MYANMAR LANGUAGE CONTINUOUS SPEECH RECOGNITION USING CONVOLUTIONAL NEURAL NETWORK (CNN)

AYE NYEIN MON

UNIVERSITY OF COMPUTER STUDIES, YANGON

JANUARY, 2019

Myanmar Language Continuous Speech Recognition Using Convolutional Neural Network (CNN)

Aye Nyein Mon

University of Computer Studies, Yangon

A thesis submitted to the University of Computer Studies, Yangon in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

January, 2019

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

…..…… ……..……

Date Aye Nyein Mon

ACKNOWLEDGEMENTS

First and foremost, I would like to thank the Union Minister, the Ministry of Education, for allowing me the advanced study and providing full facilities during the Ph.D Course at the University of Computer Studies, Yangon.

Secondly, I would like to thank Dr. Mie Mie Thet Thwin, Rector of the University of Computer Studies, Yangon, for giving me an opportunity to do this doctoral research.

I sincerely wish to express my greatest pleasure and special thanks to my supervisor, Dr. Win Pa Pa, Professor, Natural Language Processing Lab., the University of Computer Studies, Yangon, for her keen interest, invaluable guidance, suggestions, constructive comments and encouragement throughout the course of this study. I would like to mention my special thanks to my co-supervisor, Dr. Ye Kyaw Thu, a Visiting Professor of Language and Semantic Technology Research Team (LST), Artificial Intelligence Research Unit (AINRU), National Electronics and Computer Technology Center (NECTEC), for his kindness, invaluable suggestions and assistances to my effective research completion throughout my study period.

I deeply and specially thank the external examiner, Professor Dr. Nwe Nwe Win, Vice-President, Myanmar Computer Federation (MCF), for her patience in critical reading, valuable suggestions and comments in the preparation of thesis.

I wish to extend special thanks to Dr. Khine Moe Nwe, Professor of the University of Computer Studies, Yangon, for her careful guidance and encouragement during my study. I am very thankful to Dr. Khin Mar Soe, Professor, Head of Natural Language Processing Lab., the University of Computer Studies, Yangon, for her valuable supports in doing research.

I would like to express my respectful gratitude to Daw Aye Aye Khine, Head of English Department, the University of Computer Studies, Yangon, for her overall supporting throughout of my Ph.D course work and doing research.

Last but not least, my deepest and heartfelt appreciation goes to my beloved parents and my younger sister for their constant encouragement, patience, financial and moral supports, and understanding throughout my study.

ABSTRACT

Researchers of many nations have developed automatic speech recognition (ASR) to show their national improvement in information and communication technology for their languages.

The dissertation aims to develop good quality Myanmar language automatic speech recognition on read speech. Myanmar language is being considered as a low-resourced language. Thus, there is no speech corpus which is freely and commercially available for ASR research. Therefore, a speech corpus named "University of Computer Studies Yangon - Speech Corpus (UCSY-SC1)" which is essential for Myanmar ASR research is constructed. The speech corpus is developed by using two types of domains: web news and daily conversations. The news is collected from the Internet and the conversational data is recorded by ourselves. This corpus is applied to build the Myanmar ASR.

Myanmar language is one of the tonal languages and different types of tones convey the difference in meanings. Therefore, like the other tonal languages such as Mandarin, Vietnamese and Thai, tone information is significantly played to improve the Myanmar ASR performance. Moreover, syllable is the basic unit of Myanmar language. Thus, in this work, the effect of tones is explored on both syllable and word-based ASR models. The comparison of syllable-based ASR model and word-based ASR model is also done.

In this work, Myanmar ASR is built by applying state-of-the-art acoustic model, Convolutional Neural Network (CNN). In low-resourced condition, CNN is better than Deep Neural Network (DNN) because the fully connected nature of the DNN can cause overfitting. And it degrades the ASR performance for low-resourced languages where there is a limited amount of training data. CNN can alleviate these problems and it is very useful for a low-resourced language such as Myanmar. Furthermore, CNN can model well tone patterns because it can reduce spectral variations and model spectral correlations existing in the signal. In this task, it showed that CNN outperformed DNN and Gaussian Mixture Model (GMM)-Hidden Markov Model (HMM). The best accuracy is achieved with CNN-based model in Myanmar ASR.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

Speech is the most natural form of communication among humans. Numerous spoken languages are employed throughout the world. As communication among human beings is mostly done vocally, it is natural for people to expect speech interfaces with the computer.

Researchers are trying to build system which can record, interpret and understand human speech as early 1960's. The use of speech for interacting with the computer may assist the developing nations as the language technologies can be implemented for the e-governance system.

1.1 Speech Recognition Research

Speech recognition (SR) means the conversion of spoken words to the text or commands. Speech recognition systems development has reached new heights, however, robustness and noise tolerant recognition systems are few of the problems that create speech recognition systems difficult to apply. A lot of research is currently being conducted all over the world for the development of robust automatic speech recognition systems [23].

Automatic speech recognition (ASR) researches have been carried out by many researchers for their particular languages and the accuracy of automatic speech recognition (ASR) systems has been increased by investigating the new architecture or applying particular properties of the target language. Deep learning techniques have also shown a great success in many large vocabulary continuous speech recognition (LVCSR) tasks [15] [42] [43] [44] [48] for English and European languages. Moreover, there are many ASR researches that apply the particular features of the target language. Tone-based languages, for example Mandarin, Thai, Vietnamese, etc., they augmented the tone related information in building acoustic models to improve the accuracy of their ASR [17] [18] [38].

For low-resourced languages, such as Polish [63], Hungarian [37], Bengali [27], Bulgarian [14], Portuguese [46], etc., they showed ASR development in their languages by collecting the data from scratch.

According to the above motivations, Myanmar LVCSR is necessary to develop in speech recognition area and it should be created by using state-of-the-art technology and by applying the particular features of Myanmar language. Therefore, this research is for the purpose of ASR technology development in Myanmar language.

1.2 Objectives of the Thesis

Many researchers are trying to develop the Automatic Speech Recognition (ASR) for their languages to improve their nations in language technologies. The main purpose of this research is to build good quality Myanmar automatic speech recognition on read speech. Acoustic model plays a crucial role to improve the ASR performance. Therefore, the next objective is to develop more efficient acoustic model to get the better accuracy of Myanmar ASR.

Myanmar language is one of the tonal and syllable-timed languages and therefore, the next objective is to explore the effect of tones for Myanmar language on both syllable and word levels. The other objective is to provide Myanmar speech processing systems such as dictation, telephony, speech to speech translation, automatic question and answering, voice commanding, and some robotic applications.

Speech processing systems can help individuals in the disability community. Speech to text processing can assist hearing impaired persons in reading online news. Moreover, it is also convenient and useful to news reporter in transcribing the news. Thus, this is one of the objectives of developing Myanmar ASR. Finally, this research is done for the purpose of ASR technology development in Myanmar language.

1.3 Contribution of the Thesis

There are three main contributions in this research.

The first contribution is developing a speech corpus for Myanmar ASR. In order to develop ASR systems, several hours of recorded speech are required for training purposes. Speech corpus creation is the first step for any automatic speech recognition research. Although most of the speech corpus for well-resourced languages such as English is widely available, there is no pre-created speech corpus for low-resourced languages. Myanmar language is a low-resourced language and there is no freely and commercially available corpus for speech processing research.

The speech corpus building is essential and it is very important for developing Myanmar ASR. Therefore, a speech corpus named “UCSY-SC1” is built by using two methods. The first method is that the recorded speech data is collected from the web news and manually transcribe them into text. The second one is that the daily conversational texts are designed first and then, record the speech by reading the transcriptions. Hence, two types of domain: web news and daily conversions include in the speech corpus.

The second contribution is showing the importance of tones in Myanmar language on both syllable-based and word-based ASR models. Currently, the automatic speech recognition (ASR) systems performance is improved by exploring the new architecture and utilizing particular properties of the target language. There are many ASR work done that apply the particular features of the target language. Tonal languages such as Mandarin, Thai, Vietnamese, etc., used the particular features of their languages which mean the tonal information was added to acoustic modeling to increase their ASR accuracy. Myanmar language is regarded as a tonal language and there are four different tones in it. The different types of tones have different meanings. Thus, accurate tone recognition plays a crucial role in automatic Myanmar speech recognition. In this task, tone-based questions are applied to construct the phonetic decision tree so that to develop more sophisticated tone modeling. In addition, the tone effects are shown on both word-based and syllable-based ASR models. The evaluation results of word-based and syllable-based ASR models are also compared.

The final contribution is building an acoustic model. The acoustic model is one of the main components of the ASR and it is significant to improve the ASR performance. Better accuracy can be achieved if the efficient acoustic modeling approach is applied. Deep Neural Network (DNN) has attained tremendous achievement for large vocabulary continuous speech recognition (LVSR) works, showing significant improvements over the conventional Gaussian Mixture Model-Hidden Markov Model (GMM-HMM). Recently, Convolution Neural Network (CNN) has given state-of-the-art results and demonstrated powerful acoustic modeling abilities because of its capability to structural locality in the feature space. Thus, CNN-based acoustic model is built for developing the Myanmar ASR. The hyperparameters of CNN are optimized to improve the ASR performance for

Myanmar language. The CNN-based model is compared with GMM and DNN models. And, it showed that the best accuracy is achieved with the CNN-based model.

1.4 Organization of the Thesis

This dissertation is organized with seven chapters, including background theory of speech recognition, building speech corpus, nature of Myanmar language, developing HMM-SGMM based acoustic model, exploring the effect of tones using CNN, error analysis of the evaluation results, and conclusion and future research on Myanmar ASR.

Chapter 1 states the introduction to speech recognition, motivation, objectives and contributions of the research work. The literature reviews on automatic speech recognition (ASR) and related works concerning Myanmar ASR researches are surveyed in Chapter 2. In Chapter 3, speech corpus building for Myanmar ASR is described. The speech corpus statistics and phone coverage of the corpus are also depicted in this chapter. Chapter 4 presents the nature of Myanmar language and basic phonemes of Myanmar language. Moreover, the structure of Myanmar language is explained and Myanmar tones are also discussed.

In Chapter 5, Gaussian Mixture Model (GMM)-Hidden Markov Model (HMM) and Subspace Gaussian Mixture Model (SGMM)-based acoustic models, Mel Frequency Cepstral Coefficient (MFCC) feature extraction technique, building language model for Myanmar language and decoder are explained. The evaluation results are shown according to the amount of training data, language model and number of Gaussians using GMM-HMM and SGMM. Furthermore, the importance of language model and acoustic model are also explored in this chapter.

Deep learning techniques, deep neural network (DNN) and convolutional neural network (CNN) for automatic speech recognition are described in Chapter 6. The Myanmar ASR performance is improved by optimization of CNN parameters such as number of feature maps and pooling size. In addition, in this chapter, the effect of tones is analyzed at both syllable and word-based ASR models by using the different acoustic models (GMM, DNN, and CNN). The comparison of syllable and word-based ASR models are also described. Moreover, the error analysis on the best hypothesis text of Myanmar ASR is done in this chapter.

Chapter 7 presents the summarization of the research work. The advantages and limitations of the work are also described. It also indicates promising avenues for future research on Myanmar ASR in this chapter.

CHAPTER 2

LITERATURE REVIEWS AND RELATED WORK

This chapter discusses literature reviews on automatic speech recognition (ASR) and recent publications in Myanmar ASR.

2.1 Introduction to Automatic Speech Recognition

Automatic speech recognition by machine has been a goal of research for more than four decades. In the world of science, computer has always understood human mimics. The idea which caused for making speech recognition system is to be convenient for humans to interact with a computer, robot or any machine by speech or vocalization rather than difficult instructions.

The basic idea of speech recognition is the converting of sound into text and commands. Speech recognition is a process by which computer maps an acoustic speech signal to some form of abstract meaning of the speech [45].

2.2 Classification of Speech Recognition Systems

The speech recognition systems can be categorized in different types based on different classes. The speech recognition system can be divided based on the type of utterances, vocabulary size and speaker dependency.

2.2.1 Classification on the basis of Utterances

2.2.1.1 Isolated Words

Isolated word recognition system can recognize single words or single utterance at a time. These systems have “Listen/Not-Listen states”, in case the speaker has to wait between utterances.

2.2.1.2 Connected Words

Connected word systems are similar to isolated words. However, it allows separate utterances to be ‘run- together’ with a minimal pause between them.

2.2.1.3 Continuous Speech

Continuous speech systems permit users to speak almost naturally and the content is determined by the computer. The continuous speech recognizers are difficult to develop as they used special methods to determine the boundaries of the utterance.

2.2.1.4 Spontaneous Speech

A spontaneous speech recognition system is able to handle a variety of natural speech features such as words being run together, “ums” and “ahs”, and even slight stutters.

2.2.2 Classification on the basis of Vocabulary Size

2.2.2.1 Small Vocabulary

The speech recognition systems that can recognize only a limited number of vocabularies are identified as small vocabulary speech recognition system.

2.2.2.2 Medium Vocabulary

The speech recognition system which can recognize a considerable number of vocabularies are called medium vocabulary speech recognition system.

