Automatic speech recognition for resource-scarce environments

NT Kleynhans

22950478

Thesis submitted in fulfillment of the requirements for the degree *Philosophiae Doctor* in Computer and Electronics Engineering at the Potchefstroom Campus of the North-West University

Promoter: Prof E Barnard

September 2013

Automatic speech recognition for resource-scarce environments

By

Neil Taylor Kleynhans

Submitted in partial fulfilment of the requirements for the degree

Philosophiae Doctor (Computer and Electronic Engineering)

in the

Faculty of Engineering on the Potchefstroom Campus

at the

North-West University

Advisor: Professor Etienne Barnard

May 2013

Automatic speech recognition for resource-scarce environments

Automatic speech recognition (ASR) technology has matured over the past few decades and has made significant impacts in a variety of fields, from assistive technologies to commercial products. However, ASR system development is a resource intensive activity and requires language resources in the form of text annotated audio recordings and pronunciation dictionaries. Unfortunately, many languages found in the developing world fall into the resource-scarce category and due to this resource scarcity the deployment of ASR systems in the developing world is severely inhibited. In this thesis we present research into developing techniques and tools to (1) harvest audio data, (2) rapidly adapt ASR systems and (3) select "useful" training samples in order to assist with resource-scarce ASR system development.

We demonstrate an automatic audio harvesting approach which efficiently creates a speech recognition corpus by harvesting an easily available audio resource. We show that by starting with bootstrapped acoustic models, trained with language data obtain from a dialect, and then running through a few iterations of an alignment-filter-retrain phase it is possible to create an accurate speech recognition corpus. As a demonstration we create a South African English speech recognition corpus by using our approach and harvesting an internet website which provides audio and approximate transcriptions. The acoustic models developed from harvested data are evaluated on independent corpora and show that the proposed harvesting approach provides a robust means to create ASR resources.

As there are many acoustic model adaptation techniques which can be implemented by an ASR system developer it becomes a costly endeavour to select the best adaptation technique. We investigate the dependence of the adaptation data amount and various adaptation techniques by systematically varying the adaptation data amount and comparing the performance of various adaptation techniques. We establish a guideline which can be used by an ASR developer to chose the best adaptation technique given a size constraint on the adaptation data, for the scenario where adaptation between narrow- and wide-band corpora must be performed. In addition, we investigate the effectiveness of a novel channel normalisation technique and compare the performance with standard normalisation and adaptation techniques.

Lastly, we propose a new data selection framework which can be used to design a speech recognition corpus. We show for limited data sets, independent of language and bandwidth, the most effective strategy for data selection is frequency-matched selection and that the widely-used maximum entropy methods generally produced the least promising results. In our model, the frequency-matched selection method corresponds to a logarithmic relationship between accuracy and corpus size; we also investigated other model relationships, and found that a hyperbolic relationship (as suggested from simple asymptotic arguments in learning theory) may lead to somewhat better performance under certain conditions.

Keywords: automatic speech recognition, data harvesting, acoustic model adaptation, feature normalisation, data selection, corpus design, resource-scarce, language technology resource development.

ACKNOWLEDGEMENTS

A special thank you to my supervisor Etienne Barnard for his guidance and sharing his vast knowledge and wisdom with me throughout my studies.

Thank you to all at the Human Language Technologies (HLT) group (past and present members) and the Meraka Institute for supporting me and granting me the opportunity to pursue my postgraduate studies.

A big thanks to the entire Sandbaai/MuST team.

To the staff at the Centre for High Performance Computing (CHPC) thank you for your assistance and technical support.

Gratitude to my family and friends.

To my parents – thank you for your enduring encouragement, support and understanding throughout my endeavour.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ADPS	Average dynamic Programming Score
AM	Acoustic Model
ASR	Automatic Speech Recognition
AST	African Speech Technology
ARMA	Auto-Regressive Moving Average
BIC	Bayesian Information Criterion
BN	Broadcast News
BP	Telephone Band Pass
CMLLR	Constrained Maximum Likelihood Linear Regression
CMN	Cepstral Mean Normalisation
CMS	Cepstral Mean subtraction
CMVN	Cepstral Mean and Variance Normalisation
CSR	Continuous Speech Recognition
D	number of deletions
DD	day
DP	Dynamic Programming
EM	Expectation-Maximisation
Garbage (%)	percentage data absorbed by garbage model
G2P	Grapheme to Phoneme prediction
GMM	Gaussian Mixture Model
HLT	Human Language Technologies
HMM	Hidden Markov Model
HTK	Hidden Markov Model Toolkit
I	the number of insertions
LM	Language Model
LOGL	Log-likelihood
MAP	Maximum a-Posteriori
MAP_MV	Mean and Variance Maximum a-Posteriori adaptation
MAP_WMV	Weight, Mean and Variance Maximum a-Posteriori adaptation
MFCC	Mel-Frequency Cepstral Coefficients
ML	Maximum Likelihood
MLE	Maximum Likelihood Estimation
MLLR	Maximum Likelihood Linear Regression
MLLR_MV	Mean and Variance Maximum Likelihood Linear Regression
MM	month
MP3	MPEG Audio layer III lossy compression format
MPEG	Moving Picture Experts Group
MVA	Mean and Variance Normalisation with ARMA filtering
N	the total number
OALD	Oxford Advanced Learner's Dictionary (of Current English)
PCA	Principal Component Analysis
PDF	Probability Density Function
PER	Phone Error Rate
Phn Acc	Phone Accuracy Percent
Phn Cor	Phone Correctness Percent
PLP	Perceptual Linear Prediction
PPR	Parallel Phone Recognition
PRLM	Phone Recognition followed by Language Modelling
S	number of substitutions
SAE	South African English
SLID	Spoken Language Identification
SP	short-pause model
SVM	Support Vector Machine
TFF	Transfer-Function Filtering
TTS	Text-to-Speech
WER	Word Error Rate
WSJ	Wall Street Journal
YY	year

CHAPTER ONE

INTRODUCTION

Speech technologies are playing an increasingly important role in the daily lives of many people. For instance, applications such as Google Voice Search [3] performing spoken web searches, telephone services using Automatic Speech Recognition (ASR) to acquire account information [4], access control systems utilising speaker recognition in a host of security checks [5] and multi-lingual spoken dialog systems employing Spoken Language Identification (SLID) [6] have all made significant contributions to the technology landscape. In some cases, these types of systems can perform their related tasks many times more cost efficiently than humans, and for limited domain applications even achieve performance levels exceeding that of humans.

Given the variety of speech-based applications, it is generally the case that an ASR system serves as the foundation whereupon applications are built and specialized. Although ASR technologies have matured over recent years, ASR development is still a resource intensive process. The process often requires large volumes of language resources such as annotated audio corpora and pronunciation dictionaries. This large initial resource requirement places a constraint on the development of ASR systems in the developing world, where most languages are subject to a scarcity of resources and are often termed *resource-scarce*.

As a contribution to rectifying this situation and supporting ASR deployment in the developing world, this thesis reports on research in the areas of (1) data harvesting, (2) rapid ASR system adaptation and (3) training data selection. Progress in these domains will hopefully contribute to the creation of speech-based applications in the developing world.

1.1 PROBLEM STATEMENTS

1.1.1 AUDIO DATA HARVESTING

When developing an ASR-based application, the general practice is to acquire a suitable corpus or to collect a significant amount of application-specific audio samples. However, it may not be possible to purchase a corpus due to cost, language or dialect availability, operating environment, and vocabulary factors. Driving a corpus collection process may not be feasible as often the task is highly resource intensive. Additionally, in a resource-scarce environment these problems are compounded.

An alternative option that can be pursued in certain circumstances is to automatically *harvest* the required language resources. An abundant supply of language resources can often be found on the Internet where, for example, transcribed podcasts can be accessed. Usually, the podcasts are published by government and news agencies, radio broadcasters, universities (lecture recordings) and private individuals. The podcasts vary considerably in quality – the text transcriptions vary in accuracy often containing spelling and grammar errors while the audio recordings regularly contain non-speech artefacts (music, tones, noise) – and require processing to convert the data into a consistent format suitable for ASR system development.

Thus, developing a tool set to automatically process a raw language resource, containing audio and annotations, into a useful ASR corpus can benefit ASR system development tremendously.

1.1.2 ASR SYSTEM ADAPTATION

Task-specific corpora are often difficult to come by and for resource-scarce languages the choices are severely limited. Given access to a language-specific corpus, it would be highly efficient to train acoustic models with the available data and then apply task-specific optimisations. When moving between different operating environments, the optimisations would have to take into consideration the data mismatch which leads to performance degradations. Currently, feature normalisation and model adaptation techniques are employed to reduce the acoustic level mismatches. In general, model-based adaptations perform better than feature normalisation approaches but require transcriptions to estimate the class-specific mismatches and apply the appropriate transforms.

Thus, we will investigate unsupervised techniques for environment normalisation, which can be applied to mismatched data applications. In addition, current ASR model adaptation techniques learn a set of transformations or update acoustic model parameters from provided adaptation data and the performance gains which are attained by the various techniques are dependent on the amount of adaptation data from which the statistics are estimated. Therefore, a comparative investigation will be performed to determine the effectiveness of current model adaptation techniques based on the amount of available adaptation data. The specific scenario that will be investigated is one in which plentiful speech data resources of either telephone bandwidth or high bandwidth are available. We will investigate how feature normalisation and adaptation techniques can increase the ASR system performance gains given increasing amounts of adaptation data from the less-resourced bandwidth.

1.1.3 TRAINING PROMPT SELECTION

A general ASR tenet is that the training of robust acoustic models, to achieve high system accuracies, requires large training corpora. The reasoning is the following: to cover the variability present in speech, many training examples are needed to properly estimate the model parameters. However, for a resource-scarce language such corpora are generally not readily available, which often necessitates the creation of a larger corpus by sourcing data from smaller similar corpora. In addition, it has been shown by [7] that large corpora contain redundant information which implies that a smaller sub-corpus can be created which contains sufficient examples to cover the variability. We therefore intend to answer the following question: if it is feasible to collect a limited amount of data with a focused corpus design, which data should be selected to aid in the collection or design efficiency?

Thus, we will investigate if it is possible to develop a data selection strategy that selects a targeted dataset which maximises ASR system accuracy.

1.2 THESIS OVERVIEW

The aim of the thesis is to investigate various methods which will facilitate the use of automatic speech recognition in resource-scarce environments. The goals can be summarised as follows:

1. to develop an automatic data harvesting procedure that transforms audio and corresponding approximate annotations into an accurate ASR corpus;
2. to investigate the application of an unsupervised channel normalisation technique for data mismatch reduction;
3. to analyse the performance of current ASR adaptation techniques as a function of the amount of data available; and
4. to develop a data selection framework and implementation which optimises ASR system performance.

The thesis is structured in the following manner:

- Chapter 2 discusses relevant literature on speech recognition theory, data harvesting approaches, ASR system adaptation and normalisation and text selection strategies.
- Chapter 3 describes our specific automatic data harvesting approach and demonstrates the effectiveness of the approach by applying it to a South African English ASR corpus creation task.
- Chapter 4 presents our analysis of the data dependence of current feature normalisation and model adaptation techniques. We demonstrate in graphical format the data dependence of various adaptation techniques and provide a guideline on which technique to use, for a given

data amount, that will result in the best performance gain. In addition, the performance of an unsupervised channel adaptation technique is investigated and compared to current state-of-the-art adaptation methods.

- Chapter 5 discusses our theoretical framework for data selection and provides an analysis on the effectiveness of the theory to select appropriate training data examples.
- In Chapter 6 we summarize our results, provide conclusions and highlight the contribution contained in this thesis.

Lastly, appendix (F) contains the nomenclature used in each chapter.

CHAPTER TWO

BACKGROUND

2.1 INTRODUCTION

This chapter describes relevant research in the field of speech technologies, on which the research presented in this thesis builds. The main topics under discussion are:

- Automatic Speech Recognition (section 2.2) - describes current automatic speech recognition theory and approaches.
- Data Harvesting and Automatic Processing (section 2.3) - discusses automatic data harvesting techniques to recover large ASR corpora.
- Normalisation and Adaptation (section 2.4) - presents information concerning system adaptation and feature normalisation for speech recognition.
- Text Selection - (section 2.5) - addresses approaches in the text selection domain, and their relevance to speech recognition.

2.2 AUTOMATIC SPEECH RECOGNITION

Current "state-of-the-art" speech recognition systems are based on a Hidden Markov Model (HMM) architecture [8]; such architectures are implied when referring to a standard automatic speech recognition (ASR) systems. The most widely cited reference on the application of the HMM paradigm in speech recognition is Rabiner [9], which is a clear introduction to the issues that must be addressed in this application. The software tool utilised in this research to develop HMM-based ASR systems was HTK [10]. The toolkit provides a set of stand-alone applications which assist in the creation of HMM acoustic models and performing recognition (decoding) tasks. Additional applications are included to

help transform data into consistent formats and implement all aspects involved in creating a complete system. The broad tasks involved in creating a basic HMM-based ASR system can be summarised:

- Front end processing - speech waveforms are parametrised into feature vectors, which are used for both of the following tasks.
- HMM parameter estimation and refinement - HMM parameters are estimated; this is typically a staged process, with a maximum likelihood (ML) technique used to compute initial models, which are then refined to increase accuracy.
- Search - Find the best possible word sequence given the acoustic models, language model and input acoustic vectors.

A speech recognition task (search) uses a decoder to postulate the most likely set of acoustic events the occur in the audio data. The probabilistic decoding framework uses a combination of the acoustic model likelihoods which are generally weighted by the probability of the event occurring. The probability of an acoustic event is described by a language model. The remainder of the section will present a discussion on the model creation and decoding processes as well as related tasks. Our focus is on the basic approaches followed for each of the steps; in each case, more sophisticated algorithms have appeared in the literature, and some – such as the use of bottleneck features [11] or discriminative training [12] are employed widely. However, those refinements are orthogonal to the issues considered in this thesis, and we will not be devoting much attention to such topics.

2.2.1 FRONT END PROCESSING

The conversion of speech audio into feature vectors is motivated by the need to compactly represent the audio stream (effectively reducing the dimensionality) and provide slowly-varying discrete data samples for Hidden Markov modelling [13]. The conversion process is largely based on speech coding principles and human auditory psychoacoustic processing research [8].

The first stage of processing is based on the speech coding theory. The audio is blocked or broken up at 10 ms intervals and converted to a spectral representation via a Fast Fourier Transform (FFT) using an analysis window of 20 to 30 ms. The standard length is 25 ms. For this limited amount of time the spectral characteristics of the speech are assumed to be stationary. A windowing function such as the Hanning or Hamming window is applied to the analysis window to reduce spectral leakage. A pre-emphasis filter, $S_n^{new} = S_n - \alpha S_{n-1}$ where $\alpha \in [0.95, 0.99]$, is applied to speech samples which increases the amplitude of the higher frequency components [10]. This compensates for the fact that the higher spectral components are attenuated at a rate of -6 dB/oct due to the radiation of speech from the lips [13]. At this stage we have a rather large number of linear spectral values.

Next, to reduce the number of components and achieve an increase in recognition performance, human psychoacoustic processing principles are applied to the linear spectrum [8, 13]. Two leading approaches are Mel-Frequency Cepstral Coefficients (MFCC) [14] and Perceptual Linear Predictive

(PLP) analysis of speech [15]. The PLP methodology follows the human psychoacoustic research more closely than the MFCC approach, but is has been shown that both techniques provide increases in performance (relative to raw spectral values) which are approximately the same [16, 17]. For historical reasons, we employ MFCC feature vectors in our research.

The process of converting the linear spectral samples into MFCC involves calculating Mel-spaced filter bank energies, compressing the energies and applying a discrete cosine transform. Figure 2.1 shows the Mel-spaced filter bank coefficients used to determine the filter bank energies. The overlapped filters simulate, to a limited degree, the masking effect of the human auditory processing mechanics where large-amplitude frequency components mask nearby surrounding lower-magnitude components.

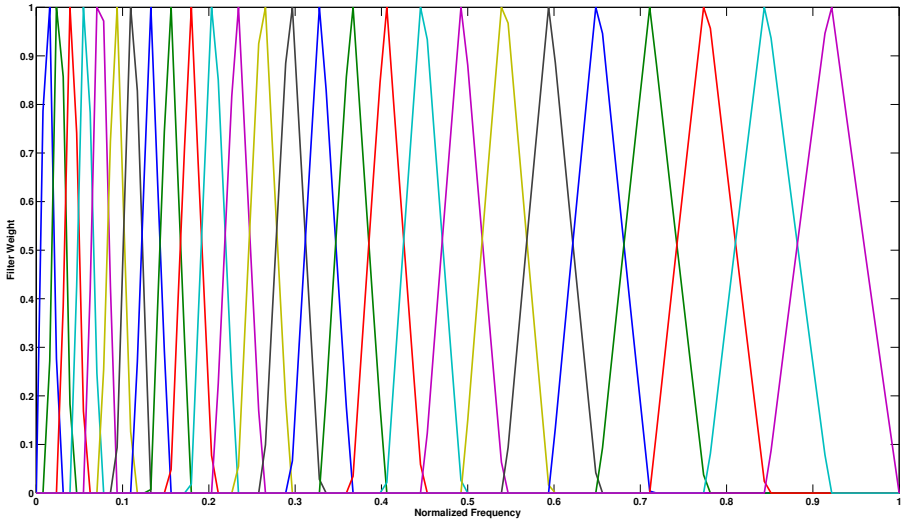

Figure 2.1: *26 Mel-spaced filter bank coefficients.*

The Mel-scale is defined by

$$Mel(f) = 2595 * \log_{10}(1 + \frac{f}{700}), \tag{2.1}$$

which has roughly linear scaling below 1000 Hz and logarithmic scaling above. The scaling is increased as one moves to higher frequencies which simulates the loss of frequency resolution of the ear [8]. The filter bank energies are calculated by performing a spectral integration of the spectral components that contribute to the specific filter bank. After calculating the filter bank energies a compression function – usually the natural logarithm – is applied. The compression simulates the human perceived loudness characteristics [8]. The final step is to calculate the cepstral coefficients, which are obtained by discrete cosine transforming the filter bank energies. The applied formula takes the form of

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j \cos\left(\frac{\pi i}{N}(j - 0.5)\right), \tag{2.2}$$

where c_i is the i^{th} cepstral coefficient, m_j is the m^{th} filter bank energy and N is the total number of filter bank energies. The role of the discrete cosine transform is two-fold: (1) spectral information is compressed into lower coefficients, and, (2) the resulting coefficients are largely decorrelated compared to filter bank energies. The decorrelating effect is necessary to approximate the assumption of statistical independence which simplifies the task of density estimation for the HMMs. The cepstral coefficients are referred to as the static features. Instead of including a frame energy value, the 0'th cepstral coefficient can be used. Usually, cepstral liftering [18] is applied to the cepstral coefficients to smooth the representation and boost the variance of the higher order coefficients.

An improvement in performance can be obtained if dynamic cepstral representations are included [19]. The first-order cepstral time derivatives (dynamic features) are calculated using the regression formula [10, 13],

$$\Delta_t = \frac{\sum_{\tau=1}^{D} \tau(c_{t+\tau} - c_{t-\tau})}{2\sum_{\tau=1}^{D} \tau^2}, \tag{2.3}$$

where c_t is the static cepstral coefficients, τ is the time-shift and D is the number of frames used in the calculation. The second-order time derivatives (acceleration features) are also estimated using the regression formula, applied to the dynamic features. The final feature vector is constructed by appending the static, dynamic and acceleration features.

An important point made by Young [13] is the fact that the entire feature extraction process has been optimised for the HMM pattern-matching task which assumes conditional statistical independence of feature vectors at different times – hence, the speech recognition process can be characterised by a Markov system.

2.2.2 HMM FORMULATION

To begin the formulation of the task of speech recognition we start with a number of speech vector observations,

$$\mathbf{O} = \mathbf{o}_1, \mathbf{o}_2, \mathbf{o}_3, \cdots, \mathbf{o}_T, \tag{2.4}$$

where $\mathbf{o}_t$ is the speech vector observation occurring at time t, and the sequence of observations represents the spoken words in an utterance. The problem of determining the most probable word sequence $\hat{W}$ can be written as,

$$\hat{\mathbf{W}} = \arg\max P(\mathbf{W} \mid \mathbf{O}). \tag{2.5}$$

We can re-formulate the problem and decompose the probability by using Bayes' Rule which gives,

$$\hat{\mathbf{W}} = \arg\max P(\mathbf{W} \mid \mathbf{O}) = \arg\max \frac{P(\mathbf{O} \mid \mathbf{W})P(\mathbf{W})}{P(\mathbf{O})}, \tag{2.6}$$

where $P(\mathbf{W} \mid \mathbf{O})$ is the probability of the word sequence given the observed speech vector sequence and $P(\mathbf{W})$ is the probability of the word sequence occurring. The denominator can be ignored as

$P(\mathbf{O})$, the probability of the sequence will remain constant independent of tested word sequences. The probability of the words, $P(\mathbf{W})$, can be estimated via a language model. The probability of the word sequence given the observed speech vector sequence, $P(\mathbf{W} \mid \mathbf{O})$, is given by a composite model created by concatenating word or sub-words HMMs [10, 13].

For small-vocabulary applications, word-level HMMs can be utilised, but for large-vocabulary applications it is more feasible to use sub-word HMMs. According to Michel *et al.* [20] the English language as of 2000 contain just over 1.4 million unique words which would relate to a rather large number of unique HMMs for word-level modelling. However, the English language contains roughly 45 phonemes which potentially can represent all current and future words. From a practical point of view the approach of modelling the phonemes is more feasible and does not require a reworking of the formulated speech recognition problem whenever the vocabulary changes. An extra step of breaking the words into a phoneme representation is needed as well as a pronunciation dictionary which contains the mappings.

Independently of the unit being modelled, the typical HMM defined in [9, 10, 13] contains three basic elements;

- null or non-emitting entry and exit states,
- internal states which produce output probabilities, and,
- a matrix of transition probabilities governing the state transitions.

Figure 2.2 shows the left-to-right HMM topology used to model the acoustic units. In the figure, states one and five are non-emitting states and facilitate the creation of the concatenated composite model by merging successive entry and exit states, a_{ii} and a_{ij} are transition probabilities, and states two to four are internal states which produce state output probabilities $b_i(\mathbf{o}_t)$.

The state output probabilities are modelled with a continuous density Gaussian mixture model represented by

$$b_j(\mathbf{o}_t) = \sum_{m=1}^{M} w_{mj}\mathcal{N}(\mathbf{o}_t; \boldsymbol{\mu}_{mj}, \boldsymbol{\Sigma}_{mj}), \tag{2.7}$$

where w_{mj} is the mixture weight, $\boldsymbol{\mu}_{mj}$ is the mixture mean and $\boldsymbol{\Sigma}_{mj}$ is the covariance matrix (generally diagonal). The state output probability represents the probability of the specific state generating the observed vector. To calculate the joint probability that a given set of HMMs (M) will generate the observed (O) and state (X) sequences $P(\mathbf{O}, \mathbf{X} \mid \mathbf{M})$, we need to calculate the product of the state transition probabilities and the state output probabilities which is given by,

$$P(\mathbf{O}, \mathbf{X} \mid \mathbf{M}) = a_{12}b_2(\mathbf{o}_1)a_{22}b_2(\mathbf{o}_2)\cdots \tag{2.8}$$

In general, given some observation $\mathbf{O}$ and a state sequence which realises the observations $\mathbf{X}(t) =$

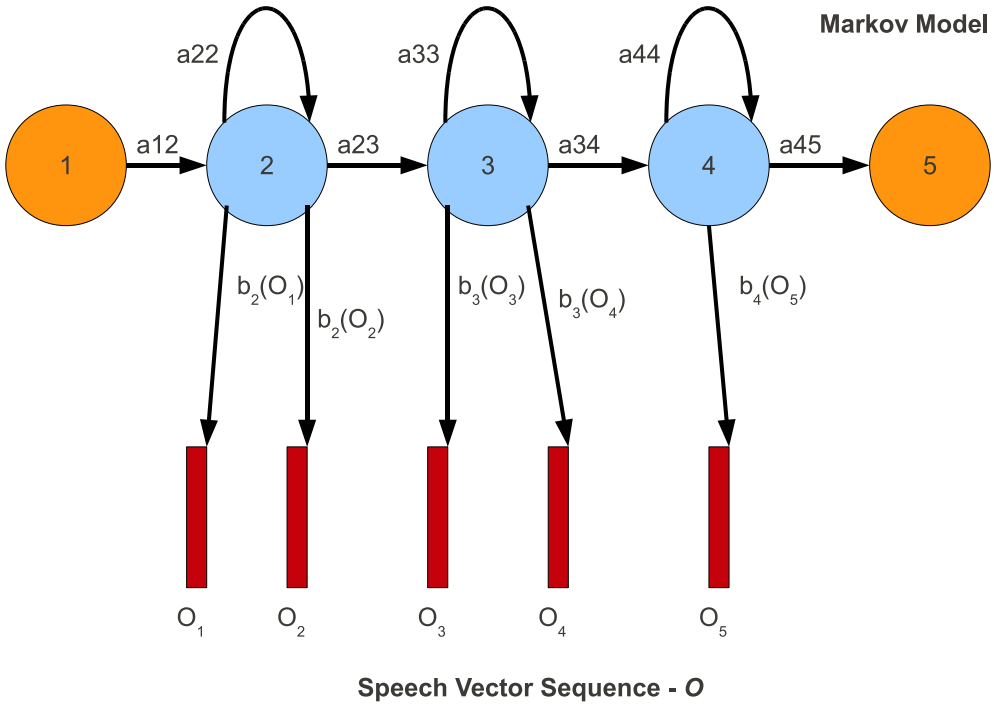

Figure 2.2: *Left-to-right Hidden Markov Model topology.*

$\mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(T)$, the joint probability is stated as,

$$P(\mathbf{O}, \mathbf{X} \mid \mathbf{M}) = a_{\mathbf{x}(0)\mathbf{x}(1)} \prod_{t=1}^{T} b_{\mathbf{x}(t)}(\mathbf{o}_t) a_{\mathbf{x}(t)\mathbf{x}(t+1)}, \tag{2.9}$$

where $\mathbf{x}(0)$ and $\mathbf{x}(T+1)$ are composite model entry and exit null states. For a recognition task, we only have access to the model set ($\mathbf{M}$) and observations ($\mathbf{O}$), so effectively recognition performs a search for the best possible state sequence $\mathbf{X}$. Currently, the most utilised algorithm for determining the most likely state sequence is the *Viterbi* algorithm [21], which approximates $P(\mathbf{O} \mid \mathbf{M})$ by maximizing equation (2.9).

2.2.3 HMM ESTIMATION

The HMM parameters are estimated using the Baum-Welch re-estimation formula which can be interpreted as an Expectation-Maximisation (EM) maximum-likelihood parameter estimation procedure for HMMs [9, 22]. This implies that the model parameters are iteratively estimated and converge to a local maximum of a likelihood function. To start the estimation process, all state means and variances are initialised to the same values, which are derived from the global data statistics as;

$$\boldsymbol{\mu}_j = \frac{1}{T}\sum_{t=1}^{T}\mathbf{o}_t, \tag{2.10}$$

$$\boldsymbol{\Sigma}_j = \frac{1}{T}\sum_{t=1}^{T}(\mathbf{o}_t - \boldsymbol{\mu}_j)(\mathbf{o}_t - \boldsymbol{\mu}_j)^T, \tag{2.11}$$

where $\boldsymbol{\mu}_j$ is the state mean and $\boldsymbol{\Sigma}_j$ is the state covariance. Then, for each Gaussian component of each state, the following update formulas are used to re-estimate the parameters:

$$\boldsymbol{\mu}_{jm} = \frac{\sum_{t=1}^{T} L_{jm}(t)\mathbf{o}_t}{\sum_{t=1}^{T} L_{jm}(t)}, \tag{2.12}$$

$$\boldsymbol{\Sigma}_{jm} = \frac{\sum_{t=1}^{T} L_{jm}(t)(\mathbf{o}_t - \boldsymbol{\mu}_{jm})(\mathbf{o}_t - \boldsymbol{\mu}_{jm})^T}{\sum_{t=1}^{T} L_{jm}(t)}, \tag{2.13}$$

where $L_{jm}(t)$ is the *state component occupation probability* or the probability of occupying the specific state at time t. The occupation probabilities can be calculated recursively using the *Forward-Backward* algorithm [9, 13]. The forward probabilities are given by

$$\alpha_j(t) = \Big[\sum_{i=1}^{N-1} \alpha_i(t-1)a_{ij}\Big] b_j(\mathbf{o}_t), \tag{2.14}$$

while the backward probabilities are given by

$$\beta_i(t) = \sum_{j=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1})\beta_j(t+1). \tag{2.15}$$

After calculating the forward and backward probabilities the occupation probabilities are given by

$$L_j(t) = \frac{\alpha_j(t)\beta_j(t)}{\alpha_N(T)}. \tag{2.16}$$

The HTK toolkit employs an *embedded* training procedure to estimate the HMM parameters. During training, each observation sequence has a corresponding orthographic transcription. Using this information, a composite HMM model can be created by concatenating the models in sequence that occur in the transcription. Then for each file the occupation probabilities for each state that is found in the composite HMM are calculated and added to an accumulator. Once all the files have been processed, the occupation counts are normalised and the model parameters updated.

2.2.4 PARAMETER TYING

Context-independent models, such as phoneme-based HMMs, are an effective means to bypass the impractical approach of creating word-based HMMs. However, to obtain substantially higher accu-

racy levels context-dependent models have to be employed [23]. The benefit they offer is the ability to model the spectral variations caused by the co-articulation induced by the phonetic context in which each phone is spoken. Context-dependent models such as biphones, triphones or cross-word triphones provide a good level of sound class discrimination. If a context-independent phone sequence is defined by *sil sh iy hh ae*, the equivalent triphone representation would be *sil-sh+iy sh-iy+hh iy-hh+ae*. However, porting the phonemes to triphones leads to an explosion in the number of models. For example, if a language contains 45 phonemes then the triphone count would be roughly $45^3 = 91125$. If we couple the number of triphones with the number of states and mixtures per Gaussian, the models contain a rather large model parameter count. This rise in model complexity incurs a data shortage penalty where some models will have no or insufficient data to train on, for realistic distributions of acoustic classes and corpus sizes.

To overcome the data insufficiency problem Young *et al.* [24] initially proposed a data-clustering approach, which finds similar states and collapses each cluster of such states into a single state. This effectively pools similar data samples and creates a larger training dataset per model. A shortcoming of the approach is that training examples must exist for state-tying to occur. Thus *unseen* triphones are excluded from the tying process and will be excluded from the final model set. To accommodate the inclusion of unseen triphones, Young [23] introduced a phonetic decision-tree-based clustering scheme. This approach shows performance levels comparable to that of the data driven technique.

The phonetic decision tree based state-tying requires as input a list of yes/no questions which make inquiry about the immediate left or right context of a phoneme. A typical question would ask: "Is the phone context to the right a fricative?". Generally, broad phonetic classes such as nasals, vowels, glides, etc. are questioned. To start the tying process, all states are pooled into a root node and the log-likelihood of the data calculated. At this stage all states are regarded as being tied. The node is then split by finding the question that results in the largest gain in the log-likelihood. The splitting process is repeated until the log-likelihood reaches a predefined threshold. To ensure that each node contains a sufficient data amount an occupation count threshold is also defined which prevents splitting of nodes with low data counts.

2.2.5 LANGUAGE MODEL

In equation (2.6), the probability of a word or word sequence $P(W)$ is generally estimated using a statistical model such as an *N-gram* language model [13]. The N-gram model provides a method of estimating the probability of a word w_N given the preceding $N-1$ words $w_1, \cdots, w_{N-1}$. The advantage of using N-grams are: the probabilities are estimated from text data, encode many language attributes such as semantics and pragmatics and do not require linguistic knowledge as input [25].

A disadvantage of N-grams is the volume of data which is required to robustly estimate the probabilities. For instance, if an ASR system has a test vocabulary of V words then one would have to estimate probabilities of V^2 bigrams (2-gram) and V^3 trigrams (3-gram). A small or medium sized text corpus would in all likelihood have a limited number of training examples for many bigrams and

trigrams; many acceptable trigrams may not occur in the corpus. Two methods of dealing with this data sparsity are discounting and backing-off [25].

For the discounting approach, the counts of the most frequently occurring N-grams are reduced and redistributed amongst the least occurring N-grams. The Back-off method estimates a probability for an N-gram that has limited training examples, by scaling the relevant N-1-gram probability. The scaling factor ensures that the N-grams are normalised correctly.

2.2.6 SEARCH

When an unknown utterance is presented to a fully-trained ASR system, the system tries to recognise the most likely sequence of words by maximising equation (2.6). To do this, the system uses a decoder to perform a search through all word sequence possibilities. The language model is also used by the decoder and constrains the search space by weighting more likely word combinations. The HTK system makes use of the efficient Viterbi algorithm to perform the search [10, 21, 25].

The basis for the Viterbi algorithm is set out in the recursive equation (2.17)

$$\phi_j(t) = \max_i\{\phi_i(t-1) + log(a_{ij})\} + log(b_j(\mathbf{o}_t)), \tag{2.17}$$

where $\phi_j(t)$ is the partial log-likelihood of state j at time t, $\phi_i(t-1)$ is the partial log-likelihood of state j at time $t-1$, a_{ij} is the transition probability from state i to state j and $b_j(\mathbf{o}_t)$ is the likelihood of observation o_t given state b_j. Given a static network, with observations on a horizontal axis and states on a vertical axis, the Viterbi algorithm tries to find the best path through the network using equation (2.17) to update the state log-likelihoods.

The use of static networks in large vocabulary systems is not computationally feasible, as the networks experience dramatic increases in size with increasing vocabulary. To create a practical search algorithm, the HTK decoder makes use of *pruning* and *tree-structure networks* [21]. To limit the number of network paths which have to be searched, a path pruning strategy is employed, using a search beam to discard the least likely paths. Paths whose likelihoods fall below a certain threshold, measured relative to the most likely path, are removed. To reduce the computational time and space requirements HTK makes use of tree-structured networks which are dynamically grown and pruned.

Lastly, the HTK decoder implements a *token passing algorithm* [10, 21, 25] which helps recast the search problem. The path leading from the start node to any point in the tree can be evaluated by summing the log probabilities of the state transitions, state outputs and language model. The path thus can be represented by a movable token which contains the path score and path sequence history. The algorithm places a single token into the root node. As the input vectors are processed, the tokens are copied to the connecting nodes and all information updated. If multiple tokens are assigned to a node, the token with the best scored is retained. After processing all the vectors, the surviving token with the best score contains the most likely path through the network and thus the most probable word sequence.

The packaged HTK Viterbi decoder is a single pass decoder which can utilise variable length language models and context-dependent phone HMMs.

2.3 DATA HARVESTING AND AUTOMATIC PROCESSING

It is generally known that speech recognition acoustic models (AM) require large amounts of transcribed audio data for robust training [26–29]. Collecting and accurately transcribing the audio data is an expensive process which is time-consuming and incurs high costs [26–28]. Pre-existing corpora may be purchased from vendors, such as *Linguistic Data Consortium* (http://www.ldc.upenn.edu/), but the corpora costs are generally high, as the aforementioned collection and transcribing costs must be covered, and the available corpora may not suit the application type [27] – noise conditions, microphone-type and word coverage.

Most developing-world languages do not have access to readily available corpora [29] which limits the development of ASR applications [29, 30]. In addition to the costs of collecting and transcribing audio data, developing-world contexts generally lack the necessary infrastructure to facilitate a collection process [29] – lack of computer networks and first language speakers who possess the relevant skills and experience.

To overcome the hurdles in developing large-data ASR corpora, automatic methods can be used to harvest data from alternate data sources. Across the internet there are many sources of transcribed audio data, in many languages, which can potentially be used for the development of acoustic models. These sources include audio-visual lecture recordings and broadcast news. Utilising an automatic method to efficiently collect and process the audio and transcriptions into a suitable ASR corpus can greatly reduce ASR development costs and hopefully increase the number of deployments of ASR-enabled applications in the developing-world.

To ensure proper AM training, transcriptions in an ASR corpus must accurately describe the spoken audio. Transcriptions which accompany audio not designed for ASR purposes tend to contain a rather loose representation of the spoken text. These imperfect transcriptions are referred to as "approximate transcriptions" [31] and are usually created quickly and cheaply. In the process of producing the transcriptions rapidly a number of artefacts are introduced [26]. Generally, the following information is not found in the approximate transcriptions:

- Speech disfluencies, hesitations, repetitions and grammatical errors indications.
- Non-speech event markings such as music, coughing, respiration, throat clearing.
- Speaker turn identifications.
- Acoustic characteristics information, for instance, noise levels and background music or speech.

In addition, due to the speed of transcribing, many errors are present in the transcriptions such as word insertions, deletions, substitutions, order switching and non-transcribed portions. Given the

errors which are commonly found in these transcriptions, one might be dissuaded from using these audio sources. However, the abundance of such data (which is large compared to ASR standards) makes it an attractive option to process the data and select accurately-transcribed portions, to build a usable ASR corpus. To select reliable data portions, the transcriptions need to be time-aligned to the speech and to this end several automatic methods have been proposed.

Hazen [31] developed an automatic procedure to correct approximate manual transcriptions sourced from lecture recordings provided by the MIT OpenCourseWare initiative. As pointed out by Hazen [31], it is far more efficient to process approximate transcriptions which are generally produced at speeds 3 to 5 times real-time than to invest in highly accurate transcriptions which are produced in the order of 50 times real-time. The proposed automatic approach performs the following tasks:

- The audio data is passed through a speech-recognition system. The ASR system uses a hybrid trigram language model which is a combination of a generic model and the manual transcription-specific language models. After decoding the data, two word anchor points are found in text which are determined by correlations between the manual and recognised transcriptions.
- Using the anchor points, forced alignment is performed on text occurring between the anchor points. The forced alignment is termed "pseudo forced alignment", since the process is allowed to insert and substitute phoneme fillers during the alignment.
- An editing phase is run where audio portions that have been marked with insertion, substitution and deletion are passed through a speech recogniser to identify the most probable words.

Using the automatic alignment procedure, outlined above, an ASR system trained on the lecture data managed to reduce the error rate from 24.3 % to 8.8 %.

Moreno *et al.* [32] proposed an iterative procedure which converts the problem of aligning large audio utterances into a recursive speech recognition task. Initially, a language model is trained on the entire transcription and then audio is decoded using the language model and a large-vocabulary ASR system. Next, a dynamic programming approach is followed to globally align the decoded text and the transcription. From the alignment, anchor points are found which confidently show agreement between the decoded and manual transcriptions. The utterance is then split into aligned and unaligned portions based on anchor points. The described process is then repeated for each unaligned segment – at each stage the language model is retrained on the text that occurs within the segment. The results showed that after running the automatic alignment procedure 98.5 % of the words were within 0.5 seconds of their true alignments.

Lamel *et al.* [26] approached the problem of automatically training acoustic models by defining a lightly-supervised technique. The first step is to normalise all text sources into a consistent format which is then used to train an n-gram language model. The text sources include the approximate

transcriptions plus any additional text sources which have similar characteristics. Next, a data segmentation task is performed which partitions the raw audio data into homogeneous segments. Each audio segment has the same audio attributes such as speaker, gender and bandwidth. To facilitate automatic audio transcribing of the raw training audio an acoustic model set is needed. These models can be sourced from pre-existing acoustic models or trained using data from the audio source. If new models are trained, only an hour or less of manually annotated data is needed. The acoustic models in conjunction with the topic-dependent language models are used to generate automatic transcriptions for the raw training data. After generating the transcriptions a data filtering phase can be run which filters the data by checking the alignments between the approximate and automatically generated transcriptions. Data segments are removed where there is a disagreement between the transcription pairs. The remaining data which survives the filtering process is used to develop new acoustic models. The sub-process from generating automatic transcriptions to training new acoustic models is iterated over a few times to improve the transcription quality and produce better acoustic models.

As mentioned previously, the audio collected from uncontrolled environments may contain speech disfluencies (repeats, repetitions and hesitations) and other non-speech events (noise, lip smacks, breathing). Although not used in the alignment of approximate transcriptions ten Bosch and Boves [33] showed that *garbage models* could be used to absorb the speech anomalies and improve ASR accuracy.

2.4 NORMALISATION AND ADAPTATION

Speech-recognition systems lose accuracy when there is a mismatch between the acoustic models and testing data. To reduce the mismatch, feature normalisation and model adaptation techniques are employed.

2.4.1 FEATURE NORMALISATION

Some of the earliest work on environmental adaptation was performed by Moreno and Stern [34], who showed that matching the bandwidths of the mismatched data can improve the results considerably for adaptation between the TIMIT and NTIMIT corpora (which we describe in more detail below). This can be achieved by bandpass filtering the audio signal or if a filterbank-style spectral analysis is performed on the audio, the filterbank can be band limited to the appropriate spectral region.

If the feature vector is cepstral-based, the simplest normalisation technique is Cepstral Mean Normalisation (CMN) or Cepstral Mean Subtraction (CMS) where an average cepstral vector is subtracted from the individual cepstral vectors. The average is usually calculated over an utterance or segment but speaker-based or corpus-wide normalisation are also possible. This technique removes static (time-invariant) components in the cepstra, which (over a sufficiently long time window) are primarily caused by the static characteristics of the speaker and the recording environment. A logical extension is cepstral variance normalisation, which is applied after mean normalisation. Cepstral

mean variance normalisation (CMVN) transforms the mean to zero and the variance to one for a set of cepstral vectors. Chen and Bilmes [35] added an auto-regression moving average (ARMA) filtering to the CMVN process, referring to the combination as MVA. The ARMA filtering limits the modulation spectrum of cepstral vectors to speech-like characteristics. MVA showed gains in system accuracy for highly-mismatch conditions [35].

Similar to cepstral-based filtering, it is possible to remove non-speech artefacts by log spectral subtraction as shown by Gelbart and Morgan [36]. In their investigation, they tried to improve the performance of ASR systems which were used in reverberant environments by subtracting a long-term spectral average from a spectral analysis frame. The long-term spectral average was estimated on ten windows before and after the current spectral analysis frame. The results showed that their log spectral subtraction performed better when the average log spectrum was estimated on speech segments about 12 seconds long, with analysis windows of one to two seconds in duration.

The cepstral vectors can also be processed in a non-linear manner as shown by Segura *et al.* [37] who experimented with cepstral-domain Histogram Equalisation (HEQ). This technique transforms the cepstral vectors into a domain which is less affected by the channel distortion. The results obtained in a multi-conditioned scenario all improved on the baseline, but were comparable to CMVN results.

2.4.2 MODEL ADAPTATION

Maximum Likelihood Linear Regression (MLLR), proposed by Leggetter and Woodland [38], provides a means to update acoustic models without having to retrain the models. The initial implementation performed mean-only adaptation and was applied to speaker adaptation. The technique estimates a set of linear-regression matrix transforms which are applied to the mean vectors of the acoustic models. To determine the transform matrix, $\mathbf{M}$, an adapted vector $\hat{\boldsymbol{\mu}}$ is defined as;

$$\hat{\boldsymbol{\mu}} = \mathbf{M}\xi, \tag{2.18}$$

where $\mathbf{M}$ is the $n \times (n+1)$ transformation matrix and ξ is the $n+1$ extended mean vector $\xi = [b, \mu_1, ..., \mu_n]$. b in the first position of the extended mean vector is set to one if an offset is to be estimated or zero to ignore the offset. The parameters of the transformation matrix $\mathbf{M}$ are estimated by maximizing the auxiliary function used in the Expectation-Maximisation (EM) algorithm derivation. The mean values in the auxiliary function are replaced by the adapted mean given in equation (2.18) and the function is optimised in terms of transformation matrix parameters.

Gales and Woodland [39] extended the framework to include variance adaptation. The covariance matrices in the acoustic models are transformed into adapted covariance matrices given by

$$\hat{\Sigma} = \mathbf{B}^T\mathbf{H}\mathbf{B}, \tag{2.19}$$

where $\mathbf{H}$ is the transformation matrix, $\mathbf{B}$ is the inverse of the Cholesky factors of the inverse covariance matrix, $\boldsymbol{\Sigma}^{-1}$ ($\boldsymbol{\Sigma}^{-1} = \mathbf{C}\mathbf{C}^T$ and $\mathbf{B} = \mathbf{C}^{-1}$). Similar to the mean transform parameters'

estimation, the auxiliary function is optimised in terms of the variance transform $\mathbf{H}$ to determine the parameters.

The MLLR adaptation technique utilises a regression class tree to ensure proper parameter estimation. The regression class tree defines a set of classes which contain similar acoustic models. The build process employs a centroid-splitting algorithm [10] which creates the tree by performing the following steps:

- Select a terminal node. The root node contains all acoustic models initially.
- Calculate the parent node's overall mean and variance based on the acoustic models associated with the node.
- Split the node in two and assign perturbed means to the child nodes.
- Iterate over the acoustic model components, clustered to the parent node, and assign them to one of the child nodes based on the Euclidean distance.
- Update the child's means and variances after all the components have been assigned.
- Repeat the reassignment process using the new means and variances until assignments have stabilised.
- Continue the entire process until the defined number of nodes have been reached.

Gauvain and Lee [40] proposed the use of a Maximum a posteriori (MAP) measure to perform parameter smoothing and model adaptation. The MAP technique differs from Maximum Likelihood Estimation (MLE) by including an informative prior to aid in HMM parameters adaptation. The results for speaker adaptation showed that MAP successfully adapted speaker-independent models with relatively small amounts of adaptation data compared to the MLE techniques. However, once the adaptation data grew large enough MAP and MLE delivered the same performance. In this adaptation scenario, the speaker-independent models served as the informative priors.

Gales [41] introduced semi-tied transforms with the goal of better decorrelating the feature vectors and improving ASR accuracies. The technique is model-based and uses MLE to determine the decorrelating matrix transforms. A regression class tree is used to create data groups and the transformation matrices are estimated for these groups. The semi-tied method replaces a model's diagonal covariance matrix by a semi-tied covariance matrix given by

$$\hat{\mathbf{\Sigma}} = \mathbf{H}\mathbf{\Sigma}_{diag}\mathbf{H}^T, \tag{2.20}$$

where $\mathbf{H}$ is the semi-tied transform matrix and $\mathbf{\Sigma}_{diag}$ is the model diagonal covariance matrix. To determine the transformation parameters, an auxiliary function (based on the expectation-maximisation approach) is optimised in terms of the transforming parameters which leads to a set of iterative equations. Numerical methods are employed to solve for the parameters.

Lastly, acoustic model adaptation techniques can be used in either a supervised or unsupervised mode. In a supervised setup, the adaptation technique makes use of hand normalised transcriptions while in a unsupervised mode the adaptation approach uses automatically generated transcriptions usually created by using a speech recognition system.

2.4.3 MODEL ADAPTATION AND ADAPTATION DATA AMOUNT

Leggetter and Woodland [38] showed that for a speaker adaptation task and using a single MLLR global transformation matrix, the improvements in word error rate only began after approximately 11 s of speech (3 adaptation utterances) were used to estimate the transform and saturated at about 15 adaptation utterances. They suggest increasing the number of transformation classes, to achieve better performance gains, but the number of classes must be chosen with care as sufficient data is needed to produce robust estimates.

