waibel, "Towards universal speech recognition," in *Proc. 4th IEEE International Conference on Multimodal Interfaces (ICMI)*, Pittsburgh, PA, USA, Oct 2002, pp. 247–252.

[41] Bo-June Hsu, "Generalized linear interpolation of language models," in *Proc. IEEE workshop on Automatic Speech Recognition and Understanding (ASRU)*, Kyoto, Japan, Dec. 2007, pp. 136–140.

[42] P. R. Clarkson and R. Rosenfeld, "Statistical language modeling using the CMU-Cambridge toolkit," in *Proc. European Conference on Speech Communication and Technology (Eurospeech)*, Rhodes, Greece, Sep. 1997, pp. 2707–2710.

[43] Bo-June Hsu and James Glass, "Iterative language model estimation: Efficient data structure & algorithms," in *Proc. Interspeech*, Brisbane, Australia., Sept. 2008, pp. 841–844.

[44] Edward W. D. Whittaker, *Statistical Language Modelling for Automatic Speech Recognition of Russian and English*, Ph.D. thesis, University of Cambridge, Cambridge, UK, 2000.

[45] Nic J. de Vries, Jaco Badenhorst, Marelie H. Davel, Etienne Barnard, and Alta de Waal, "Woefzela - an open-source platform for ASR data collection in the developing world," in *Proc. Interspeech*, 2011, pp. 3177–3180.

[46] D. Lyu, T. Tan, E. Chng, and H. Li, "SEAME: a Mandarin-English code-switching corpus in South-East Asia," in *Proc. Interspeech*, 2010, pp. 1986–1989.

[47] J. Y. C. Chan, P. C. Ching, and T. Lee, "Development of a Cantonese-English code-mixing speech corpus," in *Proc. Interspeech*, 2005, pp. 1533–1536.

[48] Y. Li, Y. Yu, and P. Fung, "A Mandarin-English code-switching corpus," in *Proc. Eighth International conference on Language Resources and Evaluation (LREC)*, 2012, pp. 2515–2519.

[49] Han-Ping Shen, Chung-Hsien Wu, Yan-Ting Yang, and Chun-Shan Hsu, "CECOS: A Chinese-English code-switching speech database," in *Proc. Speech Database and Assessments (Oriental COCOSDA)*, Hsinchu, Taiwan, Oct 2011, pp. 120–123.

[50] Laurent Besacier, Etienne Barnard, Alexey Karpov, and Tanja Schultz, "Automatic speech recognition for under-resourced languages: A survey," *Speech Communication*, vol. 56, pp. 85–100, 2014.

[51] Dong Yu, Li Deng, Peng Liu, Jian Wu, Yifan Gong, and Alex Acero, "Cross-lingual speech recognition under runtime resource constraints," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4193–4196.

[52] K. C. Sim and H. Li, "Context-sensitive probabilistic phone mapping model for cross-lingual speech recognition," in *Proc. Interspeech*, 2008, pp. 2715–2718.

[53] Viet Bac Le, Laurent Besacier, and Tanja Schultz, "Acoustic-phonetic unit similarities for context dependent acoustic model portability," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006, pp. 1101–1104.

[54] J. J. Sooful and E. C. Botha, "Comparison of acoustic distance measures for automatic cross-language phoneme mapping," in *Proc. International Conference on Spoken Language Processing (ICSLP)*, 2002, pp. 521–524.

[55] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li, "Context-dependent phone mapping for acoustic modeling of under-resourced languages," *International Journal of Asian Language Processing*, vol. 23, no. 1, pp. 21–33, 2015.

[56] S. M. Witt and S. J. Young, "Phone-level pronunciation scoring and assessment for interactive language learning," *Speech communication*, vol. 30, no. 2-3, pp. 95–108, 2000.

[57] Marelie H. Davel, Charl J. van Heerden, and Etienne Barnard, "Validating smartphone-collected speech corpora," in *Proc. International Workshop on Spoken Languages Technologies for Under-Resourced Languages (SLTU)*, 2012, pp. 68–75.

[58] Nir Friedman, Dan Geiger, and Moises Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, pp. 131–163, 1997.

[59] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. F. M. Ng, B. Liu, P. S. Yu, Z. H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.

[60] Pedro Domingos and Michael Pazzani, "On the optimality of the simple Bayesian classifier," *Machine Learning*, vol. 29, pp. 103–130, 1997.

[61] M. Meteer and J. R. Rohlicek, "Statistical language modeling combining n-gram and context-free grammars," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1993, vol. 2, pp. 37–40.

[62] Andreas Stolcke, "SRILM - An extensible language modeling toolkit," in *Proc. Intl. Conf. Spoken Language Processing (ICSLP)*, Denver, Colorado, September 2002, pp. 901–904.

[63] E. Barnard, M. Davel, and C. van Heerden, "ASR corpus design for resource-scarce languages," in *Proc. Interspeech*, Brighton, UK, Sept. 2009, pp. 2847–2850.

[64] M. H. Davel and O. Martirosian, "Pronunciation dictionary development in resource-scarce environments," in *Proc. Interspeech*, 2009, pp. 2851–2854.