2.2.2.3 Large Vocabulary

The speech recognition system that can recognize a large number of vocabularies and these systems are defined as large vocabulary speech recognition system.

2.2.3 Classification on the basis of Speaker Mode

2.2.3.1 Speaker Dependent

A speaker dependent system is created to recognize only a single speaker. It is typically easier to develop, cheaper to buy and more accurate, however, it is not as flexible as speaker adaptive or speaker independent systems.

2.2.3.2 Speaker Independent

A speaker independent system is built to recognize for any speaker. These systems are the most difficult to create and most expensive.

2.2.3.3 Speaker Adaptive

The speaker dependent data are utilized by the speaker adaptive systems. The best appropriated speaker is adapted to recognize the speech and the error rates are reduced by adaption [23].

2.3 Application Areas of ASR

One of the major application areas is the interaction of human with computer. Although many tasks are better solved with visual or pointing interfaces, speech has the possible to be a better interface than the keyboard for works. This consists of hands-busy or eyes-busy applications. Another significant application area is telephony, in case spoken dialogue systems for entering digits, recognizing to receive collect calls, finding out airplane or train information, and call-routing. In some applications, the combination of speech with a multimodal interface can be more capable than a graphical user interface without speech. Finally, in dictation, ASR is also used and it is very useful in areas such as law [20].

2.4 Dimension of Variations in ASR

The first dimension of variation in speech recognition tasks is the size of vocabulary. Speech recognition is easier if the size of vocabulary to recognize is smaller. Therefore, tasks with very small vocabularies, such as “yes” or “no” recognition, digits recognition are relatively easy. On the other hand, tasks with large vocabularies, like recognizing human-human telephone conversations, or automatic transcribing broadcast news are more difficult.

The second dimension of variation is how fluent, natural, or conversational the speech is. **Isolated word** recognition, which can only recognize a single word, is much easier than recognizing **continuous speech**. For example, recognizing humans speeches that input to machines, either reading loudly in **read speech** (that mimics the dictation task), or communicating with speech dialogue systems, is quite easy.

The third dimension of variation is channel and noise. Commercial ASR systems such as dictation systems, and researches of speech recognition in the laboratory, are conducted with head mounted microphones that have high quality. Head mounted microphones remove the distortion that happens in a table microphone. Any kind of noise also makes recognition accuracy lower. So recognizing a speaker uttering in a quiet place is much easier than recognizing a speaker speaking in a noisy environment such as in a car on the highway with the window open.

The final dimension of variation is accent or speaker-class characteristics. Speech can easier to recognize if the speaker is uttering a standard dialect, or in general one that matches the training data. The foreign-accented speech or speech of children is harder to recognize unless the speeches do not have in the training data [23].

2.5 History of Automatic Speech Recognition

The first ASR system is an isolated digit recognition system that can only recognize a single speaker. It was constructed by Davis, Biddulph, and Balashek of Bell Laboratories in 1952 [7]. The speech recognition systems have a dramatic improvement due to technology development over the last 60 years. Juang and Rabiner [19] express the development during the first four decades.

During 1960's, isolated words recognition system with small vocabularies (10 - 100 words) can able to recognize with the advantage of filter-bank analyses and simple time normalization approaches.

In 1970's, speaker independent speech recognition systems which can recognize medium vocabularies (100 - 1000 words) are able to build by applying simple template-based of pattern recognition methods. Large vocabulary (1000 - unlimited number of words) speech recognition system was developed by using Hidden Markov Models (HMM) and stochastic language models during 1980's. Neural networks became popular as acoustic modeling technique in ASR in the late 1980s. From then, neural networks have been applied in many speech recognition tasks such as phoneme classification [57], isolated word recognition [59], audiovisual speech recognition, etc.

In 2010, industrial researchers are combined with academic researchers and they developed a success of a deep feed forward neural network (DNN) in LVCSR tasks in case large output layers of the DNN constructed on context dependent HMM states were adopted [6] [8] [61]. As of October 2014, the comprehensive reviews of this development, the related background of automatic speech recognition and the effect of several machine learning paradigms were found in the articles [9] [60].

End-to-end ASR research has been much attention since 2014. It has achieved competitive results compared to conventional hybrid Hidden Markov Model-deep neural network model-based automatic speech recognition (ASR) systems. Such E2E systems are attractive because they do not need initial alignments between input acoustic features and output graphemes or words. The first effort of end-to-end ASR was with Connectionist Temporal Classification (CTC) based systems developed by Alex Graves [et.al.] in 2014 [11].

Another approach to CTC-based models is attention-based models. Attention-based ASR models were presented simultaneously by Chan [et al.] in 2016 [4]. By the end of 2016, the attention-based models have seen greater success than the CTC models [1] [58]. Currently, progress in end-to-end ASR technology has been made [58] [62] [64] [65] and has significantly improved the recognition rate of ASR systems.

2.6 Performance of Speech Recognition Systems

Accuracy and speed are the two most common criteria for determining speech recognition system performance. Word Error Rate (WER) is typically applied for accuracy measurement and speed is commonly rated with Real Time Factor (RTF).

WER can be computed by using Equation (2.1):

$$WER = \frac{S+D+I}{N} \qquad \text{Equation (2.1)}$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions and N is the number of words in the reference.

If the inputs of duration I require time P to process, RTF can be computed by using Equation (2.2):

$$RTF = \frac{P}{I}$$ Equation (2.2)

Other performance measurement includes Concept Error Rate (CER), Single Word Error Rate (SWER) and Command Success Rate (CSR) [23].

2.7 Automatic Speech Recognition Architecture

There are three main stages of automatic speech recognition.

In the feature extraction stage, the sound waveform is sampled into frames that are transformed into spectral features. This step is required for classification of sounds because the raw speech signal has both information and the linguistic message. It is also a high dimensionality. These characteristics of the raw speech signal would be impractical for the classification and cause in high WER. Commonly used feature extraction techniques are Mel Frequency Cepstral Coefficient (MFCC), Linear Predictive Coding (LPC), Linear Prediction Cepstral Coefficient (LPCC), Perceptual Linear Prediction (PLP), Linear Discriminant Analysis (LDA), Discrete Wavelet Transform (DWT), Relative Spectral (RASTA-PLP) and Principal Component Analysis (PCA).

In the phone likelihood stage, the system computes the likelihood of the observed spectral feature vectors depends on linguistic units (words, phones, subparts of phones).

Final stage of ASR, the decoding is the procedure to compute which words sequence is most likely to match to the input speech signal that is characterized by the feature vectors. In other words, it is searching a huge HMM network for determining the most likely path given the acoustic observations. Lattice rescoring method is a standard decoding framework for state-of-the-art LVCSR systems.

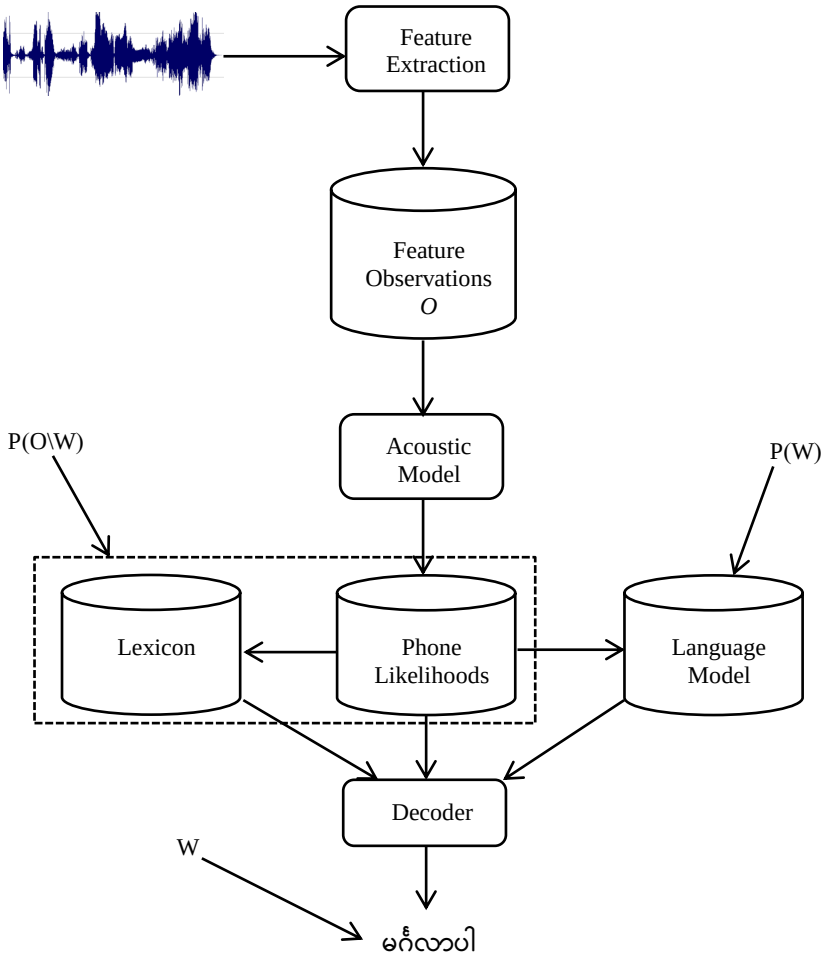


Figure 2.1 Overview of Automatic Speech Recognition System

The most likely sentence $\hat{W}$ depends on some observation sequence O can be calculated by using the product of two probabilities for each sentence and choosing the sentence for which this product is greatest. This is described by using Equation (2.3):

$$\hat{W} = \underset{W \epsilon L}{argmax} P(O|W)P(W) \qquad \text{Equation (2.3)}$$

In the above equation, the acoustic model can be computed by the observation likelihood, P (O\W). The language model can be gained for computing the prior probability, P (W) [20].

2.8 Automatic Speech Recognition Researches on Myanmar Language

There are some Myanmar ASRs recently found in publications.

Myanmar automatic speech recognition was built in combination of several acoustic models using multi-scale features by Thandar Soe [et.al,] [49]. In this work, they built a robust automatic speech recognizer using deep convolutional neural networks (CNNs). The multiple acoustic models were developed with various acoustic feature scales. The multi-CNN acoustic models were combined based on a Recognizer Output Voting Error Reduction (ROVER) algorithm for final speech recognition experiments. They showed that integration of temporal multi-scale features in model training achieved a 4.32% relative word error rate (WER) reduction over the best individual system on one temporal scale feature.

Automatic speech recognition on spontaneous interview speech was developed by Hay Mar Soe Naing [et.al,] [36]. In this study, the author built a recognizer for Myanmar Interview speech by using the classical Gaussian Mixture Model based Hidden Markov Model (HMM-GMM) approach. The speech corpus has 5 Hrs data (3.5 Hrs for training set, 39 mins for development set, and 47 mins for test set) with 600 utterances. The duration of each utterance is 3~60 seconds and contains average 40 words in one sentence. It explored the effect of variation on the nature of acoustic features. Moreover, the importance of the number of senones and Gaussians were adjusted and the best WER, 20.47% was achieved on speaker dependent triphone model.

Syllable-based Myanmar continuous automatic speech recognition was proposed by Wunna Soe [et.al,] [50]. In this work, syllable-based Myanmar language model was used to predict the order of syllable sequence in speech recognition process. The acoustic models for Myanmar ASR are created based on HMM and GMM. Two types of acoustic models were also contributed in this research: speaker dependent and speaker independent model.

Syllable-based phonetic dictionary and language model are used in this task. The syllable-based language model was open-vocabulary language model and the author also contributed the grammatical based word segmentation algorithm to build the word-based language model. Moreover, the author compared the syllable-based language model with word-based language model using different language modeling toolkit (SRILM and CMU) and it showed that syllable-based language model

achieved better accuracy than word-based language model. Moreover, both speaker dependent and speaker independent were analyzed by using three parameters: amount of training data (estimated total hours training), number of tied states (senones), and number of densities (number of GMMs).

Myanmar language speech recognition using hybrid artificial neural network and hidden markov model was presented by Thin Thin Nwe [et.al,] [39]. This work used syllable-based segmentation. Mel Frequency Cepstral Coefficient (MFCC), Linear Predictive Cepstral Coding (LPCC) and Perceptual Linear Prediction (PLP) were used in feature extraction techniques. ANN has a good discriminative ability and flexible, it is not adapted for sequential data like speech. The benefits of HMMs-based systems are the acoustic speech signal is represented statistically and they are stochastic processes that able to perform modeling sequential data. Nevertheless, standard HMMs have some weaknesses in developing a large vocabulary speaker independent continuous ASR system. It has poor discrimination power because of unsupervised learning in case the model parameters are estimated by maximum likelihood (ML estimation). Thus, in this work, hybrid ANN/HMM system is proposed to augment ASR performance. Training database comprised 260 female speaker's utterances. In this work, hybrid ANN/HMM method is applied for building automatic speech recognition with a medium size vocabulary.

Hay Mar Soe Naing [et.al,] [35] presented large vocabulary continuous speech recognition for Myanmar on travel data. Phonemically-balanced corpus has 4,000 sentences and 40 hours of the corpus was used for training data. An open test set with 100 utterances, uttered by 25 speakers, was conducted. Moreover, a Myanmar pronunciation lexicon with a vocabulary of 34K words, together with a G2P converter was constructed to be applied.