Gauvain and Lee [40] managed to achieved significant improvements in word error rates using MAP adapted models compared to MLE trained models. In their experiments they used three model types: (1) speaker-dependent (SD) models trained on a specific speaker's data only using MLE training, (2) speaker-adapted models (SA-1) which were created by MAP adapting a speaker-independent model – a model trained on data sourced from many speakers –, and, (3) a second set of speaker-adapted models (SA-2), created by MAP adapting gender-dependent models – models trained on female or male data only. At two minutes of training / adaptation data, the word error rates (WER) for the various models were SD 31.5 %, SA-1 8.7 % and SA-2 7.5 % and at five minutes the WER were SD 12.1 %, SA-1 6.9 % and SA-2 6.0 %. At thirty minutes of data the models produce similar results, around 3.5 % WER. The two and five minutes results show that the MAP adaptation effectively produces robust speaker-adapted models on sparse data.

Goronzy and Kompe [42] evaluated a speaker adaptation approach using a modified version of MLLR, normal MAP and a combination of the two to improve a system's sentence error rate. The modified MLLR technique uses a MAP approach to update the means by assigning a new mean which is calculated by weighting the previous and newly estimated means – the weights are dynamically estimated based on the amount of adaptation data seen. The speech database contained German commands which are used to control consumer devices – thus, a small-vocabulary application. Comparing the system accuracy improvements obtained by utilising MLLR, MAP and MLLR plus MAP and incrementally adding more adaptation data, the MAP plus MLLR techniques only started providing the best accuracies at about 100 utterances, while MAP surpassed MLLR at about 125 utterances.

Wang *et. al.* [43] investigated the improvement of speech recognition systems utilised in processing non-native speech data. The methods which they investigated were bilingual models, speaker adaptation, acoustic model interpolation and polyphone decision tree specialisation. For the experiment which made use of the speaker adaptation techniques, MLLR and MAP were used to adapt the acoustic models and reduce the mismatch between the German acoustic models and German speech data generated by non-native speakers. The MLLR transforms and MAP adaptations were estimated

on increasing amounts of adaptation data starting from seven minutes and concluding at 52 minutes. The experiment further investigated the effect the adaptation data composition had on adaptation performance. This was done by using data from a set or varying number of speakers. The results show that the MLLR transform estimated on the fixed-speaker dataset perform better compared to the MLLR transform estimated on the varied-speaker dataset for data amounts less than 42 minutes; after that, the performances were similar. The MAP adapted acoustic models, using the varied-speaker dataset, initially performed worse compared to both MLLR approaches but produced better gains after the 17 minute adaptation data mark. The fixed-speaker MAP adapted acoustic models consistently performed better than the MLLR transforms and provided a gain over the speaker-varied MAP adapted acoustic models until 42 minutes of adaptation data where after the performances were comparable.

Caon *et. al.* [44] made use of a 20-fold cross-validation approach to investigate supervised acoustic model adaptation using maximum-likelihood (ML) training, MAP adaptation and MLLR transforms. The seed acoustic models were trained on 40 hours of French broadcast news audio data. The adaptation corpora contained speech collected from non-native speakers and comprised three data types, namely, read, interview and distressed speech. The adaptation data was split into four separate corpora: single speaker read speech (SDR), single speaker female interview speech (SDIf), single speaker male interview speech (SDIm) and a 21 speaker distressed expression speech (FDE). The SDR corpus contained 162 utterances and 1572 words, SDIf and SDIm contained 103 utterances and 512 words each, and, FDE contained 2646 utterances and 10080 words. The phone error rate (PER) results showed that for the SDR and FDE corpora the order of improvements for the various adaptation approaches were (in order least to most): MLLR, MAP, ML *not* updating transition probabilities and ML updating transition probabilities. For the SDIf and SDIm corpora using ML training produced results which were worse compared to non-adapted models. The MAP and MLLR adaptation performed better compared to the non-adapted models but MLLR produced a slightly lower PER for the female corpus (SDIf) while for the male corpus (SDIm) MAP obtained a lower PER compared to MLLR.

Wallace *et. al.* [45] investigated various supervised and unsupervised adaptation techniques to improve automatic transcription generation using speech recognition to extract transcriptions from telephony-quality audio data. Their experiments focused on speaker-dependent adaptation. The supervised adaptation experiments used hand normalised transcriptions while the unsupervised adaptation approach used transcriptions generated by the baseline non-adapted acoustic models. The supervised techniques showed continued word error rate reductions for the following order of adaptation techniques: global MLLR, regression tree MLLR, MAP, cascaded global plus regression tree MLLR, and, cascaded global plus regression tree MLLR plus MAP. For the unsupervised adaptation experiments the order is somewhat different: global MLLR, MAP, regression tree MLLR, cascaded global plus regression tree MLLR, and, cascaded global plus regression tree MLLR plus MAP. The unsupervised experiments highlights the sensitivity of MAP to inaccurate transcriptions and robustness

of the MLLR approach. The adaptation amount experiments, which used 10, 30 and 60 minutes of adaptation data, showed for both supervised and unsupervised adaptation, the global MLLR approach could not provide further performance gains after 30 minutes of data. The cascaded global plus regression tree MLLR and cascaded global plus regression tree MLLR plus MAP showed continued improvements as more adaptation was data added.

Bocchieri *et. al.* [46] proposed the use of several approaches to adapt out-of-domain acoustic models to an in-domain application. Two corpora were created by recording data from two spoken-language customer assistance applications referred to as "OpServ" (out-of-domain) and "CustCare" (in-domain). Utterances were selected from the in-domain corpus to construct two adaptation datasets, one containing approximately 2 hours of speech data and the other containing about 9 hours. They showed that MAP adaptation of the out-of-domain acoustic models using both adaptation sets produced better accuracies compared to that of MLLR adaptation. Retraining models using 2 hours of in-domain data and all the out-of-domain data produced better accuracies than just adapting the out-of-domain models using the 2 hours of data. Similarly, retraining models using only the 9 hours of in-domain speech provided the best accuracies compared to all adaptation and training in- and out-of-domain combinations. Interestingly, a slight increase in performance was achieved by adapting the retrained in-domain acoustic models using out-of-domain data – the acoustic models' state output densities are smoothed thus producing better results. To further improve the in-domain acoustic model accuracies, out-of-domain mixture components, obtained from out-of-domain acoustic models, were merged with in-domain acoustic models. This final hybrid model produced the best result.

Lastly, Bocchieri *et al.* [46] described a strategy to port existing acoustic models to new applications. The approach for increasing amounts of in-domain data *t* is (adapted from [46]):

- If $0 < t < t_{mllr}$ use out-of-domain acoustic models,
- If $t_{mllr} < t < t_{map}$ use MLLR adapted out-of-domain acoustic models.
- If $t_{map} < t < t_{cntx}$ use MAP adapted out-of-domain acoustic models.
- If $t_{cntx} < t < t_{new}$ retrain acoustic models on in-domain and out-of-domain data (building context-trees on in-domain data).
- If $t_{new} < t$ retrain acoustic models on in-domain data.

The authors speculated that the transition values t_x are application-specific and depend on factors such as triphonic contents of the in- and out-domain data. For the "OpServ" and "CustCare" corpora Bocchieri *et al.* [46] found that t_{map} was below 1500 utterances (2 hours), t_{cntx} was around the 1500 utterance count and t_{new} was between t_{cntx} and 6000 utterances (9 hours). No value was given for t_{mllr} as MLLR and MAP were only compared on 1500 and 6000 utterances – MAP performed better on both datasets. Using 6000 in-domain utterances the, MAP adaptation increased the accuracy of the out-of-domain acoustic models from 60.1 % to 64.9 % while retraining the models on the same

data improved the accuracy from 60.1 % to 66.1 %. Retraining on the 6000 in-domain utterances and then applying MAP adaptation using the out-of-domain data resulted in an overall accuracy of 66.2 %. Finally, training acoustic models on the 6000 in-domain utterances and adding mixture components obtained from acoustic models trained on out-of-domain data, provided the best performance at 67.3 %.

2.5 TEXT SELECTION

It is generally assumed that large amounts of audio data are needed to train truly robust acoustic models. But as more data is added to large training sets, the observed gains in accuracy tend to become smaller, which implies that the data contains redundant information [7, 47, 48]. Moore [48] showed that, for HMM-based ASR systems, there is a linear relationship between the word error rate (WER) and the logarithm of the training data amount. In addition, different experimental configurations influence the starting WER but not the slope of WER decreases across training data amount. Based on the results, it would seem to show that the vocabulary and language model used during the recognition phase play an important role in determining the starting WER. Thus, the question on enough data really depends at which WER level you would want to operate the ASR system and the constraints on the resource investment. Using linear extrapolation Moore [48] showed hypothetically, that for a particular ASR system, training data in the range of 3,000,0000 – 10,000,000 hours would produce an effective WER of 0 %. Similarly, an ASR using another configuration would require 600,000 – 800,000 hours of speech data to achieve a 0% WER. Collecting such data amounts does seem impractical and due to the logarithmic relationship between training data amount and WER, simply adding more and more data does not seem to be a plausible solution when trying to achieve lower WERs.

Wu *et. al.* [7] and Nagorski *et. al.* [47] have shown, however, that it is possible to select a smaller sub-corpus from a large corpus and train ASR models which provide performance comparable to that of models train on all the data. Thus, with proper data selection, a sub-corpus can be created which contains sufficient data variation to produce robust ASR models.

Wu *et. al.* [7] proposed a maximum entropy principle data selection algorithm that would select a sample of the data from a larger corpus which was motivated to reduce training time but still provide robust ASR acoustic models. Their selection criteria were based on sampling the data to create a corpus which approximately contained uniform counts for either words, words plus characters or words plus phones. The greedy selection algorithm was employed to select the units and used a maximum entropy principle to guide the selection. Wu *et. al.* [7] argued that this type of data selection produced an optimal acoustic model training approach. Their results showed for 150 hours of data the random selection obtained 25 % error rate while the uniform word plus phone selection achieved an error rate of 24.4 %. Interestingly, training on 840 hours only achieved an error rate of 24.3 %. They also showed that selecting by word distributions at 30, 50, and 100 hour training data intervals, the maximum entropy selection performed on average better than a random selection.

Nagorski *et. al.* [47] proposed an optimal data selection algorithm to select a small data subset from a larger corpus that would effectively represent the entire data statistics. The motivation in selecting a sub-corpus was to reduce training times. Their selection algorithm performed the following tasks:

- Create a 810-dimensional utterance supervector (18 phones $\times$ length 15 static MFCCs $\times$ 3 HMM states). The supervector is created by firstly, calculating, on a per utterance basis, the average of the static cepstral coefficients per HMM state – state alignments are determined by forced alignment. Then, for each of the eighteen context-independent phone HMMs (three states per HMM), the state's average cepstral vectors are appended. Finally, the appended HMM cepstral vectors are combined to form the supervector.
- Apply Principal Component Analysis (PCA) to reduce the 810-dimensional supervector to 48-dimensional vector.
- Use K-means clustering to cluster the reduced vectors into a phone-specific number of clusters.
- Select a pre-defined number of supervectors which are close to the cluster centroids and add the associated utterances to the training corpus. Adjusting the number of selected supervectors determines the size of the training corpus.

The results show that, for limited-data applications, their optimal selection algorithm outperforms random selection consistently. However, as more and more data is added the randomly selected utterances produced a performance which approaches the optimal selection performance.

An important concept for corpus design is *coverage*, highlighted by van Santen and Buchsmans, [49], which plays a large role in determining the corpus suitability for specific ASR applications. An ASR system will perform poorly if the training and testing unit distributions are vastly different – for instance training a digit recognition system to perform a proper name recognition task. The impact of unit distribution dissimilarity can be overcome if one limits the training and evaluation sets to be extracted from the same data source. It is to be expected that the training and evaluation distributions should tend to be similar given large data amounts. But if this is not possible a selection process is needed to reduce the unit distribution differences. As highlighted in [49], two possible selection criteria are to (1) cover all units or (2) base the coverage on unit frequencies. Each has associated weaknesses: for full unit coverage, it becomes difficult to limit the total size of the corpus while trying to include all rare units, and for selection based on unit frequencies, unit frequencies for sub-domain texts are typically quite varied.

Assuming one has a target unit distribution, Gouvea and Davel [50] showed the importance of matching the training and evaluation distributions and that the KL-divergence metric is an effective data selection tool. In their ASR-specific experiments, they designed a few target n-gram distributions (representing target domains) by randomly selecting utterances, from an evaluation dataset, to generate a set of target n-gram distributions and then created the target distributions by selecting 500

utterances which had similar n-gram distributions. Then, to train ASR systems, a 1000 utterances were selected from a training dataset using one of three selection criteria: (1) selecting to match the target distribution, (2) uniform n-gram selection, and, (3) random selection. In addition, KL-divergence selection was performed on unigrams and trigrams. Their results showed that the targeted distribution ASR systems performed better compared to random and uniform selection and interestingly, the random selection performed better than the uniform selection. Lastly, the trigram selection proved a better choice compared to unigram selection.

2.6 CONCLUSION

In this chapter we discussed prior research pertaining to the body of work captured in this thesis. Importantly, the majority of experiments performed in this research utilised automatic speech recognition systems and the particular system design choices used will be described in more detail in the subsequent chapters. Our particular data harvesting approach, which employed several of the ideas described above as well our own innovations, will be discussed further in the Chapter 3. Chapter 4 provides a comparison of current feature normalisation and model adaptation methods and highlights the data dependence of each technique, again extending and reflecting on similar findings in the literature. Lastly, speech recognition corpus creation and optimal data selection for improved ASR performance will be presented in Chapter 5; we introduce a novel approach in which the selection criterion is based on unit accuracies given the number of training examples, rather than an approximation of the full-data statistics.

CHAPTER THREE

DATA HARVESTING FOR RESOURCE-SCARCE ENVIRONMENTS

3.1 INTRODUCTION

As highlighted in section (2.3), using automatic methods to harvest audio and process their approximate transcriptions is an efficient and relatively cheap method that can be utilised to create new ASR corpora. This specific task has received significant attention for well-resourced languages and has been shown to be effective in processing broadcast news and lecture transcriptions, as shown in [26, 31]. To apply these techniques effectively, however, one does require suitable and robustly trained ASR systems and language models as shown by Meng *et. al.* [28]. In under-resourced environments this is generally not a valid assumption, as many developing-world languages do not have such resources available.

Therefore, the main goal of this research is to investigate whether it is feasible to create an ASR corpus, for an under-resource language, by harvesting an audio source, which provides audio data and accompanying approximate transcriptions, without having access to directly relevant ASR resources. In addition, the automatic method should be developed with the constraint that the approach does not require extensive manual intervention. This constraint is imposed since many developing-world contexts do not have the necessary language processing expertise, as highlighted by Barnard *et. al.* [29].

To narrow the scope and focus the research task, we will develop an ASR corpus for broadband South African English (SAE) and build a recogniser using the harvested audio data. SAE is classified as a resource-scarce dialect of English and is significantly different from major English dialects such as American or British English. These dialect differences do present substantial recognition challenges [29] and provides a motivation to develop resources for SAE. Furthermore, the research will

focus on harvesting a corpus of audio data which is accessible to the public via a website and that provides approximate transcriptions.

In addition, our research will focus on:

- investigating a number of approaches which require varying amounts of manual intervention,
- exploring the ASR development process in such a domain, and
- establishing if good recognition results are possible given limited manual intervention and limited ASR processing skill.

In this chapter we describe the chosen SAE wideband broadcast source in section (3.2) and in section (3.3) we highlight the steps needed to correctly pre-process the data. In section (3.4) we elaborate the iterative harvesting technique and describe the resource creation process needed to successfully apply the harvesting technique. The final corpus design and post-processing algorithms are described in section (3.5). The experimental setup for various experiments can be found in section (3.6). Finally, in section (3.7), we present the results and make concluding remarks concerning the work in section (3.8).

3.1.1 PUBLICATION NOTE

This work was done in collaboration with Etienne Barnard, Marelie Davel and Charl van Heerden. The chapter is based on the joint conference paper "*Efficient harvesting of Internet audio for resource-scarce ASR*" [51]; the current author was involved in all aspects of that work, and had primary responsibility for all tasks related to data pre-processing and the creation and verification of manual transcriptions.

3.2 MONEYWEB DATA SOURCE

Nowadays internet broadcasts are commonplace, with news providers and radio stations uploading podcasts of their shows for the general public to access at their convenience. Sometimes, text transcriptions accompany the audio which allows a greater reach of the broadcast, for example, for persons who prefer the written format or who live with disabilities. Internet sources which contain audio and transcriptions provide a good starting point for the creation of a speech corpus. As the SAE dialect is currently classified as resource-scarce, any readily available data source which could aid in the growth of the language resources would prove to be most beneficial.

We investigate one such source, namely *MoneyWeb*, which is accessible from the internet address *www.MoneyWeb.co.za*. The producers of the show upload podcasts of the broadcasts and include an accompanying approximate text transcription (an imperfect transcription suitable for human reading). The contents of the podcasts are limited to financial, business and investment news topics and the style of the audio varies from interviews, with in-studio as well as phone-in guests, to read news reports.

The audio also contains music usually played at the introduction and conclusion of the shows and advertisements which may be completely spoken or contain a musical component.

3.2.1 AUDIO DATA

The audio is provided in the MPEG Audio layer III (MP3) lossy compression format [52]. Due to the compression scheme used and the recording environments, the audio quality is variable. The general recording sample rate is set at 44.1 kHz, but a few audio files have varying sample rates. The audio format is duplicated stereo meaning the two channels are identical. The recording file name convention is represented as YYMMDD-NN.MP3, where YY are the last two digits of the year, MM is for the month, DD is for the day and NN is the broadcast number for the day of recording.

The content of the audio is quite varied and contains different speaking styles and dialects, as well as music and other non-speech events. The recordings contain a mix of spontaneous and read speech and depending on the speaker the speech is either well articulated or marred with many disfluencies. The disfluencies encompass false starts, repetitions, fillers and repaired utterances. In addition, given the nature of radio interviews many recordings contain varying levels of speaker overlap and large portions of the audio contain low-bandwidth telephone-quality speech.

3.2.2 TEXT DATA

The transcription for a podcast can be obtained by following a hyperlink to the desired podcast, which forwards the user to a web page containing a link to the audio and a hypertext formatted transcription. A registered subscriber may obtain a copy of the transcription via email. The text contains a speaker tag at the beginning of a paragraph in bold font and terminated by a colon e.g **ALEC HOGG:**. Ideally, automated tools could be used to download the podcasts and capture the text from the website, as manually sourcing the data is a time-consuming task; however, for our exploratory experiments, we did not invest in the development of such tools.

The quality of the text content can be consider to be "professional". This means that the text typically does not have speech disfluencies indicated, the text contains a small fraction of spelling errors, speaker errors such as grammar mistakes are repaired (without such repair being indicated in any way) and are rife with transcriber inconsistencies (ambiguous date representations and numerical financial quantities).

3.2.3 INITIAL MONEYWEB CORPUS

Over the course of a couple of months, a few members of the Meraka Human Language Technologies (HLT) research group downloaded MoneyWeb podcasts and the corresponding transcriptions from broadcast years spanning 2008, 2009 and 2010. Table 3.1 shows a breakdown of the downloaded data by year.

Following the collection of the MoneyWeb data, the corpus was randomly partitioned into four

Table 3.1: *Initial MoneyWeb corpus broken down by year of broadcast.*

Year	# of recordings
2008	260
2009	268
2010	835
Total	1363

sections, namely, training, manual training, development and evaluation. Splitting a corpus into a training, development and testing sets is a standard practice in corpus design. We added an additional class, *manual training*, which was used to evaluate how much manual effort is needed in processing a raw set of audio recordings and transcriptions compared against automated techniques. Table 3.2 shows the partitioned corpus make-up, indicating the original size of each set in hours and the number of audio files per set.

Table 3.2: *Partitioned MoneyWeb corpus. Sizes are in hours.*

Data Set	# of hours	# of files
Training	99.9	1083
Manual Training	13.8	185
Development	2.7	45
Evaluation	3.8	50
Total	120.2	1363

It must be noted that the speaker distribution of the MoneyWeb data is highly skewed towards the interviewers, news readers and frequent guests. The proportion of audio data contributed by frequently occurring speakers far outweighs the overall contribution of all other speakers combined. This is in contrast to specifically designed speech recognition corpora which, as a design criterion, try to balance different speaker contributions. Additionally, speech corpora ensure mutually exclusive speaker sets which form the training, development and evaluation sets. Given the nature of the MoneyWeb data, these criteria would be impossible to achieve. Fortunately, they are also irrelevant to our primary purpose, which is to boost the language resources in a resource-scarce environment. It would be easy to return to strict design criteria once sufficient data amounts have been collected.

3.3 DATA PRE-PROCESSING

In this section we describe the pre-processing methods and tools which were used to clean up the raw audio and text data and transform the data into a consistent format. The pre-processing involves the following tasks:

- audio normalisation,

- text normalisation, and,
- missing pronunciation prediction.

3.3.1 AUDIO NORMALISATION

As stated above, the MoneyWeb audio data are available in a MP3 lossy compression format. For the HTK toolkit to process the audio files, we used Sox, an audio manipulation application [53], to transform the audio into the standard Microsoft WAVE RIFF format [54] which used signed 16 bits to represent each sample. Given that there were a varying number of sampling frequencies present in the audio data a down-sampling process was applied to the data which involved applying a low pass filter, with a steep cut-off at 8 kHz, and then re-sampling to 16 kHz. As most of the audio data was recorded in a studio environment, no amplitude normalisation was performed on the data (it was assumed the audio normalisation would be performed by the studio recording equipment). Finally, the audio files were mixed down to a single channel per file which simply meant discarding one of the channels. This option was taken after visually inspecting several audio files which revealed that the stereo channels were identical. In future a more robust audio normalisation process should be implemented which (1) performs clip-detection to ensure proper volume normalisation and basic audio quality control and (2) to detect if one of the stereo channels is missing and rather select the channel which has audio data to construct the mono channel version.

3.3.2 TEXT NORMALISATION

The initial text normalisation phase implemented a basic level of processing which was performed to reformat the raw transcriptions into a consistent format. The raw downloaded transcriptions were altered by converting all words to their lower case representations, removing speaker tags and punctuation, removing spurious text which was identified by *, = or comment (#) letters and collapsed paragraphs spread across multiple lines. The final format of the processed text file placed the text from a speaker turn on a single line.

3.3.3 PRONUNCIATION DICTIONARY

The final stage of the pre-processing phase was missing pronunciation prediction. Given the nature of the MoneyWeb text data and the fact that no large SAE pronunciation dictionaries are currently available, an automated pronunciation prediction scheme was used. The chosen technique was a grapheme-to-phone (g2p) prediction approach based on the popular Default&Refine paradigm. Evaluation of this method can be found here [55]. The rules for the Default&Refine process were extracted from an in-house SAE dictionary [56] which itself was bootstrapped from the British English dictionary, OALD.

To predict a pronunciation for an unknown word, the input words must contain only graphemes which were present at the time of extracting prediction rules. The MoneyWeb data however, contains

many numerical and financial oriented terms e.g. 20th, 2009, $5/ounce, 245c/share. Thus, an additional round of normalisation was needed to convert the numerical and financial terms into words. An in-house token normalisation software package, developed by the HLT TTS group, was modified and used to perform the necessary normalisations. The conversions are performed given a lookup table of rule-based regular expressions and matching tokens to the closest rule. The modifications were necessary to deal with the financially-oriented numerical values. Table 3.3 shows a few examples of token normalisation which converts tokens to word-level equivalents.

Table 3.3: *Token normalisation process converting numerical values to word equivalents.*

Raw text token	Equivalent words
170kg	one hundred and seventy kilograms
400000oz	four hundred thousand ounces
irp5	i r p five
$870	eight hundred and seventy dollars

Once the tokens have been converted to their word-level equivalents, the g2p algorithm can predict pronunciations for out-of-vocabulary words. After the prediction phase has completed, the pronunciations can be mapped to the tokens as they would appear in the text. Table 3.4 shows the number of unique words and number of words without dictionary pronunciations by MoneyWeb corpus set type.

Table 3.4: *The number of unique words and words which did not have dictionary pronunciations for each MoneyWeb set.*

Data Set	# of unique words	# missing pronunciations
Training	18508	3776
Manual Training	8454	1197
Development	3653	321
Evaluation	4404	426
Total	19566	4387

The pronunciation prediction can be fully automated and provides a quick means to easily supplement the corpus with new text. On occasion the text contains missing graphemes which prevents the g2p from predicting a pronunciation. In such cases manual pronunciations have to be generated.

3.4 ITERATIVE HARVESTING

Our data harvesting approach was similar to the "lightly supervised acoustic training" technique proposed by Lamel *et. al.* [26] which is roughly made up of three steps (1) data segmentation, (2) word-based alignments and (3) filtering. Our approach, however, differs in that we do not require the data segmentation step to partition the data into homogeneous data segments. We instead relied

on a garbage model (detailed in section (3.4.4)) which was utilised during the alignment and filtering phases. Using this method of processing freed us from requiring word recognition outputs and language models which are needed by the "lightly supervised acoustic training". Lastly, we did not use dynamic programming to filter the data, as used by Lamel *et. al.* [26], but relied on a time-based filtering scheme which was enabled by the garbage model. We only made use of our DP filtering approach during the final corpus creation (see section (3.5)).

The approximate MoneyWeb transcriptions contain no timing information with which to align text portions to the audio. Fortunately, phone recognisers provide a means to obtain alignment information which can be used to extract the speech segments. The challenge in a resource-scarce environment is how to develop robust acoustic models for the task. Two approaches are to (1) source the data and train acoustic models from scratch, or, (2) adapt existing acoustic models to improve their robustness.

An appropriate confidence measure is needed to give an indication of badly aligned segments and by discarding these "bad" segments we can improve the phone recogniser acoustic models. With better alignments and a robust data filtering technique we can develop a final corpus which contains reliable audio-transcription pairings.

3.4.1 PUBLICATION NOTE

This section reports on work that was done in conjunction with Marelie Davel and Charl van Heerden. Marelie Davel implemented the dynamic programming filtering algorithm detailed in section (3.5.1) and Charl van Heerden implemented the garbage modelling process described in section (3.4.4).

3.4.2 BOOTSTRAPPED ACOUSTIC MODELS

One way to generate acoustic models is to bootstrap models from a well-resourced language or dialect. The US and UK dialects of English are well-resourced and either could provide a good starting point from which to bootstrap acoustic models for South African English. Given that we had access to a high-quality read speech US English corpus, the Wall Street Journal (WSJ) corpus [57], we decided to use this corpus to train our initial acoustic model set.

Our phone recognition system was a cross-word tied-state context-dependent recogniser. Three state left-to-right Hidden Markov Models were used to model the triphone contexts and each state was modelled using eight Gaussian mixture models. The model parameters are estimated using the Baum-Welch algorithm. The acoustic data was mapped onto a Mel Frequency cepstral coefficient feature space using a 39 dimensional feature vector. The feature vector comprised 13 static, 13 first derivative and 13 second derivative coefficients. Cepstral mean normalisation of the static coefficients was applied on a per utterance basis. Semi-tied transforms were estimated for 40-classes as provided by a regression class tree. This phone recognition system was our standard system and all new acoustic models trained were based on this setup.

As US and SA English are dialects with distinct phonologies and phonetic categories, phone set mappings had to be performed. We manually merged the two phone sets by applying the following

rules;

- splitting diphthongs into monothongs
- removing stress and diacritic markings
- combining closest matching phones

The US English phone set contains three fewer phones compared to that of the SAE phone set that we employ (see [51] for details). For the initial forced alignment phase, using the WSJ acoustic models, these phones were not used. After the first retraining phase these phones were inserted and trained using the MoneyWeb data. The final phone set contained 33 phones as well as *sil* (silence) and *sp* (short pause).

3.4.3 MANUALLY DERIVED ACOUSTIC MODELS

As stated previously, a portion of the MoneyWeb data, approximately 14 hours, was set aside for manual processing. The reason for manually processing the data was to investigate whether manual labelling and correction of the source data was really needed, and what gain could be achieved in following a manual route compared to that of an automated approach. The main goals in manually processing the data, were to produce time-alignments between the approximate transcriptions and audio, and, broadly filter the audio data by removing non-transcribed, music and music plus speech audio segments.

Initially the audio was pre-processed using techniques described in section (3.3.1). The approximate transcriptions were pre-processed in a manner similar to the one described in section (3.3.2), except that the speaker tags were not removed. It was found that the speaker tags provided a good means to track speaker changes and, for single speakers, to track changes in the audio. For example, in the single speaker case, if a speaker read the news which was followed by an advert and then went into an interview, the approximate transcription would have separate entries for the news and interview – the advert would not be indicated. The separate text entries found in the approximate transcriptions (marked by speaker tags) were logical audio segments but an additional process was needed to time-align these entries to the audio and filter the data.

To aid in the alignment and filtering process, an application, *Segmenter*, was created. This application takes as input the audio file and transcription. It extracts the speaker tags from the text source and provides check boxes which are labelled with the speaker names. Check boxes for music and non-transcribed text were also provided. The application splits the audio track into segments which are a fixed user-defined duration - the typical duration for a segment used in practice was one second. The user would then press the play button and the application would continuously play the segments until the end of the track or stop if interrupted by the user. The user would then for each segment click on the check boxes and so label each second of audio with an appropriate tag. After labelling the entire audio track, the tags would be saved to a label file. This label file provides a crude alignment between the audio and text data. Figure 3.1 shows a screen capture of the *Segmenter* application.

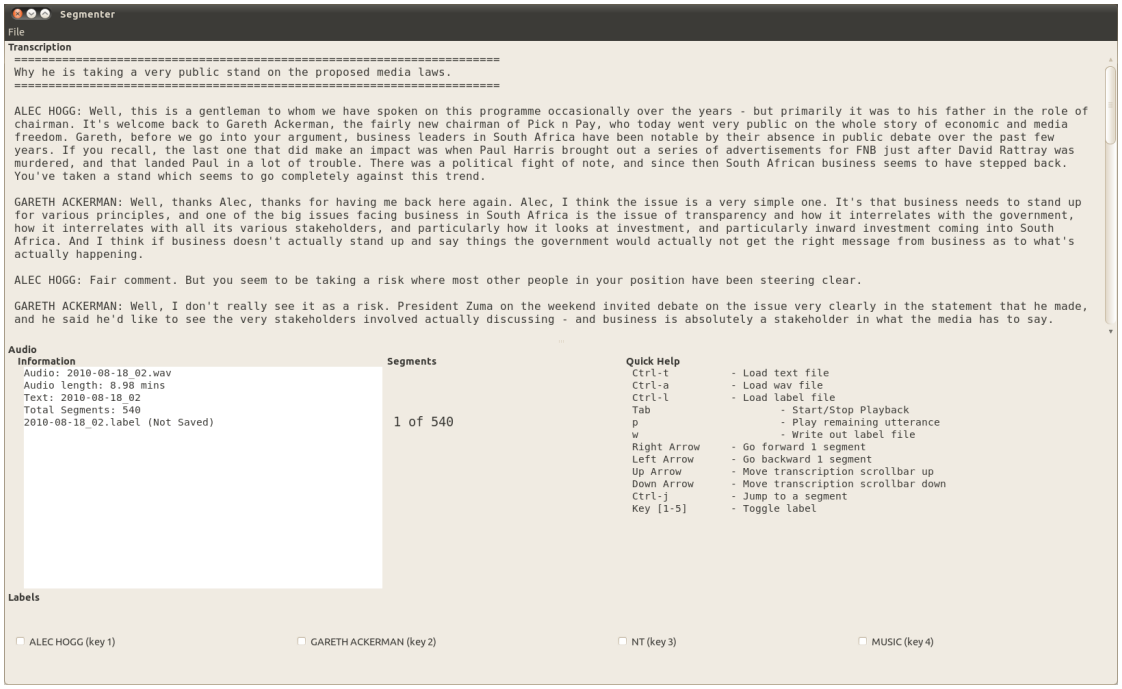

Figure 3.1: *The Segmenter application which was used to create crude alignments between the audio and transcriptions.*

Using the label files, approximate transcriptions and audio files, a small manually processed corpus was created. The approximate text entries were extracted, speaker tags removed and placed into separate text files. The time-alignments provided by the label files were then used to extract the corresponding audio segments from the utterance, thus producing a time-aligned text and audio pairs. By selecting audio, which had only speaker labels marked, a light filter mechanism was introduced which removed music and non-transcribed audio portions. This small MoneyWeb data subset was used to MAP adapt the WSJ acoustic models. Three iterations of the MAP adaptation algorithm were run in the adaptation process.

3.4.4 GARBAGE MODEL

All our acoustic models contain a *garbage* model This broad acoustic model is used to absorb speech disfluencies, non-speech audio and non-transcribed speech and in doing so aid in the refinement of the forced alignments. The garbage model is represented by a left-to-right HMM with three absorbing states and sixteen mixtures per state; it is trained on all audio data without word or phone labels. A semi-tied transform is also estimated for the model. The garbage model is inserted into HTK *sp* model which gives that model the functionality to absorb speech disfluencies, music or non-transcribed portions.

The standard *sp* model, used in HTK, is a TEE model which allows the absorption of between-word silence or can be skipped altogether [10]. It is a three state HMM with an empty entry and exit state and its middle state tied to the middle silence model state. The modified *sp* model is expanded

to six states; with the standard empty entry and exit states (state one and six), the second state tied to the middle silence state and the garbage model occupying states three, four and five. The transition probabilities are modified to allow the model to be skipped, pass straight to the garbage model, return to first state from the last garbage model state or exit the entire model from the last garbage model state. Figure 3.2 shows the hybrid *sp*-garbage model.

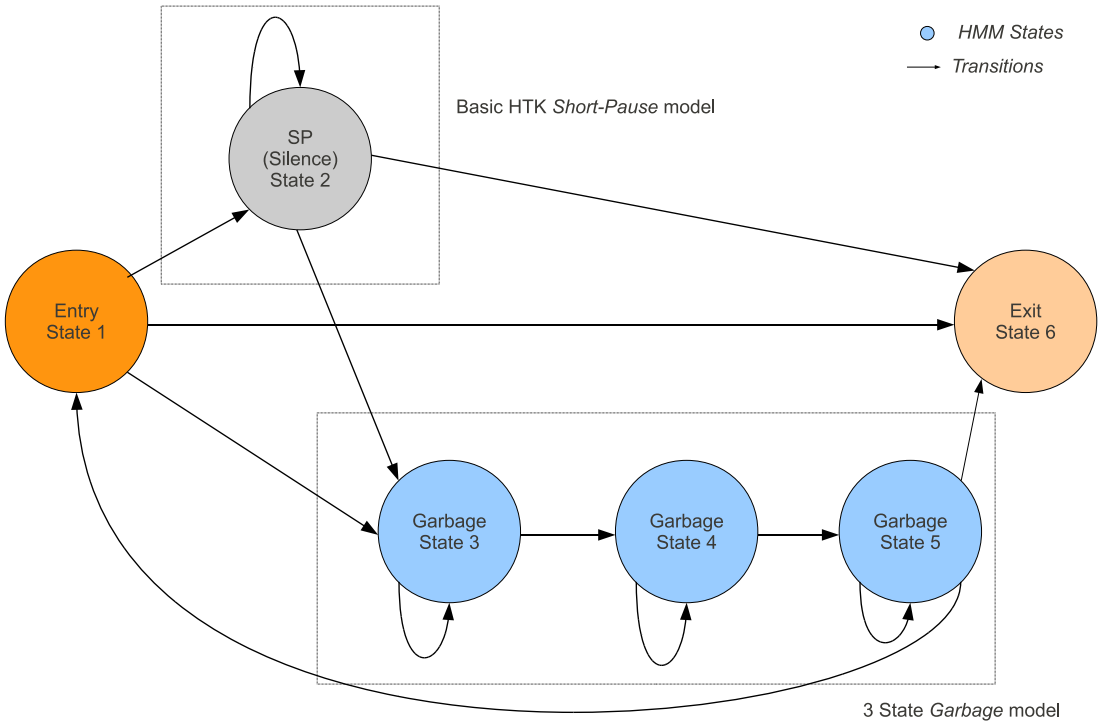

Figure 3.2: *The modified sp-garbage HMM model.*

Care must be taken when using a garbage model as it could end up absorbing all the speech. The amount of speech absorbed by the garbage model must be monitored after each iteration to make sure that useful alignments are being produced.

3.4.5 ALIGNMENT-FILTER-TRAINING CYCLE

Once all previously mentioned steps have been completed, we have reached a point to start iteratively harvesting the MoneyWeb data. The first phase (*alignment*) is to align the approximate transcriptions with the corresponding audio. To do this, acoustic models are required which can be pre-existing, for instance trained on similar data, or from the output of the alignment-filter-train cycle. Our acoustic models were updated to contain a garbage model (see section (3.4.4)) which will absorb speech disfluencies and non-speech artefacts and provided a more robust alignment. To create the alignments the HTK *HVite* tool is used to compute forced alignments [10]. It does this by firstly loading the provided transcriptions and dictionary and expanding the words into phone representations. A new network is created by concatenating the phone sequences. The network may contain multiple paths

depending on the number of word pronunciation variants. Lastly, HVite uses the Viterbi algorithm to find the best path through the network which matches the acoustic data to the acoustic models.

Following the alignment, a simple *filtering* scheme is used to extract audio segments which have good alignments. The filtering method segments the audio by cutting up the audio on silence boundaries and discards short speech segments less than 700 ms in duration. It was found for mis-aligned portions, the decoder would add many *sp*-garbage and really short speech labels while trying to squeeze in the text and that the correlation between the bad DP-scores and short speech segments was quite high. Thus, instead of using the resource-intensive DP-score filtering approach, we could achieve comparable results by using this simpler data filtering scheme. These surviving audio and *sp*-garbage segments are used to train a new set of acoustic models or adapt the existing acoustic models.

Using the source data to update the acoustic models reduces the mismatch between the data and the models and ensures better output alignments. Once the new models are available the process is repeated until some criterion is met – for instance the acoustic log-likelihoods or some other confidence measure stabilizes.

3.5 CORPUS CREATION

The audio-text alignments generated by the alignment-filter-training phase are more reliable than the initial alignments and can be used to generate an accurate corpus. To create the new corpus a more elaborate data filtering scheme was used to select the most reliable audio alignments. After compiling the corpus and adding speaker identities to the segments, a final data filtering step was run which classified the bandwidth of the audio data. This allowed us to produce two corpora, which were either suitable for low-bandwidth or high-bandwidth applications.

3.5.1 DATA FILTERING

A dynamic programming (DP) data filtering scheme was used to select audio segments which had reliable alignments. The DP algorithm compared two phone strings obtained from the forced alignments and a free decode of the audio using an flat phone grammar. The final acoustic model sets were used to generate the phone strings. The phone string comparison was performed an a per segment basis and used a variable cost matrix to find the best alignment between the phone strings. The DP score gives an indication on how well the strings are aligned and serves as a confidence measure when selecting segments. The cost matrix was manually generated and had a score distribution which favoured matched alignments. The scores were assigned as follows: 1 for correct alignments, -0.2 for easily confusable phone substitutions and -1 for all other errors (substitutions, insertions and deletions). The DP score for a segment was calculated by finding the best alignment score and normalising it by the length of the segments. A user-adjustable threshold was set and segments which possessed DP scores greater than the threshold were added to the corpus. The eventual size of the

corpus is dependent on the threshold value, with a high threshold value discarding many segments but producing segments which have a high reliability.

3.5.2 AUDIO BANDWIDTH DETECTION

The final phase of the post processing was to classify the bandwidth of the audio segments. To classify the bandwidth we needed to determine if the signal contains useful audio information above 4 kHz. In general, this would be a simple task but the MP3 lossy compression scheme complicated the process. The compression algorithm uses a psychoacoustic model to determine which components of the spectrum are necessary for proper human perception and achieves a compression by discarding non-essential components. For human hearing many speech events only require frequency components below 4 kHz for correct perception and in general lower frequency components, because of their larger energy, will mask higher components making the higher frequencies non-essential. This trend could be observed in the data as high bandwidth audio contain many instances of audio signals which had all the frequency content above 4 kHz removed. To further complicate the classification process, many of the audio files contained clipped audio. The issue of clipped audio is that the spectral envelope information is distorted which tends to spread the frequency content across the frequency band. This gives an appearance of a high-bandwidth signals. The only reliable distinguishing factor between the low- and high-bandwidth portions we could find was to calculate the ratio of energy between the frequency components above and below the 4 kHz boundary.

The initial algorithm used to segment an audio track into constant-bandwidth segments was:

- Pre-emphasis filter the audio to boost the high frequency components.
- Split the audio track into 50 ms non-overlapping segments.
- Over all the frames and for each frequency component, sum (time-wise) the component's magnitudes using 4 seconds of frames before and after the current frame.
- For each frame determine the maximum magnitude frequency component below and above the 4 kHz boundary and calculate the ratio. Convert the ratio into its dB value.
- Median filter the dB values to obtain smoothed trajectories. We used a sample window of 31.
- Define the mid-point to be the average of the maximum and minimum values of the smoothed dB values.
- Find all locations where the dB values transition through the mid-point values. These are considered to be change in bandwidth candidates.
- Eliminate all candidates whose relative change is less than a threshold value. The threshold was set to 20.
- Define audio segments as audio which lie between the change in bandwidth points.

- Classify each segment with average dB value greater than mid-point value as high-bandwidth, else classify it as low-bandwidth.

Summing the frequency component's magnitudes (time-wise sense) over 4 seconds gave good smoothing – analogously, if one refers to BIC audio segmentation, values greater than 1 or 2 seconds are generally used [58] – as better statistics are estimated over longer windows. The median filter window and candidate elimination threshold were chosen after visually inspecting a few samples. The median filter window length of 31 gave a good balance between smoothing and resolution while the elimination threshold of 20 managed to give successful classifications in the majority of cases.

The benefit of the above algorithm is that only the audio data is needed. Conversely though, the issue with the algorithm is that (1) the segments which are classified have no alignment with the transcriptions and (2) 4 seconds of audio data is needed to classify a 50 ms frame. To overcome the difficulties of the initial algorithm we decided to use a Gaussian Mixture Model (GMM) classifier approach to classify audio segments. Using the bandwidth classifications and manually labelled files we segmented the manual MoneyWeb corpus audio into 4 categories, namely, music (music), music plus speech (+music), high-bandwidth speech (hb) and low-bandwidth speech (lb). The HTK toolkit was used to train models for each of the categories using a 1 state HMM. 512 mixtures were used to model the feature distributions. The features were 39 dimensional MFCCs; 13 static, 13 first derivatives and 13 second derivatives. Based on results published by Acero and Stern [59] performing class-based cepstral normalisation significantly improves cepstral normalisation, thus we deviated from the standard HTK Cepstral Mean Normalisation (CMN) approach and estimated an average MFCC vector for each category, and then used this vector to normalise the category-specific feature vectors. The assumption is that the average cepstral estimates for the various classes are relatively different and normalising out-of-class cepstral vectors, will create vectors that are best described by a large non-zero mean normal distribution. The way in which we classified a segment of audio was to send the audio through the 4 GMM classifiers and chose the label from the classifier which gave the maximum average log-likelihood score. The benefit of the GMM approach is that we can present segments, obtained from the DP scores selection process, to the classifiers and obtain a result straight away.

Initially, it was thought that we would need a music and music plus speech classifier to separate these audio types from the utterance, but the garbage model proved to be extremely adept at absorbing these audio types. Thus, these categories were not used. In section (3.7) below we report on the performance achieved with these algorithms.

3.6 EXPERIMENTAL SETUP

In this section we describe the various corpora, performance metrics and experimental setups used to validate our harvesting approach.

3.6.1 CORPORA

3.6.1.1 MONEYWEB DEVELOPMENT AND EVALUATION CORPORA

As described above, the corpus was partitioned into separate training, manual, development and evaluation sets. The development set had 2.7 hours of audio data and 45 speakers, while the evaluation set had 3.8 hours of audio data and 50 speakers. The development sub-corpus was used during corpus development experiments and the evaluation corpus was used for final verification purposes.

3.6.1.2 LWAZI ENGLISH CORPUS

The Lwazi [29] corpus contains telephone-quality recordings and their associated transcriptions sourced from the eleven official languages of South Africa. There are approximately 200 speakers per language. For our experiments we limited ourselves to the English language. Table 3.5 shows the number files and duration in hours of audio data for the Lwazi English sub-corpus. Given its limited size, we used the entire corpus as an evaluation set.

Table 3.5: *The number of files and duration in hours for Lwazi English sub-corpus.*

# files	5843
Duration (hours)	10.05

3.6.1.3 NCHLT ENGLISH CORPUS

The NCHLT corpus [60] contains high-bandwidth speech recordings sourced from the eleven official South African languages. The text prompts were derived from large text corpora and the selection criterion was chosen to attain a coverage of the most common triphones. For our experiments we were limited to the English language sub-corpus and made use of the corpus's evaluation set. Table 3.6 shows the number of files and duration in hours for the NCHLT English evaluation set.

Table 3.6: *The number of files and duration in hours for the NCHLT English evaluation set*

# files	3232
Duration (hours)	2.3

3.6.2 PERFORMANCE METRICS

Here we describe a few proxy measures used to determine both the quality of the acoustic models trained on the MoneyWeb corpus and the audio and transcriptions that make up the harvested corpus. The use of proxy measures gives us an automated means to monitor the harvesting process

and limits the need for manual intervention during the harvesting process. The proxy measures were used throughout the corpus development phase and final evaluation, and verified with a number of independent measures as we describe below.

The average acoustic log-likelihood of the corpus data served as our first proxy measure. This measure informs us how likely it would be that the various acoustic models would generate the audio data. The log-likelihood scores (logL) are reported on per-frame basis which gives a finer resolution of measure. As stated previously, we needed to make sure that the garbage model (garbage %) did not end up consuming the entire corpus, so monitoring the amount of data absorbed by the garbage model became our next proxy measure. This measure has a direct influence on our first measure, as the average acoustic log-likelihood scores only give meaningful information if the amount of data consumed by the garbage model decreases or stays the same. The next measure used was derived from the DP scores (ADPS) which were used in the final stage of selecting the best aligned audio portions. The DP scores inform us how well the decoded phone strings fit with the phone representation of the approximate transcription. Lastly, as we did not have gold standard MoneyWeb transcriptions we used the forced alignments generated by our final acoustic model set as a proxy for these missing pristine transcriptions. Using these transcriptions as a reference set, we performed phone recognition and reported these accuracies as our last proxy measure.

Although the proxy measures provide a convenient method to automatically monitor the harvesting process, their reliability had to be verified. To investigate the proxy measures' viability, phone recognition results were also generated using the Lwazi and NCHLT corpora, which had verified phone transcriptions. In addition, the phone recognition results also provide a means to validate and verify the data harvesting process in itself, similar to the approach followed by Lamel *et. al.* [26].

3.6.3 SETUP

3.6.3.1 WSJ BOOTSTRAPPED HARVESTING

Our first experiment was designed to test the feasibility of our data harvesting approach – the alignment-filter-train cycle. The experiment starts with unmatched WSJ acoustic models and runs through three cycles of the alignment-filter-train process, producing new acoustic models at each cycle. The MoneyWeb training corpus is harvested for each cycle and the processed data used to train the acoustic models. The only manual actions needed to prepare the WSJ acoustic models were (a) to map the phone set and (b) to include a garbage model during the training process. To determine the quality of the acoustic models and track the performance of the proxy measures, all proxy measures (logL, ADPS and Garbage (%)) are reported for each cycle. In, addition phone correctness and accuracy percentages, generated on the MoneyWeb evaluation set, are listed. Lastly, to validate the proxy measures, phone correctness and accuracy percentages, generated on the Lwazi and NCHLT corpora, will be listed.

The experimental outcomes will help us verify (1) that it is possible to make use of an unmatched

acoustic model set, developed on audio data sourced from a dialect, to start the harvesting process, (2) at each cycle the produced models are of better quality compared to the previous cycle's output and (3) the proxy measures are a reliable measure with which to measure the data harvesting process performance.