[65] Steve Young, Gunnar Evermann, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Valtcho Valtchev, and Phil Woodland, "The HTK book," *Cambridge University Engineering Department*, vol. 3, pp. 175, 2002.

[66] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlcek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, December 2011.

[67] Sabine Zerbian, "Onset consonant clusters in Tswana: Cw-sequences and affricates," in *Studies in Bantu Linguistics and Languages: Papers in memory of Professor Rugatiri Mekacha. Bayreuth: Bayreuth African Studies*, 2013, pp. 143–165.

[68] N. G. Clements and E. V. Hume, "The internal organization of speech sounds," in *The handbook of phonological theory*, pp. 245–306. Blackwell, 1995.

[69] M. H. Davel and E. Barnard, "NCHLT 2013 phone set," `https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxuY2hsdHNwZWVjaGNvcnB1c3xneDoxN2RmOTJkMmU5NzQwMjE0`, 2013, Accessed 30 Nov. 2015.

[70] D.R. van Niekerk and E. Barnard, "Phonetic alignment for speech synthesis in under-resourced languages," in *Proc. Interspeech*, Brighton, UK, Sept. 2009, pp. 880–883.

[71] Etienne Barnard, Marelie H. Davel, Charl J. Van Heerden, Febe De Wet, and J. Badenhorst, "The NCHLT speech corpus of the South African languages," in *Proc. International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU)*, 2014, pp. 194–200.

[72] M. H. Davel, C. van Heerden, N. Kleynhans, and E. Barnard, "Efficient harvesting of internet audio for resource-scarce ASR," in *Proc. Interspeech*, 2011, pp. 3153–3156.

[73] Paul Boersma and David Weenink, "Praat: doing phonetics by computer (version 5.1.05) [computer program]," 2009, Retrieved May 1, 2009, from http://www.praat.org/.

[74] Christoph Burgmer, "Detecting code-switch events based on textual features," master thesis, Kalsruhe Institute of Technology, Germany, November 2009.

[75] Ngoc Thang Vu, Heike Adel, and Tanja Schultz, "An investigation of code-switching attitude dependent language modeling," in *Proc. Statistical Language and Speech Processing*, Tarragona, Spain, July 2013, pp. 297–308.

[76] Carol Myers-scotton, Ed., *Social motivations for codeswitching: evidence from Africa*, Clarendon press, Oxford, 1993.

[77] J. Y. C. Chan, H. Cao, P. C. Ching, and T. Lee, "Automatic recognition of Cantonese-English code-mixing speech," *Computational Linguistics and Chinese Language Processing*, vol. 14, pp. 281–304, 2009.

[78] Ngoc Thang Vu, Dau-Cheng Lyu, Jochen Weiner, Dominic Telaar, Tim Schlippe, Fabian Blaicher, Eng-Siong Chng, Tanja Schultz, and Haizhou Li, "A first speech recognition system for Mandarin-English code-switch conversational speech," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012.

[79] Dafydd Gibbon, Roger Moore, and Richard Winski, *Handbook of standards and resources for spoken language systems*, Walter de Gruyter, 1997.

[80] L. Loots, M. Davel, E. Barnard, and T. Niesler, "Comparing manually-developed and data-driven rules for P2P learning," in *Proc. Symposium of the Pattern Recognition Association of South Africa (PRASA)*, Stellenbosch, South Africa, Nov. 2009, pp. 35–40.

[81] C. Zarras, K. Pastiadis, G. Papadelis, and G. Papanikolaou, "Cepstrum-based estimation of resonance frequencies (formants) in high-pitch singing signals," in *Proc. DAGA*, Berlin, Germany, 2010, pp. 661–662.

[82] Marelie Davel and Etienne Barnard, "Developing consistent pronunciation models for phonemic variants," in *Interspeech 2006*, 2006, pp. 1760–1764.

[83] T. Modipa and M. H. Davel, "Pronunciation modelling of foreign words for Sepedi ASR," in *Proc. 21st Annual Symposium of the Pattern Recognition Association of South Africa (PRASA)*, 2010, pp. 185–189.

[84] T. I. Modipa, M. H. Davel, and F. De Wet, "Context-dependent modelling of English vowels in Sepedi code-switched speech," in *Proc. 23rd Annual Symposium of the Pattern Recognition Association of South Africa (PRASA)*, 2012, pp. 6–9.

[85] T. Modipa, M. H. Davel, and F. de Wet, "Acoustic modelling of Sepedi affricates for ASR," in *Proc. Annual Research Conference of the South African Institute of Computer Scientist and Information Technologists (SAICSIT 2010)*, 2010, pp. 394–398.

[86] T. Modipa, M. Davel, and F. de Wet, "Acoustic modelling of Sepedi affricates for ASR," in *Proc. Annual Research Conference of the South African Institute of Computer Scientist and Information Technologists (SAICSIT)*, 2010, pp. 394–398.

[87] T. I. Modipa and M. H. Davel, "Predicting vowel substitution in code-switched speech," in *Proc. Annual Symposium of the Pattern Recognition Association of South Africa (PRASA-RobMech)*, 2015, pp. 154–159.