In this work, three types of acoustic models were developed: Gaussian mixture model (GMM), DNN (Cross Entropy), and DNN (state-level minimum Bayes risk (sMBR) and compared their performance. The tones and pitch features were incorporated to acoustic modeling and experiments were conducted with and without those features. The word-based language model (LM) and syllable-based LM were compared and their difference is also explored. With sequence discriminative training DNN (sMBR), the best WER of 15.63% and the best SER of 10.87% were achieved using tone and pitch features.

Myanmar continuous speech recognition using Dynamic Time Warping (DTW) and HMM was presented by Ingyin Khaing [et.al,] [22]. In this research work, they addressed the issue of automatic word/sentence boundary detection in both quiet and noisy environment. The combinations of Linear Predictive Coding (LPC), MFCC and Gammatone Cepstral Coefficients (GTCC) techniques were applied in feature extraction. MFCC has the advantage of good discrimination of speech signal. And, LPC provides an accurate estimation of the parameters of speech. In addition, it is also a capable computational model of speech. Moreover, DTW was used in the feature clustering so that to solve the lack of discrimination in the Markov model. In this work, HMM was used for recognition process. They used 558 utterances as training data, and only 10 utterances of 1 female speaker for evaluation.

CHAPTER 3

SPEECH CORPUS BUILDING

Speech corpus is significant for statistical model based automatic speech recognition and it affects the performance of a speech recognizer. Current ASR system use statistical models constructed on speech data. Hence, the statistical-based ASR systems greatly depend on speech corpora. Therefore, speech corpora are needed for ASR training.

Speech corpus is a large collection of audio recordings of spoken language and it also includes transcriptions of the speech. Speech corpus can be classified into two types: Read speech and Spontaneous speech. For example, broadcasts news, Word lists, Number sequences, etc., are contained in Read speech. Spontaneous speech type includes Narratives, interview speeches, etc.

Building speech corpora is a mandatory task for training any automatic speech recognition system. It is also the first step in building ASR system especially for low-resourced languages where there is no pre-created speech corpus. For well-resourced languages, like in English, such resources are well known and widely available. It has rich of collected speech data. For low- resourced languages, it has to develop speech corpus from the scratch since there are no pre-created speech corpora [34].

Myanmar language can be considered as a low-resourced language regarding the linguistic resources available for NLP. Lack of proper data is the main problem when it comes to speech recognition research in the Myanmar Language. Therefore, speech corpus is needed to build for developing Myanmar ASR. In this task, a Myanmar speech corpus named UCSY-SC1 is constructed by using two types of domain: web news and daily conversations.

3.1 Speech Data Collection in Low-Resourced Languages

There are some efforts in developing the speech corpus for low-resourced languages.

AGH corpus for Polish speech was built by Piotr Zelasko, et.al, [63]. It was developed for automatic speech recognition (ASR) and text-to-speech (TTS) systems applications. The corpus has several groups of recordings: read sentences, spoken

commands, a phonetically balanced TTS training corpus, telephonic speech and others. The length of recordings is above 25 hr. There are 166 unique speakers. The common age group is 20–35 and one third of them are females. By using the corpus, SARMATA ASR system gains phrase recognition rate of 91.9 %.

The speech corpus for Bulgarian language was built by Neli Hateva, et.al, [14]. It was created for the purpose of ASR technology development. The corpus includes speech read from selected declarative and interrogative sentences. The total number of speakers is 147 and there are 85 females and 62 males. It consists of 21,891 sentences in the corpus. Their total length is around 32 hours. The recordings have been done in a soundproof room. All experiments are done using a speaker independent (SI) acoustic model trained on the corpus by varying the beam width for the beam search between 1000 and 3000 states by a step of 500. The test set contains each 50 long legal utterances that are recorded by 9 speakers. The best word accuracy is achieved at 93.85% with beam 3000.

A Large Spontaneous Speech Database for Agglutinative Hungarian Language was created by Tilda Neuberger, et.al, [37]. It is a phonetically-based multi-purpose database and it consists of several types of spontaneous and read speech from 333 monolingual speakers (about 50 minutes of speech sample per speaker). It consists of 184 female and 149 male speakers.

A speech corpus for Bengali language was constructed by Sandipan Mandal, et.al, [27]. A continuous read speech corpus of young and old people was built. It is developed for standard colloquial Bengali language which is mostly spoken in West Bengal, India. Hidden Markov Model Toolkit (HTK) was applied for aligning the speech data. It can be used for age detection network. To check the speech corpus quality, phoneme recognition and continuous word recognition performance are observed.

Spontaneous speech corpus for European Portuguese was developed by Tiago Freitas, et.al, [46]. The purpose of building the corpus is to be accessed for the training of speech synthesis and recognition systems in addition to phonetic, phonological, lexical, morphological and syntactic studies. Sociolinguistic and pragmatic research is also contained in designing the corpus. The data consists of unscripted and unprompted face-to-face dialogues between family, friends, colleagues

and unacquainted participants. All recordings are orthographically transcribed and prosodically annotated. The total recording time is 53 hours. The participants in recording are male and female speakers of standard EP. They are between the ages of 17 and 74 years. This corpus is available on the web in two different ways: Spock, a spoken corpus access tool and IMDI database.

3.2 UCSY-SC1: A Myanmar Speech Corpus Building

A Myanmar speech corpus, UCSY-SC1, is developed for the purpose of ASR training. Generally, a speech corpus can be built in two approaches. One approach is collecting the speech that is already been recorded and they are manually transcribed them into text. The second approach is designing the text corpus first and recording the speech by uttering the collected text [34]. The Myanmar speech corpus is built by using the two ways.

3.2.1 Collecting Data from the Online Resources

Nowadays, a lot of speech data are found on the Internet and the speech data can be collected from the web in developing our speech corpus. Therefore, the first way is applied to create our speech corpus for broadcast news domain.

Today, the Internet is a source that has various resource types: social media for instance Facebook or Twitter gives videos files and short, colloquial texts, whereas blogs and news portals have more formal and longer texts news, and audio files. Furthermore, they are freely accessible on the Internet. It is proved that the corpora constructed on online resources provide promising results. In this work [47], text corpora are developed for many low-resourced languages by crawling the web. The Internet resources are utilized in some of the example works such as creating an n-gram model in [66] and developing a model for irony detection in short texts [2]. Both spontaneous and read speech data are found on the Internet. Among them, the read speech type, the broadcast news data, is gathered from the Web for Myanmar continuous speech recognition. The duration of the web data collecting process lasts one year and it has involved two persons including me.

3.2.1.1 Speech Corpus Preparation

Myanmar news is available on many web sites and for our speech corpus, the speech data is collected from the sites of Myanmar Radio and Television (MRTV)[1], Voice of America (VOA)[2] sites. Additionally, the speech data from social media, Facebook, of Eleven broadcasting[3], 7days TV[4], ForInfo news[5], GoodMorning Myanmar[6], BBC Burmese[7], and breakfast news[8] are also gathered. The speech corpus contains both local and foreign news. They are about politics, health, speech, education, crime, sports, weather, and business news, etc. The format of the speech files is converted to .wav files format. Then, a single channel (mono) type with 16 kHz sampling rate is set to the audio files. The long wave files are cut into short length audio files. For speech segmentation, Praat[9] tool is used. Silence and background noise are discarded in segmenting the audio files. The length of the audio files is between 2 sec and 60 sec.

3.2.1.2 Speaker Information

The news presenters are professional, well-experienced and well-trained. Therefore, they have clear voice in news broadcasting. Female news presenters are mostly found in the web news. Hence, in this corpus, male speakers are involved less than females. The Figure 3.1 describes the speaker information of web news data with respect to gender. The ages of the speakers are under 35.

[1] http://www.mrtv.gov.mm
[2] https://burmese.voanews.com/
[3] https://www.facebook.com/elevenbroadcasting
[4] https://www.facebook.com/7DayOnlineTV/
[5] https://www.facebook.com/forinfo/
[6] https://www.facebook.com/GoodMorningMyanmarLive/
[7] https://www.facebook.com/bbcburmese/
[8] https://www.facebook.com/bmrtv/
[9] http://www.fon.hum.uva.nl/praat/

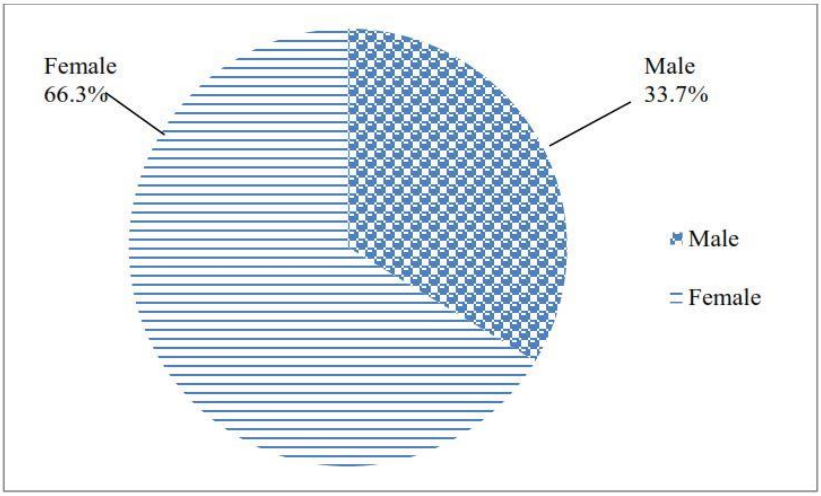


Figure 3.1 Speaker Distribution of Web News Data with respect to Gender

3.2.1.3 Speech Utterance

Some of the news from the Internet already has transcription but, some does not have. Therefore, they are manually transcribed into text if the transcription is not available. In Myanmar language, it is necessary to segment the text as there is no space between words when writing. So, the texts are segmented into words using the segmentation tool [40]. Besides, the word segmentation is manually checked again to be correct. MLC dictionary is applied to check the spelling of the words. The longest transcription has 105 words and 171 syllables. The shortest transcription has only 2 words and 3 syllables. For web news data, it has 8,973 unique sentences, 11,040 unique words. Bootstrapping technique is applied in increasing the text corpus size. Myanmar 3 Unicode font is used for the text corpus. The example news sentences from the corpus are as depicted in Table 3.1. The format of each sentence is the utterance-id followed by the transcription of each sentence.

Table 3.1 Example Sentences of the Corpus on News

ucsy-mrtv-aungmyothu 1000 နှစ် ထောင့် ဆယ့် ခြောက် ခုနှစ် မြန်မာ့ ရုပ်ရှင် အကယ်ဒမီ ထူးချွန်ဆု ပေးပွဲ အခမ်းအနား သို့ နိုင်ငံတော် ၏ အတိုင်ပင်ခံ ပုဂ္ဂိုလ် ဒေါ် အောင်ဆန်းစုကြည် မှ သဝဏ်လွှာ တစ် စောင် ပေးပို့ ခဲ့ ပါတယ်
ucsy-eleven-chosetpaing 2002 ကမ္ဘာ့ ရွှေ ဈေး မြင့်တက် လာ မှု ကြောင့် ပြည်တွင်း ရွှေ ဈေးကွက် အမြင့် ဘက် သို့ ဦးတည် နေ ကြောင်း ဒေသ ရွှေ ဈေးကွက် က ဆို ပါတယ်
ucsy-forinfonews-yuparkhaing_6642 လာအို တင်းနစ် အဖွဲ့ချုပ် မှ လာမည့် ဩဂုတ် လ မလေးရှား နိုင်ငံ တွင် ကျင်းပ မည့် နှစ် ဆယ့် ကိုး ကြိမ်မြောက် အရှေ့တောင် အာရှ ဆီးဂိမ်း တွင် ပါဝင် ယှဉ်ပြိုင် ရန် အသင်း အတွက် အားကစားသမား များ ကို ရွေးချယ် နေ ပါတယ်

3.2.2 Recording Daily Conversations

The second approach (designing the text corpus first and recording the speech by reading the collected text) is used for collecting the conversational data. It took 3 months for data recording and 11 persons involved for the speech and text segmentation.

3.2.2.1 Text Corpus Preparation

First, the daily English conversations which include Myanmar translations are collected from the web. It involves the conversations in hotels, restaurants, streets, telephones, etc. The translated Myanmar sentences are used to develop the corpus. And then, the spelling of the text is manually checked and the words are segmented as the news data. The sentences contained in the corpus are shorter than that of news domain. The longest sentence has 35 words and 55 syllables. The shortest sentence has only 1 word and 1 syllable. For conversational data, it contains 2,156 unique sentences, 1,740 unique words. The example sentences for the daily conversational domain are as shown in Table 3.2. The format of each sentence is similar to news domain (utterance id followed by each utterance).

Table 3.2 Example Sentences of the Conversational Data

ucsy-record-ayechanmay 16201 အဝတ် တွေ က ဈေးပေါ တယ် လို့ ထင် တယ်
ucsy-record-ayechanmay 16202 သူငယ်ချင်း တွေ ရောက် လာ ချိန် မှ စားသောက်ဆိုင် က ထွက် တော့ မယ်
ucsy-record-ayechanmay 16203 ကျေးဇူးပြု ၍ အကူအညီ လို ရင် ပြော ပါ

3.2.2.2 Speaker Information

The sentences are recorded by 4 male speakers and 42 female speakers of the faculties and students of University of Computer Studies, Yangon, Myanmar. Since the females outnumber males in our university, many female speakers are found in the corpus. The speaker distribution with respect to age is depicted in Figure 3.2. The speakers are between 19-40 aged groups.