3.6.3.2 MANUAL DATA PROCESSING

Our automatic data harvesting approach initially uses unmatched acoustic models to start the process and develops acoustic models at each alignment-filter-train cycle. Lamel *et. al* [26] showed that their technique could be started by using acoustic models trained on one hour or less of manually transcribed data sourced from the raw audio data. Thus, a question that arose was: could our automated process perform better if the initial models that were used to start the process incorporated some audio data from the source?

Thus, to better understand the effect of using in-domain data, we experimented with different initial acoustic models which were created by MAP adapting the WSJ models with varying amounts of in-domain data. Specifically, we used 30 min, 1 hour, 2 hours and 4 hours of adaptation audio data where the adaptation data was sourced from the manual training set. Due to the nature of the MoneyWeb data, which contains large amounts of speech data from a relatively few speakers, we had to employ a selection algorithm to try and balance the speaker contributions. The simple algorithm tried to add data uniformly from the various speakers, with the maximal amount from any single speaker being limited to approximately 5 % of the corpus. The process was terminated when the target data amount had been reached.

For this experiment two alignment-filter-train cycles were run and the four MAP adapted models and the WSJ model used for initialization. The MoneyWeb training corpus was harvested and at the end of each cycle the surviving data was used to train new acoustic models.

The quality of the resulting acoustic models was measured by the suite of proxy measures generated by the development set and phone correctness and accuracy percentages, generated on the Lwazi and NCHLT corpora. The experiment was designed to verify if (1) manual intervention is really needed by the automatic data harvesting process and (2) if so, how much intervention is required.

3.6.3.3 CORPUS SIZE

The final automatic harvesting experiment tested the stability of our approach with respect to the amount of data which had to be harvested. In this experiment, 10% and 20% sub-corpora, randomly chosen from the MoneyWeb training corpus, were created and used to determine the stability of the automatic harvesting process. The experiment will run two alignment-filter-train cycles on each training corpus – 10%, 20% and 100% – and at the end of each cycle a new set of acoustic models will be trained on the corpus-specific harvested data. To monitor the quality of acoustic models proxy measures will be extracted from the MoneyWeb development set and Lwazi and NCHLT corpora phone correctness and accuracy percentages are reported.

The experimental outcome will help verify if our data harvesting approach can be successfully applied to various data sizes and thus establish if the process is roughly independent of data size.

3.6.3.4 BANDWIDTH CLASSIFICATION

To investigate the accuracy of the high-/low-bandwidth speech classifier we randomly selected ten files from the evaluation dataset and hand labelled the boundaries of high- and low-bandwidth speech. Table 3.7 shows the number of frames and duration in minutes for low-bandwidth and high-bandwidth segments present in the ten hand labelled evaluation files. It was felt that choosing ten files was a reasonable trade-off between manual effort and reliable results and were representative of the evaluation set.

Table 3.7: *The number of frames and duration in minutes for the low-bandwidth and high-bandwidth segments in subset of hand labelled evaluation files.*

Class	# frames	Duration (Min)
Low-Bandwidth (lb)	67009	11.18
High-Bandwidth (hb)	217428	36.27
Total Duration (Minutes)	47.45	

To measure the bandwidth classification performance, the audio files were passed to the classifier and the classifier marked the boundaries of low and high bandwidths. The classifier accuracy was determined by aligning the manual labels and classifier outputs and totalling the durations of correct and incorrect classifications. The manual labels were created by a three person team, using Praat [61] textgrids and finally, all textgrids verified by a single individual.

The outcome of the experiment was to establish the accuracy of the bandwidth classifier.

3.6.3.5 4-CLASS CLASSIFIER

To establish the performance level of the 4-class GMM-based classifier, we used the same ten audio files, selected from the evaluation set, used to test the low-/high-bandwidth classifier. These files were manually labelled by recording the boundaries of music (music), music plus speech (+music), high-bandwidth speech (hb) and low-bandwidth speech (lb). Table 3.8 shows the number of frames for low-bandwidth, high-bandwidth, music plus speech and music segments found in the manually labelled test set.

To analyse the performance of the classifier each frame of testing audio was assigned a class label, based on the highest log-likelihood score, and compared to the manually verified labels. The performance of the classifier can be determined by how accurately the different classes are correctly recognised and how many cross-class errors are made (when an incorrect class labelled is assigned to a frame)

Table 3.8: *The number of frames for the low-bandwidth, high-bandwidth, music plus speech and music segments found in subset of hand labelled evaluation files.*

Class	# frames
Low-Bandwidth (LB)	67009
High-Bandwidth (HB)	217428
Speech plus Music (+MUSIC)	3537
Music (MUSIC)	17184
Total Duration (Minutes)	50.8

The experiment outcome will establish how accurately the 4-class classifier can distinguish between low-bandwidth, high-bandwidth, music plus speech and music audio segments.

3.7 RESULTS

In this section we present experimental results for the experimental setups discussed in section (3.6.3).

3.7.1 WSJ BOOTSTRAPPED HARVESTING

The results for starting the automatic data harvesting process with unmatched WSJ acoustic models and running three cycles of the alignment-filter-train process are shown in table 3.9. The results shown are the acoustic log-likelihoods scores (logL), average dynamic programming score (ADPS), the percentage of data absorbed by the garbage model (Garbage (%)), the phone accuracy percentage (Phn Acc) and the phone correctness percentage (Phn Cor). The phone accuracies and correctness were obtained using the MoneyWeb evaluation set while the remaining proxy measure results were obtained from the training set. Table 3.10 shows phone correctness and accuracy results using the acoustic models generated at each stage of the alignment-filter-train process and recognising the Lwazi and NCHLT data.

Table 3.9: *Improvements in the proxy measures and phone correctness and accuracies for three alignment-filter-train cycles and initially using the bootstrapped WSJ acoustic models. Results obtained on the MoneyWeb evaluation set.*

Model	logL	Garbage (%)	ADPS	Phn Acc (%)	Phn Cor (%)
WSJ	-87.019	37.62	0.139	36.45	45.22
retrain 1	-79.109	32.78	0.337	53.85	60.93
retrain 2	-78.436	31.08	0.358	55.67	62.22
retrain 3	-78.264	30.76	0.359	56.40	62.84

The harvesting measures in table 3.9 stabilize relatively quickly and converge after the 3 iterations. The original training data set contained approximately 99 hours of audio data (see table 3.2) and after 3 iterations of processing contained 68 hours of mixed high- and low-bandwidth speech data only. The

Table 3.10: *Phone correctness and accuracy measures for the iterative alignment-filter-train which initially used the bootstrapped WSJ acoustic models. Results were obtained using the Lwazi and NCHLT corpora.*

	Lwazi		NCHLT	
Model	Phn Acc (%)	Phn Cor (%)	Phn Acc (%)	Phn Cor (%)
WSJ	22.35	43.04	38.55	56.62
retrain 1	28.83	48.37	39.08	55.95
retrain 2	30.82	49.62	39.42	56.19
retrain 3	31.57	49.99	39.11	56.42

phone accuracies are low given the size of the harvested corpus (compare, for example, WSJ itself, where phone accuracies of approximately 82 % are typically achieved [62]), but mixed bandwidth speech data and the spontaneous speech quality play a large role in reducing the accuracy. Selecting more reliable alignments, by setting a higher DP control threshold when selecting audio portions, and by splitting the corpus by bandwidth would improve these accuracies.

Table 3.10 shows similar increasing trends for the phone correctness and accuracies compared to table 3.9. The absolute values are relatively low but the acoustic model training data and testing data (Lwazi and NCHLT) were recorded in different environments which is known to severely impact recognition results. The increasing trends fit the proxy measure trends well, suggesting that our proxy measures are functioning as planned.

3.7.2 MANUAL DATA PROCESSING

Results for using the WSJ and different MAP adapted acoustic models as initial acoustic model sets for the automatic harvesting process are presented. Table 3.11 shows the acoustic log-likelihood scores (logL), average dynamic programming score (ADPS) and the percentage of data absorbed by the garbage model (Garbage (%)), obtained using the MoneyWeb development set, for the WSJ and 0.5, 1, 2 and 4 hour MAP adapted acoustic models and for two retrain cycles. Table 3.12 shows phone correctness and accuracies generated using the Lwazi and NCHLT corpora, for the WSJ and 0.5, 1, 2 and 4 hour MAP adapted acoustic models and for two retrain cycles.

Table 3.11: *Comparing the results of adding various adaptation data amounts to update the initial and MAP adapted WSJ acoustic models. Results obtained on the development set.*

	initial			retrain1			retrain 2		
Model	logL	ADPS	Garbage (%)	logL	ADPS	Garbage (%)	logL	ADPS	Garbage (%)
WSJ	-84.62	0.197	38.675	-77.23	0.402	35.029	-76.70	0.413	33.649
0.5 hr MAP	-81.93	0.263	38.045	-77.27	0.403	35.165	-	-	
1hr MAP	-81.17	0.298	37.203	-77.19	0.408	35.031	-	-	
2hr MAP	-80.12	0.336	36.204	-77.14	0.410	35.059	-	-	
4hr MAP	-79.17	0.365	34.967	-77.06	0.413	34.847	-76.56	0.416	33.426

Table 3.12: *Comparing the results of adding various adaptation data amounts to update the initial WSJ acoustic models. Results obtained using the Lwazi and NCHLT corpora.*

		Lwazi		NCHLT	
	Model	Phn Acc (%)	Phn Cor (%)	Phn Acc (%)	Phn Cor (%)
Initial	WSJ	22.35	43.04	38.55	56.62
	0.5 hr MAP	26.84	45.92	38.77	55.93
	1hr MAP	28.42	46.95	39.36	56.63
	2hr MAP	30.46	48.42	39.69	56.49
	4hr MAP	32.75	49.63	40.28	56.99
Retrain1	WSJ	28.94	48.35	39.02	56.04
	0.5 hr MAP	30.14	48.87	39.63	56.41
	1hr MAP	29.99	49.15	39.58	56.21
	2hr MAP	30.54	49.3	39.8	56.37
	4hr MAP	32.75	49.63	39.43	55.98
Retrain2	WSJ	30.8	49.5	39.49	56.05
	4hr MAP	31.37	49.91	39.11	55.89

The measures shown in Table 3.11 stabilize after 2 alignment-filter-train iterations and follow similar trends to the WSJ bootstrapped method used to harvest the training data set. Considering the initial measure, we can observe the models do benefit from more adaptation data, given the low log-likelihood values, high average DP scores and low Garbage absorption values, which is consistent as more training examples are available. But, as we run through the retrain cycles the benefits are lost quite quickly. Interestingly, the bootstrapped WSJ model performs equally well compared to the 4 hour MAP adapted model after just 2 retrain cycles.

Table 3.12 show the same trends as observed by the proxy measures: initially the models perform much better when adapted with increasing amounts of in-domain data, but as the retrain cycles increase the benefit of using adaptation data diminishes.

3.7.3 CORPUS SIZE

Table 3.13 shows the acoustic log-likelihoods scores (logL), average dynamic programming score (ADPS) and the percentage of data absorbed by the garbage model (Garbage (%)) proxy measures, obtained using the MoneyWeb development set, for harvesting 10%, 20% and 100% of the training corpus. Table 3.14 show Lwazi and NCHLT phone correctness and accuracies using acoustic models developed on 10%, 20% and 100% of the training data and for two retrain cycles.

As with our other harvesting experiments, consistent convergence trends are followed. The general trend in table 3.13 shows that with more data we are able to achieve better absolute proxy measure scores but the relative changes in convergence rates are approximately the same.

In table 3.14, the Lwazi phone results validate the overall proxy measure trends which show that the acoustic model quality is improved when more data is used to train the models. The NCHLT results show similar improvement trends but the initial WSJ models have quite high phone correctness and accuracy values, which limits the margin of improvement.

Table 3.13: *Comparing the efficiency of the harvesting approach on restricted total data sizes. Proxy measures obtained on the development set.*

	initial			retrain 1			retrain2		
model	logL	ADPS	Garbage (%)	logL	ADPS	Garbage (%)	logL	ADPS	Garbage (%)
10 %	-84.62	0.197	38.675	-79.17	0.296	37.535	-78.47	0.323	36.249
20 %	-84.62	0.197	38.675	-78.43	0.340	36.412	-77.67	0.378	35.227
100 %	-84.62	0.197	38.675	-77.23	0.402	35.029	-76.70	0.413	33.649

Table 3.14: *Comparing the results of adding various adaptation data amounts to update the initial WSJ acoustic models. Results obtained using the Lwazi and NCHLT corpora.*

		Lwazi		NCHLT	
	Model	Phn Acc (%)	Phn Cor (%)	Phn Acc (%)	Phn Cor (%)
Initial	WSJ	22.35	43.04	38.55	56.62
Retrain1	10%	21.78	43.75	31.17	50.4
	20%	25.52	46.02	34.59	52.33
	100 %	28.94	48.35	39.02	56.04
Retrain2	10 %	24.03	45.01	32.83	51.79
	20 %	27.22	47.42	36.86	53.8
	100 %	30.8	49.5	39.49	56.05

3.7.4 BANDWIDTH CLASSIFICATION

Table 3.15 shows the accuracy of the bandwidth classifier to correctly identify high-bandwidth (hb) and low-bandwidth (lb) audio segments.

Table 3.15: *The percentage accuracy and errors made by the high-low bandwidth classifier.*

		Classified Class	
		lb	hb
True Class	lb	94.49	5.51
	hb	0.72	99.28

The ability of the classifier to correctly identity between the two classes is quite high. The task of identifying low-bandwidth speech seems to be slightly more difficult as more errors were made in trying to identify this class. The proposed algorithm seems to work well and clearly overcomes the MP3 and clipping artefacts.

3.7.5 4-CLASS CLASSIFIER

Table 3.16 shows the percentage of correctly identified frames and errors made by assigning frames to the incorrect class, of the 4-Class classifier. The four classes are music, music plus speech (+music), low-bandwidth (lb) audio and high-bandwidth (hb) audio.

Table 3.16: *The percentage of correctly identified frames and frames in error made by the 4-Class classifier.*

		Classified Class			
		music	+music	hb	lb
True Class	music	85.04	11.51	3.08	0.37
	+music	7.30	66.70	24.10	1.91
	hb	2.85	32.26	61.68	3.20
	lb	0.31	1.66	3.76	94.27

Considering the table of percentages, the classifier readily distinguishes low-bandwidth speech and music audio with a great accuracy level. Unfortunately, it makes a considerable amount of errors between high-bandwidth speech and music plus speech audio. This task appears to be rather difficult for 16 kHz sampled audio and a benefit may be achieved if the sampling rate is increased; as music contains useful frequency components above the 8 kHz bandwidth and can be used for greater class separation. Fortunately, this distinction is not critical to the rest of our processing, since the garbage model is able to reject speech which is heavily influenced by the presence of music.

3.8 CONCLUSION

The work presented in this chapter shows an automated and efficient data harvesting process which takes publicly available audio data with approximate transcriptions and creates a reliable speech recognition corpus. Significant contributions to emerge from the research are;

- The process of alignment-filter-train, to harvest audio data with approximate transcriptions, can be efficiently automated.
- Using the Lwazi and NCHLT English corpora we showed that the proxy measures, log-likelihood, average dynamic programming scores, and the percentage of data absorbed by the garbage provided useful methods of automatically monitoring the harvesting process.
- Bootstrapped acoustic models trained on out-of-dialect audio data can be used to provide good audio-transcription alignments.
- Retraining acoustic models on the harvested data provides better models to generate more accurate alignments and iterating the alignment-filter-train cycle provides better models after each iteration.
- A garbage model can be used to good effect in order to improve the reliability of the alignments by absorbing non-speech audio and speech disfluencies.

- A dynamic programming algorithm can be used to robustly filter the harvested data and create a corpus which has quite reliable alignments between text and audio. Additionally, the DP scores serve as a good metric with which to monitor the harvesting process.
- Manually processing data to create an adaptation corpus, which can be used to adapt a set of acoustic models, is not needed. The win obtained by manually processing the data is lost after the first retrain cycle.
- The data size, of the to be harvested data, affects the performance metric values – more data results in better values. However, the performance metric convergence rates exhibit a low correlation to the data size and the relative improvements are approximately the same.
- The proposed algorithm to detect the bandwidth of the speech data overcomes the MP3 encoding and audio clipping artefacts.
- A GMM-based classifier, which classifies frames of audio data as music, music plus speech, low-bandwidth speech and high-bandwidth speech, can easily distinguish the low-bandwidth and music class. It, however, shows some difficulty in classifying the high-bandwidth and music plus speech classes.

Besides these findings, and the software tools that have been developed, an important output from this work is the corpus itself – creating a new corpus for an under-resourced language is an important step to enable the further development of speech processing tools. Depending on the exact threshold chosen, the corpus contains around 70 hours of transcribed speech, with a variety of speaking styles. This corpus has been used for the training of acoustic and language models, for research on the influence of speaking style on recogniser accuracy, and for the analysis of keyword-spotting algorithms. Unfortunately, due to contract obligations to a client the work performed on the database cannot be released to the public but a spoken-term-detection system was successfully built using the harvested corpus. The spoken-term-detection system is described in [63] but the analysis was performed on a different – and more challenging – corpus.

For future work, the data pre- and post-processing tools can be improved – specifically, an audio-clip detector which can be used by the bandwidth detector to ignore clipped audio portions. In addition, a channel detector should be added to make sure we are selecting an audio channel which has data. The 4-class classifier could also be improved by adding GMM smoothing and updating the CMN vectors with speech only.

CHAPTER FOUR

CROSS CHANNEL ADAPTATION

4.1 INTRODUCTION

It is well known that speech recognition systems perform poorly when there is a mismatch between the acoustic models and testing audio data. The mismatch can manifest itself in several ways; the leading causes are environmental noise, channel differences, various speaking styles and different dialects. A system's performance can be greatly increased if the mismatch is sufficiently reduced. Currently, acoustic model adaptation and feature normalisation techniques provide a means to reduce the mismatch and play a crucial role in speech recognition system deployment. Feature normalisation improves the feature robustness by trying to remove channel or environmental distortions while acoustic model adaptation shifts the model's means and scales the variances to accommodate for the change in data statistics. In a resource-scarce environment, adaptation techniques are particularly important as one often has to develop ASR systems using available data which is not matched to the application: for instance, using high-bandwidth data to train acoustic models for an application that operates on telephone quality audio data – over the course of using the system, application-specific data can be collected and used to adapt or create new acoustic models and thereby improve the performance.

Generally, feature normalisation is applied to the feature vectors independent of ASR task which improves system robustness and performance. To further improve the ASR system accuracies more complex adaptation schemes are used in cascade such as MLLR and MAP, and, finally, when enough data is collected retraining the acoustic models becomes viable. Using the results obtained by Chen and Bilmes [35], Goronzy and Kompe [42], Wang *et. al.* [43] , Wallace *et. al.* [45], and, Bocchieri *et. al.* [46], (as described in section (2.4.1) and section (2.4.3)), we can in summary state that the expected order of increasing performance gains provided by the use of various adaptation techniques

is : feature normalisation, adaptation by MLLR transformation, MAP adaptation and retraining the models.

For the specific ASR scenario describe by Bocchieri *et. al.* [46], MAP adaptation provided the best results for fewer than 1500 sentences, which equated to 3.5 hours of audio data and roughly 2 hours of speech data. Between 1500 and 6000 sentences, training the context-trees on in-domain data and estimating the state distributions on both in- and out-domain data resulted in the best performance – 6000 sentences corresponded to 14.4 hours of audio data and approximately 9 hours of speech data. For 6000 sentences and above, retraining the HMM acoustic models on the in-domain data provided the best results. However, no data threshold was provided for the use of MLLR. Leggetter and Woodland [38] showed in their speaker-adaptation experiments that mean-only MLLR adaptation, using a full global regression matrix, yielded improvements after three adaptation utterances (roughly 11 seconds of speech). The performance gain saturates at about 15 utterances. To make better use of the additional adaptation data, additional regression classes are needed.

Wang *et. al.* [43] showed that for non-native speaker adaptation and for a fixed number of speakers MAP adaptation consistently performed better than MLLR, independent of adaptation data amount. The only scenario where MLLR provided a performance gain over MAP was when the number of speakers found in the adaptation data were varied; however, even in that case MAP proved a better choice once the adaptation data amount exceeded 20 minutes. The investigation performed by Wallace *et. al.* [45] showed when applying the adaptation techniques in isolation and on 60 minutes of data, MAP produced the best gain followed by regression tree MLLR and lastly global MLLR. Cascading global plus regression tree MLLR plus MAP obtained the best performance gain at 60 minutes of data as well as at varied adaptation data amounts of 10 and 30 minutes.

From the relevant literature survey we can see that the boundaries where one would chose a specific adaptation technique over another are quite varied. The transition boundaries are dependent on the ASR task where parameters such as speaker number and adaptation type (environment, dialect or speaker) have an influence over the transitions. The boundary measure is usually specified by duration, generally measured in minutes or hours or audio data. This can be misleading, as adaptation performance is likely to depend on how much speech data is actually available within the audio data. For instance read, conversational or distressed speech would all have different ratios of speech to non-speech. In terms of current HMM-based ASR systems a more informative unit would be the total number of words, phones or triphones found in the adaptation data.

Collecting speech resources is time consuming, incurs high costs and requires linguistic expertise [29]. These costs can be reduced in a number of ways by using data collection techniques such as harvesting of internet resources, utilising speech collection tools like Woefzela [60] or telephony-based collection. For under-resourced languages the primary drive is to aggressively collect data to boost the resources with less emphasis being placed on the collection environment. Thus, for these languages it is possible that several corpora can be created relatively easily by sourcing data from different environments which may result in quite varied data characteristics across the entire dataset.

For example, in South Africa the high-bandwidth NCHLT [60] and telephony Lwazi [29, 30] corpora are available which contain language data for eleven South African languages. If a telephony ASR application is required one could make use of the vastly larger speech resources found in the NCHLT corpus to improve the acoustic models developed solely on the Lwazi corpus.

In an under-resourced language environment it becomes necessary to make use of standard adaptation techniques to improve ASR system performance, however, the available data may be limited. The performance gain achieved in using standard adaptation techniques is dependent on the amount of adaptation data and each standard adaptation technique has a specific performance curve. Therefore the aim of this research is to:

- Investigate several standard adaptation techniques performance gain curves, dependent on the adaptation data amount, for mixed low-bandwidth telephone-quality and high-bandwidth high-quality audio data scenario.
- Set up a guideline that indicates which adaptation method is most appropriate given an adaptation data amount, for this type of application.

For the course of our experiments we limited ourselves to the use of standard normalisation and adaptation techniques and report on additional evidence regarding the relative contributions of different adaptation methods. Some of these findings confirm facts that are already known in the literature. The amount of adaptation data will be specified by the number of triphones which provides a more relevant unit for ASR systems. As stated, our experiments will focus on mixed low-bandwidth telephone-quality and high-bandwidth high-quality audio data applications and investigate how to port acoustic models starting from a low-bandwidth scenario and progress towards a high-bandwidth one and vice versa – a scenario of great relevance in developing-world contexts.

A description of the various feature normalisation and adaptation techniques used in our experiments can be found in section (4.2). The corpora used and the data selection strategy are described in section (4.3). Results are presented in section (4.4) and concluding remarks can be found in section (4.5).

4.2 NORMALISATION AND ADAPTATION METHODS

Generally speaking, the process of improving a speech recognition system's performance can be applied in the feature and model domains [38]. Both approaches strive to reduce the mismatch between the acoustic models and data feature vectors. With feature normalisation techniques, the feature data statistics are either normalised to a standard set of values or transformed to the training values. Model-based adaptations are realised by updating the acoustic model parameter directly either by re-estimating the parameter values or estimating a set of transforms which indirectly provides the updated model parameters. In our adaptation experiments we used a standard set of feature normalisation and model adaptation methods which are currently available to the speech recognition system

developer. We will start off by describing the a few feature normalisation techniques and then elaborate on various model adaptation approaches.

4.2.1 FEATURE NORMALISATION

Feature normalisation applies a set of transforms on the feature vectors, which reduces the mismatch between the feature vectors and acoustic models. In general, the normalisation techniques are unsupervised, meaning the method has no reliance on knowledge concerning the audio content, and assumes certain distributions for the feature vectors and distorting phenomena (such as noise and channel responses).

4.2.1.1 BAND LIMITING

A simple feature normalisation strategy is to band-limit the spectral content of the audio signals. Moreno and Stern [34] demonstrated the importance of matching the portion of speech bandwidth which is used to extract speech features on the Timit and NTimit corpora. This suggests that for unknown channels a process should be run to determine the bandwidth of useful speech information. Incorporating this knowledge into acoustic model design can result in better system performance. Their work also showed that band limiting by linear filtering the audio signal does not take into account other sources of signal degradation introduced by real-world channels. These sources include additive noise and non-linear gain and phase distortions. More elaborate feature normalisation or model adaptation techniques are needed to reduce these non-linear and noisy effects.

4.2.1.2 CEPSTRAL MEAN NORMALISATION

One of the first compensation techniques to be used widely was cepstral mean normalisation. The method estimates an average cepstral vector over a set of cepstral observations and removes the bias from each vector. This simple technique performs well in removing convolutional noise and constant channel distortions. HTK [10] applies CMN to static cepstral coefficients only, leaving the derivative components unaltered, and estimates the average cepstral vector on a per utterance basis. An online-based cepstral mean normalisation can also be employed in live applications, where a global cepstral vector is estimated prior to system deployment and is continually updated as new feature vectors are processed.

4.2.1.3 MVA

Chen and Bilmes [35] showed through their in-depth analysis that cepstral mean normalisation (subtraction) worked well at removing convolution noise but performed poorly in removing additive noise. It was further shown that the effects of additive noise, depending on the noise level, can be reduced if variance normalisation was applied to the cepstral coefficients. Lastly, Chen and Bilmes re-introduced

filtering of the cepstral coefficients which limits the modulation frequencies and improves the dynamic range of the cepstral trajectories by suppressing noise effects. The cepstral trajectories were filtered using a finite length auto-regression moving average (ARMA) filter. The process of applying mean subtraction, variance normalisation and ARMA filtering is known as MVA. The steps involved in transforming a cepstral vector are represented mathematically by

$$c_m^i(t) = c^i(t) - \mu \tag{4.1}$$

$$c_v^i(t) = \frac{c_m^i(t)}{\sigma} \tag{4.2}$$

$$c_a^i(t) = \frac{c_a^i(t-M) + ... + c_a^i(t-1) + c_v^i(t) + ... + c_v^i(t+M)}{2M+1}, \tag{4.3}$$

where μ is the cepstral mean, σ is the cepstral standard deviation, t is the discrete time instance, i is the dimension and M is the ARMA filter length. M was set to 2 and all cepstral coefficients (static and derivative) were normalised as suggested in [35].

4.2.1.4 NORMALISATION LENGTH

Feature normalisation can be applied at varying feature set sizes. The most common normalisation length is the utterance, however, the normalisation can be applied at the speaker-level (all utterances from a speaker) or in online operation. Both Segura *et. al.* [37] and Viikki and Laurila [64] showed, in a digit recognition setup, that over 50 frames of feature vectors (more than 0.5 s) are needed to robustly normalise the means and variances of the features vectors. The experiments performed by Segura *et. al.* [37] showed that mean and variance normalisation out-performed mean normalisation only. Chen and Bilmes [35] showed that MVA produced better results when compared to MV normalisation only, in both an online and utterance-based normalisation mode. For their digit recognition setup, it was empirically found that at least 30 speech frames are needed before commencing online MVA feature normalisation.

4.2.1.5 TRANSFER-FUNCTION FILTERING

Gelbart and Morgan [36] showed that feature normalisation can be achieved by removing a long-term average log spectral estimate from spectral analysis frames. Their technique, however, required lengthy speech segments to estimate the average log spectrum and relatively longer analysis windows. Such delays would be impractical for real-time ASR systems. It was shown previously [65] that channel normalisation can be performed by inverse filtering the short-term spectra. Starting with the basic idea we formulated a slightly different approach. If it is assumed that the discrete cosine transform of the logarithmic short-term spectra are drawn from a multivariate Gaussian distribution, then channel normalisation can be realised by the mapping of normal distributions. The first step is to estimate the *mean* ($\boldsymbol{\mu}$) and *covariance* ($\boldsymbol{\Sigma}$) statistical moments, which, using the maximum-likelihood estimates, are given by

$$E[\mathbf{X}] = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_i \tag{4.4}$$

$$\boldsymbol{\mu} = E[\mathbf{X}] \tag{4.5}$$

$$\boldsymbol{\Sigma} = E[\mathbf{X}^2] - E[\mathbf{X}]^2, \tag{4.6}$$

where $\boldsymbol{\mu}$ is the mean value, $\boldsymbol{\Sigma}$ is the covariance and $E[]$ is the expected value operator. The measures are extracted from channel-specific data and require no transcriptions. The manner in which the estimates are obtained are as follows;

- Block the audio in 25 ms frames and overlap consecutive frames by 10 ms (standard values used in current ASR feature extraction).
- Firstly apply the logarithmic transform to short-term frame spectra then apply the discrete cosine transform.
- Update the mean and covariance accumulators.
- Once all frames have been processed, calculate the final mean and covariance values.

Given various mean and covariance statistics estimated from different channels, one set of feature vectors (in our case the discrete cosine transform of the logarithmically mapped short-term spectra) can be normalised to another distribution by firstly normalising the feature vectors to zero mean and unit covariance distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, then applying an affine transform to the feature vectors to shift their statistics such that they will produce the target mean and covariance measures. This is achieved by applying the following steps to each feature vector:

- $\mathbf{Z}_{zero} = \mathbf{A}_{src}^{-1}(\mathbf{Z}_{src} - \boldsymbol{\mu}_{src})$, where $\mathbf{A}_{src}$ is given by the Cholesky decomposition of $\boldsymbol{\Sigma}_{src}$, $\mathbf{A}_{src}\mathbf{A}_{src}^T = \boldsymbol{\Sigma}_{src}$.
- $\mathbf{Z}_{tgt} = \mathbf{A}_{tgt}\mathbf{Z}_{zero} + \boldsymbol{\mu}_{tgt}$, where $\mathbf{A}_{tgt}$ is given by the Cholesky decomposition of $\boldsymbol{\Sigma}_{tgt}$, $\mathbf{A}_{tgt}\mathbf{A}_{tgt}^T = \boldsymbol{\Sigma}_{tgt}$.

After transforming a feature vector to the most likely target vector, the inverse discrete transform is applied and the values mapped by the exponential function. This channel normalised linear spectrum is sent through to the feature extraction unit for final processing. This technique is similar to the one proposed by Gelbart and Morgan [36] but differs in the following ways: (1) applied to short-term spectrum 25 ms, and, (2) extends log spectral subtraction by assuming the analysis frames are drawn from a Gaussian distribution and applies mean and variance normalisation.

4.2.2 MODEL ADAPTATION

Model-based adaptations reduce the mismatch between model and acoustics by updating the model parameters. Given current techniques, these parameters can be updated directly through parameter (re-)estimation or indirectly by estimating a set of linear transforms. Maximum A-Posteriori (MAP) adaptation is currently the most popular method to directly update model parameters, while Maximum Likelihood Linear Regression (MLLR) provides an indirect update mechanism.

4.2.2.1 MAXIMUM LIKELIHOOD LINEAR REGRESSION

First proposed for speaker adaptation [38], MLLR was easily extended to environmental adaptation and has since become a standard technique. The initial framework was limited to mean-only adaptation but Gales and Woodland [39] expanded the framework to include variance adaptation. The technique estimates a set of linear transforms from adaptation data, then updates the model parameters by applying the transforms to the mean and variance parameters. An advantage of the technique is that the transforms are applied to the models during the decode phase which removes the need to store newly adapted models. The technique also requires relatively small amounts of adaptation data since it uses binary regression class trees to group similar models together and thus create larger class-specific pools of adaptation data. The MLLR implementation is elegant since multiple transforms can be applied to model parameters. For instance a typical transform estimation process would initially estimate a set of mean transforms, then applying these mean transforms to the models, estimate a set of variance transforms - thus forming a cascade of transforms.

For our MLLR experiments we used the following approach;

1. Estimate a 40-class regression tree.
2. Estimate 40-class-specific semi-tied transforms.
3. Using the semi-tied transforms as parent transforms, estimate 40-class-specific mean transforms.
4. Using the mean transforms as parent transforms, estimate 40-class-specific variance transforms.

The number of regression classes was set to 40 which correlates well to the average number of sound classes in a language. These mean and variance transforms are stored in separate files and are loaded and applied to the models during recognition.

4.2.2.2 MAXIMUM A-POSTERIORI ADAPTATION

MAP adaptation provides a means to adapt the model parameters without having to retrain the models from scratch. The MAP training procedure incorporates prior information which provides a parameter estimation benefit compared to standard maximum likelihood parameter estimates [40]. The effectiveness of MAP adaptation is only fully realised at relatively large data amounts as the technique

updates different model components separately. Thus, the adaptation data must cover quite a large set of different training examples and each example a sufficient number of times. However, Gauvain and Lee [40] showed that using MAP adaptation to speaker adapt existing speaker-independent models requires much less data to gain substantial improvements in the word error rates (compared to retraining the models). Therefore, it does seem that MAP possesses a lower critical data limit than the limit needed to train robust acoustic models.

HTK [10] provides a mechanism to update the weights, means, variances and various combinations of the these. The MAP adaptation experiments that we performed either adapted the weights-means combination or weights-means-variances combination and used 10 adaptation iterations.

4.3 EXPERIMENTAL SETUP

In this section we describe the corpora used during the investigation, baseline ASR system, performance measures, data selection and cross-channel experiments.

4.3.1 CORPORA

The various feature normalisation and model adaptation experiments were performed on pairs of American English and IsiNdebele read-speech corpora. To ensure a mismatch between corpora a low- and high-bandwidth version of each language were chosen, and to simulate the typical environment for low-resource languages, we experimented using within-language cross-channel adaptation.

4.3.1.1 WALL STREET JOURNAL

The Wall Street Journal (WSJ) CSR corpus contains high-bandwidth American English read-speech utterances and orthographic transcriptions [57]. For our purposes we sourced the speaker-independent sub-corpus which contains a separate training and testing set with no speaker overlap. The audio was recorded with a Sennheiser microphone at a sample rate of 16 kHz and contains financially oriented content. The transcriptions contain three text subsets: a small set spoken by all the speakers, a few sentences which have limited speaker overlap and a unique sentence set. There are an equal number of male and female speakers. Table 4.1 shows the make-up of the WSJ corpus.

Table 4.1: *The WSJ corpus statistics for the training and testing sets.*

Set Type	# utterances	# hours	# speakers
Train	12776	24.9	101
Test	1142	2.2	10

4.3.1.2 NTIMIT

NTimit (Network Timit) is a telephone-bandwidth read-speech corpus [66]. NTimit was created by transmitting the Timit corpus data through “local” and “long-distance” telephone networks in the

United States. The purpose of the NTimit corpus was to aid in the investigation of telephone network distortions on speech. The Timit corpus is an high-bandwidth American English read-speech corpus. The main corpus design criteria ensured phonetic diversity which enables the study of general speech characteristics. The data was collected across the United States and encompassed the eight main dialect regions of the country. Each speaker contributed ten sentences; two were common to all speakers and were used to investigate dialect variations, five were selected to provide phonetic diversity and the last three were sourced from the Brown corpus. Table 4.2 shows the NTimit corpus statistics.

Table 4.2: *The NTimit corpus statistics for the training and testing sets.*

Set Type	# utterances	# hours	# speakers
Train	4617	3.9	462
Test	1675	1.4	168

4.3.1.3 NCHLT

The NCHLT corpus is high-bandwidth read-speech corpus containing audio data and transcriptions collected from eleven South African languages. The audio data was recorded using a number of high-quality mobile devices. The transcriptions contain short sentences and were derived from large text corpora in order to attain coverage of the most common triphones of the target language. The collection software performs automatic quality control and strives to ensure that the utterances are correctly recorded. For our cross-channel experiments we limited ourselves to using the IsiNdebele sub-corpus (which was the only completed sub-corpus at the initiation of our experiments). The initial corpus contained 90297 utterances collected from 209 speakers. After running pre-processing, which removed utterances that contained English words, clipped audio data and audio files containing incorrect header information, the corpus was reduced to 60687 utterances and 169 speakers. For English word detection we sourced an in-house English pronunciation dictionary and created a list of all words found in the dictionary. To detect English words, present in the IsiNdebele text, we compared each word to the list of English words and identified out-of-language words. The NCHLT corpus does not have a dedicated training and testing set; hence, five-fold cross validation was used to partition the corpus and create the desired sets. Table 4.3 shows the five-fold training corpus statistics and table 4.4 shows the five-fold testing corpus statistics.

4.3.1.4 LWAZI

The Lwazi corpus contains read and elicited speech recordings collected from eleven South African languages [29]. There are approximately 200 speakers per language and the audio data was recorded over the telephone network. Each speaker contributed thirty utterances; sixteen sentences were sourced from phonetically rich text while the remaining 14 sentences were elicited by questions that

Table 4.3: *The NCHLT-IsiNdebele training and adaptation corpora. The corpora statistics are reported by cross-validation folds.*

Fold	# utterances	# hours	# speakers
1	47348	61.78	136
2	49206	63.31	136
3	49710	63.28	136
4	49128	64.36	136
5	48847	63.05	136

Table 4.4: *The various NCHLT-IsiNdebele cross-validation testing corpora.*

Fold	# utterances	# hours	# speakers
1	13339	16.61	33
2	11481	15.08	33
3	10977	15.11	33
4	11559	14.03	33
5	11840	15.34	33

produced either short phrases or single words (e.g. yes/no answers, digits, etc...). To create a counterpart for the NCHLT corpus we chose the IsiNdebele sentences. As with the NCHLT corpus, we had to create a speaker-independent training and testing sets - we did this by partitioning the corpus into five sub-corpora. Tables 4.5 and 4.6 show the sub-corpora statistics for the training and testing sets respectively.

Table 4.5: *The Lwazi-IsiNdebele training/adaptation cross-validation corpora.*

Fold	# utterances	# hours	# speakers
1	4817	4.09	160
2	4804	4.11	160
3	4813	4.08	160
4	4807	4.11	160
5	4811	4.11	160

Table 4.6: *The Lwazi-IsiNdebele testing corpora shown by cross-validation fold.*

Fold	# utterances	# hours	# speakers
1	1196	1.03	40
2	1209	1.01	40
3	1200	1.05	40
4	1206	1.02	40
5	1202	1.02	40

4.3.1.5 DATA SELECTION

To investigate the relationship between the amount of adaptation data and the performance of each adaptation method, we needed to devise an algorithm that would grow adaptation data pools from a given data set. Additionally, we needed to obtain an average accuracy value so we decided to repeat each experiment five times, which meant five adaptation data pools had to be created. Our simple data growing algorithm performed the following steps:

- Randomly partition the data into five sub-corpora and ensure each pool has unique speakers.
- For each sub-corpus (sub-corpora are processed independently), start at the first file and sum up the number of triphones contained in each subsequent file added to the data pool. At specified triphone counts, save the file list up to that point.
- If the desired triphone counts cannot be achieved, within a given data pool, randomly select data from the other sub-corpora until the count is reached.

It must be noted that the algorithm "grows" the adaptation pool. For example, if we would like to create two lists which contain files contributing 100 and 250 triphone counts, then the 250 triphone count file list will contain all the files present in the 100 triphone counts file list as well as additional files which make up the difference.

4.3.2 BASELINE ASR SYSTEM

The speech recognition system, was based on a standard HMM-based system [13]. Firstly, the audio data was converted to a set of standard MFCC vectors. The vectors were estimated from a 25 ms audio window and a 100 vectors per second of speech were calculated. Each vector was constructed by concatenating 13 static, 13 first derivative and 13 second derivative coefficients. Cepstral mean normalisation was applied on a per utterance basis and only to the first 13 coefficients. The HMMs, used to model the cross-word context-dependent triphones, were of a 3 state left-to-right structure and each state contained 8 mixture diagonal covariance Gaussian models. A question-based tying scheme was followed to create a tied-state data sharing system [23] - where any context-dependent triphone having the same central context could be tied together. As a last step a 40-class binary regression tree was estimated and a semi-tied transform was estimated for each class [41].

4.3.3 PERFORMANCE MEASURES

The performance of the various ASR systems will be measured using phone-level accuracies. These are calculated by using a dynamic programming algorithm to generate an alignment between a phone-level reference transcription and phone-level recognition output. The phone accuracies are calculated using the following formula:

$$Accuracy = \frac{N - D - S - I}{N} \times 100\%, \tag{4.7}$$

where N is the total number of phones found in the reference transcription, D is the total number of deletions, S is the total number of substitutions and I is the total number of insertions. For all our experiments, the reported results are deletion and insertion error balanced, which is achieved by iteratively changing the decoder insertion penalty.

4.3.4 CROSS CHANNEL ADAPTATION INVESTIGATION

4.3.4.1 FEATURE NORMALISATION

To investigate the band-limiting results presented by Moreno and Stern [34] and the MVA results presented by Chen and Bilmes [35] several cross-channel ASR experiments will be performed using the WSJ and NTimit corpora. To test the performance gains produced by band-limiting the audio signals, ASR systems will be trained on data band-limited to specific bandwidths which will then be used to recognise testing data that was also band-limited to the same bandwidth as the training data. The three bandwidths that were chosen were 0-8 kHz (used 16kHz sampling rate and represented as 16k), 0-4 kHz (used 8kHz sampling rate and represented as 8k) and 250-3400 Hz (8kHz sampling rate and represent as BP). The cross-channel tests are realised by training WSJ ASR systems and testing on NTimit data and similarly, training NTimit ASR systems which will be tested on WSJ data. CMN will be applied to each training and testing utterance. The MVA normalisation method will be tested on the same ASR systems but instead of applying CMN to each bandwidth-specific training and testing utterances, MVA will be applied. The means and variances of the features will be normalised on a per-utterance basis based on results present by Chen and Bilmes [35]. Lastly, phone-level accuracies will be used to measure the ASR system performances.

4.3.4.2 ADAPTATION ACCURACIES AND PARAMETERS

To investigate the performance gains that the various adaptation techniques produce when applied to ASR systems which are used to recognise mismatched data, a set of cross-channel adaptation experiments will performed using the WSJ and NTimit corpora. The training and testing data from both corpora will be band-limited using the previously defined bandwidths categories (0-8 kHz (16k),0-4 kHz (8k) and 250-3400 Hz (BP)) and the processed data will be grouped to create bandwidth-specific corpora. MVA feature normalisation will be applied on an utterance-basis and to each utterance found in the separate datasets. Baseline ASR systems will be trained on each bandwidth-specific corpus and then used to test the following adaptation techniques: transfer-function filtering (TFF), mean and variance MLLR (MLLR_MV) and MAP. The training data from the cross-channel corpus will be used as adaptation data i.e. WSJ 16k acoustic model will be adapted with NTimit 16k training data, etc.

The MAP adaptation method updates the weights, means and variance components of the acoustic models. The parameters can be updated separately or in combination; for instance, weights and

means or means and variances. Therefore two MAP configurations will also be tested: (1) MAP adaptation of the weights and means (MAP_WM) and (2) MAP adaptation of the weights, means and variances (MAP_WMV). Investigating the MAP weight and mean adaptation was inspired by the speaker recognition results presented by Reynolds *et. al* [67].

Phone-level accuracies will be used to measure the various adapted ASR system performances.

4.3.4.3 PERFORMANCE GAIN CURVES

To create performance gain curves for the various adaptation techniques a set of cross-channel adaptation experiments will be performed using the WSJ and NTimit corpora combination as well as the NCHLT and Lwazi corpora combination. To generate the performance gain curves, the following procedure will be used:

- An ASR system will be trained on band-limited audio data sourced from one of aforementioned corpora's training set.
- A portion of adaptation data will be selected from the corresponding cross-channel training corpus set. The data selection approach is outlined in section (4.3.1.5).
- The ASR system will then be adapted using the adaptation techniques and the selected adaptation data. The adaptation techniques under investigation are transfer-function filtering, MLLR and MAP. In addition, an ASR system will be trained on the adaptation data without the use any adaptation techniques.
- The adapted and retrained ASR systems will be used to recognise the corresponding cross-channel testing dataset.
- The process will be repeated on increasing amounts of adaptation data. The data intervals are chosen for specific triphone counts and are 1000, 2000, 4000, 8000, 16000, 32000, 64000, 128000, 256000, 512000. The Lwazi and NTimit corpora can only support triphone counts up to 128000.

The adapted and retrained ASR system performances will be measured using phone-level accuracies. For a specific adaptation technique, the WSJ and NTimit experiment results will be averaged over the five adapted ASR systems created at each triphone count interval. For the NCHLT and Lwazi experiments the results will be averaged over the five folds and five adapted ASR systems created at each triphone count interval for a chosen adaptation approach.

4.4 RESULTS

4.4.1 FEATURE NORMALISATION

Table 4.7 shows the effect on system performance when limiting the audio signal bandwidth. The chosen bandwidths were 0-8kHz (16k), 0-4 kHz (8k) and 250-3400 Hz telephone bandwidth (BP).

For these experiments CMN was applied to the cepstral vectors. The results for the WSJ train and test experiments show that the accuracies fall as the high-bandwidth signal is increasingly band limited, which is to be expected as less spectral information is used to distinguish between the different speech sound classes. It is surprising that the drop in accuracy from 16k to 8k is so slight; this might indicate a need to increase the frequency resolution on the filter bank analysis. The cross-channel, WSJ train and NTimit test experiments, show the inverse, where, as the audio bandwidth is limited the results improve. The greatest gain is achieved when the WSJ models were trained on audio signals which have approximately the same bandwidth as the true NTimit bandwidth (in this case, telephone bandwidth).

Turning to the NTimit experiments, we see that bandwidth limiting for the NTimit train and test scenario shows a very slight increase in performance after 8k but a reasonable gain from 16k. This is most probably due to the removal of noisy bands from the speech feature extraction analysis. As with the WSJ cross-channel experiments, the NTimit train and WSJ test experiment shows a good gain in accuracy when the training and testing corpora have similar audio bandwidths. The results shown in table 4.7 correlate well with the results presented in Moreno and Stern [34], which highlights the need for closely matching the test and training audio bandwidths.

Table 4.7: *Cross-Channel speech recognition phone-level accuracies for various bandwidths of the WSJ and NTimit corpora. CMN was applied to the utterances.*

Train	Test					
	WSJ 16k	NTimit 16k	WSJ 8k	NTimit 8k	WSJ BP	NTimit BP
WSJ 16k	81.97	18.79				
WSJ 8k			81.17	19.56		
WSJ BP					77.71	38.80
NTimit 16k	11.61	56.12				
NTimit 8k			18.60	57.69		
NTimit BP					40.74	57.98

Table 4.8 shows the results for within-corpus and cross-channel experiments when MVA processing is applied to the cepstral vectors. Similar band-limiting trends hold when comparing the results to the CMN results (table 4.7) – as the audio bandwidth is limited we see a decrease in performance for within-corpus experiments using high-bandwidth signals, increase in performance for within-corpus experiments using low-bandwidth signals and increase in performance when running cross-channel experiments.

Comparing the results pair-wise in tables 4.7 and 4.8 we see a mixed bag of results. The WSJ train and test and the WSJ train and NTimit test experiments all achieved slight increases in accuracy. However, the NTimit within-corpus experiments only indicate a marginal increase in the 8k testing scenario while the other scenarios experienced slight decreases in performance. The cross-channel NTimit experiments did show an improvement for the 16k and 8k scenarios but failed to achieve a gain in the BP experiment. The performance gains MVA provided over CMN follow the same trends as seen in the results presented by Chen and Bilmes [35].