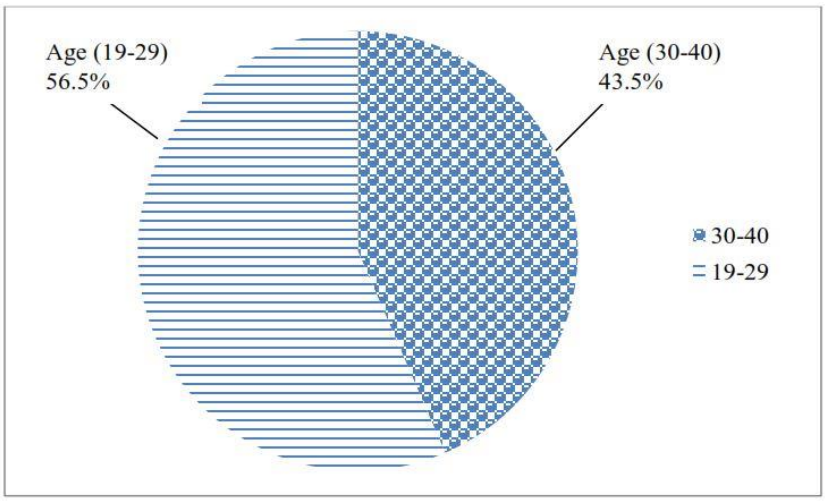


Figure 3.2 Speaker Distribution of Recorded Conversational Data with respect to Age

3.2.2.3 Recording Platform

The recording work has been done in a laboratory of our university. It is a very quiet place with no effects from the room like echo and background noises. It is also a healthy place to work creatively because people can breathe well and feel relaxed. Tascam DR-100MKIII was used for speech recording. It is intended to be used for audio designers and engineers. It has an easy-to-use interface with robust reliability. The file can be recorded into .wav .bwf and .mp3. It can choose mono or stereo channels. In this corpus, the audio files were recorded into .wav (PCM) format.

3.2.2.4 Speech Segmentation and Recording

The audio files have formatted with sample frequency 16 kHz and mono channel with 16 bits encoding. The recorded files are segmented using audacity tool[10]. Moreover, the silence portion of each utterance is discarded. In a speech corpus, audio and text data should be aligned. Thus, each recorded sentence is manually listened and checked them with their corresponding text transcription and made necessary corrections. If the speakers do not have a clear voice, the recordings are done repeatedly until they are correct and smooth. All speakers read at normal pace.

3.2.3 Transcription Normalization

Some of the transcriptions of broadcast news and daily conversions that got from online consist of non-standard words. They are numbers, dates, abbreviations acronyms, symbols, and English names such as name of organization, things, persons, animals, social media, etc. The pronunciation of the words cannot get from the

dictionary. Therefore, they need to be done text normalization and transliteration into Myanmar language. In this work, those words are manually transcribed into Myanmar words by listening the corresponding audios. Table 3.3 shows the example words that need to be normalized.

Table 3.3 Example of Text Normalization

Description	**Example**	**Normalization**
Date	၂၀၁၆-၂၀၁၇	နှစ် ထောင့် တစ် ဆယ့် ခြောက် နှစ် ထောင့် တစ် ဆယ့် ခုနစ်
Time	၃ နာရီ ၅၅ မိနစ်	သုံး နာရီ ငါး ဆယ့် ငါး မိနစ်
Number	၁၁၄ ဦး	တစ် ရာ တစ် ဆယ့် လေး ဦး
Digit	09-448045577	သုည ကိုး လေး လေး ရှစ် သုည လေး ငါး ငါး ခုနစ် ခုနစ်
Symbols	/ %	မျဉ်းစောင်း ရာခိုင်နှုန်း
Organization Name	Community Center	ကွန်မြူနတီ စင်တာ
Social Media Name	Facebook	ဖေ့စ်ဘွတ်
Book Name	Party Animals	ပါတီ အဲ နီးမဲ
Acronyms	FDA	အက်ဖ် ဒီ အေ
Person Name	Mr. Filippno Grandi	မစ္စတာ ဖီလစ်နို ဂရမ်းဒီ

3.3 Statistics of Corpus

The speech corpus consists of two types of domain: web news and conversational data. Both are read speech data types. The detailed information of the corpus is as described in Table 3.4.

For news, it has 25 hrs and 20 mins speech data spoken by 261 speakers (177 females and 84 males) with 9,066 utterances. For conversational data, the duration of the recorded speech is 17 hrs and 19 mins. The total number of utterances is 22,048 which are recorded by 42 females and 4 males.

The corpus consists of 306,088 words. 11,696 words are unique and about 37% occurs only once. And, about 5% of unique words appear between 100 and 1,000 times. Only nearly 1% is found more than 1,000 times in the unique words.

Table 3.4 UCSY-SC1 Corpus Statistics

Data	Size	Speakers			Utterance	Unique Word
		Female	Male	Total		
Web News	25 Hrs 20 Mins	177	84	261	9,066	9,956
Daily Conversations	17 Hrs 19 Mins	42	4	46	22,048	1,740
Total	42 Hrs 39 Mins	219	88	307	31,114	11,696

The 20 most occurrence words in UCSY-SC1 corpus with their percentage of the occurrence count are presented in Table 3.5. It is observed that most frequent words are postpositional markers, prepositions, particles, and conjunctions. This is because most of the Myanmar sentences included in the corpus are complex sentences and they are constructed with clause makers such as postpositions, particles or conjunctions.

Table 3.5 Top 20 Most Frequent Words and their Occurrence Frequency in UCSY-SC1 Corpus

Word	Translation	Occurrence in UCSY-SC1 corpus (%)
ကို	postpositional marker (PPM) to indicate objective case, destination, etc.	2.72
မှာ	at, on, in, under, by, etc.	2.54
က	dance or PPM to indicate nominative case, locative case, etc.	2.28
တွေ	particle suffixed to a noun to denote numerousness, diversity	1.70
မ	title prefixed to the proper name of a female or particle prefixed to a verb to convey a negative sense	1.48
တာ	particle suffixed to verbs and adjectives to form nouns	1.35
တဲ့	same as the conjunction 'that' in English or particle suffixed to a phrase or sentence to denote reported speech	1.33
လို့	particle used in conjunction with interrogatives or particle	1.28

	suffixed to a verb for emphasis, etc.	
နဲ့	same as conjunction 'and' or PPM suffixed to a noun to indicate the instrumental case, etc.	1.24
ပါတယ်	used as verb suffixes	1.17
နေ	the sun or stay or particle suffixed to a verb to denote a continuing process	1.16
ခဲ့	particle suffixed to verbs to emphasize definitiveness of an action or condition	1.10
ဖြစ်	become or make or word suffixed to a verb to denote ability, etc.	1.06
တစ်	one	1.03
တယ်	colloquial form of the sentence final	1.02
ရှိ	be or have	0.99
ရ	obtain or particle suffixed to verbs and collocating with the postpositional marker, etc.	0.97
နှစ်	year	0.95
ပြီး	finish or word to indicate the completion of an act, etc.	0.86
မှု	noun forming particle	0.83

3.4 Phones Coverage in Speech Corpus

Phone coverage is vital for improving the ASR accuracy. Myanmar-English dictionary that was developed by Myanmar Language Commission (MLC) [31] is used as the baseline and this dictionary is extended with the vocabularies of the speech corpus. There are about 38,376 words in the lexicon. The training set has 67 phonemes and it covers 94.37% of phonemes. Table 3.6 describes an example of Myanmar lexicon.

Table 3.6 Example of Myanmar Lexicon

Myanmar Word	Phoneme
အ	a.
အားကစား	a: g a- z a:
အာကာသ	a k a th a.
အပ်နှင်း	a' n h in:

The distributions of phonemes for both consonant and vowel phonemes occurring on the speech corpus were analyzed. The frequency data on consonant distribution of the corpus are given in Figure 3.3. The phoneme /j/ is the most occurrences in the corpus. This is because the phoneme represents some medials such as '္ယ', 'ျ and the consonants 'ရ' and 'ဃ' are defined as /j/ phoneme. The second most occurrences is the phoneme /d/ because the consonants 'ဒ', 'ဓ' , 'ဍ' and 'ဎ' are represented by the same phoneme /d/. The Myanmar word 'တြိ' is rarely appeared in Myanmar language. Therefore, the pronunciation phoneme of the word, /tr/ phoneme, was found 1 time in the texts. A few nasal phonemes, /ng/ and /nj/, are found.

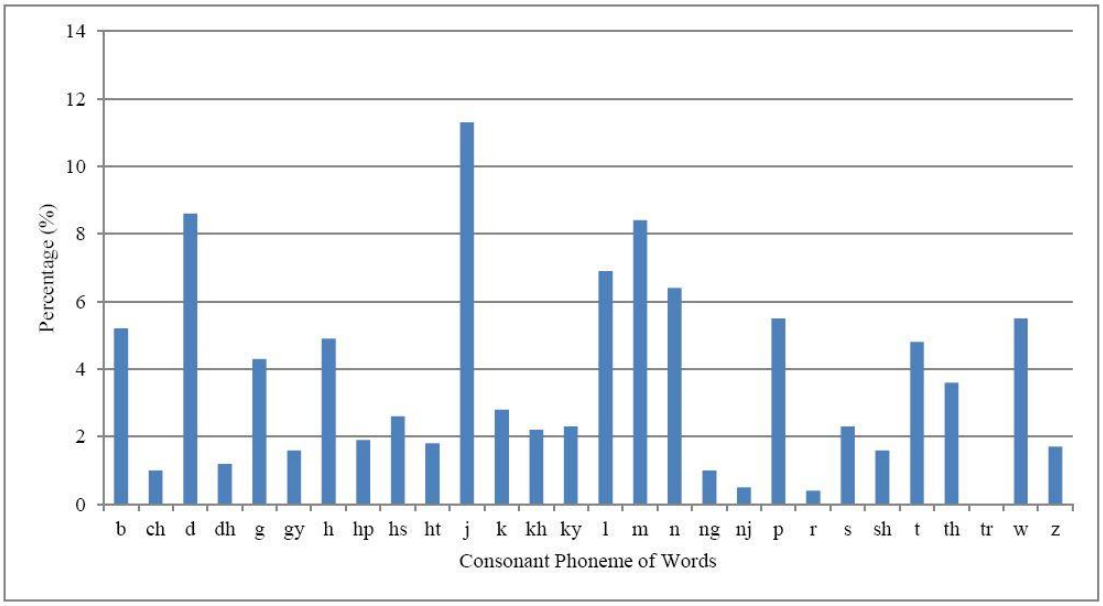


Figure 3.3 Consonant Phonemes Distribution of UCSY-SC1 Corpus

The frequency data on vowel distribution of the corpus are shown in Figure 3.4.

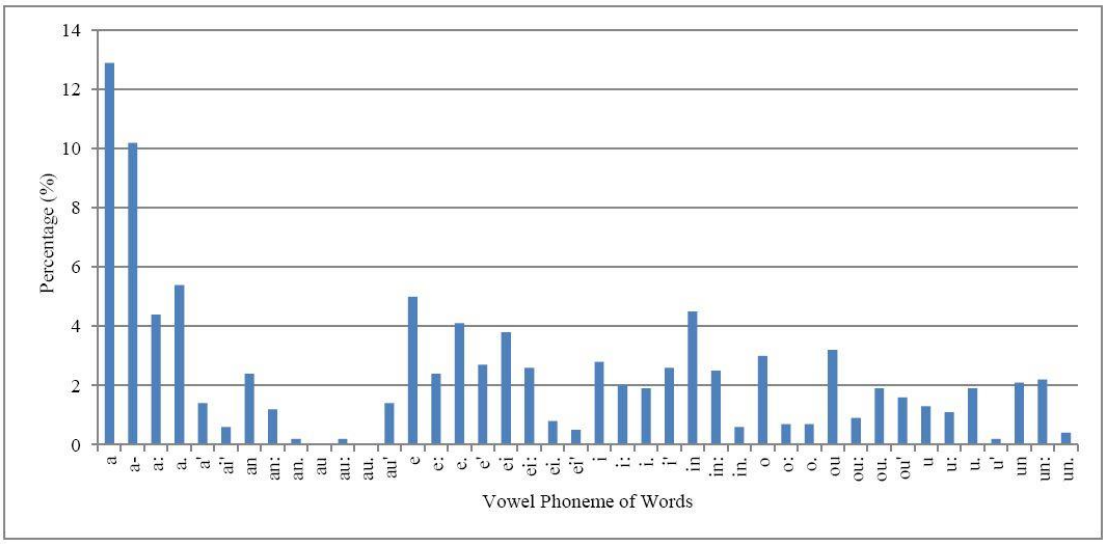


Figure 3.4 Vowel Phonemes Distribution of UCSY-SC1 Corpus

All vowel phonemes are appeared in the corpus. The most frequent phoneme is the phoneme /a/ with tone1 and most of the pronunciation of the words is formed with the vowel phoneme. For example, the words 'ကောင်း' is composed of the phonemes of /k/ + /a/+/un:/ and 'ကိုင်း' is formed by the combination of the phonemes /k/+/a/+/in:/.