Table 4.8: *Cross-channel experiment phone-level accuracies obtained from the WSJ-NTimit corpora and using MVA feature normalisation.*

Train	Test					
	WSJ 16k	NTimit 16k	WSJ 8k	NTimit 8k	WSJ BP	NTimit BP
WSJ 16k	82.54	19.84				
WSJ 8k			81.31	21.99		
WSJ BP					78.02	40.27
NTimit 16k	12.52	55.89				
NTimit 8k			22.26	57.84		
NTimit BP					40.49	57.90

To get a sense of what improvement MVA does provide table 4.9 shows the total number of testing utterances the recogniser managed to actually decode and find a phone sequence. The NTimit test set contained 1675 utterances while the WSJ test set contained 1142 utterances in total. The results in table 4.9 show two values per cell; the first and second values indicate the total number of testing utterances the decoder managed to find a phone sequence for when CMN and MVA were applied respectively. For the WSJ train and NTimit test experiments MVA only recovered two utterances during the 16 kHz decoding phase. MVA did prove useful in recovering a large number of sentences for the NTimit train and WSJ test experiments during the 16k and 8k decodes. The results do show that at approximately matched bandwidths MVA loses its edge over CMN. It must be noted, however, that Chen and Bilmes [35] did allude to the fact that MVA might require tuning as it is application specific. All results presented from here on utilized the MVA feature normalisation, as for mismatched conditions MVA results in less recognition failures and on average a slightly better cross-channel system accuracy.

Table 4.9: *The total number of testing utterances and the number of utterances actually decoded for the cross-channel WSJ-NTimit experiments.*

Train	Test		
	NTimit 16k	NTimit 8k	NTimit BP
WSJ 16k	1673 / 1675		
WSJ 8k		1675 / 1675	
WSJ BP			1675 /1675
	WSJ 16k	WSJ 8k	WSJ BP
NTimit 16k	106 / 357		
NTimit 8k		840 / 1118	
NTimit BP			1142 / 1142

4.4.2 ADAPTATION ACCURACIES AND PARAMETERS

Table 4.10 shows accuracies obtained when using NTimit adapted WSJ acoustic models to recognize NTimit testing data. All the NTimit training data was used to adapt the acoustic models. In the table, *TFF* refers to transfer-function filtering, *MLLR_MV* refers to MLLR mean and variance

transformations, *MAP_WM* refers to MAP adaptation of the weights and means, *MAP_WMV* refers to MAP adaptation of weights, means and variances and *NTimit Train* refers to models trained with the NTimit training data. The results show a trend of improved performance as one moves from left to right column-wise (i.e. from *None* to *Train*). An exception is the MAP_WMV result for the telephone bandpass audio (BP) - which is surprising. These improvements fit well with the known adaptation method trends, as one moves from the *None* to *Train* and more acoustic model parameters are updated which gives a great ability to reduce the mismatch between the acoustic models and the testing data. Interestingly, the gain experienced from MAP_WM to MAP_WMV is marginal which might suggest that channel effects play a bigger role in shifting the model means than affecting the cepstral variations.

Table 4.10: *The cross-channel experiment phone-level accuracies obtained using WSJ trained models adapted using various adaptation techniques and all NTimit training data.*

	Adaptation Method					
Train / Test	None	TFF	MLLR_MV	MAP_WM	MAP_WMV	NTimit Train
WSJ 16k / NTimit 16k	19.84	-	42.15	49.94	50.25	55.89
WSJ 8k / NTimit 8k	21.99	-	48.43	51.53	52.99	57.84
WSJ BP / NTimit BP	40.27	43.28	50.76	55.17	53.45	57.90

Table 4.11 shows phone-level accuracies for WSJ testing data decodes using NTimit acoustic models adapted using various adaptation methods and specific WSJ training data bandwidths. In the table *WSJ Train* refers to acoustic models trained with WSJ training data. For the 8k and BP testing scenarios we see a similar trend compared to the previous set of experiments (table 4.11), where moving from *None* to *Train* provides better gains in performance. However, the 16k scenario shows us that MLLR_MV and MAP_WM perform quite poorly at reducing the channel introduced mismatch but MAP_WMV adaptation appears to be capable of reducing the mismatch quite substantially - here it seems the models' covariances play a more significant role. The importance of model variance adaptation is supported by the 8k and BP tests which show large gains when moving from MAP_WM to MAP_WMV.

Table 4.11: *The cross-channel experiment phone-level accuracies obtained using NTimit trained models adapted using various adaptation techniques and all WSJ training data.*

	Adaptation Method					
Train/Test	None	TFF	MLLR_MV	MAP_WM	MAP_WMV	WSJ Train
NTimit 16k / WSJ 16k	12.52	-	21.22	19.20	61.65	82.54
NTimit 8k / WSJ 8k	22.26	-	49.77	60.44	70.44	81.31
NTimit BP / WSJ BP	40.49	45.27	50.89	63.99	71.14	78.02

Tables 4.10 and 4.11 show that MAP_WMV adaptation does perform well, given sufficient adaptation data amounts, in updating acoustic models but there is still a performance gap between the MAP_WMV adapted models and retrained models. This suggests that during the model creation process the techniques (state tying, the number of physical models, etc...) used to form the models

provides a performance gain. In addition, if one considers the results column-wise (specific adaptation techniques), except for the WSJ retraining case, the results improve as the signal bandwidth matches which confirms the findings of Moreno and Stern [34].

4.4.3 PERFORMANCE GAIN WSJ - NTIMIT

Figure 4.1 shows the accuracies obtained from NTimit acoustic models trained on band-limited audio data and adapted using different adaptation techniques and various amounts of adaptation data sourced from the WSJ training data. The experiments show a hypothetical scenario where an ASR system initially uses acoustic models trained on low-bandwidth telephone-quality data and the application has to recognize high-bandwidth high-quality data. For all experiments MVA feature normalisation and band-limiting was utilized unless otherwise stated. To obtain an average accuracy value for an experiment the process was repeated five times on different adaptation or training sets. In the figure the following tags appear in the legend:

- **NTIMIT_WSJ_BP** - Acoustic models trained on all the band-limited NTimit training data and recognised band-limited WSJ test data.
- **WSJ_BP** - Acoustic models trained on all the band-limited WSJ training data and recognised band-limited WSJ test data.
- **WSJ_16k** - Acoustic models trained on all the 16 kHz WSJ training data and recognised 16 kHz WSJ test data.
- **NTIMIT_WSJ_TFF** - Acoustic models trained on band-limited NTimit data which was normalised using transfer-function filtering which uses increasing amounts of WSJ to estimate the filtering function. The test data was band-limited WSJ data.
- **NTIMIT_WSJ_MLLR_BP** - Acoustic models trained on band-limited NTimit data and then adapted using MLLR which is estimated on increasing amounts of band-limited WSJ data. The test data was band-limited WSJ data.
- **NTIMIT_WSJ_MLLR_16k** - Acoustic models trained on band-limited NTimit data and then adapted using MLLR which is estimated on increasing amounts of 16 kHz WSJ data. The test data was 16kHz WSJ data.
- **NTIMIT_WSJ_MAP_BP** - Acoustic models trained on band-limited NTimit data and then adapted using MAP for increasing amounts of band-limited WSJ data. The test data was band-limited WSJ data.
- **NTIMIT_WSJ_MAP_16k** - Acoustic models trained on band-limited NTimit data and then adapted using MAP for increasing amounts of 16 kHz WSJ data. The test data was 16kHz WSJ data.

- **WSJ_RETRAIN_16k** - Acoustic models trained on increasing amounts of 16 kHz WSJ training data and recognised 16kHz WSJ test data.

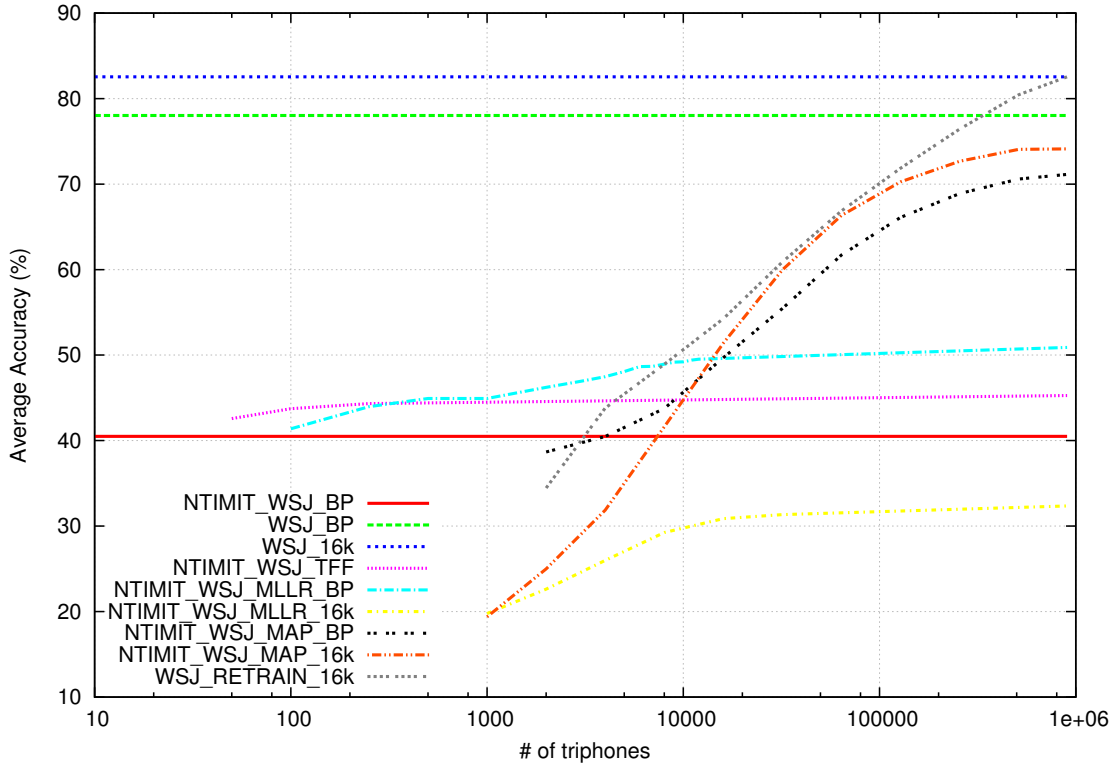

Figure 4.1: *A low-bandwidth to high-bandwidth scenario and accuracies obtained using various acoustic models and adaptation techniques.*

Interpreting the plots we can see at really low adaptation data levels (fewer than 400 triphone examples) the transfer-function feature normalisation gives the best performance gain. Around 400 triphone examples MLLR starts to give better performance gains as the transfer-function feature normalisation gain has saturated. MLLR continues to give the best gain until 7000 triphone examples where retraining the acoustic models with the 16 kHz WSJ data starts to deliver the best possible performance. The 16 kHz WSJ acoustic models performance improves considerably between 7000 and 200000 triphone training examples. Surprisingly the MAP adaptation method did not out-perform the retrained models at any stage. Even though the MAP adapted models did not give the desired performance (the expected TFF $\rightarrow$ MLLR $\rightarrow$ MAP $\rightarrow$ RETRAIN transition), the MAP adapted band-limited acoustic models initially performed better (from about 2000-10000 triphone examples) compared to the MAP adapted 16k models. It is also interesting to see how quickly the MAP adaptation performance gain plateaus: the phase of linear accuracy improvements as data increases starts to end around 80000 triphones. Lastly, the MLLR adaptation using the 16 kHz WSJ data did not perform well at all – producing accuracies well below the non-adapted ASR setup NTIMIT_WSJ_BP. This shows that MLLR performs better when there is a smaller mismatch between acoustic models and adaptation data whereas MAP has a better ability to deal with large data mismatches.

Figure 4.2 shows accuracies obtained using WSJ acoustic models trained on band-limited data and adapted using increasing amounts of NTimit training data and various adaptation techniques. The experiments show a hypothetical scenario where an ASR system initially uses acoustic models trained on band-limited high-quality audio data and the application has to recognise low-bandwidth telephone-quality data. For these experiments all audio was band-limited and MVA feature processing was applied. Each experiment was repeated five times to obtain average accuracy values. The legend tags presented in the figure mean the following;

- **WSJ_NTIMIT_BP** - Trained acoustic models on all the band-limited WSJ training data and decoded band-limited NTimit test data.
- **NTIMIT_BP** - A complete system trained and tested on band-limited NTimit data.
- **WSJ_NTIMIT_TFF** - Acoustic models trained on transfer-function filtered band-limited WSJ data which used increasing amounts of band-limited NTimit data to estimate the filtering function and tested on band-limited NTimit testing data.
- **WSJ_NTIMIT_MLLR_BP** - MLLR adapted WSJ acoustic models, using increasing amounts of band-limited NTimit training data, tested on band-limited NTimit testing data.
- **WSJ_NTIMIT_MAP_BP** - MAP adapted WSJ acoustic models, using increasing amounts of band-limited NTimit training data, tested on band-limited NTimit testing data.
- **NTIMIT_RETRAIN** - A NTimit system trained on increasing portions of band-limited NTimit training data and tested on band-limited NTimit testing data.

As with the low- to high-bandwidth scenario we see regular trends. The transfer-function feature normalisation performs the best at low triphone counts. Around 100 triphone examples MLLR starts producing better gains and continues as the best option till around 35000 training examples. At this point the retrained models start delivering the best gains. Somewhat improved MAP performance may be achievable with additional parameter experimentation, as we discuss in section (4.4.5), but we again expect only marginal utility compared to MLLR, on the one hand, and retraining, on the other

All the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for the curves can be found in appendix (B) , subsection (B.1).

4.4.4 PERFORMANCE GAIN NCHLT - LWAZI

To corroborate the data dependence trends obtained with the WSJ-NTimit corpora, we repeated the cross-channel experiments on the NCHLT-Lwazi corpora. The only difference is that the transfer-function normalisation was dropped as the MLLR appears to give approximately the same performance gains at really low data amounts. As with the WSJ-NTimit experiments, each experiment was

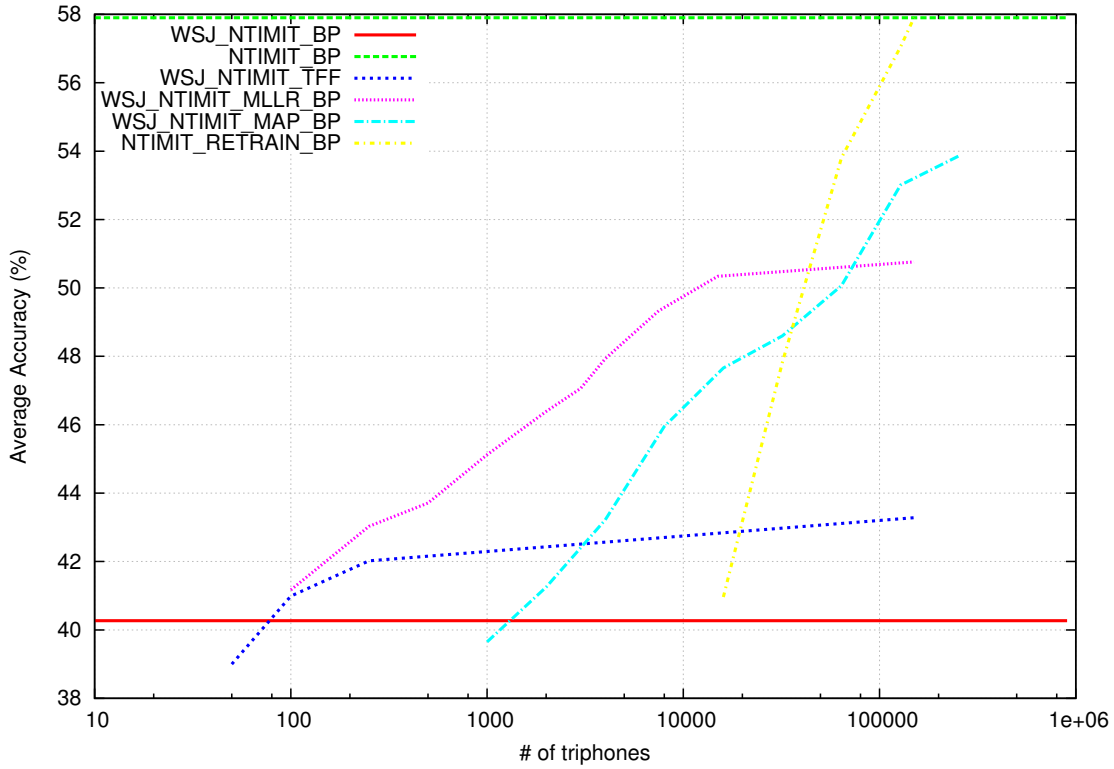

Figure 4.2: *A high-bandwidth to low-bandwidth scenario and accuracies obtained using various acoustic models and adaptation techniques.*

repeated five times and in addition, the experiments were run independently on each cross-validation fold. Figure 4.3 shows the average improvement in accuracies (across folds) as more adaptation data is used to adapt the high-bandwidth acoustic models to the low-bandwidth environment using various techniques. The explanation for the legend tags which appear in the graph are;

- **NCHLT_LWAZI_BP** - Acoustic models trained on all NCHLT data, ASR system tested on Lwazi data and datasets limited in bandwidth to 250-3400 Hz.
- **LWAZI_LWAZI_BP** - Acoustic models trained on all Lwazi data, ASR system tested on Lwazi data and datasets limited in bandwidth to 250-3400 Hz.
- **NCHLT_LWAZI_MLLR_BP** - Acoustic models trained on NCHLT data and MLLR adapted using increasing amounts of Lwazi data, ASR system tested on Lwazi data and datasets limited in bandwidth to 250-3400 Hz.
- **NCHLT_LWAZI_MAP_BP** - Acoustic models trained on NCHLT data and MAP adapted using increasing amounts of Lwazi data, ASR system tested on Lwazi data and datasets limited in bandwidth to 250-3400 Hz.
- **LWAZI_RETRAIN_BP** - Acoustic models trained gradually increased amounts of Lwazi data, ASR system tested on Lwazi data and datasets limited in bandwidth to 250-3400 Hz.

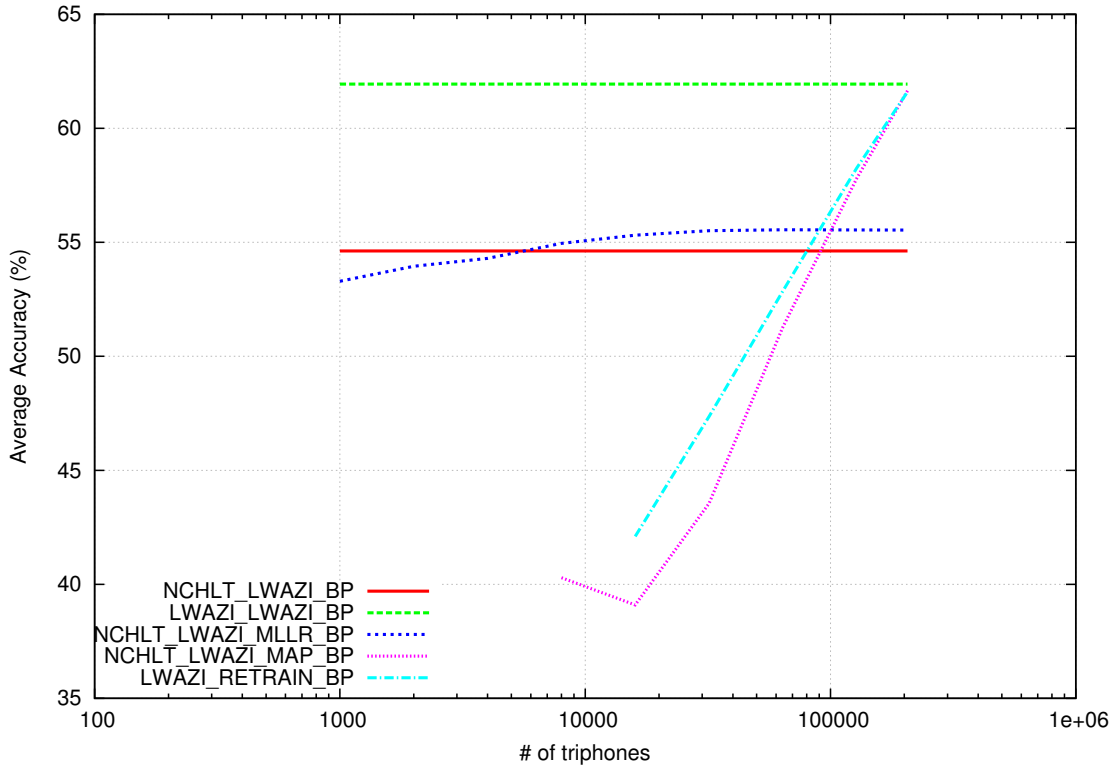

Figure 4.3: *The average accuracies obtained using various adaptation methods to port high-bandwidth (NCHLT) acoustic models to low-bandwidth (Lwazi) telephonic environment.*

As can be seen in figure 4.3, unexpectedly, the MLLR (mean and variance) initial performance is worse than applying no model adaptation, which implies that the limited adaptation data does not generalize well. For a triphone count between 6000 to 9000 the MLLR starts producing a performance gain but saturates relatively quickly around 12000 triphones. As with the WSJ-NTimit results the retrained acoustic models out-perform the MAP adapted models. The retrained acoustic models start to produce better results around 70000-80000 triphones.

Figure 4.4 shows the average performance gains, as the adaptation data amount is systematically increased and used to adapt the low-bandwidth acoustic models to high-bandwidth environment. The legend tags have the following meaning;

- **LWAZI_NCHLT_BP** - Acoustic models trained on all Lwazi training data, recognition performed on NCHLT testing data and both datasets band limited to 250-3400 Hz.
- **NCHLT_NCHLT_BP** - Acoustic models trained on all NCHLT training data, recognition performed on NCHLT testing data and both datasets band limited to 250-3400 Hz.
- **NCHLT_NCHLT_16k** - Acoustic models trained on all NCHLT training data, recognition performed on NCHLT testing data and both datasets used original 16 kHz sampled data.
- **LWAZI_NCHLT_MLLR_BP** - Acoustic models trained on Lwazi training data and MLLR adapted using increasing amounts of NCHLT training data, recognition performed on NCHLT

data and both datasets band limited to 250-3400 Hz.

- **LWAZI_NCHLT_MAP_BP** - Acoustic models trained on Lwazi training data and MAP adapted using increasing amounts of NCHLT training data, recognition performed on NCHLT data and both datasets band limited to 250-3400 Hz.
- **LWAZI_NCHLT_MAP_16k** - Acoustic models trained on Lwazi training data and MAP adapted using increasing amounts of NCHLT training data, recognition performed on NCHLT data. The Lwazi data was band limited and the NCHLT data was left at full bandwidth.
- **NCHLT_RETRAIN_16k** - The acoustic models were trained on increasing amounts of NCHLT training data and recognition was performed on NCHLT testing data. The full bandwidth of the NCHLT data was used.

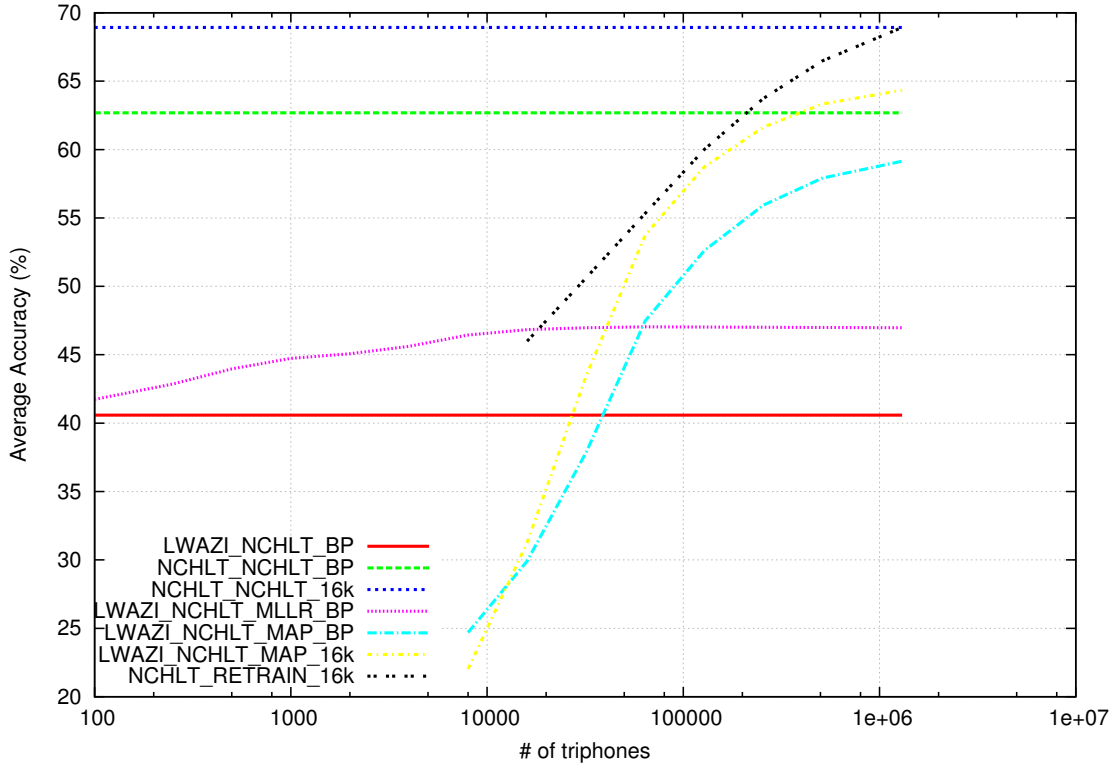

Figure 4.4: *The average accuracies obtained using various adaptation methods to port low-bandwidth (Lwazi) telephonic acoustic models to high-bandwidth (NCHLT) clean environment.*

Figure 4.4 is quite similar to the NTimit to WSJ transition experiments. At 100 triphone counts, MLLR provides a gain in performance and continues to produce the best gain in performance until a triphone count of around 18000, where the retrained models start providing the best accuracy. Again, the 16k MAP performs better than its band-limited counterpart but does not improve on the retrained models.

All the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for the curves can be found in appendix (B) , subsection (B.2).

4.4.5 MAP PERFORMANCE INVESTIGATION

To establish whether or not the perceived poor MAP adaptation performance was related to the choice of adaptation parameters, we ran a few MAP adaptation experiments using various parameter values. The aim was to find parameter combinations that would produce the best possible MAP performance for various adaptation data amounts. The MAP training procedure has two degrees of freedom; (1) τ the prior information weight and (2) the number of iterations used to update the model parameters. To aid in the investigation of the effect of adaptation parameter choice we chose to MAP adapt the band-limited NTimit acoustic models using 16 kHz WSJ data. For τ we choose values around the value of 10 as suggested by HTK [10], $\tau \in [5, 10, 20]$ – the smaller the value the less weight is given to the prior information. Then, for each τ value we ran experiments using a set of iterations chosen to be $[1, 2, 3, 5, 10, 20]$. For each iteration count we adapted the acoustic models using various data amounts; the selected triphone counts were $[1000, 2000, 4000, 8000, 16000, 32000, 64000, 128000, 256000, 512000]$. Again, each experiment was repeated five times to obtain an average accuracy value.

The band-limited NTimit acoustic models were MAP adapted and retrained on high-bandwidth WSJ 16kHz data and the MLLR adaptation was estimated on band-limited WSJ data. These choices were based on results presented in section (4.4.3) for the low-bandwidth to high-bandwidth scenario. The MLLR technique produced improved accuracies when estimating the transforms on band-limited audio data and was not effective in reducing the mismatch when trying to match band-limited telephone-quality trained acoustic models to high-bandwidth data. Similarly, the MAP adaptation approach produced improved accuracy gains when adapting band-limited telephone-quality acoustic models to high-bandwidth testing data.

Table 4.12 shows the acoustic model accuracies when MAP adapting using 16 kHz WSJ data, MLLR adapting using BP WSJ data and retraining the acoustic models on 16 kHz WSJ data. The accuracy values shown in column one (MAP 16 kHz WSJ) are the best accuracy values obtained for the specific triphone counts over all combinations of informative prior weight τ, iteration counts and triphone counts. For a full listing of the results refer to tables A.1, A.2 and A.3 which can be found in the appendix (A).

The only win observed for MAP is at 32000 with an accuracy of 61.01 % which is marginally better than that of the retrained model accuracy at 60.9 %. For lower triphone counts MLLR is the best adaptation option and as the data amount increases, retraining the acoustic models produces the best results. This suggests that MAP might possibly be the best method for a narrow band of triphone counts in our experimental condition. This small gain, however, does not justify the parameter searching required, and could very well be sampling noise.

Table 4.12: *Comparison of the accuracies obtained using MAP adaptation, MLLR adaptation and retraining the acoustic models.*

Triphone count	Adaptation Method		
	MAP 16 kHz WSJ	RETRAIN 16kHz WSJ	MLLR BP WSJ
1000	22.64	-	44.92
2000	29.44	34.46	46.23
4000	36.86	43.81	47.47
8000	45.41	48.92	48.94
16000	54.01	54.26	49.52
32000	61.01	60.9	-
64000	66.47	66.86	-
128000	70.73	71.85	-
256000	73.29	76.43	-
512000	74.82	80.45	-

4.5 CONCLUSION

In this chapter we analysed the performance gains afforded by the use of several standard feature normalisation and model adaptation techniques for adapting between narrow-band and wideband speech corpora. The feature normalisation approaches investigated were cepstral mean normalisation (CMN), cepstral mean and variance normalisation with arma filtering (MVA) and a novel transfer-function filtering normalisation. Amongst the model adaptation techniques, we evaluated maximum likelihood linear regression (MLLR) for mean and variance adaptation and maximum a-posteriori (MAP) adaptation of the weights, means and variances. The main conclusions that may be drawn from the work are:

- A large performance gain can be achieved if the bandwidths of varying sources of data are properly matched. This is intuitive as we should extract speech information from relevant portions of the spectrum.
- CMN performed comparably to MVA for the WSJ-NTimit experiments. The only real benefit provided by MVA was the total number of utterances that were decoded for severely mismatched conditions.
- The novel transfer-function filtering feature normalisation approach performed comparably to MLLR for low adaptation counts but the observed gains plateau quickly as more data was added. Other benefits of the transfer-function normalisation method are that it does not require transcriptions to perform the normalisation and can be applied independently of the various model-based adaptations.
- For low adaptation data amounts MLLR provides the best accuracy gain.
- MLLR works well in reducing the mismatch for bandwidth matched adaptations but failed to achieve ASR system accuracies when transforming band limited acoustic models to full bandwidth models (16kHz).

- As the adaptation data count approaches 10000 to 100000 triphone examples, retraining the acoustic models becomes a viable option – out-performing MLLR and MAP.
- Around the 10000 to 100000 adaptation triphone count MAP starts to perform better than MLLR but never beats the retraining the acoustic models.
- Our findings are in agreement with many results in the literature, but also in conflict with some other findings; this emphasised the fact that some of the strengths and weaknesses of the various adaptation techniques depend on the particular use case (e.g. speaker adaptation vs. dialect adaptation vs. channel adaptation). The main contribution of the current chapter is to arrive at a consistent picture of the behaviour that can be expected for the specific case of adaptation between low- and high-bandwidth applications. We believe that this picture will be particularly useful for system developers in the developing world, who are likely to be confronted with this scenario in practice.

We have demonstrated the efficiency of feature normalisation and model adaptation techniques to reduce the mismatch between telephone-quality and high-bandwidth speech audio. To obtain the best results for channel mismatched scenarios one should employ bandwidth matching, MVA feature normalisation, apply MLLR mean and variance transformation at relatively low adaptation data amounts and after 10000 triphone training examples, retrain the acoustic models on data sourced from the operating environment.

Similar to previously published work we have seen MLLR provide the best adaptation for low data amounts but the observed gains become saturated relatively quickly as more data is added. At this saturation point MAP adapting and retraining the acoustic models become better adaptation options. For channel and environmental adaptation, retraining the acoustic models provides better results compared to MAP adaptation. This is contrary to the speaker-adaptation task where the channel and environment characteristics are similar and the only substantial difference is the triphonic content and speaker characteristics. In this case MAP has a much greater window of data amounts where it is the best adaptation option.

CHAPTER FIVE

EFFICIENT DATA SELECTION FOR ASR

5.1 INTRODUCTION

Current state-of-the-art speech recognition systems use HMMs to model speech acoustic event sequences. The models capture statistical information, which relates the observed acoustic event sequences to hidden unit sequences such as words or phones as well as temporal acoustic event structure. The statistical modelling by the HMM makes it reliant on the observed data, and as was mentioned in section (2.5), we would like to minimize the amount of data that is required to obtain a specified level of accuracy, in order to make the creation of speech corpora in under-resourced languages as efficient as possible.

As summarized in section (2.5), Moore [48] found that there is a logarithmic relationship between Word Error Rate (WER) and training data amount, which implies that simply adding data at random is a slow method of increasing ASR system performance and theoretically vast amounts data are needed to drastically reduce the WER. Wu *et. al.* [7] and Nagorski *et. al.* [47] showed that it is possible to select a subset of the data to achieve ASR performance comparable to using all the data. A uniform selection criterion (maximum entropy principle) can in some cases enable ASR system performance comparable to that of systems trained on much larger datasets [7]. However, the uniform selection criterion is somewhat ad-hoc: it does not take the data's unit distributions into account and does not work in all situations (as shown in [50]). It is therefore doubtful that the training strategy proposed in [7] produces an optimal system performance in general. If one had access to a target unit distribution, the KL-divergence metric could be used to select utterances from a larger corpus and create a training distribution which matches the testing distribution [50]. In practice, however, the test distribution is not known or is specialized to a specific task which will not generalize well.

One aspect, however, shared by all ASR systems and independent of ASR configuration, is the relationship between the occurrence of training units in a training data set and the accuracy the unit

achieves in the final evaluation. To our knowledge no-one has based the selection criteria on the relationship between units' accuracies and their occurrence in the training data. From this, the goal of our research is to

- Develop a theoretical framework which guides a unit selection process based on the relationship between the number of training occurrences and resulting accuracy with the goal of improving the final ASR performance.
- Create an implementation of the theory for validation purposes.

The main purpose of our work is corpus design. Creating a corpus from scratch is a resource-intensive process, including such tasks as prompt design, data collection, validation, packaging and logistical management. Using data selection criteria to improve ASR accuracy, can improve the effectiveness of the corpus design and contribute to efficient data collection which is a necessity in resource constrained environments. However, for our initial investigation into optimal data selection we will limit ourselves to the evaluation of proposed techniques on existing corpora to establish the validity of the theory and implementation.

In section (5.2) we describe the unit selection theory and implementation strategy. The experimental corpora and setup are described in section (5.3). Our results are provided in section (5.4) and final remarks are captured in section (5.5).

5.2 THEORY

5.2.1 ASR TRAINING UNITS

Currently, ASR systems use HMMs to model phones by representing a phone as a sequence of states. Phonetic context has a significant effect on the acoustic realisation of a phone [68], which is most pronounced at the transitions between phones. To better model the phonetic context effect, multi-state HMMs are used, where each state models a part of the phone. In this case, however, the initial and final state probability density functions (PDF) generally have much larger variances, compared to the centre states, due to the phone transitions [68]. To improve state PDF modelling acoustic units larger than phones are used. Schwartz *et. al.* [68] proposed the *triphone* to approximately model all possible phone contexts – a triphone is created by taking into account the left and right phones surrounding a specific phone. In effect a triphone models the centre phone conditioned on the adjacent phones and improves the modelling of acoustic effects introduced by phone contexts [68]. Triphones serve as the current standard sub-word modelling unit in ASR systems since context-dependent phone modelling improves performance drastically.

One drawback of introducing triphones is data scarcity since there is not enough data to model certain triphones robustly. The standard approach to overcome limited data is tree-based state tying [23], where HMM states are shared amongst related HMMs, which reduces the number of states and increases the data amount per state – initially all states are pooled into a single root node, then a

process of node splitting is followed which partitions the states using phone-context questions. The question which results in the greatest log-likelihood gain is used to split the node.

5.2.2 TRIPHONE TRAINING UNIT

The triphone unit is a complex entity which inherently has a context relationship with surrounding triphones and a non-linear relationship to other triphones through the tied-state sharing strategy. So can the individual triphones be used when modelling the ASR system accuracy? To investigate the suitability of using triphones in modelling an ASR system's performance, two relationships are likely to play an important role:

- The correlation between the accuracies of a triphone and the triphones immediately adjacent to it – this correlation is caused by local contextual relationships, and is not directly useful for corpus design.
- The correlation between a triphone's accuracy and the number of times the triphone occurred in the training corpus, which is a logical consequence of the relationships reported by Moore [48], and could be used to select triphone occurrence counts in order to design ASR corpora.

To investigate these correlations, the *Pearson product-moment correlation coefficient* and *Spearman's rank correlation coefficient* were used. The Pearson correlation coefficient indicates the linear dependence between variables while Spearman's rank correlation coefficient indicates the non-linear dependence between variables. For both correlation tests, the coefficient's value ranges from -1 to 1, which shows a highly negative correlation and highly positive correlation respectively. The Pearson correlation coefficient r is estimated using,

$$r = \frac{\sum_i (X_i - \overline{X})(Y_i - \overline{Y})}{\sum_i (X_i - \overline{X})^2 \sum_i (Y_i - \overline{Y})^2}, \tag{5.1}$$

where X_i and Y_i are raw scores and $\overline{X}$ and $\overline{Y}$ are the associated raw score means. The Spearman correlation coefficient ρ is estimated using,

$$\rho = \frac{\sum_i (x_i - \overline{x})(y_i - \overline{y})}{\sum_i (x_i - \overline{x})^2 \sum_i (y_i - \overline{y})^2}, \tag{5.2}$$

where x_i and y_i are the ranks of the respective raw scores X_i and Y_i, and, $\overline{x}$ and $\overline{y}$ are the average rank values. To interpret the correlation coefficient values we make use of the guidelines set out in table 5.1 and table 5.2. Table 5.1 shows the interpretations for the Pearson product-moment correlation coefficient value strengths and table 5.2 shows the interpretations for the Spearman's rank correlation coefficient value strengths.

To compute the Pearson and Spearman correlation coefficients, we used the Matlab function *corr* [69]. When calculating the statistical significance of correlation coefficients the function makes use of the Student's t distribution for Pearson estimations while for the Spearman estimates, exact

Table 5.1: *Interpretations for the various Pearson product-moment correlation coefficient strengths. Adapted from [1].*

Pearson coefficient r range	Interpretation
+0.70 or higher	very strong positive relationship
+0.40 to +0.69	strong positive relationship
+0.30 to +0.39	moderate positive relationship
+0.20 to +0.29	weak positive relationship
+0.01 to +0.19	no or negligible relationship
-0.01 to -0.19	no or negligible relationship
-0.20 to -0.29	weak negative relationship
-0.30 to -0.39	moderate negative relationship
-0.40 to -0.69	strong negative relationship
-0.70 or higher	very strong negative relationship

Table 5.2: *Interpretations for the various Spearman's rank correlation coefficient strengths. Adapted from [2].*

Spearman coefficient ρ range	Interpretation
+0.9 to +1.0	very strong positive relationship
+0.7 to +0.89	strong positive relationship
+0.5 to +0.69	moderate positive relationship
+0.3 to +0.4.9	moderate to low positive relationship
+0.16 to +0.29	weak to low positive relationship
+0.16 to -0.16	no or negligible relationship
-0.16 to -0.29	weak to low negative relationship
-0.3 to -0.49	moderate to low negative relationship
-0.5 to -0.69	moderate negative relationship
-0.7 to -0.89	strong negative relationship
-0.9 to -1.0	very strong negative relationship

permutation distributions are used for small sample sizes and large-sample approximations for large sample sizes. The returned P-Values are calculated by doubling the more significant of the two one-tailed tests.

5.2.3 TRIPHONE CORRELATION INVESTIGATION

5.2.3.1 EXPERIMENTAL SETUP

To investigate the correlation aspects of the triphones two ASR systems were created. The first was trained on a portion of the American English Broadcast News (BN) corpus, where audio segments were selected if they were marked as high-fidelity and read speech. The second system, was trained on an inflated American English read-speech Wall Street Journal (WSJ) corpus which was created by adding all the training, development and evaluation data into a single training corpus. The WSJ sub-corpus that was used was only a portion of the entire WSJ corpus and was sourced from data corpus labelled "The Continuous Speech Recognition Wall Street Journal Phase I (CSR-WSJ0) Corpus".

Table 5.3 shows the number of hours of data for the BN and WSJ corpora.

Table 5.3: *The number of hours of audio data for the BN and WSJ corpora used to investigate triphone correlation aspects.*

Corpus	# hours
WSJ	62.8
BN	21.7

The ASR systems trained on either the BN and WSJ data used three tied-state HMMs to model triphones with eight Gaussian mixtures per state as well as training 40-class semi-tied transforms. 39-dimensional Mel-Frequency Cepstral Coefficients (MFCC) were created by extracting 13 static coefficients and appending delta and double delta coefficients. Speaker-based Cepstral Mean Normalisation was applied to the cepstral vectors. Speaker-based normalisation was used to keep consistency throughout our experiments since for later experiments some corpora had short one word utterances which is not enough data to estimate reliable mean vectors on a per utterance basis.

5.2.3.2 CALCULATING TRIPHONE ACCURACY

Triphone accuracies were calculated from deletion and insertion balanced phone-level recognition outputs. The BN-trained system was used to recognise all the WSJ data and similarly the WSJ-trained system was used to recognise all the BN audio data. The phone outputs were expanded to triphones and all silence models were removed – silence and short-pause. The alignments had to be processed further by altering triphone contexts around silence markers which did not occur at the start and end of an utterance. This involved removing the silence markers from the triphone name and inserting the appropriate phone name. To calculate a specific triphone's recognition accuracy, a correct accumulator, an error accumulator and a total accumulator are created for the triphone and set to zero. Then for each utterance, a triphone's accuracy was calculated as follows:

- The triphones from the recognised output were pooled together. (Our preliminary results showed a very slight difference in triphone accuracies and errors when comparing the pooled method and DP alignments)
- The corresponding utterance from the reference set (containing the true triphone sequence), was selected.
- The total accumulator for each triphone in the reference set was incremented.
- Each triphone appearing in the reference set, was selected and a check was performed to see if it occurred in recognition output triphone pool.
- If the triphone was present, the correct accumulator was incremented. The triphone was then removed from the pool.

- If the triphone was NOT present, the associated error accumulator was incremented.
- If any triphones were left in the recognition output pool, after processing all the triphones from the reference set, then each associated triphone error accumulator was incremented.

To calculate the final triphone accuracy, the final correct accumulator was divided by the sum of the final total and final error accumulators.

5.2.3.3 TRIPHONE CORRELATIONS

The first question to be addressed is: what correlation exists between a triphone's accuracy and the accuracies of the triphones immediately adjacent to it? To measure this correlation, a two-column table containing triphone accuracy pairs was set up. A pair was created as follows: if a triphone sequence was $t_A\ t_B\ t_C$ then the first pair would be $t_A\ t_B$ and the second pair would be $t_B\ t_C$. Table 5.4 shows the Pearson and Spearman correlation coefficients which measured the correlation between a triphone's accuracy and the accuracies of triphones immediately adjacent to it for both the BN and WSJ ASR systems as well as the statistical significance P-Values. The values in table 5.4 show that for both systems and both correlation tests, there exists a low to weak correlation between a triphone's accuracy and the accuracies of the triphones immediately adjacent to it.

Table 5.4: *The Pearson and Spearman correlation coefficients, and the associated P-values, which measured the correlation between a triphone's accuracy and the accuracies of triphones immediately adjacent to it for both the BN and WSJ ASR systems.*

Training Corpus	Pearson	P-Value	Spearman	P-Value
WSJ	0.2272	0.00E+00	0.2561	0.00E+00
BN	0.2249	0.00E+00	0.2540	0.00E+00

Table 5.5 shows the Pearson and Spearman correlation coefficients, and associated statistical significance test P-Values, which measured the correlation between a triphone's accuracy and the accuracies of triphones two positions away from it for both the BN and WSJ ASR systems. For this test, a similar two column table was constructed but instead the pairs were created as follows: given a triphone sequence $t_A\ t_B\ t_C\ t_D\ t_E$, pairs $t_A\ t_C$ and $t_C\ t_E$ are inserted as tables entries. The values in table 5.5 show that for both systems and both correlation tests, an insignificant correlation exists between these triphone accuracies. This implies triphone errors fade quickly and that triphones two positions apart have independent accuracies, to a good approximation.

The second relationship which was investigated was the correlation between a triphone's accuracy and the number of times the triphone occurred in the training data. To measure the correlation a two column table was set up which contained a triphone's accuracy and the training occurrence count. Table 5.6 shows the Pearson and Spearman correlation coefficients for WSJ and BN ASR systems and the statistical significance test P-Values. The Pearson values show that there is a weak to moderate linear relationship between triphone accuracy and training count. The Spearman measure shows that

Table 5.5: *The Pearson and Spearman correlation coefficients, and the associated P-Values, which measured the correlation between a triphone's accuracy and the accuracies of triphones two positions away from it for both the BN and WSJ ASR systems.*

Training Corpus	Pearson	P-Value	Spearman	P-Value
WSJ	0.0253	0.00E+00	0.0340	0.00E+00
BN	0.0243	0.00E+00	0.0315	0.00E+00

there is a strong non-linear correlation between the two entities, which is compatible with Moore's [48] findings of a logarithmic relationship between ASR system accuracy and the amount of training data.

Table 5.6: *The Pearson and Spearman correlation coefficients, and the associated P-Values, which measured the correlation between a triphones accuracies and the number of times the triphone occurred in the training set for both the BN and WSJ ASR systems.*

Training Corpus	Pearson	P-Value	Spearman	P-Value
WSJ	0.2888	0.00E+00	0.7604	0.00E+00
BN	0.3924	0.00E+00	0.7809	0.00E+00

Based on the correlation results, triphones seem to be a good candidate unit to model an ASR system's performance as there is a strong non-linear correlation between an triphone's accuracy and the number of times the triphone occurred in the training data. There is a weak to low correlation between adjacent triphones accuracy which a model could take into account for improved modelling, but for the development of our approach we assume triphone accuracy independence which should not severely impact the triphone accuracy modelling.

5.2.4 TRIPHONE TYING

Generally, for current ASR systems there is insufficient data to train all the triphones robustly and thus state tying is used to share data amongst similar states (see section (2.2.4)). Throughout our experiments we use the question-tying scheme introduced by Young [23] which has the benefit of allowing unseen triphones. The questions we defined are basic and are questions based on the left and right contexts of the individual monophones – no additional groupings (e.g. broad class such such as nasal or vowel) are created. All the triphones which have the same central monophone are pooled and used to create the shared states. The way in which states are tied, however, is highly non-linear and physical triphones may share data from a variety of other triphones. In addition, the triphones' states may themselves share data from different triphones and not in equal counts – for instance triphone A's state 2 may share data with 5 triphones while state 3 shares data from 3 triphones. Therefore it becomes difficult to assign a training count to a triphone given state tying. For the scope of our work we do not delve into the assignments made by state-tying and treat the triphones as distinct units and derive the triphone counts from the training data.

5.2.5 FRAMEWORK

The fundamental modelling unit for current ASR systems are tied-state triphones. As motivated above, it is reasonable to presume that the overall ASR system performance is related to the triphone recognition accuracy – words are recognised from monophone sequences and monophone sequences are extracted from recognised triphone sequences. Thus, word accuracies are related to monophone accuracies which are related to triphone accuracies. Therefore, our first assumption is that the overall ASR system's accuracy is related to the individual triphone accuracies.