The second most frequent phone is the /a-/ with neural tone. In Myanmar language, the basic vowels (/i/, /ei/, /e/, /a/, /o/, /ou/, /u/) have their own properties. While these vowels are influenced by the sounds of the surrounding, they have changed to neutralized vowels when their own properties have been decreased. Therefore, most of the Myanmar words are found with neutral tone in the corpus.

For instance,

နား + ရွက် ==> နာ + ရွက်

Most of the nasalized vowels such as /ai'/, /an./, /ei'/, /in./, /u'/ and /un./ are the least frequent phones in the corpus.

CHAPTER 4

MYANMAR LANGUAGE

This chapter presents introduction to Myanmar language and basic phonemes of Myanmar language. Moreover, structure of Myanmar language and Myanmar tones are also discussed.

4.1 Introduction to Myanmar Language

Myanmar Language (formerly known as Burmese) is an official language of Myanmar. The Myanmar script descends from Brahmi script of South India. The text of Myanmar has a string of characters with no explicit word boundary markup. It is written from left to right without any spaces between words or syllables.

Myanmar characters can be divided into three groups: consonants (known as "Byee"), medials (known as "Byee Twe"), and vowels (known as "Thara"). There are 33 basic consonants, four basic medials, and six combined medials in Myanmar script. Myanmar numerals are decimal-based for counting. Myanmar language has four tones. Different tone makes different meaning for syllables with the same structure of phoneme. A tone is represented by a diacritic mark [54].

4.2 Basic Consonants, Vowels and Myanmar Phonemes

A Myanmar text is a sequence of characters without obvious word boundary markup. It is written from left to right no regular inter-word spacing. The characters in Myanmar language can be divided into three groups: consonants, medials and vowels. The Myanmar basic consonants can be multiplied by medials. Syllables or words are composed of the combination of consonants and vowels. But, some syllables can be structured by just consonants, without any vowel. The Myanmar script contains other characters such as special characters, numerals, punctuation marks and signs.

The 33 basic consonants in the Myanmar script are called as "Byee" in the Myanmar language. Consonants use as the base characters of Myanmar words. The Table 4.1 shows the basic consonants of Myanmar language and they are grouped by their pronunciation types: unaspirated, aspirated, voiced and nasal.

Table 4.1 Group of Myanmar Consonants

Grouped consonants				
Unaspirated	**Aspirated**	**Voiced**		**Nasal**
က /k/	ခ /kh	ဂ /g/	ဃ /g/	င /ng/
စ /s/	ဆ /hs/	ဇ /z/	ဈ /z/	ဉ,ည /nj/
ဋ /t/	ဌ /ht/	ဍ /d/	ဎ /d/	ဏ /n/
တ /t	ထ /ht/	ဒ /d/	ဓ /d/	န /n/
ပ /p/	ဖ /hp/	ဗ /b/	ဘ /b/	မ /m/
ယ /j/	ရ /j/l/r/	လ /l/	ဝ /w/	သ /th/
	ဟ /h/	ဠ /l/	အ /a/	

There are 23 phonemes for 33 consonant scripts; some scripts share the same pronunciations. For example, the pronunciations of “ဒ”, “ဓ”, “ဍ” and “ဎ” are the same and they are defined as the same phoneme /d/.

Basically, there are 12 vowels in Myanmar writing. These vowels are အ(a.), အာ(a), ကြ(i.), ဣ(i), ဥ(u.), ဦ(u), ဧ(ei), အဲ(e:), ဩ(o:), ဪ(o), အံ(an), အို(ou). The variation အ(a.), အာ(a), အီ(i.), ဤ(i), အု(u.), ဦ(u), ဧ(ei), အဲ(e:), ဪ(o:), ဪ(o), အံ(an), အို(ou) can also be written. These 12 basic vowels can be extended with using of tone marker (့) and (း), and also devowelizing consonants. The sequential extension of 12 vowels listed in the original Thinbon Gyi. These extension vowels are depicted in Table 4.2.

In the following table, the vowels with bold are 12 basic vowels and others are extended vowels. At first, there remained the original 11 vowels: အ(a.), အာ(a), အီ(i.), ဤ(i), အု(u.), ဦ(u), ဧ(ei), အဲ(e:), ဪ(o:), ဪ(o), အံ(an). When အို (ou) is added, the result is the basic 12 vowels in the Myanmar language [54]. Moreover, there are 18 nasalized vowels ended with င်၊ ဉ်၊ န်၊ မ် and 7 glottal stop vowels that are ended with က်၊ စ်၊ တ်၊ ပ်.

Table 4.2 Myanmar Basic Vowels and Extended Vowels

အ(a.)	**အာ(a)**	အား(a:)
အိ(i.)	**အီ(i)**	အီး(i:)
အု(u.)	**အူ(u)**	အူး(u:)
အေ့(ei.)	**အေ(ei)**	အေး(ei:)
အဲ့(e.)	အယ် (e)	**အဲ(e:)**
အော့(o.)	**အော်(o)**	**အော(o:)**
အံ့(an.)	**အံ(an)**	အန်း (an:)
အို့(ou.)	**အို(ou)**	အိုး(ou:)

Myanmar syllables are basically formed by consonant and vowel combination. Myanmar syllables involve either a vowel by itself or a consonant combined with a vowel. Myanmar syllable structure is as shown in Table 4.3.

Table 4.3 Myanmar Syllable Structure

Initial Consonant	**Glide Consonant**	**Vowel**	**Final Consonant**	**Tone**
C	(G)	V	(N/)	T

The combination of အို vowel and က consonant creates one syllable ကို as က +အို = ကို.

There are 10 consonants used for devowelizing က်, င်, စ်, ဉ်, ည်, တ်, န်, ပ်, မ်, and ယ်. Myanmar has loaned words from Pali and Sanskrit since the advent of contact with those languages. In adapting Pali and Sanskrit words different means were utilized. The final consonant was dropped. For example, the original word ဒက္ခ of Pali

changed to ဒဂ် and the final consonants ဍ was dropped. The vowel of the final syllable was dropped. For example, the original word ဓာတု changed to ဓာတ် and the vowel ဥ(အု) of တု was dropped. Thus there came to be Pali and Sanskrit loaned words like ဒဂ် and ဓာတ် with final devowelizers. Therefore, Pali and Sanskrit devowelizers came into Myanmar language to augment the ten Myanmar devowelizers. There are 27 devowelizers in present Myanmar writing system. These devowelizers are က်, ခ်, ဂ်, ဍ်, စ်, ဇ်, ဉ်, ည်, ဋ်, ဌ်, ဏ်, တ်, ထ်, ဒ်, ဓ်, န်, ပ်, ဗ်, ဘ်, မ်, ယ်, ရ်, လ်, ဝ်, သ်, ဟ်, and ဠ်. There are four basic consonant combination symbols in Myanmar writing ျ, ြ , ွ, ှ. These Myanmar characters may be joined with appropriate consonants out of the 33 such as ကျ, ပြ, စွ, လှ. These symbols can also be combined with each other in two characters or three characters as in ကြွ, မြွ.

4.3 Myanmar Grammar

It is said there are over 3000 languages, other said that over 4000 languages or 5000 languages around the world. There are many different opinions in that number of language in the world. And there are many definition of language. Language is sound. It is based on the sound that human speak. Language is meaning, and language is defining by imagination, language is “change”. And language is developing and language is difference of structured styles. Language is systematic and the sharing. Moreover, there are many other definitions of language. Nevertheless, grammar is one part of the language and it states the structure of that language.

As grammar is a major part of linguistics, it learns the rules behind languages. Precisely, the grammar aspect that does not concern meaning directly is called syntax, while the aspects that concern meaning include semantics and pragmatics. There are nine parts of speech in Myanmar grammar. These are noun, pronoun, verb, adjective, adverb, conjunction, postpositional marker, particle, and exclamation. There are two types of Myanmar sentence by defining its structure; simple sentence and compound sentence.

In the other hand, there are five types of Myanmar sentence defining semantics. These are

1. statement sentence

2. question sentence
3. negative sentence
4. urge sentence
5. desire sentence

There are two types of words in Myanmar languages; **isolated word** and **provided word**. Isolated Myanmar word is the meaningful word that does not depend on other words. Isolated words can be noun, pronoun, adjective, verb, adverb, and exclamation from Myanmar part of speech. But exclamation can be isolated words and sometimes it may be a word that combines a provided word. The provided words are the word that provides and combines to be a meaningful word in building phrases or sentences. The provided words are particle, postpositional marker, and conjunction.

The grammatical hierarchy is a valuable idea of successively comprised levels of grammatical construction operating within and between grammatical levels of analysis. This hierarchy is a compositional hierarchy in which lower levels typically are filler units for the next higher level in the hierarchy.

In Myanmar language, most sentences are mainly composed of phrases. Phrases are built with words, and words are composed of syllables. In a Myanmar sentence, a syllable is the smallest unit that has semantics. The Figure 4.1 shows how to compose a simple Myanmar sentence from syllable level to sentence level.

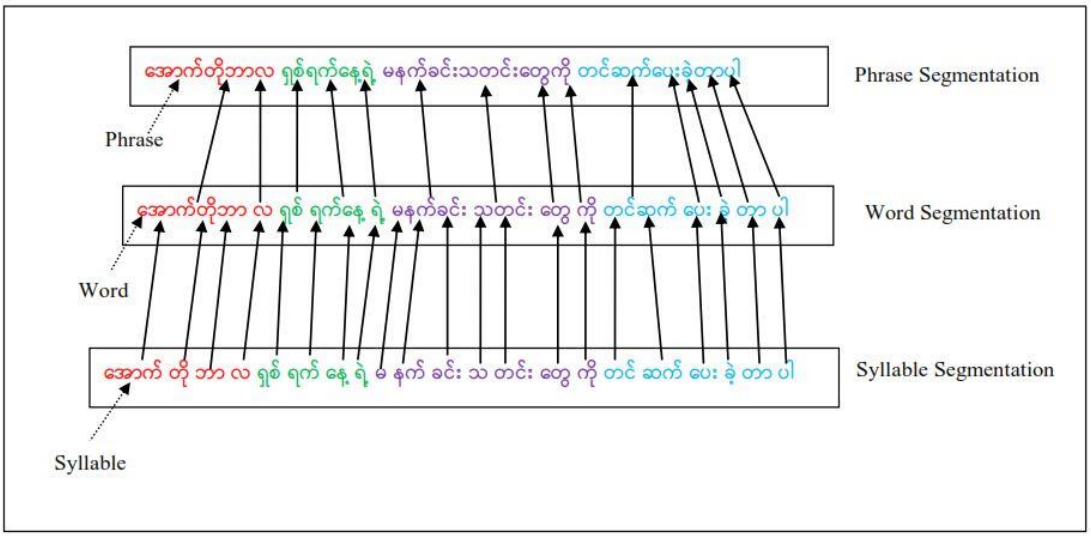


Figure 4.1 Example of Grammatical Hierarchy of Myanmar Sentence

Generally, there are four patterns of syllable structure for all languages. The patterns of syllable structure are described by symbols as the following.

1. -V-
2. CV-
3. -VC
4. CVC

Among them, CV- pattern can see in every language. There are many different languages, and some languages have both CV- and -V- and some have CV-, -V-, and -VC. Moreover some languages have CV-, -V-, -VC and CVC. In these patterns, C means consonant and V means vowel.

The structure of Myanmar syllable is a combination of initial plus medial (tone plus vowel) plus final. The initial is consonant or consonant with glides. The medial contain vowel and tone. Sometime tone does not contain in medial. The final is glottal stop or nasalized sound.

4.4 Myanmar Phonology

Speech can be produced if there are just place of articulation, articulator, and manner of articulation. The basic consonants of Myanmar are described in the following table according to the three parameters of place of articulation, manner of articulation, and articulator [55].

The phonology is the system that combines the vowel and the consonant. Myanmar phonology can be composited by just one vowel, or one vowel and consonant, consonant combination symbols. In Myanmar language, the vowels have their own sounds. Therefore, just only one vowel can produce clear sound such as အ၊အာ၊ အား. Myanmar consonants have no clear own sound and if it combines a vowel, it can produce the clear sound. Example is က+ အား =ကား. Table 4.4 shows the phonology of Myanmar consonants.

There are four phonology methods in Myanmar language.

1. Vowel phonology
2. Combining consonant, consonant combination symbols and an original vowel phonology

3. Combining consonant, consonant combination symbols and a nasalized vowel phonology
4. Combining consonant, consonant combination symbols and a glottal stop vowel phonology

Table 4.4 Phonology of Myanmar Consonants

Manner of Articulation	**Place of Articulation**						
	Bilabial	**Dental**	**Alveolar**	**Palato-alveolar**	**Palatal**	**Velar**	**Glottal**
Nasal	မ		န	ည		င	
(Stop)	မ်		န်	ည်		င်	
Stop Voiced	ဘ (ဗ)		ဒ			ဂ	
Voiceless	ပ ဖ		တ ထ			က ခ	
Fricative Voiced		သ	ဇ				
Voiceless		သ	စ ဆ	ရှ			
Affricate Voiced				ဂျ			
Voiceless				ကျ ချ			
Central Approximant Voiced	ဝ		(ရ)		ယ		
Voiceless	ဝ့						ဟ
Lateral Approximant Voiced			လ				
Voiceless			လ့				

1. **Vowel phonology**

The original extended 22 vowels, 18 nasalized vowels, and 7 glottal stop vowels are involved in vowel phonology.

Examples:

အ၊ အာ၊ အား …

အင့်၊ အင်၊ အင်း…

အက်၊ အစ်၊ အတ်…

2. **Combining consonant, consonant combination symbols and an original vowel phonology**

The original 22 vowels can be composited with consonants and consonant combination symbols. Examples are shown in Figure 4.2.

Examples:

Consonants		Vowel		Result
က	+	အာ	=	ကာ
စ	+	အီ	=	စီ

Consonants combination symbols		Vowel		Result
ကြ	+	အ	=	ကြ
မှ	+	အား	=	မှား

Figure 4.2 Myanmar Phonology of Combining Consonants, Consonant Combination Symbols and an Original Vowel

3. **Combining consonant, consonant combination symbols and a nasalized vowel phonology**

The third phonology method is the combining consonants, consonant combination symbols and a nasalized vowel. Examples are described in Figure 4.3.

Examples:

Consonants		Nasalized vowel		Result
ခ	+	အင်း	=	ခင်း
မ	+	အောင်း	=	မောင်း

Consonants combination symbols		Nasalized vowel		Result
ကွ	+	အင်း	=	ကွင်း
မှ	+	အန်	=	မှန်

Figure 4.3 Myanmar Phonology of Combining Consonants, Consonant Combination Symbols and a Nasalized Vowel

4. Combining consonant, consonant combination symbols and a glottal stop vowel phonology

The fourth phonology method is the combining consonants, consonant combination symbols and a glottal stop vowel [32]. Examples are depicted in Figure 4.4.

Examples:

Consonants		Glottal stop vowel		Result
စ	+	အစ်	=	စစ်
တ	+	အိတ်	=	တိတ်

Consonant combination symbols		Glottal stop vowel		Result
ကြ	+	အစ်	=	ကြစ်
မျ	+	အောက်	=	မျောက်

Figure 4.4 Myanmar Phonology of Combining Consonants, Consonant Combination Symbols and a Glottal Stop Vowel

4.5 Myanmar Tones

Myanmar tone is conveyed by syllable and is characterized by both fundamental frequency and syllable duration. There are four tones in written Myanmar: low, high, creaky and checked. Table 4.5 displays the characteristics of Myanmar tones. The fundamental frequency for the four types of tones of the phoneme 'a'(အ) is shown in Figure 4.5. Different tones have different in meanings. For instance,

Tone1	(အာ တယ်)	(a te)	[to be wide open, to talk too much]
Tone2	(အား တယ်)	(a: te)	[to be free]
Tone3	(အ တယ်)	(a. te)	[to be dumb or dull]
Tone4	(အပ် တယ်)	(a' te)	[to place an order]

Table 4.5 Characteristics of Myanmar Tones

Tone		Phonation	Length	Pitch
Low	a	Modal voice	Medium	Low
High	a:	Breathy voice	Long	High
Creaky	a.	Creaky voice	Medium	High
Checked	a?	Final glottal stop	Short	Varies

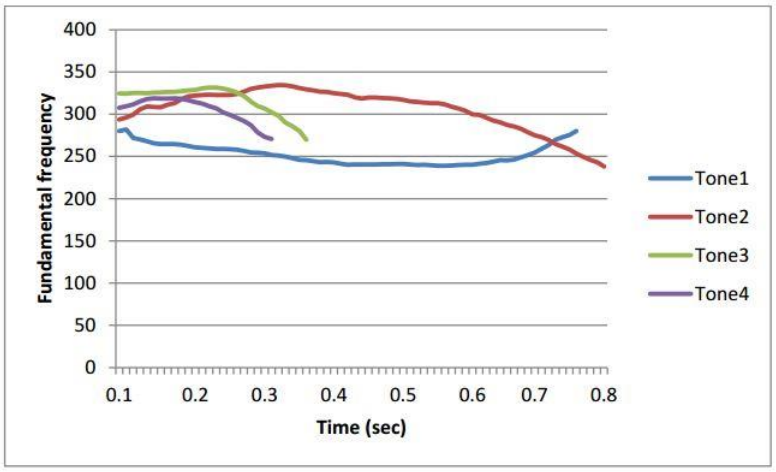


Figure 4.5 Examples of Four Tones of the Myanmar Syllable 'a'(အ)

4.5.1 The Low Tone [a]

The Low tone is invariably described as low-pitched, but with a contour that is either level, bears a slight fall or has a final rise. Low tones are also produced without the stress ascribed to the other tones. Low tone syllables are moderately long and produced with regular, modal glottal vibration. In the orthography, on most vowels it is denoted by the absence of a diacritic or trailing character. Low-toned syllables may be either open (CV) or closed with a sonorant nasal segment (CVN).

4.5.2 The High Tone [a:]

Many of the contradictory phonetic claims regarding Burmese tones concern the production of the High tone. In particular, whether it is breathy-voiced or not and whether it is consistently a falling tone or simply high-pitched. High tone vowels are long, being either longer or equally long to Low vowels. They are also produced with greater intensity than Low and reduced vowels, but comparison with Creaky and Checked tone intensity is not as straightforward. It is sometimes referred to as the "Heavy" tone due to this long, stressed syllable. Like the Low tone, High tone syllables may be either open (a CV syllable) or closed with a sonorant nasal segment (CVN).

4.5.3 The Creaky Tone [a.]

The Creaky tone bears a pitch contour which starts high and falls, though the steepness of this fall is not always agreed upon. The Creaky tone also bears high peak intensity early in the vowel and a brief duration (or, at least shorter than Low or High syllables). The Creaky tone is consistently identified with creaky-voicing or a slow glottal closure. Creaky tone syllables may also be either open (CV) or closed with a sonorant nasal segment (CVN).

4.5.4 The Checked Tone [àʔ]

Checked syllables bear the shortest vowel of all the tones, one that is cut off by the quick glottal closure of the final stop. A common alternative label for the tone used in Burmese grammars is the "Killed tone", in reference to this abrupt closure and to the orthographic character used with final stops, {်}, which has the name /ʔə.θàʔ/ "the killer". The vowel bears a very high pitch and high peak intensity [12].

When the sounds of vowels are produced, the imagined axes inside the speaker's mouth are drowning for showing the far-off places where the height of the body of the tongue reached and moved. According to those axes, and the following vowel quadrilateral and the situations of the tongue, the vowels are seen in the below figure.

	Front	Central	Back
High	အီ (I)		အူ (U)
High	အေ (EI)		အို (OU)
Mid			
Low	အယ် (E)		အော် (O)
Low		အ (A.)	

Figure 4.6 Vowels and Vowel Quadrilateral

The basic phonemes of Myanmar vowels are described with four tone levels in Table 4.6. There are 50 phonemes of Myanmar vowel in the table but number of phonemes may be over 50. According to the Table, there is not just postscript double dot (ः) in the tone level II. The vowel number 16 အဲ and 31ဪ are also contained in this tone II although they do not have the postscript double dot. Furthermore, the nature of their phonemes is the same with other vowels that have postscript double dot in this column [32].

Table 4.6 Myanmar Vowels with Tone Level

Basic Symbol	Non-nasalized Vowels				Nasalized Vowels		
	Tone I	Tone II	Tone III	Tone IV	Tone I	Tone II	Tone III
အီ	1.အီ	2.အီး	3.အိ	4.အစ်	5.အင်	6.အင်း	7.အင့်
အေ	8.အေ	9.အေး	10.အေ့	11.အိတ်	12.အိန်	13.အိန်း	14.အိန့်
အယ်	15.အယ်	16.အဲ	17.အယ့်	18.အက်	19.အိုင်	20.အိုင်း	21.အိုင့်
				22.အိုက်			
အာ	23.အာ	24.အား	25.အ	26.အတ်	27.အန်	28.အန်း	29.အန့်
အော်	30.အော်	31.အော	32.အော့	33.အောက်	34.အောင်	35.အောင်း	36.အောင့်
အို	37.အို	38.အိုး	39.အို့	40.အုပ်	41.အုန်	42.အုန်း	43.အုန့်
အူ	44.အူ	45.အူး	46.အု	47.အုတ်	48.အွန်	49.အွန်း	50.အွန့်

CHAPTER 5

BUILDING THE BASELINE ACOUSTIC MODEL WITH GAUSSIAN MIXTURE MODEL (GMM) - HIDDEN MARKOV MODEL (HMM)

In this chapter, the baseline Gaussian Mixture Model (GMM)-Hidden Markov Model (HMM) and Subspace Gaussian Mixture Model (SGMM) acoustic models are presented. Moreover, building language model for Myanmar language is also discussed. Mel Frequency Cepstral Coefficient (MFCC) feature extraction technique and decoder used in the experiments are also described. The evaluation of Automatic Speech Recognition (ASR) performance is done according to the amount of training data, language model, and number of Gaussians using GMM-HMM and SGMM.

5.1 Hidden Markov Model (HMM) Acoustic Models

Acoustic modeling is an essential part of ASR system and is establishing the connection between the acoustic features and phonetics. Acoustic model plays a crucial role in accuracy of the system and responsible for computational load. Many acoustic modeling techniques are available. And, among them Hidden Markov Model (HMM) is commonly applied and known as it is an effective algorithm for training and recognition [45].

Hidden Markov models are generative models based on stochastic finite state networks. Markov models are stochastic state machines that have a finite set of N states. Given a pointer to the active state at time t the selection of the next state has a constant probability distribution. Therefore, states sequence is a stationary stochastic process. An n^{th} order Markov assumption is that the likelihood of entering a given state depends on the occupancy in the previous n states. In speech recognition a 1^{st} order Markov assumption is commonly applied. The probability of the state sequence $q_T = (q_1, \ldots q_T)$ is specified by:

$$P(q_T) = P(q_1) \prod_{t=2}^{T} P(q_t | q_1, \ldots, q_{t-1}) \qquad \text{Equation (5.1)}$$

and by the first-order Markov assumption this is approximated by:

$$P(q_T) \simeq P(q_1) \prod_{t=2}^{T} P(q_t | q_{t-1}) \qquad \text{Equation (5.2)}$$

The observation sequence is assumed as a series of points in vector space Y_T = {y(1),…y(*T*)} or alternatively as a series of discrete symbols. Therefore, an observation sequence probability can be expressed by:

$$p(Y_T) = \sum_{Q_T} p(Y|q_T)P(q_T) \qquad \text{Equation (5.3)}$$

where the sum $\sum_{Q_T}$ is over all possible state sequences q_T through the model and the probability of a set of observed vectors, $p(Y_T|q)$, can be described by:

$$p(Y_T|q_T) = \prod_{t=1}^{T} p(y(t)|q_t) \qquad \text{Equation (5.4)}$$

The HMMs form can be expressed by the set of parameters which describes them:

States HMMs comprised of N states in a model; the pointer (q_t=*i*) shows being in state *I* at time *t*.

Transitions The transition matrix A provides the probabilities of traversing from one state to another over a time step

$$a_{ij} = P(q_{t+1} = j|q_t = i) \qquad \text{Equation (5.5)}$$

The form of the matrix can be constrained such that certain state transitions are not allowed. In addition, the transition matrix has the constraint that

$$\sum_{j=1}^{N} a_{ij} = 1 \qquad \text{Equation (5.6)}$$

And

$$a_{ij} \geq 0 \qquad \text{Equation (5.7)}$$

State Emissions Each emitting state has associated with it a probability density function $b_j(y(t))$; the probability of emitting a given feature vector if in state *j* at time *t:*

$$b_j(y(t)) = p(y(t)|q_t = j) \qquad \text{Equation (5.8)}$$

The hidden states of HMM can be used to model speech in various means. The speech recognition using HMM models typically do not permit arbitrary transactions. They place robust restrictions on transitions depending on the sequential nature of

speech. HMMs for speech do not allow transitions from states to go to earlier states in the word. This means states can transition to themselves or to successive states [52].

For recognizing small numbers of words in speech task, using an HMM state to denote a phone is sufficient. Nevertheless, in general large vocabulary continuous speech recognition (LVCSR) tasks, a more fine-grained representation is needed.

To consider the information regarding the non-homogeneous nature of phones over time, in LVCSR a phone is normally modeled with more than one HMM state [21]. The most common configuration is using three HMM states, a beginning, middle, and end state. Thus, each phone has 3 emitting HMM states in place of one (plus two non-emitting states at either end), as presented in Figure 5.1.