In section (5.2.3) we showed that there exists a strong correlation between a triphone's accuracy and the number of times the triphone appeared in the training data. In addition, the correlation between the accuracies of adjacent triphones is low to weak. Thus, secondly, we assume that a triphone's accuracy is primarily determined by the number of times it occurs in the training data only and for the scope of this work we ignore adjacency effects. Given these two assumptions, we can mathematically formulate the overall system performance as,

$$A_{total} = \sum_{i=1}^{N} p_i A_i(n_i), \tag{5.3}$$

where A_{total} is the overall system accuracy, p_i is the probability of occurrence for triphone *i* and A_i is the i^{th} triphone accuracy dependent on the occurrence count. Thus, equation (5.3) states that the overall system accuracy is given by the sum over all individual triphone accuracies multiplied by the probabilities of their occurrence.

When collecting data in resource-scarce environments, there are limited resources with which to collect data. Thus, the collected corpus will contain a limited amount of data determined by the resource investment. Therefore, to represent this resource constraint we introduce a corpus design constraint, which limits the total triphone count to a specified number. To enforce this constraint, we introduce a Lagrange multiplier into equation (5.3) and rewrite it as,

$$A_{total} = \sum_{i=1}^{N} p_i A_i(n_i) + \lambda(\sum_{i=1}^{N} n_i - N), \tag{5.4}$$

where λ is the Lagrange multiplier, n_i is the i^{th} triphone count and N is the total triphone count in the training corpus. Given our equation that describes the ASR system accuracy we would like to find the optimal assignment of triphone training counts which improves the ASR system's accuracy. In order to find the optimal triphone counts, we need to calculate the first derivative of equation (5.4) and set it equal to zero,

$$\frac{\partial A_{total}}{\partial n_i} = 0, \tag{5.5}$$

which will provide the optimal assignment of the triphone counts and maximise the overall system accuracy A_{total}. Working through the derivation we obtain,

$$\frac{\partial A_{total}}{\partial n_i} = p_i \frac{\partial A_i(n_i)}{\partial n_i} + \lambda, \tag{5.6}$$

where p_i is the probability of triphone occurring, $\frac{\partial A_i(n_i)}{\partial n_i}$ is the derivative of an individual triphone accuracy with respect to its training count and λ is the Lagrange multiplier introduced by the constraint $\sum_{i=1}^{N} n_i = N$. $\frac{\partial A_i(n_i)}{\partial n_i}$ and λ are unknown and need to be calculated.

Once these values have been calculated it becomes an easy task to solve for the triphone counts n_i. The most difficult part in solving the above equation (5.6) is finding a suitable expression for the individual triphone accuracies $A_i(n_i)$. Given that we have to calculate the derivative of $A_i(n_i)$, it would be convenient to find a suitable functional form which would avoid the use of numerical derivative techniques.

5.2.6 SELECTING AN ACCURACY FUNCTION

In the previous section we have derived an expression for the optimal number of triphones to include in a corpus in order to maximise an ASR system's accuracy, summarized in equation (5.6). To solve for the optimal triphone count we must have a suitable triphone accuracy function which relates an individual triphone's accuracies to the number of times the triphone occurred in the training set. Here, we consider two theoretical distributions as well as two empirical distributions, and select a compromise that will allow us to investigate the potential of our approach.

- There are simple arguments from learning theory [70] that suggest an asymptotic functional relationship of the form

  $$A_i(n_i) = B - \frac{C}{n_i}, \tag{5.7}$$

  with constants B and C again problem-specific and algorithm-specific parameters, and with the relationship only expected to be valid for large n_i. Figure 5.1 shows a plot of equation (5.7) which relates triphone accuracy to triphone count and where we have assigned the values $B = 100$, $C = 1000$. As expected, a triphone's accuracy is initially low but increases rapidly as more data is added, reaching a plateau when n_i reaches the same order of magnitude as C. This roughly coincides with ASR systems' behaviour which generally shows a benefit when data amount is steadily increased but eventually the observed improvement diminishes.

  Using equation (5.6), substituting the derivative of equation (5.7) and setting the results to zero we can rearrange the equation to obtain an expression for the triphone counts n_i. The steps followed are,

  $$\frac{\partial A_{total}}{\partial n_i} = \frac{p_i C}{n_i^2} + \lambda, \tag{5.8}$$

  $$\frac{p_i C}{n_i^2} + \lambda = 0, \tag{5.9}$$

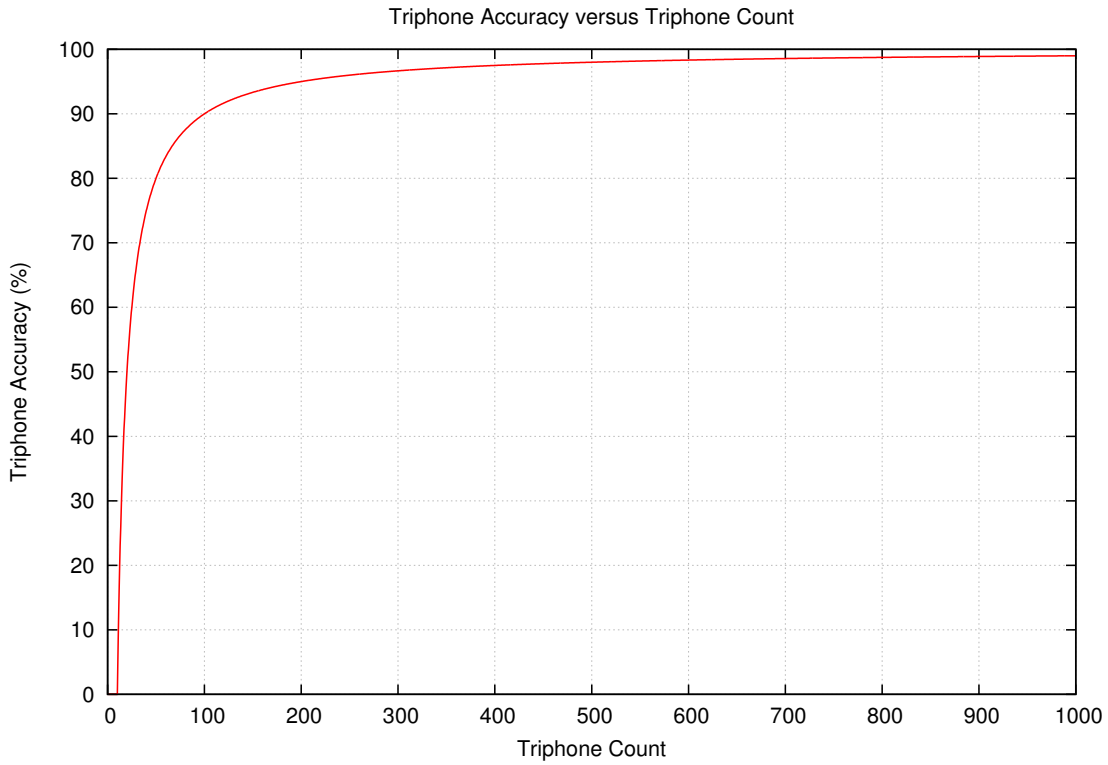

Figure 5.1: *The hypothetical asymptotic accuracy function which describes the triphone accuracy given the triphone count.*

$$n_i = \sqrt{\frac{p_i C}{-\lambda}}. \tag{5.10}$$

It would be more convenient to calculate the triphone prior which will give us a way of calculating the triphone counts independently of data size. Thus, we can rewrite the triphone counts as a function of the total triphone count as $n_i = q_i N$. Substituting this expression into equation (5.10), we obtain an expression for the triphone prior,

$$q_i N = \sqrt{\frac{p_i C}{-\lambda}}, \tag{5.11}$$

$$q_i = K\sqrt{p_i}, \tag{5.12}$$

where $K = N\sqrt{\frac{C}{-\lambda}}$. To solve for the prior q_i, which will be the optimal prior set, we use the constraint that the sum of the prior must equal 1, $\sum_{i=1}^{N} q_i = 1$, which implies that $K = \frac{1}{\sum_{i=0}^{N}\sqrt{p_i}}$. Finally, given the values of the initial training set triphone prior, we can calculate the optimal prior by taking the square root of the initial priors and then normalising their values such that they sum to equal one.

- On the other hand, Moore [48] produced a variety of evidence showing a logarithmic relation-

ship between the WER and the total amount of data used to train an ASR system. If we assume that the same trend holds for individual triphone accuracies, we obtain a relationship of the form

$$A_i(n_i) = B + C \log n_i, \tag{5.13}$$

with B and C parameters that describe the details of the logarithmic improvements suggested by Moore. Following the same steps as above, we find that this assumption requires that

$$q_i = p_i \tag{5.14}$$

in order to optimize ASR accuracy.

Thus, these two functional forms lead to data-selection approaches that range between "natural" selection (i.e. the selected triphone frequencies should match those that occur in the reference data) to "compressed" selection, where the selected frequencies are proportional to the square root of the occurrence frequencies. Which of these forms is most appropriate for speech data is an empirical question, and we investigate evidence from a few widely-studied corpora below.

5.2.7 TRIPHONE ACCURACY FUNCTION: EMPIRICAL EVIDENCE

In order to gain a better understanding of the relationship between accuracy and frequency, triphone accuracies were measured for the WSJ and BN corpora. The corpora, ASR systems and manner in which we calculated the triphone accuracies are highlighted in section (5.2.3).

Figure 5.2 (A) shows the average triphone accuracy as a function of triphone training occurrence estimated using a BN trained ASR system and recognising WSJ data, and figure 5.2 (B) shows the number of distinct triphones used to average the triphone accuracies. Similarly, figure 5.3 (A) shows the average triphone accuracy as a function of triphone training occurrence estimated using a WSJ trained ASR system and recognising BN data, and figure 5.3 (B) shows the number of triphones used to average the triphone accuracies. The average triphone accuracy figures (A) both show a logarithmic-style relationship between the triphone accuracy and the number of triphone training examples – ever increasing amounts of data are needed to improve the accuracy. These figures are relatively smooth to triphone counts of around 100, but start to fluctuate after this point. The fluctuation can largely be put down to the limited number of examples that are used to average the accuracies as shown in the (B) figures – the triphones which have high training counts usually only have one example with which to calculate an average accuracy. Hence, factors other than the triphone count (e.g. the inherent variability of a particular triphone) are excessively influential in those accuracies.

The triphone accuracy graphs shown in figures 5.2 (A) and 5.3 (A) are quite noisy and it is difficult to get a sense of what the underlying trends are above the 100 triphone training count. Therefore, to obtain a set of smooth figures a simple smoothing technique was employed – a moving average filter using 100 samples either side of each data point was used to calculate the smoothed accuracies. Figure

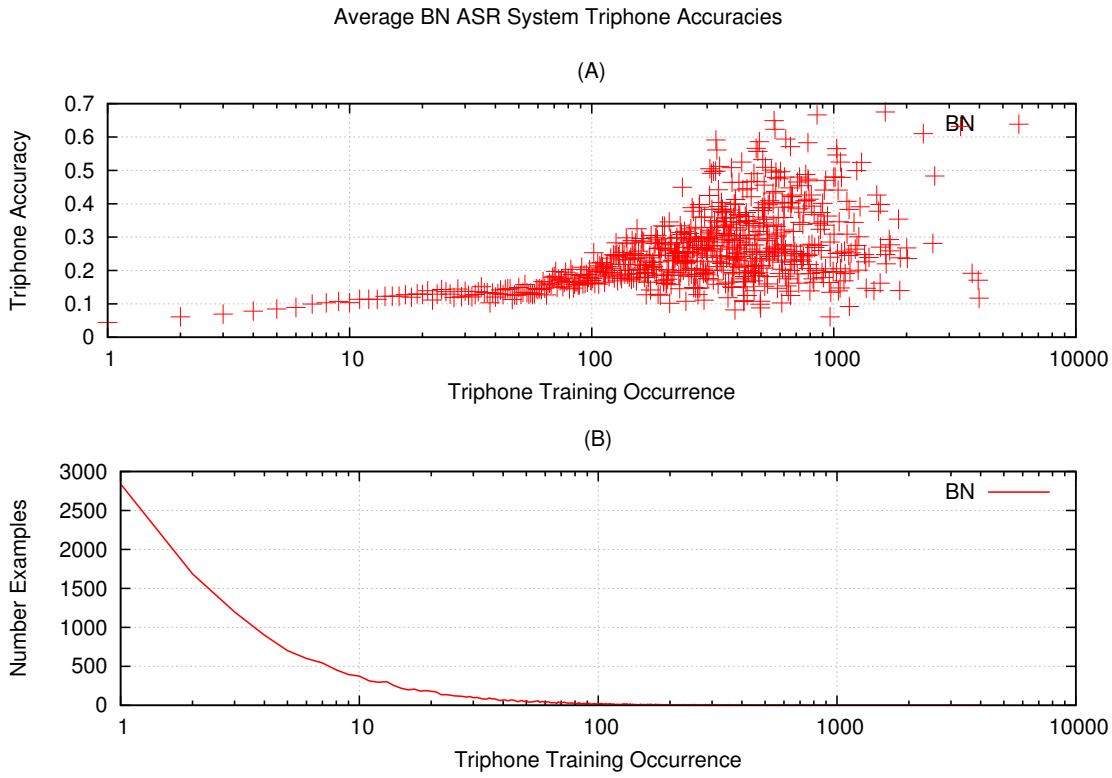

Figure 5.2: *Graph (A) shows BN-derived triphone accuracy as a function of triphone training count using the WSJ corpus as an evaluation set. Graph (B) shows the number of examples used to average the triphone accuracies.*

5.4 shows the smoothed graphs for the triphone accuracies as a function of triphone training count for both the BN and WSJ systems. Besides the artefact at the end of the graphs, where the triphone accuracies decrease, the general trend is a gentle increase in accuracies from counts 1 to about 20, then a rapid increase in accuracies from about 30 to 750 triphone count, and, finally a diminishing of the accuracy improvements above a count of 750.

These graphs are roughly compatible with both of the functional forms in the previous section, but in the large-n region, where we would hope to understand their differences in most detail, the graphs are too noisy for useful conclusions to be made. Fortunately, we can smoothly change from one form to the other by adjusting the exponent of p_i that is proportional to q_i: if it is 1.0, we obtain the optimal distribution for the logarithmic relationship, an exponent of 0.5 is optimal for the relationship derived from learning theory, and intermediate values of the exponent presumably correspond to intermediate relationships between accuracy and training count. (Another alternative is to model curves such as those in figure 5.2 and figure 5.3 in more detail; as we show in appendix (C) , such an approach does not seem promising in the current circumstances). Thus, we will set

$$q_j = \frac{p_i^r}{\sum_k p_k^r}, \tag{5.15}$$

where r is the compression factor. To obtain the target total number of training triphones, the

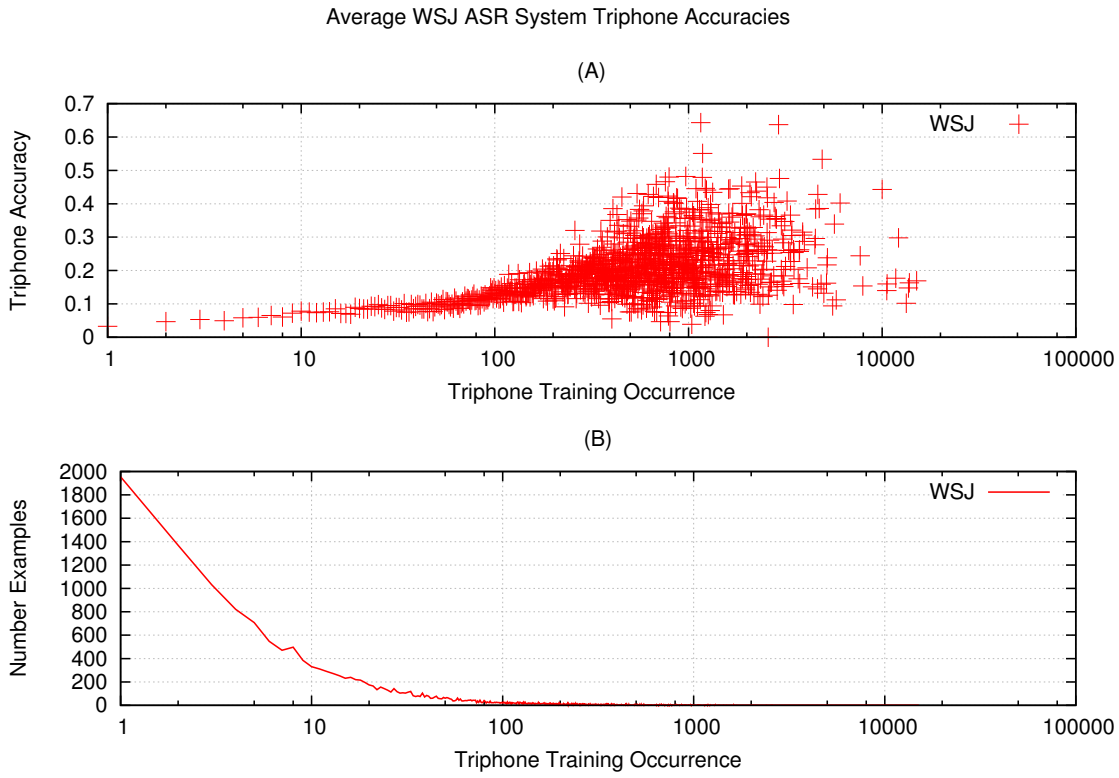

Figure 5.3: *Graph (A) shows WSJ-derived triphone accuracy as a function of triphone training count using the BN corpus as an evaluation set. Graph (B) shows the number of examples used to average the triphone accuracies.*

compressed and normalised probabilities of occurrence are multiplied by the total target count and rounded to remove fractional components.

5.2.8 GREEDY UNIT SELECTION

To select the target triphone count distribution we used a data selection approach similar to the regularised Kullback-Leibler divergence-based data selection proposed by Gouvea and Davel [50]. The regularisation is controlled by a user-specified constant. In their approach, the main goal was to select a set number of utterances (N), from a larger dataset (T), to match a target distribution of n-grams. The algorithm initialises a candidate subset by randomly selecting N utterances. For all utterances which are left in dataset T, an utterance (U) is selected and used to substitute, one at a time, all the utterances found in the candidate subset. For each substitution, the change in KL-divergence is measured. Once all possible substitutions have been made for the candidate subset utterances, the substitution which gives the greatest decrease in regularised KL-divergence is made.

Our data selection requires selecting a number of utterances which will produce the desired distribution and limit the number of training triphones to a set amount. Thus, we modified the Gouvea and Davel [50] approach in a number of ways. Our algorithm steps are:

- *Initialisation Stage*: The candidate subset is created by randomly selecting utterances and lim-

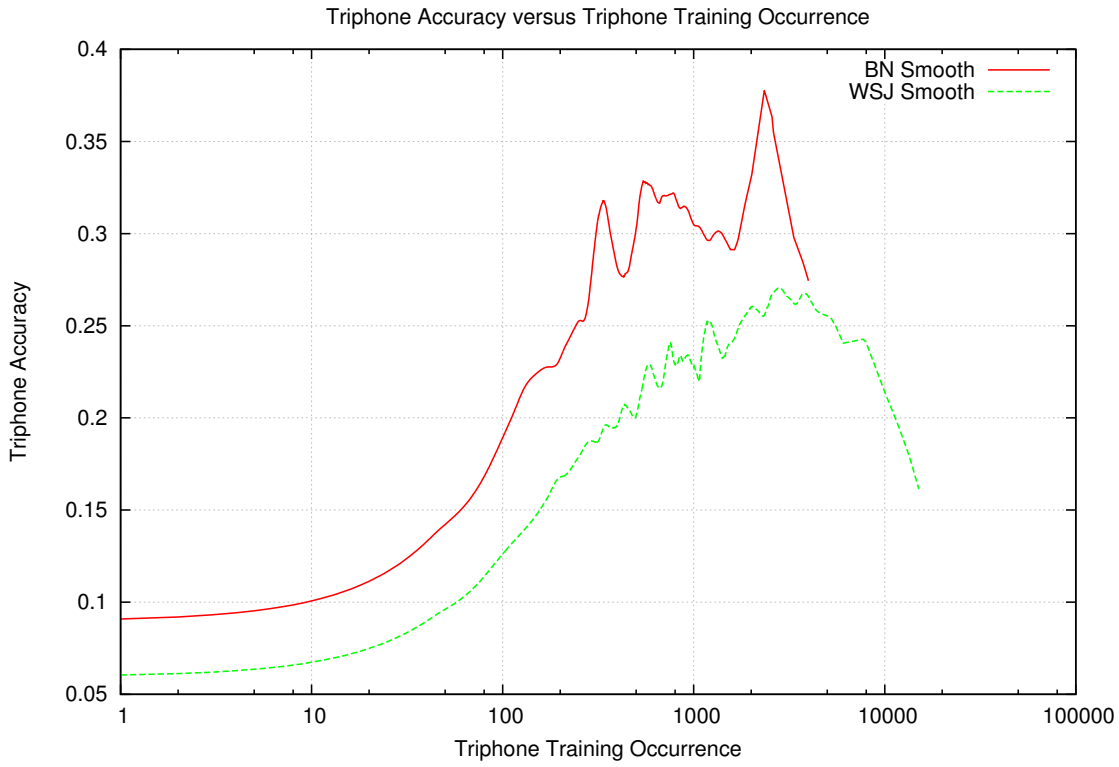

Figure 5.4: *Smoothed graphs showing triphone accuracy as a function of triphone training count for the BN and WSJ experiments.*

iting the number of utterances by the number of training triphones instead of a set utterance count.

- *Main Stage*: The main stage of the algorithm implemented an iterative two-phase selection process.
  - *Phase One Addition*: For the first phase, on a per utterance basis, each remaining training set utterance is added to the candidate subset, the KL-divergence measured, and then removed from the candidate subset. The utterance which results in the largest decrease in KL-divergence is added to the candidate subset.
  - *Phase Two Removal*: For the second phase, on a per utterance basis, each utterance in the candidate subset is removed, KL-divergence measured and the utterance placed back into the candidate subset. The utterance which results in the largest decrease in KL-divergence is removed.
- *Correction Phase*: The correction phase ensures that the candidate subset has the correct number of target triphones.
  - If the training triphone count is too high then, utterances are removed from the candidate subset, via Phase Two Removal, until the target count is reached.

– If the training triphone count is too low, utterances are added from the training set, via Phase One Addition, until the target count is reached.

5.3 EXPERIMENTAL SETUP

This section describes the corpora used in our investigations, data selections and experimental setups.

5.3.1 CORPORA

5.3.1.1 TIMIT

The Timit [71] corpus contains read-speech American English high-bandwidth audio recordings. The corpus contains 6300 utterances collected from 630 speakers each contributing 10 utterances. The speakers were selected from eight distinct dialect regions. The corpus has a 70-30 percent male-female gender split. The text prompts consisted of dialect, phonetically-compact and phonetically-diverse sentences. Two dialect sentences were read by all speakers and used to measure dialect differences. The phonetically-compact sentences were designed to cover phonetic pairs and each speaker spoke five sentences with seven speakers reading the same sentences. The phonetically-diverse sentences added phonetic diversity and were selected to maximise allophonic contexts. Each speaker read three phonetically-diverse sentences – unique to the specific speaker.

For our Timit experiments we removed the sentences read by all speakers, as their high frequency severely biases the corpus distribution and thus biases the results. Table 5.7 shows statistics for the reduced Timit corpus training and evaluation sets.

Table 5.7: *Timit corpus statistics with the dialect sentences removed.*

	Training	Evaluation
# utterances	3696	1344
# speakers	462	168
Duration (hours)	3.14	1.15

5.3.2 WALL STREET JOURNAL

The Wall Street Journal (WSJ) [57] corpus is a large American English corpus built to meet a few design criteria. The entire corpus contains a variety of audio and text, which accommodates various vocabulary sizes, language model perplexities, variable sized speaker-dependent and -independent training data amounts, read and spontaneous speech, verbalised and non-verbalised punctuations and differing recording environments. For our experiments we chose the speaker-independent read-speech training corpus with high-quality recordings, and the 5k vocabulary evaluation corpus. The text prompts were chosen from newspaper text. Similar to Timit we removed the speaker adaptation sentences. We only sourced WSJ data from "The Continuous Speech Recognition Wall Street Journal

Phase I" corpus. Table 5.8 shows statistics for the WSJ corpus training and evaluation sets, with the speaker-adaptation utterances removed.

Table 5.8: *WSJ corpus statistics with the speaker-adaptation sentences removed.*

	Training	Evaluation
# utterances	8734	1858
# speakers	101	8
Duration (hours)	18.76	4.38

5.3.2.1 LWAZI

The Lwazi [29, 72] corpus contains telephone quality recordings and their associated transcriptions covering the eleven official languages of South Africa. The read and elicited speech data was collected from approximately 200 speakers per language with each speaker contributing 30 utterances. A portion of the utterances were randomly selected from a phonetically balanced corpus and the remainder are words or short phrases. For our experiments we limited ourselves to the IsiZulu language spoken by the majority of South Africans. As the corpus does not contain dedicated training and evaluation sets, we split the corpus into ten folds. The folds were created by randomly partitioning the speakers into ten mutually exclusive sub-corpora, which served as the evaluation sets. The training sets were created by cycling through the evaluation folds and assigning all folds to the training set except for the current evaluation fold. Table 5.9 shows some statistics for the Lwazi IsiZulu sub-corpus by fold.

Table 5.9: *Corpus statistics for the ten randomly selected folds for the IsiZulu Lwazi corpus.*

	Training			Evaluation		
Fold	# utterances	# speakers	Duration (hours)	# utterances	#speakers	Duration (hours)
1	5189	179	8.29	596	20	0.88
2	5229	179	8.34	556	20	0.84
3	5189	179	8.22	596	20	0.95
4	5196	179	8.14	589	20	1.03
5	5228	179	8.30	557	20	0.88
6	5203	179	8.20	582	20	0.97
7	5213	179	8.23	572	20	0.94
8	5197	179	8.33	588	20	0.84
9	5203	179	8.23	582	20	0.94
10	5218	180	8.32	567	19	0.86

5.3.2.2 AST

The African Speech Technology (AST) corpus contains telephony-quality speech data for five South African languages [73]. The speech data was collected from over 200 speakers and the prompts were chosen to support services such as teleservice transactions, hotel booking applications and in-

formation retrieval. For our experiments we made use of the IsiZulu sub-corpus and selected the train, development and evaluation sets described in Niesler [74] but only utilised the evaluation set for our experiments. Table 5.10 shows some statistics for the AST IsiZulu sub-corpus by training, development and evaluation sets.

Table 5.10: *AST IsiZulu corpus statistics for the training, development and evaluation sets.*

	Training	Development	Evaluation
# utterances	8295	390	583
# speakers	199	10	16
Duration (hours)	7.03	0.28	0.45

5.3.3 DATA SELECTION

In section (5.2.6) we propose that the optimal data selection approach depends on the relationship between the unit's accuracy and count. To investigate our approach we define three optimal distributions: (1) "natural" selection (based on the logarithmic relationship), (2) "compressed" selection (based on learning theory) and (3) a combination of the two ("intermediate").

To produce an optimal distribution for the "natural" selection we choose utterances at **random** until a specified target total training triphone count was achieved. Throughout this chapter "natural" represents a **random** selection.

The optimal "compressed" distribution was created by:

- Estimating the triphone counts from a training utterance set.
- Calculating the triphone distribution by normalising the sum of the triphone counts to one.
- Applying the square-root operator to the triphone probabilities.
- Re-normalising the transformed triphone probabilities so that they sum to one.
- Multiplying the triphone probabilities by a target training triphone count and further normalising by rounding to the nearest integer.
- Using the KL-divergence selection algorithm (see section 5.2.8) to select the target distribution from the entire training utterance set.

To produce the "intermediate" optimal distribution, the steps which produce an optimal "compressed" distribution were followed, except the triphone probabilities are raised to a power of 0.75 instead of applying the square-root operator. It was found that the Timit, WSJ and Lwazi training corpora contained many utterance repetitions. Table 5.11 shows the number of utterances and unique utterances found in the Timit, WSJ and Lwazi training sets. Therefore, an additional investigation

was performed to determine the effect of estimating the triphone distributions on the unique sentences only but still selecting from all the training utterances to achieve the target training triphone distributions.

Table 5.11: *The total number of utterances and unique utterances found in the Timit, WSJ and Lwazi training sets.*

Corpus	# utterances	# unique utterances
Timit	3696	1731
WSJ	8734	5028
Lwazi	5786	3917

Lastly, to compare our data selection results with current selection techniques, the maximum entropy principle (max-entropy) selection was also used. We followed the max-entropy selection algorithm outlined in Wu *et. al* [7] and selected either word or triphone units. Their proposed greedy selection algorithm efficiently selects the required number of utterances by analysing the change in entropy if an utterance is added to the training pool: if the increase is above a certain threshold then the utterance is included in the training set. The chosen threshold determines the final size of the training set.

To distinguish amongst the various data selection methods, the following keys will be used for the remainder of the chapter (and relevant Appendices):

- `Natural` - "natural" data selection (random)
- `Sqrt` - "compressed" data selection
- `0.75` - "intermediate" data selection
- `MaxEnt Tri` - max-entropy selection based on triphone units
- `MaxEnt Wrd` - max-entropy selection based on word units
- `Uniq Sqrt` - "compressed" data selection using the unique utterance triphone distribution
- `Uniq 0.75` - "intermediate" data selection using the unique utterance triphone distribution

5.3.4 MATCHED-PAIRS SIGNIFICANCE TEST

To determine the statistical significance of the performance differences measured, we employed a matched-pairs statistical significance test described by Gillick and Cox [75]. Initially, the speech stream is partitioned into statistically independent segments where the segment can be sentences, speech occurring between speaker pauses or entire utterances. For our purposes we chose the entire utterance as the segments. Next, we count the number of errors, per segment, made by the two algorithms to be compared, N^i_{1or2}, where i is the segment number. In an ASR setup, the error is given by the sum of deletion, insertion and substitution errors. Given the error counts, we define a variable

$Z^i = N_1^i - N_2^{i,i=1,2,...,n}$, to be the difference in errors made in a segment and n is the total number of segments. If the algorithms perform similarly, the average difference in the number of errors made in a segment, μ_z would be close to zero, thus we would like to ascertain whether or not $\mu_z = 0$. To determine if μ_z is equal to zero, the test statistic W is defined as

$$W = \frac{\mu_z}{\frac{\sigma_z}{\sqrt{n}}}, \tag{5.16}$$

where the mean μ_z is given by

$$\mu_z = \frac{1}{N}\sum_{i=0}^{N} Z^i, \tag{5.17}$$

and the standard deviation σ_z is given by,

$$\sigma_z = \sqrt{\frac{1}{n-1}\sum_{i=1}^{N}(Z^i - \mu_z)^2}. \tag{5.18}$$

If n is large, we can make the assumption that W will approximately be normally distributed with unit variance. To set up the significance test, we define the null hypothesis as $H_0 : \mu_z = 0$ and the alternative hypothesis is defined as $H_1 : \mu_z \neq 0$.

To test the validity of the null hypothesis, we perform a two-tailed test by computing the P-Value $P = 2Pr(Z \geq \mid W \mid)$, where Z is a random variable described by a standard normal distribution - $\mathcal{N}(0, 1)$. The null hypothesis H_0 is rejected if $P < \alpha$, where α is the significance level generally set to $0.05, 0.01$ or 0.001.

5.3.5 ASR SYSTEMS

For all experiments we trained standard HMM-based ASR systems. Three state left-to-right HMMs (beginning and ending non-emitting states not counted) were used to model tied-state cross-word context-dependent triphones. Each HMM state contained eight mixture Gaussian models which modelled the state distributions. The state-tying questions were generated by creating left and right questions for each individual phone. The audio was encoded into MFCC vectors using a 25 ms window and shifting the window by 10 ms after encoding a frame. The MFCC vectors were 39 dimensional and were constructed by appending 13 static, 13 first derivative and 13 second derivative components. Speaker-based CMN was applied to each utterance. This standard HMM-based ASR setup was used throughout our experiments. The acoustic models were trained on audio data sourced either from the training corpus or the relevant cross-validation folds (for the Lwazi IsiZulu corpus).

5.3.6 TRAINING CORPORA

To test the various data selection approaches, we partitioned the various training corpora into fractional subsets and trained ASR systems on these sub-corpora. The data-selected fractional training

sub-corpora were generated by selecting a subset of training utterances which produced a specified percentage of the total number of triphones which made up the entire training set. The percentages used were 20%, 40%, 60%, and 80% e.g if a training corpus contained 100000 training triphones, then four sub-corpora were created that contained roughly 20000, 40000, 60000 and 80000 training triphones. In addition, for the "natural" and max-entropy selections, a growing selection strategy was utilised, which meant that the larger sub-corpora were created by using the previous smaller sub-corpus as a starting point and adding utterances to meet the larger training triphone counts i.e. 80% contains all 60% utterances, 60% contains all 40% utterances and 40% contains all 20% utterances. The Timit, WSJ and 10-fold Lwazi corpora training sets will be used to create the various sub-corpora.

5.3.7 PERFORMANCE MEASURES

The performance of the different ASR systems was measured using word correctness (Word Cor %) and word accuracy (Word Acc %) percentages [10]. If we define S as the number of substitution errors, D number of deletion errors, I the number of insertion errors and N the total number of labels in the reference transcriptions, then word correctness is given by

$$Correctness = \frac{N - D - S}{N} \times 100\%, \tag{5.19}$$

and, the word accuracy is given as,

$$Accuracy = \frac{N - D - S - I}{N} \times 100\%. \tag{5.20}$$

To measure the word correctness and accuracy, the evaluation sets were recognised using the acoustic models trained on the various data selections. The decoding network was built using a flat word-loop grammar and contained only the words which occurred in the evaluation set. To evaluate the statistical significance of the performance, the matched-pairs significance test was used as described in section (5.3.4). The "natural" results will serve as reference for the statistical significance tests and a significance level of 0.001 is chosen.

To verify any improvements brought about by the use of data selection methods were not merely achieved by matching training and evaluation distributions, independent evaluation corpora are used. This will ensure different triphone distributions for the training and evaluation sets. Specifically, the WSJ evaluation set will be used to validate Timit data selections, the Timit evaluation set will be used to validate WSJ data selections, and AST will be used for Lwazi data validation.

As our theory makes the assumption that the overall ASR system accuracy is given by a weighted sum of individual triphone accuracies we will also report triphone correctness and accuracy values as well as their statistical significance. The triphone results are derived from the word recognition outputs which are expanded to phone-level transcriptions which are further processed to form triphone labels.

5.4 RESULTS

In this section we present data selection results on three significantly different corpora: American English Timit, American English WSJ and IsiZulu Lwazi.

5.4.1 TIMIT

Table 5.12 shows word correctness, word accuracies and P-Value statistical significance for various ASR systems trained on percentages of selected Timit data and evaluated on the Timit evaluation set. At the 20% data percentage level the max-entropy selection methods produce significantly degraded performance compared to `Natural` selection, while the `Sqrt`, `0.75` and `Uniq` variants produce better accuracies but only the `Uniq Sqrt` approach provides a significant improvement. For the 40% level the max-entropy data selection methods produce slightly worse accuracies but the decreases in performance are insignificant compared to the `Natural` approach. The remaining data selection methods produce significantly better accuracies with `Uniq Sqrt` producing the best results when compared to natural selection. At the 60% level the `MaxEnt Tri` data selection approach produces a significant decrease in performance while the `MaxEnt Wrd` method attains an insignificant decrease with respect to natural selection. Again, the remaining selection methods achieve an increase in performance, except that the `0.75` and `Uniq Sqrt` techniques do not produce significant gains. For the last data percentage, 80%, both max-entropy and `0.75` techniques produce lower accuracies compared to natural selection with `MaxEnt Tri` producing a significant loss in performance. `Sqrt`, `Uniq Sqrt` and `Uniq 0.75` approaches attain better accuracies yet are insignificant. Overall, `MaxEnt Wrd` performs better than `MaxEnt Tri` yet not surpassing the `Natural` selection accuracies. The various compression and intermediate techniques, for the majority of cases, provide better performance compared to `Natural` and max-entropy based data selections.

Table 5.12: *Word correctness, word accuracies and P-Value results for Timit trained and Timit evaluated ASR systems using different data selection methods and percentages of the total training data.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	47.57	44.35	46.24	47.97	48.73	49.51	48.44
	Word Acc (%)	40.88	37	37.75	42.09	42	43.36	42.1
	P-Value	-	3.32E-09	3.31E-06	4.46E-02	7.42E-02	6.63E-05	4.78E-02
40%	Word Cor (%)	53.44	51.95	53.73	55.83	55.21	57.08	56.62
	Word Acc (%)	44.93	43.4	43.94	48.14	47.37	49.75	49.33
	P-Value	-	1.35E-02	1.41E-01	1.05E-07	4.48E-05	2.00E-15	8.94E-12
60%	Word Cor (%)	57.71	55.4	57.75	59.32	58.74	59	59.55
	Word Acc (%)	49.29	45.95	48.47	51.44	50.67	51.01	51.78
	P-Value	-	3.27E-08	1.86E-01	1.22E-04	1.56E-02	2.91E-03	1.15E-05
80%	Word Cor (%)	60.41	58.82	60.14	60.41	60.18	60.59	60.93
	Word Acc (%)	51.66	49.37	51.31	52.19	51.35	51.97	52.52
	P-Value	-	1.05E-04	5.47E-01	3.42E-01	5.75E-01	5.91E-01	1.28E-01

Table 5.13 presents triphone correctness, triphone accuracies and P-Value significances for Timit trained ASR systems evaluated on the Timit evaluation set using various data selection methods and

training data percentages. As with the word-based results, the `MaxEnt Tri` and `MaxEnt Wrd` data selection methods both perform consistently worse when compared to the `Natural` selection approach with the `MaxEnt Tri` approach producing significant losses for all data percentages. Only at the 20% data percentage level does the `MaxEnt Wrd` method attain a significant decrease in accuracy. For all data percentages, the word-based max-entropy selection is superior to the triphone-based unit variant. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` data selection methods attain improved accuracies for all data percentages when compared to `Natural` and max-entropy methods. The improvements, however, are not all statistically significant. The `Uniq Sqrt` method obtains the best performance for 20% and 40% data percentages, while `Uniq 0.75` is the best method for the remaining data percentages.

Table 5.13: *Triphone correctness, triphone accuracies and P-Values for various Timit systems evaluated on Timit data. The results are displayed by data selection method and percentage of training data.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	48.53	45.84	46.33	48.83	49.44	50.22	49.21
	Tri Acc (%)	44.89	40.96	42.31	45.57	46.04	47.12	45.71
	P-Value	-	4.00E-15	9.55E-07	1.23E-01	1.50E-02	3.07E-06	8.76E-02
40%	Tri Cor (%)	53.66	52.15	52.88	55.42	55.32	56.48	56.11
	Tri Acc (%)	48.52	46.59	47.56	50.65	50.39	52.07	51.71
	P-Value	-	2.65E-05	4.92E-02	1.29E-06	2.04E-05	6.00E-15	1.08E-11
60%	Tri Cor (%)	56.97	55.28	56.58	58.52	57.97	57.88	58.5
	Tri Acc (%)	51.71	49.35	51.22	53.46	52.84	52.83	53.57
	P-Value	-	5.47E-08	2.86E-01	3.09E-05	6.23E-03	9.16E-03	8.72E-06
80%	Tri Cor (%)	59.38	58.03	59.13	59.71	59.29	59.46	59.65
	Tri Acc (%)	53.97	52.23	53.6	54.39	53.98	54.12	54.59
	P-Value	-	2.78E-05	3.89E-01	2.73E-01	9.77E-01	7.22E-01	1.21E-01

Table 5.14 shows word correctness, word accuracies and P-Value results obtained using Timit ASR systems trained on data selected using a variety of data selection methods and evaluated on the WSJ evaluation set. Comparing to the `Natural` data selection approach, at the 20% data percentage level the max-entropy based methods produce significantly lower accuracies. The remaining data selection methods attain higher accuracies but only the `0.75` method provides a significant result. For the 40% level all results are significant – max-entropy selections give degraded performances and the `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches produce better performances. At 60% the `MaxEnt Tri` and `MaxEnt Wrd` selections attain lower accuracies, however, the word-based selection is insignificant. The remaining selection methods produce significantly better results. Lastly, at 80% the max-entropy selections produce insignificantly decreased performances. The three data selections methods `Sqrt`, `Uniq Sqrt`, `Uniq 0.75` produce significantly better accuracies while `0.75` data selection provides an insignificant improvement. Overall, the max-entropy based data selections produce lower accuracies compared to `Natural` selection and word-based attaining better results over the triphone unit selection. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` techniques obtain better results with respect to the other data selection methods, however, not all

results are significant.

Table 5.14: *Word correctness-accuracy results and P-Value measures for Timit trained and WSJ evaluated ASR systems using different data selection methods and percentages of the total training data.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	41.45	39.38	40.49	41.71	42.24	41.58	41.62
	Word Acc (%)	34.22	31.21	32.04	34.89	35.54	35.16	34.69
	P-Value	-	1.00E-15	3.12E-09	5.89E-02	2.37E-04	7.13E-03	1.94E-01
40%	Word Cor (%)	48.88	46.87	47.58	50.9	49.85	51.64	50.25
	Word Acc (%)	40.33	38.1	37.51	43.51	42.18	44.25	42.56
	P-Value	-	7.57E-09	1.02E-13	0	4.54E-07	0	1.27E-09
60%	Word Cor (%)	52.18	49.57	52.04	53.72	53.34	53.41	53.48
	Word Acc (%)	43.67	39.81	42.62	45.61	45.23	45.19	45.11
	P-Value	-	0	4.49E-03	2.34E-08	6.90E-06	1.49E-05	6.41E-05
80%	Word Cor (%)	53.43	52.76	52.99	54.31	53.34	53.7	54.78
	Word Acc (%)	43.23	42.93	43.21	45.45	44.03	44.37	45.87
	P-Value	-	3.91E-01	9.53E-01	9.50E-11	1.49E-02	7.27E-04	1.30E-14

Table 5.15 shows triphone correctness, triphone accuracies and significance P-Values measures for ASR systems training on Timit data selected using different data selection methods and data percentages which were evaluated on the WSJ evaluation set. For the 20%, 40% and 60% data percentage levels, the `MaxEnt Tri` and `MaxEnt Wrd` data selection methods perform significantly worse compared to the `Natural` selection method, while at the 80% percentage the results are insignificantly worse. `MaxEnt Tri` selection only manages to outperform `MaxEnt Wrd` selection at the 40% data percentage. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches produce slightly improved results compared to `Natural`, `MaxEnt Tri` and `MaxEnt Wrd` methods with the winners, per data percentage, attaining significant improvements.

Table 5.15: *Triphone correctness, triphone accuracies and P-Value results obtained using Timit ASR systems trained using various data selection methods and data percentages and evaluated on the WSJ evaluation set.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	46.2	45.28	44.96	47.05	47.91	47.65	47.1
	Tri Acc (%)	42.2	40.27	40.8	43.38	44.02	43.98	42.98
	P-Value	-	5.79E-12	4.26E-07	1.33E-05	7.03E-12	2.28E-11	3.03E-03
40%	Tri Cor (%)	53.32	52.37	51.61	55.24	54.54	55.85	54.82
	Tri Acc (%)	48.06	46.39	45.63	50.12	49.32	51.12	50.03
	P-Value	-	1.15E-09	0.00E+00	1.00E-15	1.58E-06	0.00E+00	9.70E-14
60%	Tri Cor (%)	56.41	54.35	55.72	57.54	57.36	57.17	57.15
	Tri Acc (%)	50.7	48.15	49.82	51.92	51.76	51.69	51.79
	P-Value	-	0.00E+00	4.33E-04	8.83E-07	1.72E-05	5.62E-05	1.42E-05
80%	Tri Cor (%)	56.73	56.49	56.8	57.83	56.92	57.17	58.22
	Tri Acc (%)	50.67	50.17	50.57	51.89	50.92	51.15	52.46
	P-Value	-	3.35E-02	6.56E-01	1.82E-07	2.66E-01	3.73E-02	5.00E-15

5.4.2 WSJ

Table 5.16 shows word correctness, word accuracies and P-Value statistical significance values obtained on WSJ trained ASR systems evaluated on the WSJ evaluation set using various data selection methods at different data percentages. The only significant results obtained are for both the `MaxEnt Tri` and `MaxEnt Wrd` data selection methods at the 20% training data percentage where a decrease in performance was achieved. All remaining results are insignificant and achieved accuracies that are comparable to the `Natural` selection approach.

Table 5.16: *Word correctness, accuracies and P-Value results for WSJ trained and WSJ evaluated systems using various data selections methods to generate the training corpora at specific percentages of the total training triphone counts.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	60.01	58.58	59.57	59.82	60.31	59.95	59.61
	Word Acc (%)	51.61	49.62	50.35	51.67	52.03	51.65	51.11
	P-Value	-	3.58E-08	3.77E-04	8.65E-01	2.34E-01	9.23E-01	1.62E-01
40%	Word Cor (%)	63.68	63.36	64.23	63.36	63.51	63.64	64.21
	Word Acc (%)	55.32	55.24	55.83	55.11	55.22	55.62	55.82
	P-Value	-	8.09E-01	1.41E-01	5.35E-01	7.49E-01	3.83E-01	1.35E-01
60%	Word Cor (%)	66.3	65.74	66.34	66.14	66.54	66.5	66.78
	Word Acc (%)	58.76	58.07	58.36	58.3	58.93	58.64	59.24
	P-Value	-	2.79E-02	1.95E-01	1.34E-01	5.89E-01	6.90E-01	1.16E-01
80%	Word Cor (%)	67.68	67.4	67.33	67.94	67.8	67.58	68.08
	Word Acc (%)	60.42	59.87	59.63	60.54	60.35	60.2	60.84
	P-Value	-	6.53E-02	4.10E-03	6.36E-01	8.28E-01	4.34E-01	1.19E-01

Table 5.17 shows triphone correctness, triphone accuracies and P-Value statistical significance results for ASR systems trained on WSJ data created by using various data selection approaches and training data percentages which were evaluated on the WSJ evaluation set. The only significant decrease in performance was achieved by the `Sqrt` method at the 60% data percentage level. All the remaining measures presented in the table show that the various systems produce comparable results with insignificant increases or decreases in performance when compared to the `Natural` selection method.

Table 5.18 shows the word correctness and accuracies and significance P-Values for Timit evaluated ASR systems trained on WSJ sub-corpora created using various data selection approaches and on limited data portions. Comparing results with `Natural` selection as baseline, both max-entropy based-methods produce significantly lower accuracies for 20% and 60% training data percentages. Interestingly, the `MaxEnt Wrd` data selection method manages to attain a slight improvement at the 40% level, however, not significant. The remaining data selection methods (`Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75`) do not produce significant gains or losses, but the majority of the accuracies are slightly higher compared to the `Natural` selection method.