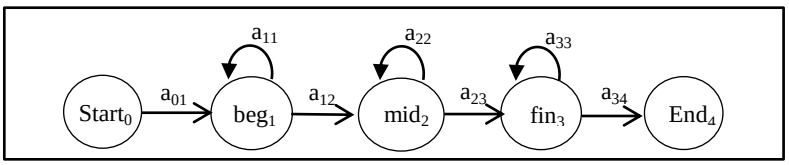


Figure 5.1 A Standard 5-state HMM Model for a Phone

To construct a HMM for an entire word applying these more complex phone models, each phone of the word model is simply replaced with a 3-state phone HMM. The non-emitting start and end states for each phone model are replaced with transitions directly to the emitting state of the preceding and following phone, leaving only two non-emitting states for the entire word as shown in Figure 5.2.

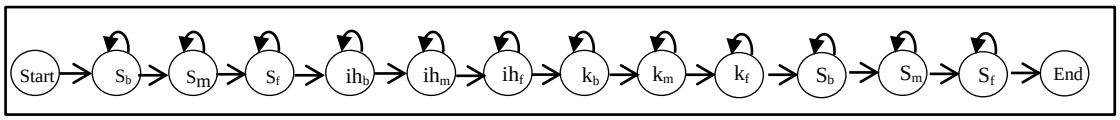


Figure 5.2 A Composite Word Model for "six" [s ih k s]

5.2 Gaussian Mixture Model (Output Probability Distributions)

In speech recognition works, continuous features are most usually applied. And, they are modeled with continuous density output probability functions. If the output distributions are continuous density probability functions in the case of continuous density HMMs (CDHMMs), a mixture of Gaussians function is used to describe [56].

A Gaussian distribution is a function parameterized by a mean, or average value, and a variance. μ specify the mean and σ^2 specify the variance, the following formula is given for a Gaussian function:

$$f(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad \text{Equation (5.9)}$$

The mean of a random variable X is the expected value of X. For a continuous variable X, it is the integral of the values of X. For a discrete variable X, this is the weighted sum over the values of X and the equation is described as below:

$$\mu = E(X) = \sum_{i=1}^{N} p(X_i)X_i \quad \text{Equation (5.10)}$$

The variance of a random variable X is the weighted squared average deviation from the mean:

$$\sigma^2 = E(X_i - E(X))^2 = \sum_{i=1}^{N} p(X_i)(X_i - E(X))^2 \quad \text{Equation (5.11)}$$

A univariate Gaussian pdf used to estimate the probability that a particular HMM state j generates the value of a single dimension of a feature vector by assuming that the possible values of observation feature vector y_t are normally distributed. Each state j has associated with it a mean value μ_j and variance σ^2_j, the system compute the likelihood $b_j(y_t)$ using the following equation for a Gaussian pdf:

$$b_j(y_t) = \frac{1}{\sqrt{2\pi\sigma^2{}_j}} exp\left(-\frac{(y_t-\mu_j)^2}{2\sigma^2{}_j}\right) \quad \text{Equation (5.12)}$$

The above equation is how to apply a Gaussian to calculate the likelihood of an acoustic for a single cepstral feature. As an acoustic observation has 39 features vector, a multivariate Gaussian is needed to use to assign a probability of 39-valued vector. For a given HMM state with mean vector μ_j and covariance matrix $\sum_j$, and a given observation vector y_t, the multivariate Gaussian probability estimate is

$$b_j(y_t) = \frac{1}{(2\pi)^{\frac{D}{2}}|\sum|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(y_t-\mu_j)^T \sum_j^{-1}(y_t-\mu_j)\right) \quad \text{Equation (5.13)}$$

In this equation, D is defined the number of dimensions and it has 39 dimensions in computing phone likelihood. The covariance matrix $\sum_j$ states the variance between each pair of feature dimensions.

A multivariate Gaussian model assigns a likelihood score to an acoustic feature vector observation. For a non-normal distribution, the system does not always

model the observation likelihood with a single multivariate Gaussian; instead, it is modeled with a weighted mixture of multivariate Gaussians. This type of a model is called a Gaussian mixture model (GMM). The equation for the GMM is stated at the following:

$$f(x|\mu,\sum) = \sum_{k=1}^{M} C_k \frac{1}{\sqrt{2\pi|\sum_k|}} exp[(x-\mu_k)^T \sum^{-1}(x-\mu_k)] \qquad \text{Equation (5.14)}$$

The output likelihood function $b_j(y_t)$ as the GMM is shown in the following:

$$b_j(y_t) = \sum_{m=1}^{M} c_{jm} \frac{1}{\sqrt{2\pi|\sum_{jm}|}} exp\left[\left(x-\mu_{jm}\right)^T \sum_{jm}^{-1}\left(y_t-\mu_{jm}\right)\right] \qquad \text{Equation (5.15)}$$

To train the GMM likelihood function, Baum-Welch is used to express the probability of a certain mixture accounting for the observation. And then, it iteratively updates this probability.

ξ function is applied to support in computing the state probability. $\xi_{tm}(j)$ is defined as the probability of being in state *j* at time *t* with the m^{th} mixture component accounting for the output observation y_t. $\xi_{tm}(j)$ as follows:

$$\xi_{tm}(j) = \frac{\sum_{i=1} N\alpha_{t-1}(j) a_{ij} c_{jm} b_{jm}(y_t) \beta_t(j)}{\alpha_T(F)} \qquad \text{Equation (5.16)}$$

If the value of ξ is got from a previous iteration of Baum-Welch, $\xi_{tm}(j)$ is used to recalculate the mean, mixture weight, and covariance by the following formulas:

$$\hat{\mu}_{im} = \frac{\sum_{t=1}^{T} \xi_{tm}(i) y_t}{\sum_{t=1}^{T} \sum_{m=1}^{M} \xi_{tm}(i)} \qquad \text{Equation (5.17)}$$

$$\hat{c}_{im} = \frac{\sum_{t=1}^{T} \xi_{tm}(i)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \xi_{tk}(i)} \qquad \text{Equation (5.18)}$$

$$\hat{\Sigma}_{im} = \frac{\sum_{t=1}^{T} \xi_t(i)(y_t-\mu_{im})(y_t-\mu_{im})^T}{\sum_{t=1}^{T} \sum_{k=1}^{M} \xi_{tm}(i)} \qquad \text{Equation (5.19)}$$

5.3 Recognition Using Hidden Markov Models

The role of an acoustic model in a speech recognition system is finding the observed data probability Y_T given a hypothesized set of word units W. The word string is mapped to the relevant set of HMM models M and thus the search is over $p(Y_T|M)$. As the continuous probability density functions provide emission probabilities, the objective of the search is to maximize the likelihood of the data given the model set.

The probability for a given state sequence $q_T = \{q_0,\ldots,q_T\}$ and observations Y_T is calculated by the product of the transition and output probabilities:

$$p(Y_T, q_T) = a_{q_1,q_2} \prod_{t=2}^{T} b_{q_t}\big(y(t)\big) a_{q_{t-1}q_t} \qquad \text{Equation (5.20)}$$

The total likelihood can be obtained by all possible state sequences (or paths) are summed in the given model that ends at the suitable state. Therefore, the observation sequence likelihood ending in the final state N is given by:

$$p(Y_T|M) = \sum_{q_T \in Q} a_{q_{TN}} \prod_{t=2}^{T} a_{q_{t-1}q_t} b_{q_t}(y(t)) \qquad \text{Equation (5.21)}$$

where Q is the set of all possible state sequences, M is the model set and q_t is the state occupied at time t in path q_T.

5.4 Training HMM: Forward-Backward Algorithm

The forward-backward algorithm is a method for efficiently computing the likelihood of generating an observation sequence depending on a set of models. The independence assumption expresses that a given observation probability relies only on the current state and not on any of the previous state sequences. Two probabilities such as the forward probability and the backward probability are introduced. The forward probability is the probability of a given model generating an observation sequence $Y_t = \{\mathbf{y}(1),\ldots\mathbf{y}(t)\}$ and being in state j at time t:

$$\alpha_j(t) = p(y(1), y(2), \ldots, y(t), q_t = j|M \qquad \text{Equation (5.22)}$$

$$= \left[\sum_{i=1}^{N} \alpha_i(t-1) a_{ij}\right] b_j\big(y(t)\big) [for\ (1 < t < T) and\ (2 < j < N-1)]$$

The initial conditions for the forward probability for a HMM are described by:

$$\alpha_1(0) = 1 \qquad \text{Equation (5.23)}$$

$$\alpha_j(0) = 0 \text{ if } j \neq 1 \qquad \text{Equation (5.24)}$$

and the termination is assumed by:

$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T) a_{iN} \qquad \text{Equation (5.25)}$$

The backward probability is specified by:

$$\beta_i(t) = p(y(t+1), y(t+2), \dots, y(T) | q_t = 0, M)$$

$$= \sum_{j=1}^{N-1} a_{ij} b_j(y_{t+1}) \beta_j(t+1) \qquad \text{Equation (5.26)}$$

with initial and terminating conditions:

$$\beta_j(T) = a_{jN} \quad \text{for } 1 < j < N \qquad \text{Equation (5.27)}$$

$$\beta_N(t) = 0 \qquad \text{Equation (5.28)}$$

Therefore, the likelihood of a given observation sequence can be described by:

$$p(Y_T|M) = \alpha_N(T) = \beta_1(0) = \sum_{j=1}^{N} \alpha_j(t) \beta_j(t) \qquad \text{Equation (5.29)}$$

Furthermore, the probability of being in state *j* at time *t* can be calculated by:

$$L_i(t) = \frac{\alpha_i(t) \beta_i(t)}{p(Y_T|M)} \qquad \text{Equation (5.30)}$$

Therefore, the forward-backward algorithm achieves an effective technique for computing the frame level alignments needed for the HMM model parameters training applying the Expectation-Maximization (EM) algorithm.

5.5 Gaussian Mixture Model (GMM) Vs. Subspace Gaussian Mixture Model (SGMM)

Although HMM-GMM framework is success, there are several shortcomings that is necessary to be solved. In a traditional system, the GMM parameters for each HMM state are estimated independently given the alignment. This requires a very large number of model parameters to be trained, particularly for context-dependent acoustic models. And therefore, a large amount of training data is needed to fit the

model. Furthermore, different sources of acoustic variability such as pronunciation variation, accent, speaker factor and environmental noise can affect the recognizer accuracy. Such variations are only weakly modeled and factorized by adaptation techniques such as maximum likelihood linear regression (MLLR), maximum a posteriori adaptation (MAP) and vocal tract length normalization (VTLN). The subspace Gaussian mixture model (SGMM) was proposed to solve these two issues better [26].

In an SGMM, the parameters of models are obtained from the globally shared model subspace with very low dimensional state-dependent vectors. The model subspace takes the major variations among the phonetic states. Therefore, only a small number of additional parameters are needed to derive the state-dependent GMMs. This results in reducing model parameters numbers and permits model estimation in more accurately with a limited amount of training data. Furthermore, a speaker subspace can also be introduced which allow SGMMs to factorize the phonetic and speaker factors in the model domain.

5.6 Feature Extraction

Feature extraction is a special kind of dimensional reduction method and it is applied in reducing the data which is very large to be processed by an algorithm. The extraction of features transformed the data input into features sets that supports the appropriate information for carrying out a desired task no requirement of the full size data, however, using the reduced set. There are many possible feature representations in feature extraction process. Among them, Mel Frequency Cepstral Coefficient (MFCC) is the most common in speech recognition [24].

5.6.1 Mel Frequency Cepstral Coefficient (MFCC)

MFCC is the most common in speech recognition. These are based on the main idea of the cepstrum.

There are seven steps in extracting MFCC features as shown in Figure 5.3.

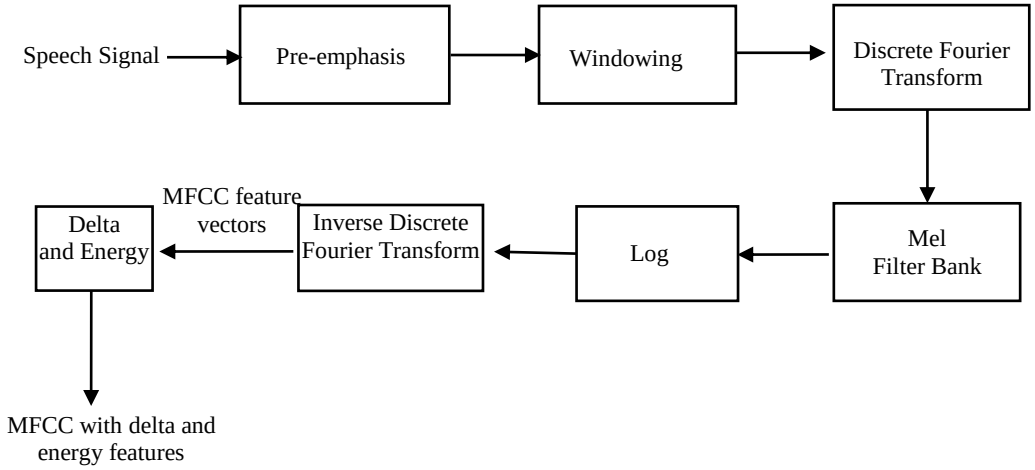


Figure 5.3 The Steps of the Mel Frequency Cepstral Coefficient Feature Extraction

5.6.1.1 Preemphasis

Pre-emphasis is the boosting the amount of energy in the high frequencies. Boosting the high frequency components improve the signal-to-noise ratio before they are transmitted or recorded onto a storage medium.