Table 5.19 shows the triphone correctness, triphone accuracies and statistical significance P-Value measures for WSJ trained ASR systems evaluated on the Timit evaluation set which were trained on

Table 5.17: *Triphone correctness, triphone accuracies and significance P-Values for WSJ trained and evaluated systems using different data selection methods and training data percentages.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	63.39	62.66	63.22	63.14	63.4	63.28	63.2
	Tri Acc (%)	58.41	57.63	57.99	58.06	58.37	58.28	58.26
	P-Value	-	1.21E-03	8.67E-02	1.67E-01	8.61E-01	6.25E-01	6.12E-01
40%	Tri Cor (%)	66.59	66.78	66.89	66.3	66.26	66.51	66.75
	Tri Acc (%)	61.41	61.69	61.76	60.95	61.01	61.29	61.57
	P-Value	-	2.05E-01	1.24E-01	4.71E-02	7.79E-02	6.00E-01	4.62E-01
60%	Tri Cor (%)	68.89	68.55	68.62	68.38	68.86	68.69	68.99
	Tri Acc (%)	63.86	63.53	63.49	63.16	63.77	63.53	64
	P-Value	-	1.23E-01	6.95E-02	6.51E-04	6.25E-01	1.02E-01	5.05E-01
80%	Tri Cor (%)	69.84	69.89	69.82	70.03	69.94	69.83	70.13
	Tri Acc (%)	64.85	64.8	64.71	65.11	64.88	64.77	65.18
	P-Value	-	8.10E-01	4.39E-01	1.56E-01	8.59E-01	6.67E-01	7.18E-02

Table 5.18: *Word correctness, word accuracies and P-Value results for WSJ trained ASR systems evaluated using the Timit evaluation set for different data selection methods and training triphone count percentages.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	55.05	52.26	53.68	55.61	55.69	55.75	55.56
	Word Acc (%)	44.37	40.41	42.17	45.4	44.9	45.73	45.35
	P-Value	-	1.69E-09	9.78E-04	1.29E-01	4.33E-01	3.41E-02	1.57E-01
40%	Word Cor (%)	58.37	56.9	58.59	59.2	59.04	59.17	59.06
	Word Acc (%)	47.27	45.4	47.42	48.28	47.7	48.16	47.56
	P-Value	-	3.52E-03	8.31E-01	1.31E-01	4.93E-01	1.87E-01	6.70E-01
60%	Word Cor (%)	60.29	58.7	58.97	60.97	60.81	61.58	61.05
	Word Acc (%)	49.74	47.6	47.09	50.47	49.6	50.84	50.19
	P-Value	-	5.73E-04	6.24E-06	2.08E-01	7.36E-01	7.21E-02	4.71E-01
80%	Word Cor (%)	61.38	60.26	60.67	61.72	61.39	61.52	61.37
	Word Acc (%)	50.9	49.37	49.22	51.29	50.59	51.1	50.58
	P-Value	-	1.83E-02	6.52E-03	4.07E-01	6.61E-01	6.46E-01	6.31E-01

data selected using different data selection techniques for various training data percentages. The `MaxEnt Tri` data selections method consistently produces lower accuracies when compared to the `Natural` selection method. The `MaxEnt Wrd` approach manages a slight insignificant improvement at the 40% level but the remaining data percentage measures are lower taking the natural selection as reference. The 20% `MaxEnt Tri`, 60% `MaxEnt Tri` and 60% The `MaxEnt Wrd` results are significantly worse. The remaining data selection methods, `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75`, produce slightly increased performances (except at 80% `0.75` and `Uniq 0.75` measures are lower) but none are significant.

Table 5.19: *Triphone correctness, triphone accuracies and P-Value results for WSJ trained ASR systems evaluated using the Timit evaluation set for different data selection methods and training triphone count percentages.*

Percentage	Metric	Selection Type						
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	53.42	51.58	52.79	54.25	54.31	54.45	54.41
	Tri Acc (%)	48.06	45.73	47.02	49.3	49	49.43	49.48
	P-Value	-	6.87E-07	2.80E-02	8.89E-03	5.20E-02	3.27E-03	2.67E-03
40%	Tri Cor (%)	56.21	55.09	56.82	57.25	56.94	57.2	56.77
	Tri Acc (%)	50.32	49.02	50.69	51.38	51.11	51.45	51
	P-Value	-	4.15E-03	4.16E-01	1.60E-02	6.77E-02	1.46E-02	1.33E-01
60%	Tri Cor (%)	58.1	57.11	57.15	58.55	58.35	59.15	58.93
	Tri Acc (%)	52.33	50.9	50.77	52.56	52.45	53.36	53.03
	P-Value	-	5.75E-04	1.45E-04	5.57E-01	8.14E-01	1.11E-02	8.91E-02
80%	Tri Cor (%)	59.38	58.03	58.63	59.46	59.02	59.56	59.22
	Tri Acc (%)	53.38	52.04	52.38	53.47	52.98	53.59	53.11
	P-Value	-	1.45E-03	1.47E-02	7.44E-01	3.31E-01	5.32E-01	5.36E-01

5.4.3 LWAZI

Table 5.20 shows the word correctness, word accuracies and significance P-Value results for Lwazi trained ASR systems evaluated on Lwazi evaluation sets using different data selection approaches and data percentages. The `MaxEnt Wrd` data selection method produces the worst results compared to all data selection methods and significantly lower performances when compared to the `Natural` data selection approach at the 20%, 40% and 60% data percentages. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` data selection methods attain lower accuracies for the 20%, 40% and 60% data percentages compared to the `Natural` technique. 20% `Sqrt`, 20% `0.75` and 40% `0.75` are insignificant decreases. At the 80% data percentage level, however, the methods produce slight improvements but are insignificant. The second best word accuracy results are: `0.75` at 20% and 40%, `Uniq Sqrt` at 60% and `Uniq 0.75` at 80%.

Table 5.20: *Word correctness, word accuracy and P-Value results for Lwazi trained ASR systems evaluated on Lwazi evaluation data. Different data percentages and data selection methods were used to create various training corpora.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	39.821	27.556	36.959	38.814	35.918	37.782
	Word Acc (%)	15.516	3.753	13.301	15.174	12.386	13.951
	P-Value	-	0.00E+00	1.09E-06	4.04E-01	1.71E-11	3.65E-04
40%	Word Cor (%)	45.754	41.994	44.699	45.193	43.972	44.069
	Word Acc (%)	19.445	15.275	17.982	18.266	17.789	18.001
	P-Value	-	0.00E+00	1.00E-03	8.00E-03	1.63E-04	9.01E-04
60%	Word Cor (%)	49.156	47.755	49.106	49.149	48.704	48.39
	Word Acc (%)	22.887	20.573	22.248	22.195	22.331	22.226
	P-Value	-	7.53E-08	9.92E-02	7.85E-02	1.50E-01	1.15E-01
80%	Word Cor (%)	51.47	51.3	51.605	51.913	51.709	51.804
	Word Acc (%)	25.081	24.99	25.309	25.874	25.551	25.997
	P-Value	-	8.80E-01	5.74E-01	2.84E-02	1.76E-01	8.97E-03

Table 5.21 shows triphone correctness, triphone accuracies and statistical significance P-Values for Lwazi ASR systems developed using different data selection methods and training data percentages which were evaluated on Lwazi evaluation sets. The `MaxEnt Wrd` data selection produces the lowest performance with only the last results being insignificant. At the 20% and 40% data percentages, the remaining data selection methods (excluding `Natural`) attain lower system accuracies compared to the `Natural` data selection approach and only `0.75` approach producing insignificant decreases. At the 60% data percentage `Sqrt` and `0.75` data selection methods achieve a slight accuracy improvement while the `Uniq` methods produce slight loses. The results are insignificant. For the 80% level, `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches all achieve a slight insignificant gain.

Table 5.21: *Triphone correctness, triphone accuracies and P-Value significances for Lwazi trained and evaluated ASR systems developed using various data selection techniques and data percentages.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	49.843	36.079	46.024	48.512	45.2	47.275
	Tri Acc (%)	43.068	28.622	40.311	42.323	39.339	41.099
	P-Value	-	0.00E+00	0.00E+00	1.01E-02	0.00E+00	4.90E-12
40%	Tri Cor (%)	56.613	51.758	54.86	55.38	54.178	54.489
	Tri Acc (%)	48.18	42.99	47.279	47.603	46.69	46.743
	P-Value	-	0.00E+00	5.98E-04	3.20E-02	1.65E-08	2.08E-08
60%	Tri Cor (%)	60.237	58.501	59.662	59.792	59.443	59.329
	Tri Acc (%)	51.171	48.988	51.216	51.231	50.995	50.703
	P-Value	-	0.00E+00	9.15E-01	9.12E-01	4.51E-01	5.05E-02
80%	Tri Cor (%)	62.793	62.306	62.592	62.819	62.787	62.937
	Tri Acc (%)	53.034	52.373	53.306	53.459	53.531	53.634
	P-Value	-	6.06E-03	2.77E-01	4.64E-02	1.70E-02	4.97E-03

Table 5.22 shows word correctness, word accuracies and P-Values for Lwazi ASR systems trained on data selected corpora and for different data percentages which were evaluated on the AST evaluation set. The `MaxEnt Wrd` data selection methods produces the lowest performances for all data percentages where for the 20% and 60% data percentage levels the results a significant. The `Sqrt` approach manages to achieve accuracy increases. however none are significant. The `0.75` data selection technique produces a significant improvement at the 60% data percentage level. The remaining results are comparable to the `Natural` selection method.

Table 5.23 shows triphone correctness, triphone accuracies and statistical significance P-Values for Lwazi trained ASR system evaluated on the AST evaluation set using various data selection approaches and training data percentages. The `MaxEnt Wrd` data selection approach produces accuracy decreases for all data percentages with only 80% level an insignificant result. The `Sqrt` approach achieves performance gains for all data levels but none are significant. The `0.75` technique produces gains for all data levels expect the 20% percentage and again none are significant. Both `Uniq` methods present no pattern but results are not significant.

If one compares the Timit, WSJ and Lwazi word correctness and accuracies, the Lwazi systems produced the highest errors. This might be explained by the telephony collection channel for the

Table 5.22: *Word correctness, word accuracies and P-Values for various Lwazi trained ASR systems evaluated on AST evaluation set using different data selection methods and training data percentages.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Word Cor (%)	31.232	21.9	30.551	29.671	28.342	29.42
	Word Acc (%)	15.844	11.522	17.086	14.7	14.696	14.234
	P-Value	-	5.88E-06	1.67E-01	2.36E-01	3.80E-01	6.20E-02
40%	Word Cor (%)	37.165	31.096	36.946	36.417	35.981	35.188
	Word Acc (%)	20.88	16.284	22.018	20.845	20.867	19.893
	P-Value	-	1.52E-03	2.97E-01	6.27E-02	4.97E-02	6.01E-01
60%	Word Cor (%)	39.696	33.998	40.092	39.865	39.598	39.238
	Word Acc (%)	23.863	15.515	24.481	24.623	23.631	23.723
	P-Value	-	5.35E-06	6.45E-02	6.24E-05	3.39E-03	2.42E-01
80%	Word Cor (%)	40.546	38.533	40.919	40.46	40.445	40.447
	Word Acc (%)	24.631	22.71	25.665	25.213	24.755	24.134
	P-Value	-	4.24E-03	1.47E-01	2.41E-01	6.36E-02	1.84E-01

Table 5.23: *Triphone correctness, triphone accuracies and P-Values for various Lwazi trained ASR systems evaluated on AST evaluation set using different data selection methods and training data percentages.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Tri Cor (%)	40.343	29.041	39.516	38.546	37.062	37.709
	Tri Acc (%)	32.815	22.903	33.238	31.776	31.129	30.937
	P-Value	-	0.00E+00	9.94E-01	7.08E-01	4.34E-01	7.46E-02
40%	Tri Cor (%)	47.053	40.084	47.287	46.483	46.09	45.412
	Tri Acc (%)	37.134	31.739	38.433	37.242	37.592	36.406
	P-Value	-	2.88E-05	2.11E-01	1.39E-01	4.56E-02	5.71E-01
60%	Tri Cor (%)	50.28	42.761	50.906	50.64	50.094	49.446
	Tri Acc (%)	38.62	31.968	40.558	40.053	39.301	39.144
	P-Value	-	1.22E-06	3.44E-03	6.34E-03	6.31E-03	4.93E-02
80%	Tri Cor (%)	51.668	49.468	52.089	51.891	51.139	50.782
	Tri Acc (%)	39.263	37.421	40.4	39.913	39.076	38.945
	P-Value	-	5.64E-02	1.82E-01	3.12E-01	1.14E-01	4.96E-01

Lwazi data and that the utterances were collected in a more natural way with less stringent collection process as compared to the strict Timit and WSJ collection setups.

5.4.4 ACCURACY CORRELATIONS AND DISTRIBUTION CORRESPONDENCE

We investigated whether the accuracy differences are caused by the correspondence between the training and evaluation corpora, and also whether our word and triphone accuracy measures are generally in agreement. The details are summarized in appendix (D) : we found that the word and triphone accuracies are indeed highly correlated, and that the divergences between the training and evaluation corpora were indeed negatively correlated with the achieved accuracies and correctness values.

5.5 CONCLUSION

The work presented in this chapter outlines a new data selection theory which provides a mechanism for choosing units based on criteria for boosting the system's overall accuracy. Contrary to other unit selection methods our approach takes into consideration the relationship between a unit's accuracy and its frequency of occurrence. We showed that the triphone is a good unit choice as there is a strong correlation between the triphone's training frequency and accuracy and in addition that there is a low to weak correlation for adjacent triphone accuracies thus allowing an independence assumption. In our theoretical framework we showed that the optimal distribution is dependent on the assumed relationships between the triphone frequency and accuracy. Based on theoretical and empirical evidence, the two relationships we investigated were logarithmic and hyperbolic. The hyperbolic relationship leads to a unit selection strategy in which the selected frequencies are proportional to the square root of the occurrence frequencies, while the logarithmic relationship leads to selected units which match the reference set *i.e.* units selected at random. A number of data selection experiments were performed to investigate the relationships and compare our approach with commonly used methods. From these we may conclude:

- In the vast majority of cases the max-entropy based data selection consistently produced the lowest performing systems and word-based max-entropy selection is superior to triphone-based unit selection. These results are consistent with results presented by Gouvea and Davel [50].
- Using our experimental setup and choosing smaller sub-corpora, the "natural" selection (random choice) is an effective strategy and is difficult to outperform in a consistent manner.
- For the Timit trained ASR systems;
  - The "compressed" and "intermediate" data selection methods have the ability to produce improved accuracies, however, not all were significantly better when compared to "natural" selection.
  - On average estimating the triphone distribution from unique utterances and then performing data selection gave slightly better system accuracies when compared with systems which estimated triphone distributions using all the data.
- For the WSJ trained ASR systems; The "compressed" and "intermediate" data selection methods performed comparably to the "natural" selection method for both WSJ and Timit evaluations.
- For the Lwazi trained ASR systems; On average, the "compressed" and "intermediate" data selection methods performed comparably to the "natural" selection method for both Lwazi and AST evaluations.

Based on the results we can see that for the majority of experiments the "compressed" and "intermediate" data selection methods achieved results comparable to that of the "natural" selection. Only for the Timit experiments did we see an improvement but the Timit corpus is somewhat artificial as the prompt selection was heavily engineered. The WSJ and Lwazi corpora are more typical of ASR data collections. The max-entropy based selections did not show any promising results which is in line with findings presented by Gouvea and Davel [50]. The main conclusion, thus is that for any data selection, matching the "natural" distribution is a competitive strategy. There are indications that the "compressed" and "intermediate" data selection methods may be useful under specific circumstances, and it is worthwhile investigating whether those methods may be preferable to "natural" selection in other practical situations.

Note, also, that some of the effects observed in our studies are not only statistically significant, but also have substantial potential impact. For example, in table 5.14 the `Uniq` methods obtain similar accuracies at 60% corpus size to what the `Natural` method obtains at 80% corpus size, implying that similar performance could be achieved with only $60/80 = 75\%$ of the collection effort. Such savings should be quite useful in practice.

As stated in the Introduction, an optimal data selection approach would be most beneficial for corpus design in resource constrained environments which would allow targeted acoustic data collection. A pragmatic approach would be to start with any available text resources which could be sourced from the internet or any other digital outlet. An initial manual effort and linguistic expertise would be needed to create a suitable phone set for the text data. Once obtained, g2p rules can be extracted to map all the words to their phonetic representations. Given the mapped text, an appropriate data selection method could be used to select from the text data the optimal set of utterances given constraints on budget and collection time.

Our proposed data selection methods rely on the triphone counts and are independent of the exact triphones defined in a languages. Thus, the methods can be applied to any language once text data has been obtained. Though van Santen and Buchsmans, [49] raised concerns on unit coverage, we have seen that for targeted data collection, the collected data should result in a distribution quite close to the chosen distribution for optimal ASR.

CHAPTER SIX

CONCLUSION

6.1 INTRODUCTION

In Chapter 1 section (1.2) we defined our overall aim as follows: to investigate various methods which will facilitate the use of automatic speech recognition (ASR) technologies in resource-scarce environments. Following from this, we devised a set of goals that we thought most appropriate in achieving our aim – these were;

- to develop an automatic data harvesting procedure that transforms audio and corresponding approximate annotations into a corpus that is usable for ASR training;
- to investigate the application of an unsupervised environment normalisation technique for data mismatch reduction, with a focus on the specific scenario of bandwidth mismatches;
- to analyse the data dependence, for specific triphone counts, of current ASR adaptation techniques in that same scenario; and
- to develop a data selection framework and implementation which optimises ASR system performance.

The remainder of the chapter summarises the thesis contributions and discusses possible future work to extend the initial investigations.

6.2 SUMMARY OF CONCLUSIONS AND CONTRIBUTION

The thesis presents several approaches that successfully expedite the development of ASR systems in resource-scarce environments. Our most important conclusions are summarised below.

- The process of alignment-filter-train, can be used to efficiently and automatically harvest audio data with approximate transcriptions.
- The proxy measures; log-likelihood, average dynamic programming scores, and the percentage of data absorbed by the garbage provided useful methods of automatically monitoring the harvesting process. The validity of the proxy measures were established using the Lwazi and NCHLT English corpora.
- Bootstrapped acoustic models trained on out-of-dialect audio data can be used to provide good audio-transcription alignments. In our specific experiments we showed that American-English acoustic models can be used in the harvesting of SAE audio data.
- The proxy measures and phone-level system accuracies (obtained using independent corpora), showed that retraining acoustic models on the harvested data provides better models to generate more accurate alignments and iterating the alignment-filter-train cycle provides better models after each iteration.
- A garbage model can be used to good effect in order to improve the reliability of the alignments by absorbing non-speech audio and speech disfluencies. In addition, the garbage model provides a means to identify mis-aligned audio and text portions which can be used to rapidly filter the data into usable and unusable portions.
- A dynamic programming algorithm can be used to robustly filter the harvested data and create a corpus which has quite reliable alignments between text and audio. Additionally, the DP scores serve as a good metric with which to monitor the harvesting process.
- Manually processing data to create an adaptation corpus, which can be used to adapt a set of acoustic models, is not needed. The win obtained by manually processing the data is lost after the first retrain cycle.
- The data size, of the data to be harvested, affects the performance metric values – more data predictable results in better values. However, the performance metric convergence rates exhibit a low correlation to the data size and the relative improvements are approximately the same.
- The transfer-function filtering feature normalisation approach performed comparably to MLLR for low adaptation counts but the observed gains plateau quickly as more data was added. Other benefits of the transfer-function normalisation method are that it does not require transcriptions to perform the normalisation and can be applied independently of the various model-based adaptations.
- MLLR works well in reducing the mismatch for bandwidth matched adaptations but failed to achieve the same ASR system accuracies when transforming band limited acoustic models to full bandwidth models (16kHz).

- Around the 10000 to 100000 adaptation triphone count MAP starts to perform better than MLLR but never beats the retraining the acoustic models.
- As the adaptation triphone count approaches 10000 to 100000 triphone examples, retraining the acoustic models becomes the best strategy which out-performs both MLLR and MAP.
- The consistent findings for the specific case of adaptation between low- and high-bandwidth applications, for example telephone-quality and smartphone-recorded speech audio, should be useful for developing world ASR system developers.
- Using our experimental setup and choosing smaller sub-corpora, the "natural" selection (random choice) is an effective strategy and is difficult to outperform in a consistent manner.
- Our comparative experiments using the max-entropy based data selection method showed results which, in the overwhelming majority of cases, produced the lowest performing ASR systems. Furthermore, the results showed that the word-based max-entropy selection is superior to triphone-based unit selection.
- The experimental evidence shows that the "compressed" (square-root compression of triphone probabilities and re-normalised) and "intermediate" (triphone probabilities raise to a power of 0.75 and re-normalised) data selection methods achieved results comparable to that of the "natural" selection in the majority of the experiments. (The Timit experiments did produce improved results with two other strategies; however, the Timit corpus does not represent a typical corpus due to specifically engineered prompt selection). The Timit results do however indicate that the "compressed" and "intermediate" data selection methods may be useful under specific circumstances. Investigating the factors which contribute to the improvements may help provide a better data selection strategy which may be preferable to "natural" selection in other practical situations.

The main contribution of our work is therefore to expand the knowledge base and tools available to those who wish to develop speech-recognition systems for under-resourced languages. In particular:

- Our tools for harvesting speech corpora from approximately transcribed data, when sufficient resources for conventional language modelling are not available, are novel and potentially useful for several under-resourced languages.
- The consistent findings for the specific case of adaptation between low- and high-bandwidth applications, for example telephone-quality and smartphone-recorded speech audio, should be useful for ASR system developers for under-resourced languages.
- Our approach to data selection has shown that the widely-used maximum entropy approach is not a good choice for ASR prompt selection. If this had been known previously, for example, the NCHLT corpus [60] could have been created in a significantly more efficient fashion.

Again, we hope that developers of speech recognition in under-resourced languages will benefit from this insight.

6.3 FUTURE WORK

To begin the automatic data harvesting iterative process we used seed acoustic models trained on American English audio data. This dialect of English is further from South African English than for example, British English, but still possessed enough commonality with SAE to limit the amount of manual effort needed to create phoneme mappings between the dialects. An interesting topic of research would be to investigate whether or not it is possible to use seed models trained on a completely different language or set of languages.

The data harvesting pre- and post-processing tools can be improved – specifically, an audio-clip detector which can be used by the bandwidth detector to ignore clipped audio portions. In addition, a channel detector should be added to make sure we are selecting an audio channel which has data. The 4-class classifier could also be improved by adding GMM smoothing and updating the CMN vectors with speech only.

The knowledge gained from the cross-channel adaptation experiments, which established the data needs of various standard adaptation methods, can potentially prove to be invaluable to a ASR developer. For refinement, more data increments can be experimented with to help produce smoother graphs and maybe even provide more data samples to model the processes. Introducing class-based transfer-function normalisations can possibly provide an added gain in performance but determining the suitable classes without transcriptions could prove to be quite challenging.

The data selection results showed that finding improvements over randomly selecting the training utterances is quite difficult. Fully understanding why the Timit results required non-uniform sampling is likely to provide additional insights, that may be useful for the creation of specialized corpora.

A number of additional experiments, which could not be completed during the current research because of computational constraints, would be useful as additional confirmation of our conclusions. In particular:

- Investigating the benefit of transfer-function filtering on the NCHLT-Lwazi cross channel experiments.
- Including error margins on the results produced for the adaptation data need experiments.
- Comparing the data selection results to those obtained using ASR system developed on all the training data for the specific corpora.

In this thesis we addressed several topics related to the application of ASR in resource-scarce environments. We developed and demonstrated approaches that can be used to assist in the design and deployment of ASR systems in limited-data scenarios. Hopefully, continuing to develop these

and new techniques will lead to the creation of more ASR applications in the developing world and lead to a greater technology penetration.

APPENDIX A

MAP ADAPTATION PARAMETER EXPERIMENTS

To establish the effect that different parameter selections have on MAP adaptation of acoustic models, we performed MAP adaptation of band limited (250-3400 Hz) NTimit acoustic models using WSJ 16 kHz data. τ, the informative prior weighting factor was set to 5, 10 and 20 – as suggested in the HTK book [10]. The iteration counts were chosen to be 1, 2, 3, 5, 10 and 20. Finally, the amount of data was systemically increased by doubling the triphone counts at each training instance – 1000, 2000, 4000, 8000, 16000, 32000, 64000, 128000, 256000 and 512000. Tables A.1, A.2 and A.3 show the accuracies obtained for tau set to 5, 10 and 20 respectively.

Table A.1: *Obtained accuracies for MAP adapted band-limited NTimit acoustic models with* $\tau = 5$.

Triphone Count	Iteration Count					
	1	2	3	5	10	20
1000	22.64	20.43	18.51	15.98	13.85	11.99
2000	29.24	28.04	25.20	19.60	16.23	14.21
4000	35.82	35.59	32.41	26.71	23.18	21.40
8000	43.22	44.63	41.99	37.52	34.93	33.72
16000	49.47	52.80	51.67	48.50	46.94	46.33
32000	54.19	59.62	59.66	58.64	58.07	57.93
64000	58.15	64.69	65.62	65.75	65.78	65.94
128000	60.46	67.50	68.95	69.79	70.33	70.53
256000	61.77	69.15	70.95	72.11	72.96	73.29
512000	62.42	70.05	72.13	73.49	74.45	74.82

Figures A.1, A.2 and A.3 show the accuracies obtained using MAP adaptation on increasing amounts of data, for various iterations counts and informative prior weights. The general trend is for low data counts using fewer training iterations is better and as more data is added updating the model parameters with more iterations produces better results. For the informative prior weights $\tau =$

Table A.2: *Obtained accuracies for MAP adapted band-limited NTimit acoustic models with* $\tau = 10$.

Triphone Count	Iteration Count					
	1	2	3	5	10	20
1000	21.92	22.00	20.12	17.06	15.30	13.59
2000	27.32	29.44	28.02	23.96	18.44	16.04
4000	33.47	36.86	35.90	31.59	25.35	23.42
8000	40.68	45.14	45.18	41.86	36.76	35.20
16000	47.15	52.78	53.50	51.86	48.42	47.62
32000	52.23	59.12	60.46	60.06	58.72	58.50
64000	56.63	64.08	65.68	66.18	66.02	66.09
128000	59.39	67.02	68.80	69.88	70.40	70.58
256000	61.12	68.78	70.76	72.06	72.95	73.28
512000	62.08	69.81	71.93	73.41	74.39	74.82

Table A.3: *Obtained accuracies for MAP adapted band-limited NTimit acoustic models with* $\tau = 20$.

Triphone Count	Iteration Count					
	1	2	3	5	10	20
1000	20.95	21.75	21.73	19.70	15.83	14.89
2000	24.96	28.25	29.37	27.89	22.57	18.05
4000	29.96	35.19	36.74	36.01	30.24	25.14
8000	37.07	43.19	45.24	45.41	41.02	36.86
16000	43.65	50.99	53.25	54.01	51.49	48.87
32000	49.12	57.40	59.94	61.01	60.17	59.22
64000	54.13	62.72	65.10	66.37	66.47	66.45
128000	57.58	66.02	68.25	69.66	70.45	70.73
256000	60.10	68.14	70.30	71.86	72.79	73.29
512000	61.47	69.38	71.60	73.21	74.21	74.79

$[5, 10, 20]$, iteration counts 2, 3 and 5 appear to produce the best balance between the accuracy and amount of adaptation data. The informative prior seems to have a marginal effect on the final system accuracy for relatively large data amounts. For those cases, the number of training iterations required for optimal performance moves towards the higher end of the range that we investigated, but for smaller training sets (where MAP adaptation could potentially be competitive), optimal performance is achieved around 5 to 10 iterations.

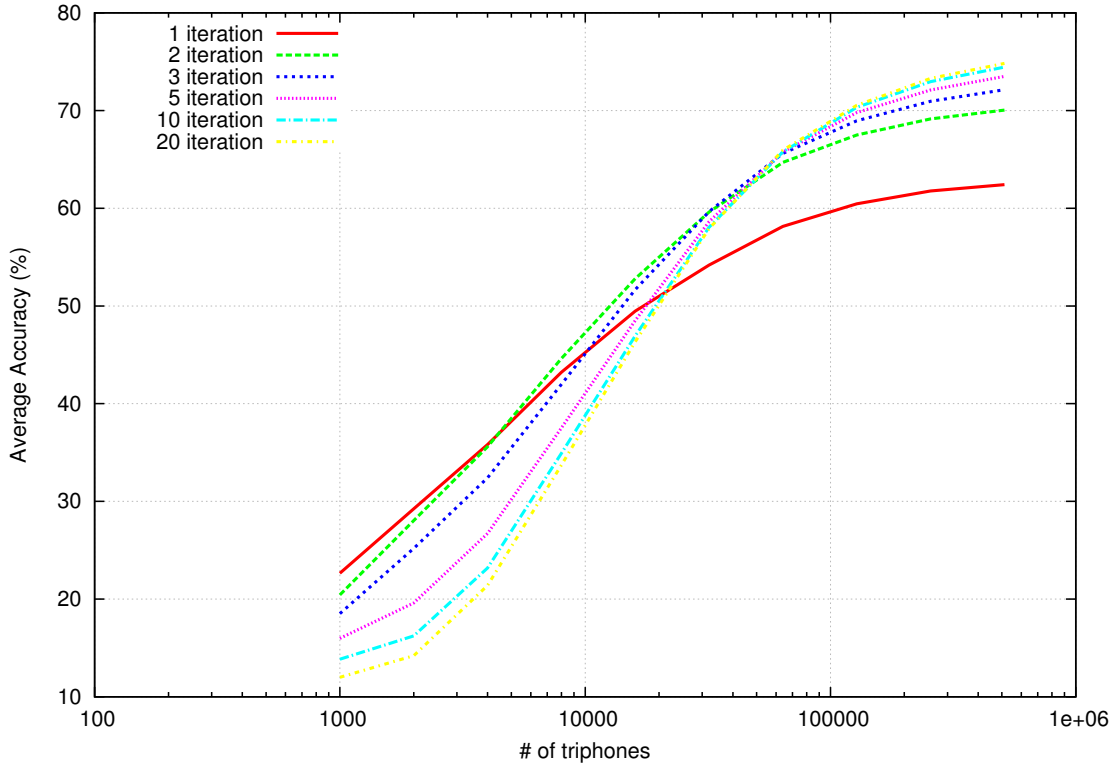

Figure A.1: *Accuracies achieved when using MAP adaptation on increasing data amounts and for various iteration counts. The informative prior weight τ was set to 5.*

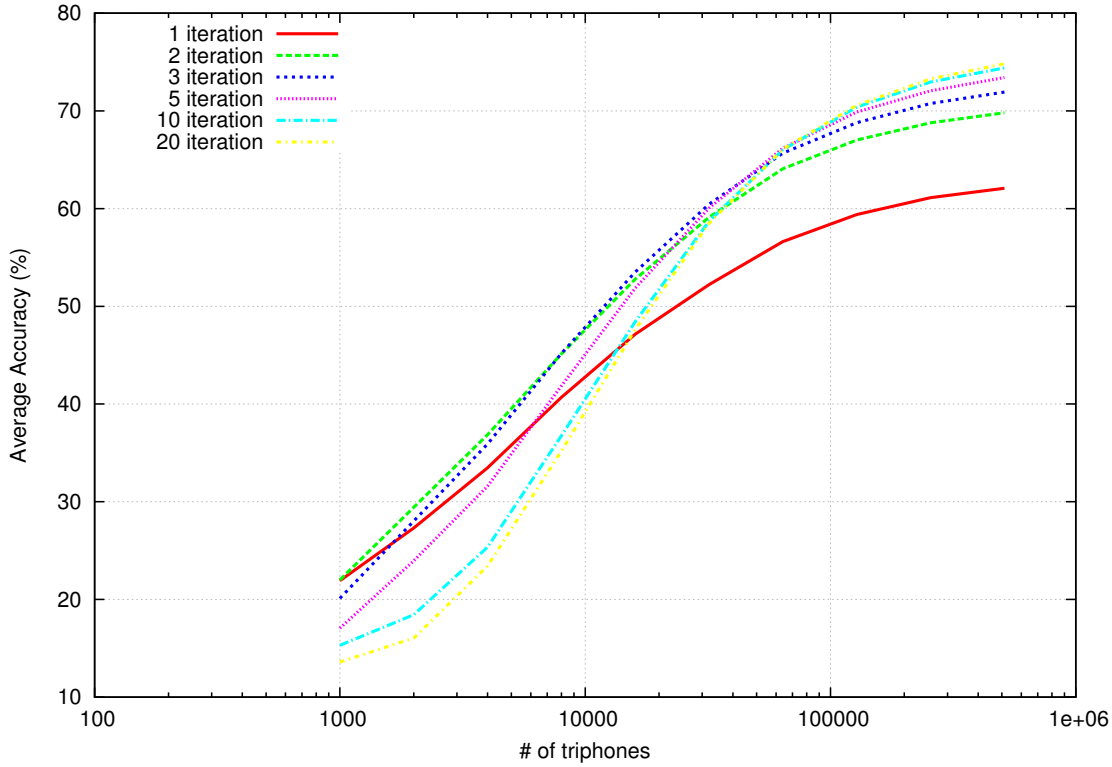

Figure A.2: *Accuracies achieved when using MAP adaptation on increasing data amounts and for various iteration counts. The informative prior weight τ was set to 10.*

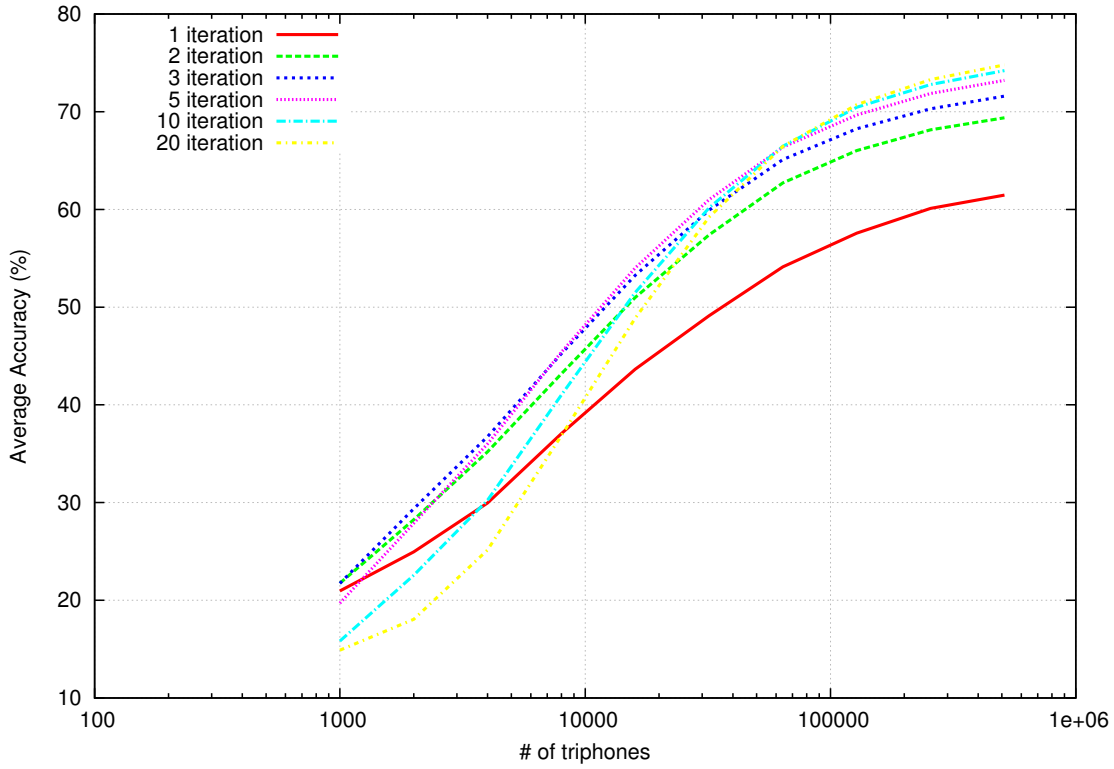

Figure A.3: *Accuracies achieved when using MAP adaptation on increasing data amounts and for various iteration counts. The informative prior weight τ was set to 20.*

APPENDIX B

ADAPTATION PERFORMANCE GAIN CURVES

In section (4.4.3) and section (4.4.4) we showed performance gain curves for various adaptation techniques applied to ASR systems and using increasing adaptation data amounts. The experiments were performed on WSJ and NTimit corpora combination and the NCHLT and Lwazi corpora combination. The phone-level accuracies used to generate the curves are presented here as well as the deletion, substitution and insertion errors made during the recognitions.

B.1 WSJ - NTIMIT EXPERIMENTS

Table B.1 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of WSJ acoustic models using NTimit band-limited (250-3400 Hz) adaptation data. The experiment is described by graph key WSJ_NTIMIT_MLLR_BP.

Table B.1: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of WSJ acoustic models using band-limited (250-3400 Hz) NTimit adaptation data. Experiment WSJ_NTIMIT_MLLR_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
100	53	41.17	30590	96786	32052
250	54.21	43.03	29848	94239	30292
500	55.01	43.71	28904	93028	30618
1000	55.95	45.12	28727	90649	29337
2000	57.03	46.39	27965	88487	28840
3000	57.78	47.06	27565	86842	29047
4000	58.23	47.93	27172	86028	27913
7500	59.68	49.33	25774	83480	28073
15000	60.12	50.34	26094	81988	26508
149860	60.45	50.76	5211	16226	5250

Table B.2 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NTimit acoustic models using WSJ band-limited (250-3400 Hz) adaptation data. The experiment is described by graph key NTIMIT_WSJ_MLLR_BP.

Table B.2: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NTimit acoustic models using band-limited (250-3400 Hz) WSJ adaptation data. Experiment key NTIMIT_WSJ_MLLR_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
100	53.48	41.39	46748	137288	47802
250	55.32	43.97	44295	132437	44894
500	55.99	44.93	44063	130047	43740
1000	56.44	44.92	43348	128982	45565
2000	57.5	46.23	42841	125274	44587
3000	58	46.94	42033	124117	43729
4000	58.58	47.47	41770	122094	43941
5000	59.04	48.08	41457	120566	43347
6000	59.21	48.65	41810	119550	41779
7000	59.41	48.69	41287	119269	42423
8000	59.47	48.94	40937	119402	41625
9000	59.93	49.17	40361	118151	42538
10000	60.06	49.22	40357	117653	42878
11000	60.01	49.42	40581	117596	41891
12000	60.03	49.52	40650	117468	41563
902894	61.01	50.89	8104	22741	8011

Table B.3 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NTimit acoustic models using 16 kHz sampled WSJ adaptation data. The experiment is described by graph key NTIMIT_WSJ_MLLR_16k.

Table B.3: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NTimit acoustic models using 16 kHz sampled WSJ adaptation data. Experiment key NTIMIT_WSJ_MLLR_16k.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
1000	35.01	19.77	66491	190088	60161
2000	37.98	22.62	60421	184397	60632
4000	41.05	25.99	58023	174751	59467
8000	43.8	29.25	54555	167367	57478
16000	44.85	30.87	53724	164084	55233
32000	45.54	31.35	52150	162929	56033
902894	45.66	32.35	10876	32048	10513

Table B.4 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NTimit acoustic models using band-limited (250-3400 Hz) WSJ adaptation data. The experiment is described by graph key NTIMIT_WSJ_MAP_BP.

Table B.5 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of WSJ acoustic models using band-limited (250-3400 Hz) NTimit adaptation data. The experiment is described by graph key

Table B.4: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NTimit acoustic models using band-limited (250-3400 Hz) WSJ adaptation data. Experiment key NTIMIT_WSJ_MAP_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
2000	51	38.67	48809	145035	48773
4000	52.43	40.48	48234	139957	47235
8000	55.97	43.75	49435	124744	48330
16000	60.76	49.7	45027	110205	43733
32000	65.19	55.45	38856	98847	38528
64000	70.16	61.71	33468	84577	33407
128000	73.64	66.11	29630	74635	29794
256000	75.83	68.9	27688	67907	27419
512000	77.03	70.59	26723	64143	25487
902894	77.75	71.14	5042	12560	5234

WSJ_NTIMIT_MAP_BP.

Table B.5: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of WSJ acoustic models using band-limited (250-3400 Hz) NTimit adaptation data. Experiment WSJ_NTIMIT_MAP_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
1000	53.01	39.65	23754	79163	29246
2000	52.53	41.25	33488	95152	30580
4000	54.2	43.22	32879	91239	29763
8000	56.86	45.95	30451	86469	29560
16000	58.68	47.66	28753	83236	29852
32000	59.55	48.6	28358	81256	29692
64000	60.52	50.07	27839	79140	28321
128000	62.51	53.01	26099	75492	25758
256000	63.23	53.87	25275	74365	25370

Table B.6 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NTimit acoustic models using 16 kHz sampled WSJ adaptation data. The experiment is described by graph key NTIMIT_WSJ_MAP_16k.

Table B.7 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for WSJ acoustic models trained on increasing amounts 16 kHz sampled WSJ training data. The experiment is described by graph key WSJ_RETRAIN_16k.

Table B.8 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for WSJ acoustic models trained on increasing amounts band-limited (250-3400 Hz) WSJ training data. The experiment is described by graph key WSJ_RETRAIN_BP.

Table B.9 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for transfer-function filtering of WSJ acoustic models using band-limited (250-3400 Hz) NTimit adaptation data. The experiment is described by graph key WSJ_NTIMIT_TFF.

Table B.10 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for transfer-function filtering of NTimit acoustic models using band-limited (250-

Table B.6: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NTimit acoustic models using 16 kHz sampled WSJ adaptation data. Experiment key NTIMIT_WSJ_MAP_16k.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
1000	32.72	19.38	92394	173710	52761
2000	36.91	25	91752	157797	47145
4000	42.89	31.86	92139	133741	43610
8000	53.49	41.62	62924	121062	46941
16000	62.6	51.41	43841	104083	44290
32000	69.18	59.97	34668	87222	36430
64000	74.03	66.34	29793	72926	30423
128000	77.16	70.27	26600	63750	27266
256000	79.01	72.68	25184	57836	25059
512000	80.13	74.06	24045	54574	23979
902894	80.04	74.12	4835	10958	4678

Table B.7: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for WSJ acoustic models trained on increasing amounts 16 kHz sampled WSJ training data. Experiment key WSJ_RETRAIN_16k.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
2000	47.14	34.46	45776	121530	40124
4000	55.12	43.81	51200	126369	44765
8000	59.35	48.92	46319	114491	41235
16000	64.58	54.26	39053	101081	40803
32000	69.46	60.9	34693	86113	33857
64000	74.55	66.86	29656	71043	30388
128000	78.34	71.85	25882	59800	25674
256000	81.87	76.43	21809	49895	21530
512000	85.21	80.45	17811	40693	18827
902894	86.92	82.54	3204	7145	3463

Table B.8: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for WSJ acoustic models trained on increasing amounts band-limited (250-3400 Hz) WSJ training data. Experiment key WSJ_RETRAIN_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
16000	52.97	40.96	31981	95451	32569
32000	58.61	47.83	29035	83130	29218
64000	63.25	53.81	25075	74510	25582
128000	65.74	57.05	23286	69557	23541
149860	66.2	57.9	4634	13683	4502

3400 Hz) WSJ adaptation data. The experiment is described by graph key NTIMIT_WSJ_TFF.

Table B.9: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for transfer-function filtering of WSJ acoustic models using band-limited (250-3400 Hz) NTimit adaptation data. Experiment key WSJ_NTIMIT_TFF.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
50	51.27	39	33750	98313	33241
100	52.84	40.99	32768	95025	32126
250	53.94	42.02	31606	93208	32306
149860	54.74	43.28	6253	18279	6208

Table B.10: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for transfer-function filtering of NTimit acoustic models using band-limited (250-3400 Hz) WSJ adaptation data. Experiment key NTIMIT_WSJ_TFF.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
50	54.66	42.58	46439	132903	47796
100	55.56	43.74	45521	130258	46769
250	56.16	44.33	44257	129149	46820
902894	57.1	45.27	8652	25288	9357

B.2 NCHLT - LWAZI EXPERIMENTS

For experiments using all the triphones (labelled "all") as adaptation data, the deletion, substitution and insertion errors will be less compared to the other triphone count-specific experiments as the later used five folds plus five random selections per fold to average the results.

Table B.11 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for Lwazi acoustic models trained on increasing amounts band-limited (250-3400 Hz) Lwazi training data. The experiment is described by graph key LWAZI_RETRAIN_BP.

Table B.11: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for Lwazi acoustic models trained on increasing amounts band-limited (250-3400 Hz) Lwazi training data. Experiment key LWAZI_TRAIN_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
16000	56.842	42.104	40155	79482	40852
32000	60.38	47.366	36782	73046	36071
64000	64.358	52.864	33305	65491	31863
128000	68.556	58.264	29685	57473	28535
all	142.708	123.2	31439	61210	31499

Table B.12 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of Lwazi acoustic models using 16 kHz sampled NCHLT adaptation data. The experiment is described by graph key LWAZI_NCHLT_MAP_16k.

Table B.13 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of Lwazi acoustic models using band-limited (250-3400 Hz) NCHLT adaptation data. The experiment is described by graph key LWAZI_NCHLT_MAP_BP.

Table B.12: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of Lwazi acoustic models using 16 kHz sampled NCHLT adaptation data. Experiment key LWAZI_NCHLT_MAP_16k.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
8000	30.268	22.02	719848	380818	129922
16000	47.942	31.262	385907	436166	263303
32000	60.464	43.466	239337	385694	267734
64000	66.148	53.756	201105	333632	196233
128000	69.694	58.73	177816	301107	173672
256000	71.882	61.63	164699	279738	162393
512000	73.172	63.334	157577	266582	155830
all	74.072	64.338	30370	51643	30817

Table B.13: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of Lwazi acoustic models using band-limited (250-3400) NCHLT adaptation data. Experiment key LWAZI_NCHLT_MAP_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
8000	46.368	24.688	322170	524528	342117
16000	49.782	29.916	302774	459720	301166
32000	57.188	37.874	234348	415440	299196
64000	61.59	47.482	228377	378335	222960
128000	65.392	52.608	202679	344103	202223
256000	67.868	55.936	188929	318862	188831
512000	69.356	57.916	180945	303476	181086
all	70.324	59.142	35133	58711	35393

Table B.14 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of Lwazi acoustic models using band-limited (250-3400 Hz) NCHLT adaptation data. The experiment is described by graph key LWAZI_NCHLT_MLLR_BP.

Table B.15 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NCHLT acoustic models using band-limited (250-3400 Hz) Lwazi adaptation data. The experiment is described by graph key NCHLT_LWAZI_MAP_BP.

Table B.16 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NCHLT acoustic models using band-limited (250-3400 Hz) Lwazi adaptation data. The experiment is described by graph key NCHLT_LWAZI_MLLR_BP.

Table B.17 shows the phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for NCHLT acoustic models trained on increasing amounts 16 kHz sampled NCHLT training data. The experiment is described by graph key NCHLT_RETRAIN_16k.

Table B.14: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of Lwazi acoustic models using band-limited (250-3400 Hz) NCHLT adaptation data. Experiment key LWAZI_NCHLT_MLLR_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
100	56.98	41.74	232589	419602	232025
250	57.882	42.862	237520	427397	237379
500	58.534	43.962	231846	422925	230463
1000	58.972	44.734	228474	419400	225394
2000	59.22	45.07	228033	415987	224012
4000	59.594	45.62	224188	413919	221229
8000	60.208	46.436	210138	393331	208549
16000	60.544	46.834	209695	389303	208829
32000	60.674	46.976	217209	403988	216731
64000	60.746	47.04	216982	403077	216836
all	60.94	46.97	42738	80669	44164

Table B.15: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for weights, means and variance MAP adaptation of NCHLT acoustic models using band-limited (250-3400 Hz) Lwazi adaptation data. Experiment key NCHLT_LWAZI_MAP_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
8000	54.452	40.29	39803	86451	39253
16000	53.664	39.094	40750	87687	40385
32000	57.31	43.55	38529	79794	38153
64000	63.418	51.286	33968	67440	33622
128000	68.618	57.812	29446	57549	29954
all	71.692	61.65	5350	10344	5565

Table B.16: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for mean and variance MLLR adaptation of NCHLT acoustic models using band-limited (250-3400 Hz) Lwazi adaptation data. Experiment key NCHLT_LWAZI_MLLR_BP.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
100	64.434	52.588	31901	66665	32822
250	64.654	53.032	31725	66243	32213
500	64.766	53.07	31604	66057	32415
1000	64.966	53.29	31664	65441	32373
2000	65.308	53.948	31157	65012	31482
4000	65.55	54.3	30941	64558	31175
8000	66.056	54.954	30551	63545	30765
16000	66.324	55.312	30370	62982	30516
32000	66.464	55.512	30290	62665	30360
64000	66.516	55.552	30327	62490	30384
all	66.696	55.538	5984	12479	6185

Table B.17: *Phone-level correctness, phone-level accuracies, deletion, substitution and insertion errors for NCHLT acoustic models trained on increasing amounts 16 kHz sampled NCHLT training data. Experiment key NCHLT_RETRAIN_16k.*

# Triphones	Phn Cor (%)	Phn Acc (%)	Del.	Sub.	Ins.
16000	60.614	45.99	226809	383423	226744
32000	63.404	50.624	220415	357421	202606
64000	66.71	55.336	201001	324685	180273
128000	70.35	60.012	167113	266999	152677
256000	73.392	63.736	162067	258379	152983
512000	75.706	66.484	150254	233769	146109
all	77.906	68.926	27406	42517	28417

APPENDIX C

DATA SELECTION VIA TRIPHONE ACCURACY EMPIRICAL MODELLING

C.1 INTRODUCTION

As stated in section (5.2.7) another approach to determine the triphone accuracy function is to empirically model the generated data which expresses the relationship between a triphone's accuracy and training amount. The difficulties in following this approach are:

- The graphs generated from this data are noisy and require smoothing to reveal the underlying structure which introduces errors into the modelling process.
- Choosing a proper model to represent the overall structure of the data is difficult and often requires non-linear optimisations to solve for parameters.

Though this approach did not yield improvements, the work will be summarised here for completeness.

C.2 EMPIRICAL TRIPHONE ACCURACY FUNCTION

To generate the data points capturing the relationship between the triphone accuracy and occurrence count, we used an ASR system trained on the BN corpus and the WSJ corpus as an evaluation set. Section (5.2.3) describes the BN and WSJ corpora, ASR system setup and approach used to calculate the triphone accuracies. Figure C.1 (A) shows the average triphone accuracy as a function of triphone training occurrence estimated using a BN trained ASR system and recognising the WSJ. Figure C.1 (B) shows the number of examples used to average the triphone accuracies. As noted previously, it

is difficult to get a sense of the underlying structure due to the later variations on the plot. To obtain a smoothed version of this graph a moving average filter was applied to the data where 100 sample-window was used. The smoothed graphs show the triphone accuracy and occurrence relationship is captured in figure C.2. The next step was to choose a functional form and estimate the parameters from the data.

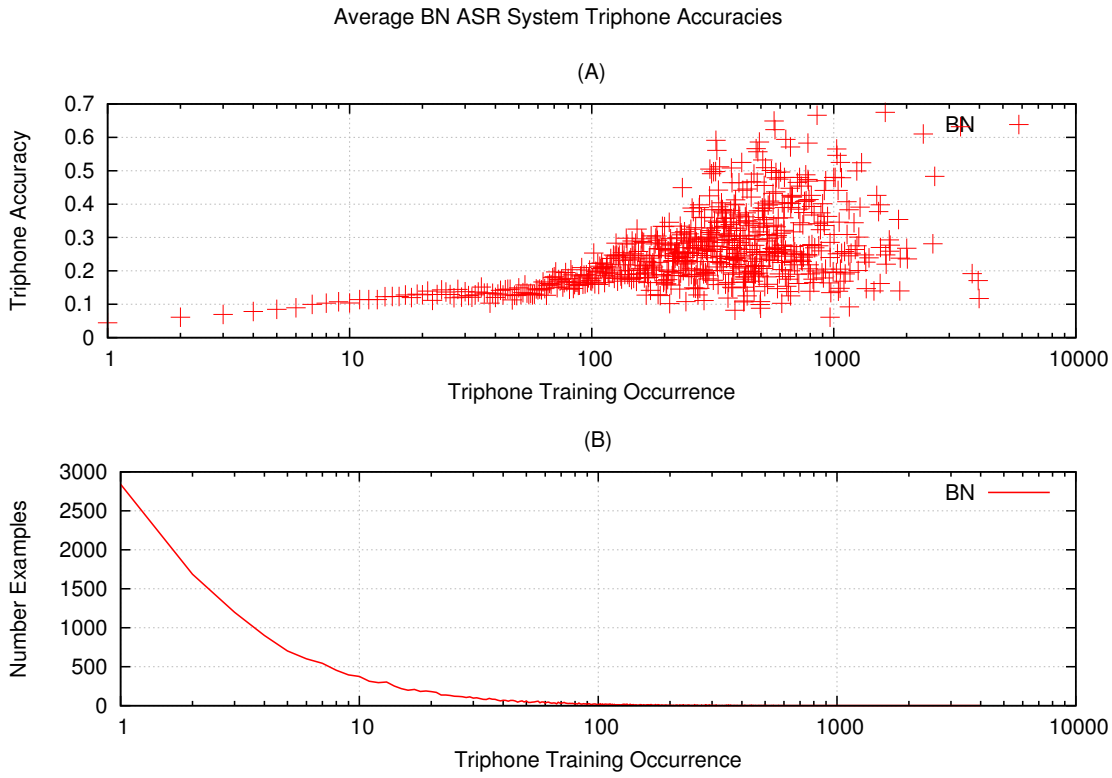

Figure C.1: *Graph (A) shows BN-derived triphone accuracy as a function of triphone training count using the WSJ corpus as an evaluation set. Graph (B) shows the number of examples used to average the triphone accuracies.*

Using the data points shown in figure C.1 (A) and a non-linear least squares curve fitting algorithm, from Matlab [76], we experimented with several functional forms and tried to attain a good fit for the data points. The final form settled on was a slight modification of the triphone accuracy function inspired by the learning theory asymptotic functional form (see equation (5.7)). The triphone accuracy function which gave the best fit was,

$$A_i(n_i) = B - \frac{C}{(n_i + D)}, \tag{C.1}$$

where the least-squares fit estimated the values $B = 0.3254, C = 32.9600, D = 129.3928$. Figure C.2 shows the resulting fit for the chosen functional form and estimated parameters.

The approximate fit (shown in green figure C.2) does model the data points quite well up to a training count of about 200. Above this, however, the large fluctuations in accuracy have a negative effect on the modelling process and the fitted function seems to underestimate the accuracies. This

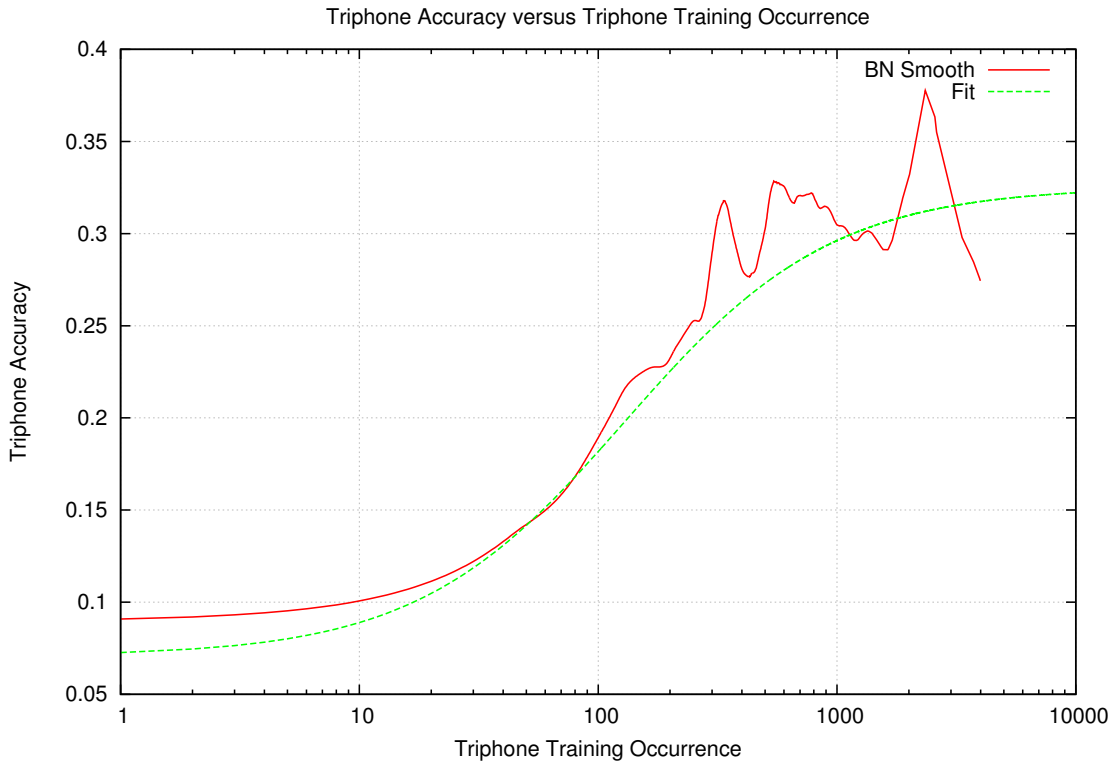

Figure C.2: *Data fit obtained using functional form given in equation (C.1) and smoothed version of figure C.1 (A) data points.*

could potentially lead to incorrect optimal triphone count assignments and may result in sub-optimal performance.

The benefit of using this functional form for the triphone accuracy function is that the derivative is easy to calculate and is given by,

$$\frac{\partial A_i(n_i)}{\partial n_i} = \frac{C}{(n_i + D)^2}. \tag{C.2}$$

To calculate the optimal individual triphone count we substitute equation (C.2) into equation (5.5). Thus:

$$\frac{p_i C}{(n_i + D)^2} + \lambda = 0. \tag{C.3}$$

Next, we expand the denominator and rearrange the equation to obtain a quadratic expression, expressed as

$$n_i^2 + 2Dn_i + (D^2 + \frac{p_i C}{\lambda}) = 0. \tag{C.4}$$

To solve for the optimal triphone frequencies we use the standard expression for the roots of a quadratic function, given by

$$n_i = \frac{-2D \pm \sqrt{(2D)^2 - 4(D^2 + \frac{p_i C}{\lambda})}}{2}. \tag{C.5}$$

This provides two potential roots. We can limit the solution to one of the roots as the triphone count n_i must be greater than or equal to zero, thus discarding all negative roots. The final triphone count n_i is dependent on the λ value which is shared by the set of triphone counts. Therefore, once we have obtained the solution for λ we can substitute this value into the set of equations and solve for each triphone count.

C.2.1 SOLVING OPTIMAL TRIPHONE COUNTS

To solve for λ we employed numerical techniques such as the *fzero* function provided by Matlab. This function tries to find a root for a single-variable continuous function [77] given an initial value or a search range. The user has to define the continuous function which has one dependent input variable, and *fzero* then alternates between bisection and interpolation to find the root of the defined function. In our case we defined the following objective function:

$$\sum_{i=1} N q_i(\lambda) - N = 0. \tag{C.6}$$

The input to the function is a λ value, which is mapped to a series of triphone counts n_i using equation (C.4) and limiting the solutions to positive integers $\mathbb{Z}^+$. Once all the triphone counts have been obtained they are summed together and subtracted from the total number of triphones that may appear in the training corpus. This value is returned and depending on the value *fzero* will terminate or alter the input λ value accordingly. After a solution is reached the triphone counts are finally calculated and mapped to the closest integer value. These are then the final optimal triphone counts.

C.3 EXPERIMENTAL SETUP

The experimental setup used to test the empirical triphone accuracy modelling approach is identical to the one described in section (5.3). For ease of reference, we assign the key `Opt Tri` to this method. We evaluated the `Opt Tri` data selection method on the Timit and WSJ corpora only.

C.4 RESULTS

A description for the max-entropy (`MaxEnt Wrd` and `MaxEnt Tri`), "compressed" (`Sqrt`, `Uniq Sqrt`) and "intermediate" (`0.75` and `Uniq 0.75`) data selection methods results are captured in section (5.4.1), section (5.4.2) and section (5.4.3). The results presented here, will therefore, be focused on the `Opt Tri` method and how this method performs compared to the other data selection approaches.

C.4.1 TIMIT

Table C.1 shows word correctness, word accuracies and statistical significance P-Values for Timit trained and evaluated systems which were trained on sub-corpora created by using various data selection methods and training data percentages. At the 20% percentage training data level the `Opt Tri` data selection methods achieves higher accuracies compared to the max-entropy based data selection approaches, however, the results are significantly worse compared to the `Natural` selection and remaining data selection approaches. For the 40%, 60% and 80% training data levels the `Opt Tri` methods produces a system word accuracy which falls between the `MaxEnt Tri` and `MaxEnt Wrd` data selection accuracies, which are all below the `Natural` data selection accuracies. At the 60% data percentage level the results are significantly worse compared to the `Natural` data selection accuracy.

Table C.1: *Word correctness, word accuracies and P-Value for Timit systems trained on various sub-corpora created using different data selection methods and data percentages and evaluated on the Timit evaluation set.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Word Cor (%)	47.57	44.35	46.24	47.97	48.73	49.51	48.44	45.96
	Word Acc (%)	40.88	37	37.75	42.09	42	43.36	42.1	38.53
	P-Value	-	3.32E-09	3.31E-06	4.46E-02	7.42E-02	6.63E-05	4.78E-02	2.82E-04
40%	Word Cor (%)	53.44	51.95	53.73	55.83	55.21	57.08	56.62	52.6
	Word Acc (%)	44.93	43.4	43.94	48.14	47.37	49.75	49.33	43.64
	P-Value	-	1.35E-02	1.41E-01	1.05E-07	4.48E-05	2.00E-15	8.94E-12	4.26E-02
60%	Word Cor (%)	57.71	55.4	57.75	59.32	58.74	59	59.55	56.21
	Word Acc (%)	49.29	45.95	48.47	51.44	50.67	51.01	51.78	46.41
	P-Value	-	3.27E-08	1.86E-01	1.22E-04	1.56E-02	2.91E-03	1.15E-05	3.42E-07
80%	Word Cor (%)	60.41	58.82	60.14	60.41	60.18	60.59	60.93	59.32
	Word Acc (%)	51.66	49.37	51.31	52.19	51.35	51.97	52.52	50.24
	P-Value	-	1.05E-04	5.47E-01	3.42E-01	5.75E-01	5.91E-01	1.28E-01	1.54E-02

Table C.2 shows triphone correctness, triphone accuracies and statistical significance P-Values for Timit trained and evaluated systems which were trained on sub-corpora created by using various data selection methods and training data percentages. At the 20% data percentage level the `Opt Tri` data selection method achieves a higher accuracy compared to `MaxEnt Tri`, `MaxEnt Wrd` approaches but the accuracy is significantly worst compared to the `Natural` technique. For the remaining data percentage levels `Opt Tri` performs better than the `MaxEnt Tri` approach but provides less of a gain compared to the `MaxEnt Wrd` data selection method. At the 60% data percentage level the decrease in triphone accuracy is significant. The `Opt Tri` approach provides consistently poorer measures when compared to `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` for all data percentages.

Table C.3 shows word correctness, word accuracies and statistical significance P-Values for ASR systems trained Timit data and evaluated on the WSJ evaluation corpus for different data selection methods and training data percentages. At 20% data percentage the `Opt Tri` data selection method produces the worst system accuracy which is a significant decrease from the `Natural` measures. For

Table C.2: *Triphone correctness, triphone accuracies and P-Value for Timit systems trained on various sub-corpora created using different data selection methods and data percentages and evaluated on the Timit evaluation set.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Tri Cor (%)	48.53	45.84	46.33	48.83	49.44	50.22	49.21	46.67
	Tri Acc (%)	44.89	40.96	42.31	45.57	46.04	47.12	45.71	42.45
	P-Value	-	4.00E-15	9.55E-07	1.23E-01	1.50E-02	3.07E-06	8.76E-02	8.25E-07
40%	Tri Cor (%)	53.66	52.15	52.88	55.42	55.32	56.48	56.11	52.5
	Tri Acc (%)	48.52	46.59	47.56	50.65	50.39	52.07	51.71	47.11
	P-Value	-	2.65E-05	4.92E-02	1.29E-06	2.04E-05	6.00E-15	1.08E-11	1.89E-03
60%	Tri Cor (%)	56.97	55.28	56.58	58.52	57.97	57.88	58.5	55.66
	Tri Acc (%)	51.71	49.35	51.22	53.46	52.84	52.83	53.57	49.73
	P-Value	-	5.47E-08	2.86E-01	3.09E-05	6.23E-03	9.16E-03	8.72E-06	1.73E-06
80%	Tri Cor (%)	59.38	58.03	59.13	59.71	59.29	59.46	59.65	58.6
	Tri Acc (%)	53.97	52.23	53.6	54.39	53.98	54.12	54.59	52.91
	P-Value	-	2.78E-05	3.89E-01	2.73E-01	9.77E-01	7.22E-01	1.21E-01	1.22E-02

40% data percentage `Opt Tri` only performs better than the max-entropy techniques but still significantly worse when compared to the `Natural` data selection method. At the 60% level `Opt Tri` achieves a system accuracy better than `MaxEnt Tri` but worse than `MaxEnt Wrd` and `Natural` methods which is significantly worse. Lastly, at the 80% `Opt Tri` data selection approach produces a gain in performance which is insignificantly better when compared to the `Natural` approach and comparable to the `0.75` and `Uniq Sqrt` techniques.

Table C.3: *Word correctness, word accuracies and P-Value results for Timit trained ASR system evaluated on the WSJ evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Word Cor (%)	41.45	39.38	40.49	41.71	42.24	41.58	41.62	38.89
	Word Acc (%)	34.22	31.21	32.04	34.89	35.54	35.16	34.69	30.72
	P-Value	-	1.00E-15	3.12E-09	5.89E-02	2.37E-04	7.13E-03	1.94E-01	0.00E+00
40%	Word Cor (%)	48.88	46.87	47.58	50.9	49.85	51.64	50.25	47
	Word Acc (%)	40.33	38.1	37.51	43.51	42.18	44.25	42.56	38.57
	P-Value	-	7.57E-09	1.02E-13	0.00E+00	4.54E-07	0.00E+00	1.27E-09	1.82E-06
60%	Word Cor (%)	52.18	49.57	52.04	53.72	53.34	53.41	53.48	50.04
	Word Acc (%)	43.67	39.81	42.62	45.61	45.23	45.19	45.11	40.89
	P-Value	-	0.00E+00	4.49E-03	2.34E-08	6.90E-06	1.49E-05	6.41E-05	4.80E-14
80%	Word Cor (%)	53.43	52.76	52.99	54.31	53.34	53.7	54.78	53.26
	Word Acc (%)	43.23	42.93	43.21	45.45	44.03	44.37	45.87	44.1
	P-Value	-	3.91E-01	9.53E-01	9.50E-11	1.49E-02	7.27E-04	1.30E-14	1.54E-02

Table C.4 shows triphone correctness, triphone accuracies and statistical significance P-Values for ASR systems trained Timit data and evaluated on the WSJ evaluation corpus for different data selection methods and training data percentages. At the 20% data percentage level the `Opt Tri` data selection method produces the lowest triphone accuracy which is significantly worst when compared to the `Natural` method. For the 40% data percentage level the `Opt Tri` approach manages to outperform both the `MaxEnt Tri` and `MaxEnt Wrd` techniques but produces a significant decrease

in triphone accuracy when compared to the `Natural` data selection method. At the 60% level the `Opt Tri` approach only beats the `MaxEnt Tri` data selection method but the accuracy is significantly worse compared to the `Natural` approach. For the last data percentage, `Opt Tri` approach produces a comparable triphone accuracy as compared to the `Natural` data selection method but is lower than the `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` methods' accuracies.

Table C.4: *Triphone correctness, triphone accuracies and P-Value results for Timit trained ASR systems evaluated on the WSJ evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Tri Cor (%)	46.2	45.28	44.96	47.05	47.91	47.65	47.1	44.33
	Tri Acc (%)	42.2	40.27	40.8	43.38	44.02	43.98	42.98	39.98
	P-Value	-	5.79E-12	4.26E-07	1.33E-05	7.03E-12	2.28E-11	3.03E-03	0.00E+00
40%	Tri Cor (%)	53.32	52.37	51.61	55.24	54.54	55.85	54.82	52.38
	Tri Acc (%)	48.06	46.39	45.63	50.12	49.32	51.12	50.03	46.57
	P-Value	-	1.15E-09	0.00E+00	1.00E-15	1.58E-06	0.00E+00	9.70E-14	1.44E-08
60%	Tri Cor (%)	56.41	54.35	55.72	57.54	57.36	57.17	57.15	54.81
	Tri Acc (%)	50.7	48.15	49.82	51.92	51.76	51.69	51.79	48.47
	P-Value	-	0.00E+00	4.33E-04	8.83E-07	1.72E-05	5.62E-05	1.42E-05	0.00E+00
80%	Tri Cor (%)	56.73	56.49	56.8	57.83	56.92	57.17	58.22	57.08
	Tri Acc (%)	50.67	50.17	50.57	51.89	50.92	51.15	52.46	50.76
	P-Value	-	3.35E-02	6.56E-01	1.82E-07	2.66E-01	3.73E-02	5.00E-15	6.98E-01

C.4.2 WSJ

Table C.5 shows word correctness, word accuracies and statistical significance P-Values for WSJ trained and evaluated ASR systems for different data selection methods and training data percentages. The `Opt Tri` data selection method produces comparable results when compared to `Natural` data selection approach with no statistically significant gains or losses. The approach also produces results which are comparable to the measures attained by the `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches.

Table C.5: *Word correctness, word accuracies and P-Value results for WSJ trained and evaluated ASR systems for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Word Cor (%)	60.01	58.58	59.57	59.82	60.31	59.95	59.61	59.51
	Word Acc (%)	51.61	49.62	50.35	51.67	52.03	51.65	51.11	51.24
	P-Value	-	3.58E-08	3.77E-04	8.65E-01	2.34E-01	9.23E-01	1.62E-01	3.05E-01
40%	Word Cor (%)	63.68	63.36	64.23	63.36	63.51	63.64	64.21	63.83
	Word Acc (%)	55.32	55.24	55.83	55.11	55.22	55.62	55.82	55.38
	P-Value	-	8.09E-01	1.41E-01	5.35E-01	7.49E-01	3.83E-01	1.35E-01	8.51E-01
60%	Word Cor (%)	66.3	65.74	66.34	66.14	66.54	66.5	66.78	65.89
	Word Acc (%)	58.76	58.07	58.36	58.3	58.93	58.64	59.24	57.81
	P-Value	-	2.79E-02	1.95E-01	1.34E-01	5.89E-01	6.90E-01	1.16E-01	1.76E-03
80%	Word Cor (%)	67.68	67.4	67.33	67.94	67.8	67.58	68.08	67.52
	Word Acc (%)	60.42	59.87	59.63	60.54	60.35	60.2	60.84	60.1
	P-Value	-	6.53E-02	4.10E-03	6.36E-01	8.28E-01	4.34E-01	1.19E-01	2.83E-01

Table C.6 shows triphone correctness, triphone accuracies and statistical significance P-Values for WSJ trained and evaluated ASR systems for different data selection methods and training data percentages. For the 20%, 40% and 80% data percentage levels the `Opt Tri` data selection methods produces comparable triphone accuracies as compared to the `Natural` approach. At the 60% data percentage level , however, it produces the worst triphone accuracy which is significantly lower than what the `Natural` data selection method achieves.

Table C.6: *Triphone correctness, triphone accuracies and P-Value results for WSJ trained and evaluated ASR systems for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Tri Cor (%)	63.39	62.66	63.22	63.14	63.4	63.28	63.2	63.45
	Tri Acc (%)	58.41	57.63	57.99	58.06	58.37	58.28	58.26	58.23
	P-Value	-	1.21E-03	8.67E-02	1.67E-01	8.61E-01	6.25E-01	6.12E-01	4.39E-01
40%	Tri Cor (%)	66.59	66.78	66.89	66.3	66.26	66.51	66.75	66.78
	Tri Acc (%)	61.41	61.69	61.76	60.95	61.01	61.29	61.57	61.37
	P-Value	-	2.05E-01	1.24E-01	4.71E-02	7.79E-02	6.00E-01	4.62E-01	8.43E-01
60%	Tri Cor (%)	68.89	68.55	68.62	68.38	68.86	68.69	68.99	68.41
	Tri Acc (%)	63.86	63.53	63.49	63.16	63.77	63.53	64	63.06
	P-Value	-	1.23E-01	6.95E-02	6.51E-04	6.25E-01	1.02E-01	5.05E-01	7.86E-05
80%	Tri Cor (%)	69.84	69.89	69.82	70.03	69.94	69.83	70.13	69.93
	Tri Acc (%)	64.85	64.8	64.71	65.11	64.88	64.77	65.18	64.77
	P-Value	-	8.10E-01	4.39E-01	1.56E-01	8.59E-01	6.67E-01	7.18E-02	6.70E-01

Table C.7 shows word correctness, word accuracies and statistical significance P-values for ASR systems trained on WSJ data and evaluated on the Timit evaluation set for various data selection approaches and training data percentages. At the 20% data percentage level the `Opt Tri` data selection method produces the second worse performance which is significant when compared to the `Natural` data selection method. The same trend is observed at the 40% level except that the decrease is not significant. For the 60% and 80% data percentage levels the `Opt Tri` performs better than the max-entropy methods but does not provide a gain over the `Natural` selection results, however, the results a comparable since the decreases are not significant.

Table C.7: *Word correctness, word accuracies and P-Value results for WSJ trained ASR systems evaluated on the Timit evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Word Cor (%)	55.05	52.26	53.68	55.61	55.69	55.75	55.56	52.99
	Word Acc (%)	44.37	40.41	42.17	45.4	44.9	45.73	45.35	42.1
	P-Value	-	1.69E-09	9.78E-04	1.29E-01	4.33E-01	3.41E-02	1.57E-01	4.27E-04
40%	Word Cor (%)	58.37	56.9	58.59	59.2	59.04	59.17	59.06	57.91
	Word Acc (%)	47.27	45.4	47.42	48.28	47.7	48.16	47.56	46.41
	P-Value	-	3.52E-03	8.31E-01	1.31E-01	4.93E-01	1.87E-01	6.70E-01	2.05E-01
60%	Word Cor (%)	60.29	58.7	58.97	60.97	60.81	61.58	61.05	59.26
	Word Acc (%)	49.74	47.6	47.09	50.47	49.6	50.84	50.19	47.94
	P-Value	-	5.73E-04	6.24E-06	2.08E-01	7.36E-01	7.21E-02	4.71E-01	2.48E-03
80%	Word Cor (%)	61.38	60.26	60.67	61.72	61.39	61.52	61.37	60.93
	Word Acc (%)	50.9	49.37	49.22	51.29	50.59	51.1	50.58	50.35
	P-Value	-	1.83E-02	6.52E-03	4.07E-01	6.61E-01	6.46E-01	6.31E-01	3.98E-01

Table C.8 shows triphone correctness, triphone accuracies and statistical significance P-values for ASR systems trained on WSJ data and evaluated on the Timit evaluation set for various data selection approaches and training data percentages. At the 20% and 40% data percentage levels the `Opt Tri` data selection method produces triphone accuracies which are better than the `MaxEnt Tri` approach but less than the accuracies produced be the `MaxEnt Wrd` approach. The accuracies are insignificantly less than the `Natural` data selection method's accuracies at those data percentage levels. At the 60% data percentage level the `Opt Tri` approach achieves the smallest triphone accuracy out of all the data selection methods which is significantly worse compared to the `Natural` technique. At the 80% mark the `Opt Tri` data selection methods produces a triphone accuracy which is comparable to the accuracy achieved by the `Natural` approach and better than the max-entropy data selection methods.

Table C.8: *Triphone correctness, triphone accuracies and P-Value results for WSJ trained ASR systems evaluated on the Timit evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Tri Cor (%)	53.42	51.58	52.79	54.25	54.31	54.45	54.41	52.15
	Tri Acc (%)	48.06	45.73	47.02	49.3	49	49.43	49.48	46.64
	P-Value	-	6.87E-07	2.80E-02	8.89E-03	5.20E-02	3.27E-03	2.67E-03	1.58E-03
40%	Tri Cor (%)	56.21	55.09	56.82	57.25	56.94	57.2	56.77	55.88
	Tri Acc (%)	50.32	49.02	50.69	51.38	51.11	51.45	51	49.49
	P-Value	-	4.15E-03	4.16E-01	1.60E-02	6.77E-02	1.46E-02	1.33E-01	7.79E-02
60%	Tri Cor (%)	58.1	57.11	57.15	58.55	58.35	59.15	58.93	56.96
	Tri Acc (%)	52.33	50.9	50.77	52.56	52.45	53.36	53.03	50.75
	P-Value	-	5.75E-04	1.45E-04	5.57E-01	8.14E-01	1.11E-02	8.91E-02	8.28E-05
80%	Tri Cor (%)	59.38	58.03	58.63	59.46	59.02	59.56	59.22	58.78
	Tri Acc (%)	53.38	52.04	52.38	53.47	52.98	53.59	53.11	52.56
	P-Value	-	1.45E-03	1.47E-02	7.44E-01	3.31E-01	5.32E-01	5.36E-01	4.55E-02

C.5 CONCLUSION

Based on the results presented in section (C.4.1) and section (C.4.2) the `Opt Tri` method is not an effective data selection approach when compared to the results obtained by the `Natural` data selection method. The performance of the `Opt Tri` data selection approach, in the majority of experiments, was comparable to the max-entropy-based data selection techniques. Considering word accuracies the proposed data selection method only managed to outperform the `Natural` data selection method for two experiments (40% WSJ train and evaluated, and, 80% Timit trained and WSJ evaluated) which were not statistically significant.

A final concern about the `Opt Tri` method is that it relies on detailed fits to the accuracy-training count curves, whereas the other methods do not require such detailed parameters. Since such values are likely to be quite different across different languages and applications, it is preferable to use methods such as `Natural` and `Sqrt` when designing a new corpus.

APPENDIX D

DATA SELECTION KL-DIVERGENCE INVESTIGATION

D.1 INTRODUCTION

In this appendix we investigate whether the observed accuracy differences, seen in the data selection results presented in section (5.4) and appendix (C) , can be explained by the correspondence between the training and evaluation distributions. In addition, we wanted to establish whether or not the word and triphone correctness measures, and, the word and triphone accuracy measures were generally in agreement.

D.2 KULLBACK – LEIBLER DIVERGENCE

The Kullback-Leibler divergence measures the difference between two distributions. The KL-divergence measure between two discrete distributions is given by,

$$D_{KL}(\mathbf{X} \parallel \mathbf{Y}) = \sum_i \log\left(\frac{X(i)}{Y(i)}\right) X(i), \tag{D.1}$$

where $X(i)$ and $Y(i)$ are the probabilities for symbol i in distributions X and Y respectively. The symmetric KL-divergence is given by $\frac{1}{2}\times[D_{KL}(\mathbf{X} \parallel \mathbf{Y})+D_{KL}(\mathbf{Y} \parallel \mathbf{X})]$ and will be used throughout this appendix to measure the similarity between the distributions.

D.3 RESULTS

In this section we present; (1) symmetric KL-divergence measures calculated between training and evaluation sets, (2) the total number of training triphones used to train the various ASR systems, and,

(3) Pearson and Spearman correlation coefficients measured between the correctness and accuracies measures as well as between the symmetric KL-divergence measures and the different correctness and accuracies. The evaluations contained in this section were performed on the various data selection experiments presented in section (5.4) and appendix (C) . The three corpora used for the evaluation were: American English Timit, American English WSJ and IsiZulu Lwazi. The experimental setup is described in section (5.3).

D.3.1 TIMIT

Table D.1 shows the symmetric KL-divergence measure between the different training and evaluation sets and the number of training triphones per data percentage and data selection method. Comparing the max-entropy columns, the distortions between the training and evaluation sets are larger compared to the natural selection method (`Natural`) – the differences tend to decrease as the data percentages increase. At the 20% and 40% training data percentages `Opt Tri` data selection method produces distributions which are less distorted compared to the max-entropy data selection methods but more distorted compared to the remaining approaches. At the 60% and 80% data percentage levels the `Opt Tri` data selection approach created less distorted distributions compared to the `MaxEnt Tri` but more distorted when compared to the remaining techniques. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` data selection methods all manage to produce lower divergence measures with the `Uniq` methods attaining slightly smaller distortions for all data percentages. `Uniq 0.75` approach produces the lowest distortions for all data percentages. The number of training triphones are all comparable with no strong deviations.

Table D.1: *Symmetric KL-divergence and the number of training triphones for Timit trained and evaluated ASR systems.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Sym KL-div	0.5749	0.6827	0.7276	0.5206	0.5238	0.5060	0.4959	0.6136
	# Training Triphones	24141	24127	24147	24136	24145	24163	24149	24138
40%	Sym KL-div	0.5620	0.6548	0.6574	0.4977	0.5261	0.4801	0.4742	0.6202
	# Training Triphones	48248	48263	48247	48266	48264	48249	48259	48285
60%	Sym KL-div	0.5555	0.6340	0.6118	0.5108	0.5227	0.4851	0.4782	0.6144
	# Training Triphones	72372	72377	72394	72404	72371	72399	72371	72374
80%	Sym KL-div	0.5530	0.6004	0.5747	0.5341	0.5293	0.5209	0.5133	0.5919
	# Training Triphones	96495	96504	96517	96514	96494	96504	96520	96501

Table D.2 shows Pearson and Spearman coefficients and the associated P-Value significances for correlations between selected measures obtained using Timit trained and evaluated systems. The Pearson correlations between the symmetric KL-divergence and word correctness, word accuracies, triphone correctness and triphone accuracies show a strong negative correlation but only the test between the word accuracies and symmetric KL-divergence passes the 0.001 significance level. The Spearman coefficients show a weak to low negative correlation between symmetric KL-divergence and word correctness and triphone correctness, while the word accuracies and triphone accuracies

show a low to moderate negative correlation towards the symmetric KL-divergence. The Spearman coefficients are insignificant. Lastly, both the Pearson and Spearman coefficients show significant very strong correlations between the word and triphone correctness and word and triphone accuracies.

Table D.2: *Pearson and Spearman correlation coefficients between selected measures obtained on Timit trained and evaluated ASR systems.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.4178	1.73E-02	-0.3488	5.03E-02
Word Acc & Sym KL-div	-0.5687	6.82E-04	-0.4527	9.89E-03
Tri Cor & Sym KL-div	-0.4410	1.15E-02	-0.3365	6.02E-02
Tri Acc & Sym KL-div	-0.5532	1.02E-03	-0.4173	1.746E-02
Word Cor & Tri Cor	0.9979	2.06E-37	0.9946	3.50E-31
Word Acc & Tri Acc	0.9978	4.29E-37	0.9911	7.64E-28

Table D.3 shows symmetric KL-divergence measures between training and evaluation sets and the number of training triphones for Timit trained ASR systems and evaluated on the WSJ evaluation set for various data selection methods and training data percentages. The triphone-based max-entropy selection method produces KL-divergence measures slightly lower than the `Natural` selection method at the 20% and 40% training data percentages but for the 60% and 80% level the distortions are higher. The word-based max-entropy data selection approach consistently attains the highest distortions. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches produce lower distortions compared to the `Natural` selection method, with the `Uniq 0.75` technique consistently producing the lowest distortions. At the 20% data percentages level `Opt Tri` data selection methods produces the second least distorted distribution, but for the remaining data percentages the `Opt Tri` data selection approach produces the second most distorted distribution. The training triphone amounts are comparable.

Table D.3: *Symmetric KL-divergence and the number of training triphones for Timit trained ASR systems evaluated on the WSJ evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Sym KL-div	0.6723	0.6596	0.7552	0.6612	0.6481	0.6354	0.6072	0.6339
	# Training Triphones	24141	24127	24147	24136	24145	24163	24149	24138
40%	Sym KL-div	0.6399	0.6398	0.7120	0.6183	0.6290	0.5911	0.5655	0.6449
	# Training Triphones	48248	48263	48247	48266	48264	48249	48259	48285
60%	Sym KL-div	0.6315	0.6452	0.6747	0.6086	0.6151	0.5723	0.5553	0.6477
	# Training Triphones	72372	72377	72394	72404	72371	72399	72371	72374
80%	Sym KL-div	0.6263	0.6345	0.6491	0.6253	0.6146	0.6025	0.5847	0.6402
	# Training Triphones	96495	96504	96517	96514	96494	96504	96520	96501

Table D.4 shows Pearson and Spearman correlation coefficients between selected performance measures obtained on Timit trained ASR systems which were evaluated on the WSJ evaluation set. The Pearson coefficients show a strong negative correlation between the word correctness, word accuracies, triphone correctness, triphone accuracies and the symmetric KL-divergence, however, only

the tests between the word accuracy and symmetric KL-divergence, and, triphone accuracy and symmetric KL-divergence are significant. The Spearman coefficients show significant moderate negative correlations exist between the word correctness and symmetric KL-divergence, word accuracies and symmetric KL-divergence, triphone correctness and symmetric KL-divergence, and, triphone accuracies and symmetric KL-divergence. Both the Pearson and Spearman coefficients show very strong correlation between the word and triphone correctness and accuracy measures.

Table D.4: *Pearson and Spearman correlation coefficients between selected measures obtained on Timit trained ASR systems evaluated on the WSJ evaluation set.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.5124	2.71E-03	-0.6039	2.51E-04
Word Acc & Sym KL-div	-0.5997	2.86E-04	-0.6656	4.94E-05
Tri Cor & Sym KL-div	-0.5424	1.34E-03	-0.6164	1.72E-04
Tri Acc & Sym KL-div	-0.6002	2.81E-04	-0.6748	3.60E-05
Word Cor & Tri Cor	0.9957	1.28E-32	0.9858	7.59E-25
Word Acc & Tri Acc	0.9976	1.43E-36	0.9956	0.00e+00

D.3.2 WSJ

Table D.5 shows symmetric KL-divergence measures between training and evaluation sets and the number of training triphones for ASR systems trained and evaluated on WSJ data for various data selection methods and training data percentages. The `Natural` data selection produces the lowest KL-divergence distortions when compared to all other data selection methods for all data percentages. For all data percentages the `Opt Tri` approach produces the third most distorted distribution which is comparable to the `Sqrt` data selection method and better compared to the `MaxEnt Tri` and `MaxEnt Wrd` data selection methods. The max-entropy data selection methods attain the highest distortions. The `Uniq` data selection methods produce slightly less distorted distributions when compared to their non-unique counterparts for all data percentages. All training triphone counts are quite similar.

Table D.5: *Symmetric KL-divergence and the number of training triphones for WSJ trained and evaluated ASR systems for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Sym KL-div	0.1441	0.2404	0.2089	0.1780	0.1487	0.1753	0.1441	0.1782
	# Training Triphones	136010	136056	136060	136002	136018	136013	135992	136004
40%	Sym KL-div	0.1322	0.1854	0.1655	0.1552	0.1412	0.1508	0.1347	0.1561
	# Training Triphones	272010	271955	271950	272005	271989	271946	271977	271947
60%	Sym KL-div	0.1282	0.1512	0.1421	0.1399	0.1354	0.1374	0.1303	0.1414
	# Training Triphones	407937	408001	407962	407967	407949	407919	407927	407933
80%	Sym KL-div	0.1258	0.1331	0.1314	0.1294	0.1282	0.1285	0.1261	0.1306
	# Training Triphones	543898	543930	543921	543970	543932	543896	543995	543906

Table D.6 shows Pearson and Spearman correlation coefficients for selected performance measures obtained on WSJ trained ASR systems evaluated on the WSJ evaluation set. Considering all comparisons made with the symmetric KL-divergence measures, the Pearson correlation coefficients show significant very strong negative correlations and similarly, the Spearman coefficients show significant strong negative correlations. The correlations between the word correctness and triphone correctness, and, the word accuracies and triphone accuracies, are very strong as reported by both the correlation coefficients.

Table D.6: *Pearson and Spearman correlation coefficients between selected measures obtained on WSJ trained and evaluated ASR systems.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.7531	6.55E-07	-0.8651	1.68E-10
Word Acc & Sym KL-div	-0.7658	3.25E-07	-0.8544	4.91E-10
Tri Cor & Sym KL-div	-0.7256	2.60E-06	-0.8182	1.06E-08
Tri Acc & Sym KL-div	-0.7289	2.22E-06	-0.8500	7.47E-10
Word Cor & Tri Cor	0.9970	5.11E-35	0.9777	6.73E-22
Word Acc & Tri Acc	0.9956	1.65E-32	0.9858	7.59E-25

Table D.7 shows a number of symmetric KL-divergence measures between the training and evaluation sets and the training triphone amount per data percentage and data selection method for WSJ

trained ASR systems evaluated on the Timit evaluation sets. The triphone-based max-entropy data selection method `MaxEnt Tri` produces training distributions which are the most distorted. For data percentages 20%, 40% and 60% `Opt Tri` data selection method generates distributions which result in the second most distorted measures, after the `MaxEnt Tri` technique. For the last data percentage, 80%, the `Opt Tri` distribution distortion is third most distorted after the `MaxEnt Tri` and `MaxEnt Wrd` approaches. The `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` methods generate distributions with lower distortions when compared to the `Natural` data selections with the non-unique selections producing slightly less distortions compared to the unique distortions. The `Sqrt` approach managed to attain the lowest distortions for all data percentages. The number of training triphones per data selection method and data percentages are comparable.

Table D.7: *Symmetric KL-divergence and the number of training triphones for WSJ trained ASR systems evaluated on the Timit evaluation set for various data selection methods and training data percentages.*

Percentage	Metric	Selection Type							
		Natural	MaxEnt Tri	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75	Opt Tri
20%	Sym KL-div	0.7076	1.0113	0.8158	0.5539	0.5841	0.5561	0.5873	0.8675
	# Training Triphones	136010	136056	136060	136002	136018	136013	135992	136004
40%	Sym KL-div	0.7196	0.9404	0.7984	0.5941	0.6131	0.5972	0.6176	0.8156
	# Training Triphones	272010	271955	271950	272005	271989	271946	271977	271947
60%	Sym KL-div	0.7275	0.8700	0.7783	0.6338	0.6415	0.6352	0.6462	0.7794
	# Training Triphones	407937	408001	407962	407967	407949	407919	407927	407933
80%	Sym KL-div	0.7336	0.8107	0.7589	0.6799	0.6806	0.6808	0.6842	0.7524
	# Training Triphones	543898	543930	543921	543970	543932	543896	543995	543906

Table D.8 shows the Pearson and Spearman correlation coefficients between selected measures which were obtained on WSJ trained ASR systems evaluated on the Timit evaluation set. The Pearson correlation coefficients between the word correctness and symmetric KL-divergence, and, between the triphone correctness and symmetric KL-divergence show insignificant weak negative correlations. The correlation between the symmetric KL-divergence and word accuracies and between the triphone accuracies, indicate an insignificant moderate negative and strong negative relationship respectively, as indicated by the Pearson correlation coefficients. The Spearman correlation coefficients show for all the symmetric KL-divergence comparisons insignificant negligible correlations. The Pearson and Spearman correlation coefficients show significant very strong correlations between the word and triphone correctness and accuracy measures.

Table D.8: *Pearson and Spearman correlation coefficients between selected measures obtained on WSJ trained ASR systems evaluated on the Timit evaluation set.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.2981	9.74E-02	-0.1818	3.17E-01
Word Acc & Sym KL-div	-0.3835	3.02E-02	-0.1895	2.98E-01
Tri Cor & Sym KL-div	-0.2895	1.07E-01	-0.1337	4.63E-01
Tri Acc & Sym KL-div	-0.4063	2.10E-02	-0.2225	2.20E-01
Word Cor & Tri Cor	0.9960	4.14E-33	0.9824	0.00E+00
Word Acc & Tri Acc	0.9942	1.28E-30	0.9862	5.12E-25

D.3.3 LWAZI

Table D.9 shows the symmetric KL-divergence measures and triphone training amounts for Lwazi trained ASR systems evaluated on the Lwazi evaluation data using various data selection approaches and training data percentages. The `MaxEnt Wrd` data selection method consistently produced distributions with the greatest distortions for all data percentages. The `Natural` data selection approaches produces distortions which are less the the `MaxEnt Wrd`, `Sqrt` and `Uniq Sqrt` methods. The `0.75` techniques achieves the lowest distortions for 20%, 40% and 60% data percentages and `Uniq 0.75` attains the lowest distortion for 80% data percentage. The triphones counts are all comparable with no major deviations from the norm.

Table D.9: *Symmetric KL-divergence measures and triphone training amounts for Lwazi trained ASR systems evaluated on Lwazi evaluation data. Different data percentages and data selection methods were used to create various training corpora.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Sym KL-div	0.1276	0.4618	0.1311	0.1042	0.1335	0.1123
	# Training Triphones	35883	35412	35452	35672	35460	35641
40%	Sym KL-div	0.1112	0.2481	0.1163	0.1011	0.1153	0.1050
	# Training Triphones	71694	71084	71040	71289	71074	71281
60%	Sym KL-div	0.1071	0.1802	0.1117	0.1022	0.1090	0.1052
	# Training Triphones	107527	106946	106884	106984	106918	107055
80%	Sym KL-div	0.1064	0.1395	0.1099	0.1052	0.1073	0.1048
	# Training Triphones	143429	142951	142867	142889	142872	142936

Table D.10 shows Pearson and Spearman correlation coefficients for selected measures obtained on Lwazi trained ASR systems and evaluated on the Lwazi evaluation sets. The Pearson correlation coefficients shows an insignificant strong negative correlation exists between the word correctness and symmetric KL-divergence, and, between the triphone correctness and symmetric KL-divergence. The correlation between the word accuracies and symmetric KL-divergence is significant strong negative relationship, while the correlation between the triphone accuracies and symmetric KL-divergence is significant and very strong. The Spearman correlation coefficients show for all the symmetric KL-divergence comparisons an insignificant moderate to low negative correlation. The reported correlations between the word correctness and triphone correctness, and, between the word accuracies and triphone accuracies are very strong and significant.

Table D.10: *Pearson and Spearman correlation coefficients between measures which were obtained on Lwazi trained and evaluated systems.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.6191	1.25E-03	-0.4644	2.22E-02
Word Acc & Sym KL-div	-0.6594	4.56E-04	-0.4618	2.30E-02
Tri Cor & Sym KL-div	-0.6177	1.29E-03	-0.4718	1.99E-02
Tri Acc & Sym KL-div	-0.7042	1.22E-04	-0.4696	2.05E-02
Word Cor & Tri Cor	0.9996	1.21E-35	0.9965	7.96E-07
Word Acc & Tri Acc	0.9907	1.41E-20	0.9869	1.00E-06

Table D.11 shows the symmetric KL-divergence measures and triphone training amounts for Lwazi trained ASR systems evaluated on AST evaluation data using various data selection approaches and training data percentages. The `MaxEnt Wrd` and `Natural` data selection approaches produce the most and second most distorted distributions respectively. The `Uniq Sqrt` data method achieves the lowest distortions and `Sqrt` the second lowest distortions. The triphones counts per data percentage are comparable.

Table D.11: *Symmetric KL-divergence measures and number of training triphones for Lwazi trained ASR systems evaluated on the AST evaluation set for different data percentages and data selection methods used to create various training corpora.*

Percentage	Metric	Selection Type					
		Natural	MaxEnt Wrd	Sqrt	0.75	Uniq Sqrt	Uniq 0.75
20%	Sym KL-div	1.0672	1.3706	0.9652	0.9934	0.9586	0.9894
	# Training Triphones	35883	35412	35452	35672	35460	35641
40%	Sym KL-div	1.1077	1.2630	1.0316	1.0511	1.0241	1.0355
	# Training Triphones	71694	71084	71040	71289	71074	71281
60%	Sym KL-div	1.1354	1.2423	1.0822	1.0894	1.0725	1.0873
	# Training Triphones	107527	106946	106884	106984	106918	107055
80%	Sym KL-div	1.1553	1.2132	1.1274	1.1304	1.1222	1.1283
	# Training Triphones	143429	142951	142867	142889	142872	142936

Table D.12 shows the Pearson and Spearman correlation coefficients between selected measures which were obtained on ASR systems trained on Lwazi data and evaluated on the AST evaluation set. The Pearson correlation coefficients show insignificant negligible relationships exist between all the symmetric KL-divergence comparisons which were compared with the word and triphone correctness and accuracy measures. The Spearman correlation coefficients, show an insignificant moderate to low correlation between, the word correctness and symmetric KL-divergence, between the word accuracies and symmetric KL-divergence, and, between the triphone correctness and symmetric KL-divergence. The relationship between the triphone accuracies and the symmetric KL-divergence is insignificantly low to weak as indicate by the Spearman correlation coefficient. Lastly, a significant very strong relationship is found between the word and triphone correctness and accuracy comparisons as reported by the Pearson and Spearman correlation coefficients.

Table D.12: *Pearson and Spearman correlation coefficients between selected measures obtained on Lwazi trained ASR systems evaluated on the AST evaluation set.*

	Pearson Coefficient	P-Value	Spearman Coefficient	P-Value
Word Cor & Sym KL-div	-0.0541	8.01E-01	0.3826	6.58E-02
Word Acc & Sym KL-div	-0.0356	8.68E-01	0.3182	1.29E-01
Tri Cor & Sym KL-div	-0.0632	7.69E-01	0.3747	7.19E-02
Tri Acc & Sym KL-div	-0.2489	2.40E-01	0.1756	4.09E-01
Word Cor & Tri Cor	0.9986	1.18E-29	0.9947	8.32E-07
Word Acc & Tri Acc	0.9593	1.43E-13	0.9452	1.98E-06

D.4 CONCLUSION

The work presented in this appendix investigated various relationships between the word and triphone correctness and accuracy results presented in section (5.4) and appendix (C) . The main aspects under investigation were;

- whether the observed ASR systems word and triphone correctness and accuracy differences, for systems trained on various data selected corpora, were related to the similarity between the training and evaluation corpora, and,
- whether the word and triphone correctness, and, word and triphone accuracies were generally in agreement.

From the results presented in section (D.3), we may conclude:

- Consistently, for all experiments it was shown that the relationship between word correctness and triphone correctness, and, between word accuracies and triphone accuracies are significant and very strong.
- For the Timit trained ASR systems;
  - For the Timit evaluated systems the correlations between the symmetric KL-divergence and the different word and triphone correctness and accuracies, ranged from weak-low (Spearman) to strong (Pearson), however, only the test between the word accuracies and symmetric KL-divergence is significant.
  - When the systems were evaluated using the WSJ evaluation set, the correlations between the symmetric KL-divergence and various correctness and accuracy measures were moderate (Spearman) to strong (Pearson) where the majority of tests are significant. Only the test between word correctness and symmetric KL-divergence, and, triphone correctness and symmetric KL-divergence were insignificant.
- For the WSJ trained ASR systems;

  - The relationship between the symmetric KL-divergence and word or triphone correctness and accuracy measures were significantly strong (Spearman) to very strong (Pearson) when using the WSJ evaluations.
  - For the Timit evaluations, the symmetric KL-divergence correlations were insignificant and ranged form negligible (Spearman) to moderate (Pearson).

- For the Lwazi trained ASR systems;

  - For Lwazi evaluations we found: (1) insignificant strong negative linear correlations between the word correctness and symmetric KL-divergence, and, between the triphone correctness and symmetric KL-divergence, (2) significant strong negative linear relationship between the word accuracies and symmetric KL-divergence, (3) significant and very strong linear correlation between the triphone accuracies and symmetric KL-divergence, and, (4) insignificant moderate to low non-linear correlations for all the symmetric KL-divergence comparisons.
  - For AST evaluations we found: insignificant negligible linear correlations for all symmetric KL-divergence comparisons, while insignificant moderate to low non-linear correlations for the same comparison, except for the symmetric KL-divergence and triphone accuracy relationship which was low to weak.

- On average the `Sqrt`, `0.75`, `Uniq Sqrt` and `Uniq 0.75` approaches produce the lowest distortions between the training and evaluations sets. The notable differences was the WSJ trained and evaluated system where the `Natural` data selection methods produced the lowest distortions.

- Regarding the `Opt Tri` data selection method: the symmetric KL-divergence measures also showed that the selected training distributions, when compared to the evaluation sets, were in the majority of cases more distorted compared to the distributions created by the `Natural` selection method. In addition, for a number of the experiments distortions were comparable to the max-entropy distribution distortions. As established in section (5.4) the max-entropy based selections were the worst performing data selection methods.

Thus, our main conclusion is: we found that the word and triphone accuracies are indeed highly correlated, and that on average the divergences between the training and evaluation corpora were indeed negatively correlated with the accuracy and correctness achieved – the lower the KL-divergence measures the higher the correctness and accuracies. However, there are also cases in which lower KL-divergence does not imply higher accuracy, so this does not seem to be the only factor that influences the accuracy achieved – this is summarised in table D.13.

Table D.13: *The data selection methods which produced the best word accuracies and lowest KL-divergence for the various training sets, evaluation sets and data percentages.*

Training Corpus	Evaluation Corpus	Data Percentage	Best Word Accuracy Method	Lowest KL-divergence Method
Timit	Timit	20	Uniq Sqrt	Uniq 0.75
Timit	Timit	40	Uniq Sqrt	Uniq 0.75
Timit	WSJ	20	0.75	Uniq 0.75
Timit	WSJ	40	Uniq Sqrt	Uniq 0.75
Timit	WSJ	60	Sqrt	Uniq 0.75
WSJ	WSJ	20	0.75	Natural / Uniq 0.75
WSJ	WSJ	40	MaxEnt Wrd	Natural
WSJ	WSJ	60	Uniq 0.75	Natural
WSJ	WSJ	80	Uniq 0.75	Natural
WSJ	Timit	20	Uniq Sqrt	Sqrt
WSJ	Timit	60	Uniq Sqrt	Sqrt
Lwazi	Lwazi	20	Natural	0.75
Lwazi	Lwazi	40	Natural	0.75
Lwazi	Lwazi	60	Natural	0.75
Lwazi	AST	20	Sqrt	Uniq Sqrt
Lwazi	AST	40	Sqrt	Uniq Sqrt
Lwazi	AST	60	0.75	Uniq Sqrt
Lwazi	AST	80	Sqrt	Uniq 0.75

APPENDIX E

LWAZI FOLD EXPERIMENTS

E.1 INTRODUCTION

In section (5.3.2.1) we introduced the IsiZulu Lwazi corpus which was used during the investigation of the various data selection approaches. The corpus does not have dedicated training, development and evaluation sets. Therefore to create an evaluation set and obtain a reliable results when using the corpus, the corpus was split into ten cross-validation folds. The statistics regarding the folds are given in section (5.3.2.1). The results presented in section (5.4.3) are averages and thus in this appendix we list the various results obtain on the individual folds.

E.2 RESULTS

The data selection approaches were evaluated on ASR systems trained on the Lwazi training folds and each ASR system was evaluated using the specific Lwazi evaluation fold and the AST IsiZulu corpus. The AST corpus details are presented in section (5.3.2.2). The experimental setup is presented in section (5.3) and the various system performance metrics are described in section (5.3.7). In this section we list the Lwazi and AST evaluation results per fold.

E.2.1 LWAZI EVALUATION

Table E.1 shows word correctness results, obtained using different data selection approaches and training data percentages, for Lwazi ASR systems trained on data sourced from the Lwazi fold-specific training sets and evaluated on the corresponding Lwazi fold-specific evaluation set.

Table E.2 shows word accuracies by fold, data selection method and training data percentage, for various Lwazi ASR systems trained on fold training sets and evaluated on the corresponding Lwazi fold evaluation set.

Table E.1: *Word correctness results for fold-specific Lwazi-trained ASR systems evaluated on the fold-specific Lwazi evaluation sets.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	45.29	45.28	45.32	40.56	32.03	33.94	36.04	38.59	38.73	42.43
	MaxEnt Wrd	30.03	32.73	29.29	30.28	24.3	21.94	26.74	25.51	25.58	29.16
	Sqrt	40.52	40.47	40.74	36.77	30.09	32.87	34.33	35.74	37.86	40.2
	0.75	44.25	43.12	42.88	38.75	32.51	33.05	35.47	37.36	39.13	41.62
	Uniq Sqrt	39.81	39.94	40.52	35.74	30.65	30.35	34.52	34.15	35.05	38.45
	Uniq 0.75	44.28	43.49	42.08	37.2	31.07	31.87	34.44	35.42	36.37	41.6
40%	Natural	52.04	52.17	49.22	47.2	38.1	38.65	40.8	44.61	45.1	49.65
	MaxEnt Wrd	47.25	47.68	44.24	44.96	35.32	35.35	38.99	40.5	40.45	45.2
	Sqrt	50.46	51.51	49.02	44.88	37.75	38.36	39.97	42.28	44.89	47.87
	0.75	51.12	51.03	50.87	44.85	37.63	38.96	40.07	42.78	45.29	49.33
	Uniq Sqrt	49.4	50.38	49.19	43.37	37.22	37.73	40.57	41.69	43.43	46.74
	Uniq 0.75	49.46	50.19	49.08	44.34	36.86	36	40.45	41.8	44.05	48.46
60%	Natural	54.9	54.95	52.86	50.07	42.71	41.74	44.8	47.55	49.13	52.85
	MaxEnt Wrd	52.72	54.6	51.44	49.48	39.93	40.77	44.32	46.06	46.95	51.28
	Sqrt	54.9	54.76	53.85	49.85	42.48	40.59	45.69	47.38	49.46	52.1
	0.75	55.04	55.16	53.85	49.53	42.3	41.27	45.55	46.16	50	52.63
	Uniq Sqrt	53	55.57	53.71	49.48	40.67	40.93	44.6	47.11	49.85	52.12
	Uniq 0.75	54.54	53.85	54.9	49.1	40.05	41.11	44.03	46.49	48.52	51.31
80%	Natural	57.17	58.13	56.44	52.32	44.54	43.26	47.78	49.79	51.17	54.1
	MaxEnt Wrd	57.52	58.47	54.71	52.66	43.72	43.96	47.09	49.32	51.16	54.39
	Sqrt	57.49	58.56	56.78	53.15	43.88	43.18	46.95	49.97	51.83	54.26
	0.75	57.57	58.36	57.29	52.61	44.28	42.94	47.87	50.45	53.13	54.63
	Uniq Sqrt	57.44	58.83	56.95	52.74	43.3	43.44	47.18	49.47	52.76	54.98
	Uniq 0.75	57.74	57.88	56.98	53.49	44.2	43.47	47.38	50.16	52.67	54.07

Table E.3 shows the number of training triphones used to train the Lwazi ASR systems. The triphone counts are dependent on the training data percentage, data selection method and fold.

Table E.4 shows the symmetric KL-divergence measures between the Lwazi training datasets and Lwazi evaluation sets.

Table E.2: *Word accuracies results for fold-specific Lwazi-trained ASR systems evaluated on the fold-specific Lwazi evaluation sets.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	26.38	18.49	22.95	11.59	6.86	6.62	10.82	18.61	11.48	21.36
	MaxEnt Wrd	8.12	6.02	2.27	1.93	-0.21	1.05	1.29	8.62	-1.19	9.63
	Sqrt	20.82	12.26	15.84	9.47	7.74	7.96	9.88	17.38	11.97	19.69
	0.75	23.65	18.35	17	11.34	10.56	8.35	9.36	18.31	13.57	21.25
	Uniq Sqrt	17.82	12.45	15.5	8.73	7.42	8.33	9.97	15.33	11.1	17.21
	Uniq 0.75	23.11	16.63	18.06	9.47	9.02	7.91	9.68	15.48	9.18	20.97
40%	Natural	30.19	24.59	24.03	16.12	12.68	9.74	11.91	23.29	16.03	25.87
	MaxEnt Wrd	23.92	18.75	16.75	13.56	6.62	10.5	10.31	18.99	11.49	21.86
	Sqrt	29.1	23.53	21.54	12.41	10.73	11.18	9.22	20.85	16.41	24.85
	0.75	28.8	22.26	22.96	13.64	10.85	10.42	9.17	23.04	16.71	24.81
	Uniq Sqrt	27.71	23.58	22.55	13.31	10.35	9.69	10.65	21.24	15.03	23.78
	Uniq 0.75	27.33	22.7	23.8	12.33	12.65	10.4	10.2	19.71	14.84	26.05
60%	Natural	35.12	26.4	26.47	18.96	16.62	13.75	14.46	26.28	20.18	30.63
	MaxEnt Wrd	30.11	26.37	24.4	17.6	11.88	11.94	13.11	25.81	17.36	27.15
	Sqrt	31.8	27.74	26.33	19.15	15.79	11.44	15.01	25.76	20.82	28.64
	0.75	32.86	26.46	26.36	17.92	16.15	11.78	16.27	24.45	19.83	29.87
	Uniq Sqrt	31.58	28.42	27.58	18.72	12.66	13.04	15.52	26.41	20.83	28.55
	Uniq 0.75	32.92	24.86	29.6	18.11	12	13.33	16.16	25.05	21.36	28.87
80%	Natural	35.97	31.12	29.91	21.19	17.79	13.43	19.41	29.3	21.67	31.02
	MaxEnt Wrd	37.87	31.86	27.95	21.56	15.73	14.69	17.15	28.87	22.9	31.32
	Sqrt	35.94	32.79	30.4	22.3	16.82	14.22	17.69	29.12	22.86	30.95
	0.75	37.11	31.09	31.05	21.91	16.89	14.38	18.71	30.8	26.11	30.69
	Uniq Sqrt	36.24	32.51	29.57	22.01	16.14	13.35	18.72	28.88	25.49	32.6
	Uniq 0.75	37.68	30.02	31.7	23.72	18.54	14.51	18.84	30.21	24.92	29.83

E.2.2 AST EVALUATION

Table E.5 shows word correctness results, obtained using different data selection approaches and training data percentages, for Lwazi ASR systems trained on data sourced from the Lwazi fold-specific training sets and evaluated on the AST evaluation set.

Table E.6 shows word accuracies by fold, data selection method and training data percentage, for various Lwazi ASR systems trained on fold-specific training sets and evaluated on the AST evaluation set.

Table E.7 shows the symmetric KL-divergence measures between the Lwazi fold-specific training datasets and the AST evaluation set.

Table E.3: *The number of training triphones per fold and for various data selection methods used to train the Lwazi ASR systems.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	35897	36077	35842	35729	36285	35618	35942	35884	35717	35846
	MaxEnt Wrd	35266	35820	35445	35222	35562	35328	35511	35324	35218	35429
	Sqrt	35314	35782	35530	35228	35704	35258	35518	35413	35295	35481
	0.75	35533	35979	35721	35429	35942	35471	35810	35602	35506	35727
	Uniq Sqrt	35352	35812	35550	35210	35696	35307	35575	35349	35287	35463
	Uniq 0.75	35530	35981	35631	35433	35916	35495	35714	35553	35477	35683
40%	Natural	71529	72102	71876	71219	72250	71382	71782	71554	71416	71830
	MaxEnt Wrd	70872	71900	71089	70620	71504	70730	71263	70867	70684	71315
	Sqrt	70854	71746	71096	70534	71581	70689	71177	70864	70704	71163
	0.75	71069	71982	71369	70818	71803	70912	71443	71146	70976	71378
	Uniq Sqrt	70880	71739	71128	70596	71615	70707	71273	70870	70777	71162
	Uniq 0.75	71098	71989	71440	70779	71768	70937	71421	71075	70950	71357
60%	Natural	107154	108441	107807	106813	108404	106882	107608	107313	107019	107832
	MaxEnt Wrd	106614	108002	107091	106087	107682	106278	107267	106582	106583	107282
	Sqrt	106501	107901	106998	106118	107728	106339	107171	106641	106416	107030
	0.75	106630	107989	107068	106279	107793	106446	107224	106744	106526	107148
	Uniq Sqrt	106568	107990	106986	106172	107747	106374	107176	106671	106426	107076
	Uniq 0.75	106741	108100	107146	106342	107793	106522	107329	106802	106557	107223
80%	Natural	143074	144834	143446	142401	144561	142687	143732	143058	142751	143752
	MaxEnt Wrd	142444	144409	143054	141963	143947	142233	143382	142519	142360	143202
	Sqrt	142381	144269	142951	141909	143931	142108	143206	142540	142282	143098
	0.75	142378	144313	143019	141928	143956	142088	143216	142562	142284	143147
	Uniq Sqrt	142411	144269	143000	141884	143910	142104	143244	142571	142274	143057
	Uniq 0.75	142520	144353	143088	141986	143961	142192	143309	142596	142286	143076

Table E.4: *The symmetric KL-divergence measures for various Lwazi fold-specific training and evaluation sets.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	0.1326	0.1322	0.1320	0.1413	0.1299	0.1231	0.1173	0.1193	0.1239	0.1247
	MaxEnt Wrd	0.4512	0.4580	0.4798	0.4382	0.4547	0.4844	0.4392	0.4618	0.4642	0.4865
	Sqrt	0.1272	0.1299	0.1313	0.1487	0.1257	0.1319	0.1268	0.1283	0.1283	0.1326
	0.75	0.0998	0.1053	0.1043	0.1171	0.1036	0.1106	0.0972	0.1025	0.0992	0.1028
	Uniq Sqrt	0.1311	0.1326	0.1312	0.1508	0.1280	0.1333	0.1308	0.1296	0.1324	0.1348
	Uniq 0.75	0.1053	0.1155	0.1158	0.1233	0.1122	0.1154	0.1050	0.1084	0.1094	0.1122
40%	Natural	0.1079	0.1192	0.1182	0.1208	0.1135	0.1167	0.1078	0.1016	0.1051	0.1012
	MaxEnt Wrd	0.2349	0.2589	0.2533	0.2571	0.2347	0.2572	0.2377	0.2485	0.2464	0.2522
	Sqrt	0.1109	0.1156	0.1143	0.1316	0.1162	0.1164	0.1139	0.1133	0.1138	0.1171
	0.75	0.0956	0.1029	0.1029	0.1133	0.1021	0.1043	0.0965	0.0988	0.0961	0.0980
	Uniq Sqrt	0.1088	0.1150	0.1128	0.1298	0.1138	0.1152	0.1127	0.1120	0.1150	0.1178
	Uniq 0.75	0.0986	0.1083	0.1056	0.1159	0.1064	0.1088	0.0982	0.1031	0.1005	0.1042
60%	Natural	0.1017	0.1147	0.1089	0.1161	0.1107	0.1163	0.1021	0.0999	0.0994	0.1012
	MaxEnt Wrd	0.1719	0.1887	0.1814	0.1840	0.1754	0.1967	0.1720	0.1752	0.1795	0.1777
	Sqrt	0.1034	0.1127	0.1100	0.1263	0.1119	0.1139	0.1108	0.1082	0.1076	0.1124
	0.75	0.0951	0.1044	0.1031	0.1139	0.1046	0.1069	0.0999	0.0985	0.0965	0.0994
	Uniq Sqrt	0.0991	0.1107	0.1088	0.1220	0.1081	0.1115	0.1073	0.1064	0.1060	0.1101
	Uniq 0.75	0.0972	0.1088	0.1075	0.1144	0.1074	0.1108	0.0995	0.1020	0.1003	0.1038
80%	Natural	0.0996	0.1139	0.1097	0.1130	0.1101	0.1182	0.1012	0.0993	0.0984	0.1005
	MaxEnt Wrd	0.1288	0.1519	0.1438	0.1443	0.1388	0.1476	0.1319	0.1370	0.1371	0.1342
	Sqrt	0.1002	0.1138	0.1105	0.1209	0.1104	0.1155	0.1088	0.1058	0.1047	0.1080
	0.75	0.0965	0.1091	0.1069	0.1154	0.1073	0.1121	0.1023	0.1017	0.0982	0.1021
	Uniq Sqrt	0.0976	0.1112	0.1096	0.1168	0.1089	0.1132	0.1049	0.1035	0.1018	0.1056
	Uniq 0.75	0.0959	0.1088	0.1087	0.1123	0.1076	0.1120	0.0996	0.1016	0.0989	0.1024

Table E.5: *Word correctness results for fold-specific Lwazi-trained ASR systems evaluated on the AST evaluation set.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	32.87	31.66	32.27	34.35	28.96	31.02	29.45	29.43	31.23	31.08
	MaxEnt Wrd	18.88	21.01	21.68	22.94	23.52	24.76	23.14	23.02	19.59	20.46
	Sqrt	33.53	28.36	31.22	30.36	29.53	29.75	31.53	28.64	31.17	31.42
	0.75	29.81	30.76	29.31	29.73	29.56	29.47	28.98	30.17	29.07	29.85
	Uniq Sqrt	27.09	27.68	27.19	30.09	27.59	28.24	29.08	29.94	27.38	29.14
	Uniq 0.75	29.15	31.43	29.46	28.65	27.04	30.59	29.42	29.55	30.99	27.92
40%	Natural	38.61	35.98	37.28	38.55	36.73	36.39	37.98	37.59	35.35	37.19
	MaxEnt Wrd	30.23	31.75	31.08	30.23	30.68	33.57	31.84	30.8	30.3	30.48
	Sqrt	36.77	37.75	37.69	35.84	37.77	36.15	36.74	36.37	37.78	36.6
	0.75	35.05	34.14	36.6	36.85	37.62	37.73	36.86	36.06	36.03	37.23
	Uniq Sqrt	35.85	33.64	34.76	34.75	37.32	36.6	35.93	36.56	37.04	37.36
	Uniq 0.75	34.76	33.63	34.54	36.31	36.07	35.41	35.48	35.35	35	35.33
60%	Natural	36.69	38.66	40.43	40.85	40.87	38.36	40.62	41.35	40.13	39
	MaxEnt Wrd	34.59	34.89	33.92	34.41	33.61	33.19	33.71	33.42	35.03	33.21
	Sqrt	40.55	40.08	41.31	40.02	39.54	38.46	38.44	40.42	41.51	40.59
	0.75	39.97	39.26	40.79	40.39	38.37	38.47	39.46	39.42	40.8	41.72
	Uniq Sqrt	38.41	38.24	40.42	39.42	40.31	40.43	39.49	37.9	40.13	41.23
	Uniq 0.75	37.99	40.24	40.29	38.66	38.45	38.08	40.16	39.55	39.07	39.89
80%	Natural	40.8	39.44	39.92	41.26	39.92	40.9	40.95	39.99	41.05	41.23
	MaxEnt Wrd	39.24	38.91	38.71	38.53	37.92	36.75	39.38	38.32	39.04	38.53
	Sqrt	40.89	40.46	41.1	41.36	41.22	40.57	41.6	40.25	41.43	40.31
	0.75	41.13	40	40.97	40.24	40.47	41.36	40.89	39.2	40.26	40.08
	Uniq Sqrt	40.55	40.27	41.13	39.5	41.56	41.05	40.77	39.51	40.17	39.94
	Uniq 0.75	38.72	39.83	41.48	39.79	40.16	41.44	40.8	41.17	40.02	41.06

Table E.6: *Word accuracies results for fold-specific Lwazi trained ASR systems evaluated on the AST evaluation set.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	18.7	17.69	15.27	21.06	12.38	14.1	14.72	13.79	14.06	16.67
	MaxEnt Wrd	7.89	9.64	11.28	11.91	14.42	13.43	12.49	13.63	10.3	10.23
	Sqrt	20.59	14.6	17.83	15.51	17.6	14.23	19.29	14.97	17.53	18.71
	0.75	14.99	15.48	13	17.99	12.33	14.9	14.18	15.53	13.63	14.97
	Uniq Sqrt	13.08	14.7	13.43	18.46	11.99	13.18	15.99	16.27	14.51	15.35
	Uniq 0.75	13.67	15.74	12.55	13.63	12.62	15.02	14.35	14.15	16.44	14.17
40%	Natural	22.35	21.7	19.77	20.93	22.35	20.59	21.63	22.71	16.35	20.42
	MaxEnt Wrd	15.66	16.88	17.39	12.89	16.36	21.15	17.28	14.75	14.86	15.62
	Sqrt	21.3	23.87	23.27	20.23	22.89	20.24	20.6	22.17	23.59	22.02
	0.75	18.26	18.16	20.21	21.01	22.66	22.32	20.67	21.12	21.02	23.02
	Uniq Sqrt	20.36	19.2	18.16	20.39	22.17	19.96	21.26	20.8	23.15	23.22
	Uniq 0.75	20.34	19.37	17.92	21.01	20.65	19.05	19.17	19.98	20.18	21.26
60%	Natural	20.74	23.49	25.99	23.55	26.45	21.3	25.34	25.28	23.94	22.55
	MaxEnt Wrd	17.78	16.24	16.54	16.45	14.98	13.84	13.38	13.56	16.57	15.81
	Sqrt	24.7	24.86	26.45	25.34	24.8	22.46	21.94	23.15	26.16	24.95
	0.75	24.99	23.96	25.4	24.83	23.09	21.45	24.57	24.74	25.49	27.71
	Uniq Sqrt	22.09	22.18	25.28	23.85	23.96	23.35	23.13	21.42	24.69	26.36
	Uniq 0.75	21.97	25.51	26.32	22	23.57	21.02	23.42	24.69	24.73	24
80%	Natural	25.07	22.93	23.1	26.51	24.32	24.58	25.63	23.46	24.45	26.26
	MaxEnt Wrd	24.5	24.59	22.4	21.84	23.33	16.9	24.11	22.77	24.14	22.52
	Sqrt	25.99	25.74	25.82	26.15	27.17	25.16	26.11	24.21	25.79	24.51
	0.75	26.37	25.67	26.18	24.82	25.57	24.88	24.53	23.54	25.78	24.79
	Uniq Sqrt	24.18	24.17	26.42	23.47	25.57	25.42	24.93	24.34	25.28	23.77
	Uniq 0.75	22.86	24.36	25.24	22.94	24.35	23.95	24.22	25.32	23.67	24.43

Table E.7: *The symmetric KL-divergence measures for fold-specific Lwazi training and the AST evaluation set for various data percentages and data selection methods.*

Percentage	Selection Type	Folds									
		1	2	3	4	5	6	7	8	9	10
20%	Natural	1.0546	1.0592	1.0581	1.0729	1.0784	1.0629	1.0581	1.0805	1.0743	1.0730
	MaxEnt Wrd	1.3641	1.3574	1.3784	1.3649	1.3796	1.3761	1.3608	1.3770	1.3695	1.3777
	Sqrt	0.9629	0.9603	0.9701	0.9645	0.9607	0.9686	0.9695	0.9640	0.9662	0.9650
	0.75	0.9896	1.0008	0.9979	0.9963	0.9908	0.9934	0.9917	0.9869	0.9896	0.9974
	Uniq Sqrt	0.9584	0.9560	0.9663	0.9570	0.9583	0.9613	0.9578	0.9585	0.9556	0.9570
	Uniq 0.75	0.9861	0.9899	0.9970	0.9857	0.9873	0.9907	0.9925	0.9882	0.9863	0.9907
40%	Natural	1.1041	1.0994	1.1074	1.1089	1.1176	1.1044	1.1117	1.1043	1.1101	1.1088
	MaxEnt Wrd	1.2614	1.2511	1.2645	1.2714	1.2647	1.2672	1.2595	1.2650	1.2608	1.2642
	Sqrt	1.0290	1.0321	1.0345	1.0307	1.0328	1.0314	1.0302	1.0321	1.0323	1.0306
	0.75	1.0517	1.0499	1.0525	1.0498	1.0502	1.0502	1.0530	1.0508	1.0516	1.0515
	Uniq Sqrt	1.0227	1.0237	1.0267	1.0219	1.0260	1.0212	1.0250	1.0254	1.0246	1.0241
	Uniq 0.75	1.0342	1.0348	1.0386	1.0362	1.0369	1.0334	1.0334	1.0357	1.0367	1.0354
60%	Natural	1.1325	1.1276	1.1336	1.1369	1.1408	1.1451	1.1343	1.1288	1.1370	1.1374
	MaxEnt Wrd	1.2476	1.2359	1.2382	1.2430	1.2429	1.2461	1.2413	1.2441	1.2422	1.2420
	Sqrt	1.0800	1.0830	1.0819	1.0814	1.0837	1.0812	1.0828	1.0819	1.0830	1.0828
	0.75	1.0876	1.0901	1.0896	1.0894	1.0902	1.0889	1.0899	1.0901	1.0895	1.0890
	Uniq Sqrt	1.0702	1.0723	1.0756	1.0691	1.0753	1.0706	1.0729	1.0718	1.0748	1.0728
	Uniq 0.75	1.0855	1.0868	1.0899	1.0859	1.0889	1.0864	1.0871	1.0864	1.0893	1.0873
80%	Natural	1.1521	1.1536	1.1527	1.1557	1.1578	1.1613	1.1541	1.1476	1.1584	1.1597
	MaxEnt Wrd	1.2132	1.2116	1.2111	1.2115	1.2134	1.2135	1.2142	1.2161	1.2143	1.2132
	Sqrt	1.1252	1.1264	1.1279	1.1258	1.1285	1.1286	1.1280	1.1269	1.1287	1.1279
	0.75	1.1284	1.1296	1.1307	1.1297	1.1313	1.1310	1.1312	1.1304	1.1319	1.1296
	Uniq Sqrt	1.1204	1.1218	1.1235	1.1206	1.1234	1.1224	1.1226	1.1217	1.1236	1.1224
	Uniq 0.75	1.1263	1.1271	1.1306	1.1271	1.1297	1.1284	1.1280	1.1277	1.1296	1.1290

APPENDIX F

LIST OF MATHEMATICAL SYMBOLS

Symbols used in the "Background" chapter

S_n	n^{th} speech sample
c_i	i^{th} cepstral coefficient
Mel(f)	Mel value at frequency f
m_j	j^{th} filter bank energy
Δ_t	cepstral derivative at time t
τ	time-shift parameter used in cepstral derivative calculation
$\mathbf{O}$	speech vector observations
$\mathbf{o}_t$	speech vector at time t
$\hat{\mathbf{W}}$	probable word sequence
$P(\mathbf{W} \mid \mathbf{O})$	probability of the word sequence given the observed speech vector sequence
$P(\mathbf{W})$	probability of the word sequence occurring
$\boldsymbol{\mu}$	mean vector
$\boldsymbol{\Sigma}$	covariance matrix
$\boldsymbol{\mu}_j$	j^{th} mixture mean vector
$\boldsymbol{\Sigma}_j$	j^{th} mixture covariance matrix
$b_j(\mathbf{o}_t)$	j state output probability given observation $\mathbf{o}_t$
a_{ij}	transition probability from state i to state j
$\boldsymbol{\mu}_j$	state j mean vector
$\boldsymbol{\Sigma}_j$	state j covariance matrix
$L_{jm}(t)$	state j component m occupation probability
$\alpha_i(t)$	state i forward probability at time t
$\beta_j(t)$	state j backward probability at time t
$\phi_j(t)$	partial log likelihood of state j at time t
w_N	word at position N
V	vocabulary
$\mathbf{M}$	mean adaptation matrix
ξ	extended mean vector
$\mathbf{H}$	covariance transformation matrix
$\mathbf{C}$	Cholesky factors of an inverse covariance matrix
$\mathbf{B}$	inverse of the Cholesky factors of an inverse covariance matrix
$\boldsymbol{\Sigma}_{diag}$	diagonal covariance matrix

Symbols used in the "Cross channel adaptation" chapter

$c_m^i(t)$	zero mean i^{th} cepstral coefficient at time t
$c_v^i(t)$	zero mean and variance normalised to one i^{th} cepstral coefficient at time t
$c_a^i(t)$	zero mean, variance normalised to one and ARMA filtered i^{th} cepstral coefficient at time t
$\boldsymbol{X}$	data vector series
$\boldsymbol{X}_i$	i^{th} data vector
$E[\mathbf{X}]$	the expected value of X
$\boldsymbol{\mu}$	mean vector
$\boldsymbol{\Sigma}$	covariance matrix
$\mathcal{N}(\mathbf{0}, \mathbf{I})$	Normal distribution with zero mean vector and unit covariance matrix
$\mathbf{Z}_{src}$	source data vector
$\mathbf{Z}_{tgt}$	target data vector
$\mathbf{Z}_{zero}$	normalized data vector
$\mathbf{A}_{src}$	lower triangular matrix of Cholesky decomposition of source covariance matrix
$\mathbf{A}_{tgt}$	lower triangular matrix of Cholesky decomposition of target covariance matrix
$\mathbf{A}_{src}^{-1}$	inverse lower triangular matrix of Cholesky decomposition of source covariance matrix
$\mathbf{A}_{tgt}^{-1}$	inverse lower triangular matrix of Cholesky decomposition of target covariance matrix
$\mathbf{A}_{src}^{T}$	transposed lower triangular matrix of Cholesky decomposition of source covariance matrix
$\mathbf{A}_{tgt}^{T}$	transposed lower triangular matrix of Cholesky decomposition of target covariance matrix
$\boldsymbol{\mu}_{src}$	mean vector estimated on source data
$\boldsymbol{\mu}_{tgt}$	mean vector estimated on target data
$\boldsymbol{\Sigma}_{src}$	covariance matrix estimated on source data
$\boldsymbol{\Sigma}_{tgt}$	covariance matrix estimated on target data
τ	prior information weight

Symbols used in the "Efficient data selection for ASR" chapter

r	Pearson correlation coefficient
ρ	Spearman correlation coefficient
X_i	i^{th} raw score
Y_i	i^{th} raw score
$\overline{X}$	mean of X raw scores
$\overline{Y}$	mean of Y raw scores
x_i	rank of the i^{th} raw scores X
y_i	rank of the i^{th} raw scores Y
$\overline{x}$	average x rank values
$\overline{y}$	average y rank values
t_x	triphone x
A_{total}	overall ASR system accuracy
A_i	accuracy of the i^{th} triphone
$\frac{\partial A_{total}}{\partial n_i}$	partial derivative of total accuracy A_{total} with respect to triphone count n_i
$\frac{\partial A_i(n_i)}{\partial n_i}$	partial derivative of triphone accuracy A_i with respect to triphone count n_i
N	total triphone count
n_i	i^{th} triphone count
p_i	occurrence probability of the triphone *i*
q_i	modified occurrence probability of the triphone *i*
λ	Lagrange multiplier
Z	random variable for matched-pairs statistical significance test
mu_z	average difference in the number of errors made in a segment
σ_z	standard deviation of the difference in the number of errors made in a segment
$\mathcal{N}(0,1)$	standard normal distribution
H_0	null hypothesis
H_1	alternative hypothesis

REFERENCES

[1] Quinnipiac University Department of Political Science, "Pearsons R Correlation," 2013.

[2] BOLD Educational Software, "Spearman Rank Correlation," 2013.

[3] B. Erol, J. Cohen, M. Etoh, H. W. Hon, J. Luo, and J. Schalkwyk, "Mobile media search," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Taipei, Taiwan, May 2009, IEEE, pp. 4897–4900.

[4] L. R. Rabiner, "Applications of speech recognition in the area of telecommunications," in *Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on*, Santa Barbara, California, USA, December 1997, IEEE, pp. 501–510.

[5] D. A. Reynolds, "Automatic speaker recognition: Current approaches and future trends," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Salt Lake City, Utah, USA, May 2001, IEEE, pp. 1–6.

[6] J. Navratil, "Spoken language recognition-a step toward multilinguality in speech processing," *Speech and Audio Processing, IEEE Transactions on*, vol. 9, no. 6, pp. 678–685, 2001.

[7] Y. Wu, R. Zhang, and A. Rudnicky, "Data selection for speech recognition," in *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, Pittsburgh, Pennsylvania, USA, December 2007, IEEE, pp. 562–565.

[8] H. Bourlard, H. Hermansky, and N. Morgan, "Towards increasing speech recognition error rates," *Speech communication*, vol. 18, no. 3, pp. 205–231, 1996.

[9] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[10] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D.. Povey, V. Valtchev, and P. Woodland, "The HTK Book. revised for HTK version 3.4," March 2009, `http://htk.eng.cam.ac.uk//`.

[11] F. Grézl and P. Fousek, "Optimizing bottle-neck features for LVCSR," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, Nevada, USA, March 2008, IEEE, pp. 4729–4732.

[12] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Orlando, Florida, USA, May 2002, IEEE, vol. 1, pp. 105–108.

[13] S. Young, "Acoustic modelling for large vocabulary continuous speech recognition," *NATO ASI SERIES F COMPUTER AND SYSTEMS SCIENCES*, vol. 169, pp. 18–39, 1999.

[14] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 4, pp. 357–366, 1980.

[15] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *The Journal of the Acoustical Society of America*, vol. 87, pp. 1738, 1990.

[16] B. Milner, "A comparison of front-end configurations for robust speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Norwich, UK, May 2002, IEEE, pp. 797 – 800.

[17] M. Frikha1, A. B. Hamida, and M. Lahiani, "Hidden markov models (HMMs) isolated word recognizer with the optimization of acoustical analysis and modeling techniques," *International Journal of the Physical Sciences*, vol. 6, no. 22, pp. 5064–5074, 2011.

[18] B. H. Juang, L. Rabiner, and J. Wilpon, "On the use of bandpass liftering in speech recognition," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, no. 7, pp. 947–954, 1987.

[19] B. A. Hanson and T. H. Applebaum, "Robust speaker-independent word recognition using static, dynamic and acceleration features: Experiments with lombard and noisy speech," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Mexico, USA, April 1990, IEEE, pp. 857–860.

[20] J. B. Michel, Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, J. P. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. A. Nowak, and E. L. Aiden, "Quantitative analysis of culture using millions of digitized books," *science*, vol. 331, no. 6014, pp. 176, 2011.

[21] J. J. Odell, V. Valtchev, P. C. Woodland, and S. J. Young, "A one pass decoder design for large vocabulary recognition," in *Proceedings of the workshop on Human Language Technology*, Plainsboro, NJ, USA, March 1994, Association for Computational Linguistics, pp. 405–410.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[23] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1994, pp. 307–312.

[24] S. J. Young, "The general use of tying in phoneme-based HMM speech recognisers," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, California, USA, March 1992, IEEE, vol. 1, pp. 569–572.

[25] S. Young, "A review of large-vocabulary continuous-speech," *Signal Processing Magazine, IEEE*, vol. 13, no. 5, pp. 45, 1996.

[26] L. Lamel, J. L. Gauvain, and G. Adda, "Lightly supervised and unsupervised acoustic model training," *Computer Speech & Language*, vol. 16, no. 1, pp. 115–129, 2002.

[27] T. Hughes, K. Nakajima, L. Ha, A. Vasu, P. Moreno, and M. LeBeau, "Building transcribed speech corpora quickly and cheaply for many languages," in *Proceedings of INTERSPEECH*, Makuhari, Japan, September 2010, pp. 1914–1917.

[28] M. Meng, S. Wang, J. Liang, P. Ding, and B. Xu, "Full utilization of closed-captions in broadcast news recognition," *Proc. IS-CSLP*, December 2006.

[29] E. Barnard, M. Davel, and C. van Heerden, "ASR corpus design for resource-scarce languages," in *Proceedings of INTERSPEECH*, Brighton, United Kingdom, September 2009, ISCA, pp. 2847–2850.

[30] C. van Heerden, E. Barnard, and M. H. Davel, "Basic speech recognition for spoken dialogues," in *Proceedings of INTERSPEECH*, Brighton, UK, September 2009, pp. 3003–3006.

[31] T. J. Hazen, "Automatic alignment and error correction of human generated transcripts for long speech recordings," in *Proceedings of INTERSPEECH*, Pittsburgh, Pennsylvania, USA, September 2006, ISCA, pp. 1606–1609.

[32] P. J. Moreno, C. Joerg, J. M. V. Thong, and O. Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Proceedings of INTERSPEECH*, Sydney, Australia, November 1998, ISCA.

[33] L. ten Bosch and L. Boves, "Survey of spontaneous speech phenomena in a multimodal dialogue system and some implications for ASR," in *Proceedings of INTERSPEECH*, Jeju Island, Korea, October 2004.

[34] P. J. Moreno and R. M. Stern, "Sources of degradation of speech recognition in the telephone network," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Adelaide, Australia, April 1994, IEEE, vol. 1, pp. 109–112.

[35] C. P. Chen and J. A. Bilmes, "MVA processing of speech features," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 257–270, 2007.

[36] D. Gelbart and N. Morgan, "Evaluating long-term spectral subtraction for reverberant ASR," in *Automatic Speech Recognition and Understanding, 2001. ASRU'01. IEEE Workshop on*. IEEE, December 2001, pp. 103–106.

[37] J. C. Segura, C. Benítez, A. de La Torre, A. J. Rubio, and J. Ramírez, "Cepstral domain segmental nonlinear feature transformations for robust speech recognition," *Signal Processing Letters, IEEE*, vol. 11, no. 5, pp. 517–520, 2004.

[38] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer speech and language*, vol. 9, no. 2, pp. 171, 1995.

[39] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, no. 4, pp. 249–264, 1996.

[40] J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *Speech and Audio Processing, IEEE Transactions on*, vol. 2, no. 2, pp. 291–298, 1994.

[41] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," *Speech and Audio Processing, IEEE Transactions on*, vol. 7, no. 3, pp. 272–281, 1999.

[42] S. Goronzy and R. Kompe, "A combined MAP+MLLR approach for speaker adaptation," in *In Proceedings of the the Sony Research Forum*, 1999, vol. 99, pp. 9–14.

[43] Z. Wang, T. Schultz, and A. Waibel, "Comparison of acoustic model adaptation techniques on non-native speech," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hong Kong, Hong Kong, April 2003, IEEE, vol. 1, pp. 540–543.

[44] D. R. S. Caon, A. Amehraye, J. Razik, G. Chollet, R. V. Andreao, and C. Mokbel, "Experiments on acoustic model supervised adaptation and evaluation by k-fold cross validation technique," in *I/V Communications and Mobile Network (ISVC), 2010 5th International Symposium on*, Rabat, Morocco, September 2010, IEEE, pp. 1–4.

[45] R. Wallace, K. Thambiratnam, and F. Seide, "Unsupervised speaker adaptation for telephone call transcription," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009, IEEE, pp. 4393–4396.

[46] E. Bocchieri, M. Riley, and M. Saraclar, "Methods for task adaptation of acoustic models with limited transcribed in-domain data," in *Proceedings of INTERSPEECH*, Jeju Island, Korea, October 2004, ISCA, pp. 2953–2956.

[47] A. Nagórski, L. Boves, and H. Steeneken, "Optimal selection of speech data for automatic speech recognition systems," in *Proceedings of INTERSPEECH*, Denver, Colorado, USA, September 2002, ISCA, pp. 2473–2476.

[48] R. K. Moore, "A comparison of the data requirements of automatic speech recognition systems and human listeners," in *Proceedings of EUROSPEECH*, Geneva, Switzerland, September 2003, pp. 2582–2584.

[49] J. P. H. Santen and A. L. Buchsbaum, "Methods for optimal text selection," in *Proceedings of EUROSPEECH*, Rhodes, Greece, September 1997, ISCA, pp. 553–556.

[50] E. Gouvêa and M. H. Davel, "Kullback-Leibler divergence-based ASR training data selection," in *Proceedings of INTERSPEECH*, Florence, Italy, August 2011, pp. 2297–2300.

[51] M. H. Davel, C. van Heerden, N. Kleynhans, and E. Barnard, "Efficient harvesting of Internet audio for resource-scarce ASR," in *Proceedings of INTERSPEECH*, Florence, Italy, August 2011, ISCA, pp. 3153–3156.

[52] K. Brandenburg and G. Stoll, "The ISO-MPEG-1 audio: A generic standard for coding of high-quality digital audio," *Journal of the Audio Engineering Society*, vol. 42, no. 10, pp. 780–792, 1994.

[53] "Sox eXchange," 24 April 2012, `http://sox.sourceforge.net/Main/HomePage`.

[54] "Resource Interchange File Format Services," 24 April 2012, `http://msdn.microsoft.com/en-us/library/ms713231`.

[55] M. Davel and E. Barnard, "Pronunciation prediction with Default&Refine," *Computer Speech & Language*, vol. 22, no. 4, pp. 374–393, 2008.

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

[57] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[58] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian Information Criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*. Virginia, USA, 1998, p. 8.

[59] A. Acero and R. M. Stern, "Environmental robustness in automatic speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Albuquerque, New Mexico, USA, April 1990, IEEE, pp. 849–852.

[60] N.J. de Vries, J. Badenhorst, M. H. Davel, E. Barnard, and A. de Waal, "Woefzela - An open-source platform for ASR data collection in the developing world," in *Proceedings of INTERSPEECH*, Florence, Italy, August 2011, pp. 3176–3179.

[61] P. Boersma and D. Weenink, "Praat: doing phonetics by computer [computer program]," *Version*, vol. 5, pp. 21, 2005.

[62] L. F. Lamel and J. L. Gauvain, "High performance speaker-independent phone recognition using CDHMM," in *Proceedings of EUROSPEECH*, Berlin, Germany, September 1993, ISCA, pp. 121–124.

[63] E. Barnard, M. Davel, C. van Heerden, N. Kleynhans, and K. Bali, "Phone recognition for spoken web search," in *CEUR Workshop Proceedings, MediaEval 2011 Multimedia Benchmark Workshop*, Pisa, Italy, September 2011, Virginia, USA, vol. 807, p. 8.

[64] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, no. 1, pp. 133–147, 1998.

[65] N. Kleynhans and E. Barnard, "A channel normalization technique for speech recognition in mismatched conditions," in *Proceedings of the Annual Symposium of the Pattern Recognition Association of South Africa*, Cape Town, South Africa, November 2008, pp. 115–118.

[66] C. Jankowski, A. Kalyanswamy, S. Basson, and J. Spitz, "NTIMIT: A phonetically balanced, continuous speech, telephone bandwidth speech database," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Albuquerque, New Mexico, USA, April 1990, IEEE, pp. 109–112.

[67] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital signal processing*, vol. 10, no. 1, pp. 19–41, 2000.

[68] R. Schwartz, Y. Chow, S. Roucos, M. Krasner, and J. Makhoul, "Improved hidden Markov modeling of phonemes for continuous speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, San Diego, California, USA, March 1984, vol. 9, pp. 21–24.

[69] MATLAB, "Matlab Function: corr - linear or rank correlation," 2013.

[70] E. Barnard, "A model for nonpolynomial decrease in error rate with increasing sample size," *Neural Networks, IEEE Transactions on*, vol. 5, no. 6, pp. 994–997, 1994.

[71] W. M. Fisher, G. R. Doddington, and K. M. Goudie-Marshall, "The DARPA speech recognition research database: specifications and status," in *Proc. DARPA Workshop on Speech Recognition*, 1986, pp. 93–99.

[72] Meraka Institute, "Lwazi ASR corpus," 2013.

[73] J. C. Roux, P. H. Louw, and T. R. Niesler, "The African Speech Technology project: an assessment," in *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May 2004, pp. 93–96.

[74] T. R. Niesler, "Language-dependent state clustering for multilingual acoustic modelling," *Speech Communication*, vol. 49, no. 6, pp. 453–463, June 2007.

[75] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Glasgow, Scotland, May 1989, IEEE, vol. 1, pp. 532–535.

[76] "lsqcurvefit," 24 April 2012, `http://www.mathworks.com/help/toolbox/optim/ug/lsqcurvefit.html`.

[77] "fzero," 24 April 2012, `http://www.mathworks.com/help/techdoc/ref/fzero.html`.