The formula for pre-emphasis filter is:

$$s_2(n) = s(n) - a * s(n-1) \qquad \text{Equation (5.31)}$$

where a is the pre-emphasis coefficient that should be in the range between 0.9 and 1, $s_2(n)$ is the output signal, $s(n)$ is input signal and $s(n-1)$ is the last input signal [29].

5.6.1.2 Windowing

Windowing is the process that creates a small speech window which describes a specific subphone, to be a stationary signal.

The windowing can be done by multiplying the signal value at time *n*, $s[n]$, with the value of the window at time *n*, $w[n]$:

$$y[n] = w[n] * s[n] \qquad \text{Equation (5.32)}$$

The simplest window is the rectangular window. But, the hamming window is more appropriate than the rectangular window for MFCC features extraction. It shrinks the values of the signal toward zero at the window boundaries, avoiding

discontinuities. Each window has a frame and the frame size is the number of milliseconds in the frame. The frame shift is the value in milliseconds between the left edges of successive windows.

The equations for rectangular window and the Hamming window are as described (assuming a window that is *L* frames long):

$$Rectangular\ w[n] = \begin{cases} 1 & 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases} \qquad \text{Equation (5.33)}$$

$$Hamming\ w[n] = \begin{cases} 0.54 - 0.46 \times \cos\left(\frac{2n\pi}{L}\right) & 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases} \qquad \text{Equation (5.34)}$$

5.6.1.3 Discrete Fourier Transform

The Discrete Fourier Transform (DFT) is applied to extract spectral information for discrete frequency bands for a discrete-time signal. The input to the DFT is a windowed signal x[n]…x[m]. The output, for each of N discrete frequency bands, is a complex number X[k] representing the magnitude and phase of that frequency component in the original signal. The DFT is defined as follows:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn} \qquad \text{Equation (5.35)}$$

Fast Fourier transform (FFT) is a commonly applied algorithm for computing the DFT. This DFT implementation is very effective; however, it only works for values of N that are powers of 2.

5.6.1.4 Mel Filter Bank

The outcome of the FFT [24] is information regarding the energy amount at each frequency band. Hearing of Human is not equally sensitive at all frequency bands. It is less sensitive above 1000 hertz of high frequencies. It proves that modeling of human hearing property through feature extraction increases speech recognition accuracy. The form of the model applied in MFCCs is to wrap the output of frequencies by the DFT onto the mel scale. A mel is a pitch unit. The sound pairs that are perceptually equidistant in pitch are separated by an equal number of mels. The mel frequency m can be calculated from that raw acoustic frequency like this:

$$mel(f) = 1127\ln(1 + \frac{f}{700}) \quad \text{Equation (5.36)}$$

5.6.1.5 Computing Log

During MFCC computation, the system performs that intuition by making filters bank that collect energy from each frequency band, in this case, 10 filters are spaced linearly below 1000 Hz and the remaining filters spread logarithmically above 1000 Hz. In this step, it takes the log of each of the mel spectrum values [24].

5.6.1.6 The Cepstrum: Inverse Discrete Fourier Transform

The next stage of MFCC feature extraction is the cepstrum computing. The cepstrum can be assumed as the spectrum of the log of the spectrum.

The cepstrum has a number of valuable processing benefits. And, it also significantly increases the performance of phone recognition.

The cepstrum c[n] is denoted as the inverse DFT of the log magnitude of the DFT of a signal, for a windowed frame of speech x[n], as follows;

$$c[n] = \sum_{n=0}^{N-1} \log(\left|\sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}\right|) e^{j\frac{2\pi}{N}kn} \quad \text{Equation (5.37)}$$

5.6.1.7 Deltas

A delta is the velocity feature and a double delta is the acceleration feature of a frame.

The delta value d(t) for a particular cepstral value c(t) at time t can be calculated as

$$d(t) = \frac{c(t+1) - c(t-1)}{2} \quad \text{Equation (5.38)}$$

5.6.2 Pitch Features

Pitch features are crucial for tonal languages because it brings phonological (tone) information. Rising, falling or pitch contours need to be aware of defining tones. Although pitch features are not enough to classify all the phonemes of a language, it needs to be modeled explicitly and it can be the most discriminating feature between two sounds [30].

Pitch features are extracted using Kaldi pitch tracker [10]. In these features, there are 3-dimensional pitch features (normalized pitch, delta-pitch, and voicing

features). It is a highly modified version of getf0 (RAPT) algorithm. It does not decide if any given frame is voiced or unvoiced. As an alternative, it allocates a pitch even to unvoiced frames whereas constraining the pitch trajectory to be continuous. Probability of voicing information is got from SAcC [25].

5.7 Language Model

The language model describes what is likely to be spoken in a particular context. Language modeling can be applied in many application areas of natural language processing technology where text is generated as output. It also plays a significant part in speech processing tasks such as ASR and tagging in natural language processing (containing part-of-speech tagging, word segmentation, etc.).

There are two types of language models: that are utilized in speech recognition tasks - grammars and statistical language models. The grammar-type language model describes very simple types of languages for command and control. They are typically written manually or produced automatically with plain code. The statistical language model uses stochastic approach called n-gram language model [28]. An n-gram is an n-token words sequences: a b2-gram (bigram) is a two-word sequence; a 3-gram (trigram) is a three-word sequence.

An n-gram model makes estimation of the probability of a length-N sentence w as

$$P(W) \approx \prod_{i=1}^{N+1} P(w_i | w_{i-n+1} \dots w_{i-1}) \qquad \text{Equation (5.39)}$$

where w_{N+1} and w_j for j < 1 are defined as special "sentence-boundary" tokens.

Many software packages for statistical-based language modeling are available and used for many years. One of the packages, the CMU-Cambridge Statistical Language Modeling toolkit [5], has been widely used in research works and has been helped the creating and testing of statistical language models.

Another popular statistical language modeling toolkit is SRI Language Modeling (SRILM) [51]. It gets advantages of from using and enhancing in the summer workshops of Johns Hopkins University/CLSP.

In this work, the language model was constructed by using the SRI Language Modeling (SRILM) language modeling toolkit.

5.7.1 Building Language Model Using SRILM

SRILM is for building statistical-based language models (LMs) toolkit, mostly for applying in machine translation, speech recognition, statistical tagging and segmentation [51]. SRILM is publicly available for noncommercial purposes. The main function of SRILM is to provide both estimation and evaluation of language model. Estimation means the model construction from training data and evaluation means calculating the test corpus probability, test set perplexity.

The command for estimating n-gram models in SRILM is *ngram-count*. For developing the language model for Myanmar language, 3-gram language model with default good-turing discounting is used. It takes training data as input text file and writes the output file in ARPA format.

The other discounting techniques can be applied to smooth the language model. They are absolute discounting, witten-bell discounting and modified kneser-ney discounting. One of these methods can be selected and used in SRILM toolkit.

Figure 5.4 describes an example of word-based n-gram language model file with ARPA format. In the header of ARPA format for n-gram backoff models, it shows how many unique n-gram types were observed of each order n up to the maximum order of the mode. After that, n-grams are listed one per line and they are grouped by n-gram order.

```
\data\
ngram 1=8,273
ngram 2=58,217
ngram 3=22,029
\1-grams:
-1.511959	</s>
-99	<s>	-0.6343088
-1.688553	က	-0.4625799
-4.48339 ကက်ရှ်စမီးယား	-0.5957683
…
\2-grams:
-4.134333	<s> ကက်ရှ်စမီးယား
-2.749582	<s> ကချင်-0.3827
…
\3-grams:
-0.2613638	ကိစ္စရပ် က တော့
-1.021788	ကုမ္ပဏီ က ဂလက်စီ
-1.136425	ကုမ္ပဏီ က စကား
…
\end\
```

Figure 5.4 Example of Word-based N-gram Language Model File with ARPA Format

5.7.2 Calculating Model Perplexity with SRILM

The perplexity is used for evaluation of the language model. The perplexity of a language model on a test set is the probability function that the language model assigns to that test set. For a test set, $W = w_1 w_2 \ldots w_N$ (words sequences of a vocabulary set, W) and the probability of a symbol w_i is based on the previous symbol

w_1,. . ., w_{i-1}.The perplexity W can be computed as the following equation:

$$pp(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_1..w_{i-1})}} \qquad \text{Equation (5.40)}$$

The perplexity on a new dataset can be calculated using SRILM's *ngram* command, using the *-lm* option to specify the language model file and the *-ppl* option to specify the test-set file [51].

Example output of the perplexity evaluation using this command is:

```
file test_corpus.txt: 2 sentences, 6 words, 0 OOVs
0 zeroprobs, logprob= -2.59329 ppl= 2.10941 ppl1= 2.70529
```

5.7.3 Smoothing Techniques in SRILM

The zero probability can be assigned to n-gram if it has not been occurred in the training data. To avoid the zero probabilities, some probability mass is taken from the observed n-gram and distributed it to unobserved n-gram. Such redistribution is identified as smoothing or discounting [17].

SRILM[1] offers many smoothing or discounting techniques. They are good-turning discounting, absolute discounting, kneser-ney discounting, witten-bell discounting and natural discounting. In this experiment, the smoothing techniques with backoff model are applied.

5.8 Decoding for ASR

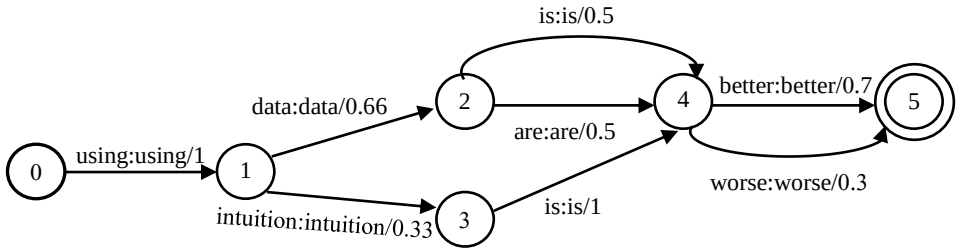
Currently, most of the large-vocabulary speech recognition system is based on models like HMMs, tree lexicons, or n-gram language models that are finite-state. It can be characterized by weighted finite-state transducers [33].

[1] https://www.srilm.com/ngram-discount.html

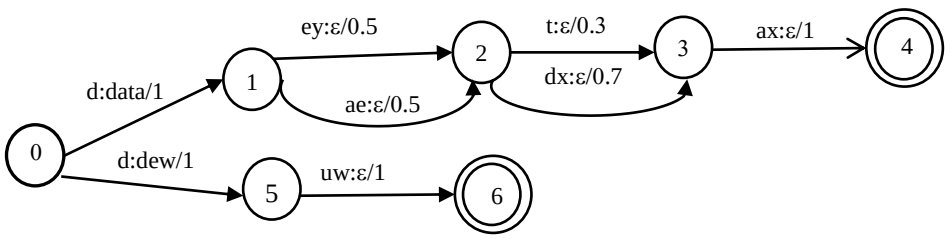
5.8.1 Weighted Transducers

Weighted finite-state transducers (WFSTs) associate pairs of symbol sequences and weights, that is, it represents a weighted binary relation between symbol sequences. A transition can be denoted by an arc from the source state to the destination state. It has the input label, the output label and the weight. The output label of a path is the concatenation of output labels of its transitions. Figure 5.5a represents is a toy finite-state language model by giving each transition identical input and output labels. This augments no new information; however, is an appropriate way of interpreting any acceptor as a transducer.

Figure 5.5b shows a toy pronunciation lexicon with a mapping from phone sequences to words in the lexicon. In the example data and dew, with probabilities representing the likelihoods of alternative pronunciations. Meanwhile the pronunciation of a word may be several phone sequences; the path corresponding to each pronunciation has ϱ-output labels on all but the word-initial transition. This transducer takes more information than the weighted finite-state acceptor (WFSA). The output label encodes the words and therefore, it is probable the pronunciation transducers can be combined for more than one word without losing word identity.



5.5 (a) A Toy Finite-State Language Model



5.5 (b) A Toy Finite-State Pronunciation Lexicon

Figure 5.5 Weighted Finite-State Transducer Examples

5.9 Experiments

In this experiment, the evaluation results on the train data, language model and the Gaussians numbers are discussed by using GMM and SGMM approaches.

5.9.1 Experimental Setup

The details of the experimental setup for data sets, acoustic and language models are dealt with in this section. Four different data sizes -10 hrs, 20 hrs, 30 hrs, and 42 hrs - are used for incremental training. The detailed statistics on the train and test sets are displayed in Table 5.1. TestSet1 is the open test data, which is web news data. TestSet2 is also open test data and it is the conversational data from natives recorded with voice recorders and microphones.

Table 5.1 Statistics on Training and Test Data

Data	Size	Domain	Speaker			Utterance
			Male	Female	Total	
TrainSet	10 Hrs 5 Mins	Web News	79	23	102	3,530
	20 Hrs 2 Mins	Web News	126	52	178	7,332
	30 Hrs 3 Mins	Web News + Conversational Data	174	86	260	15,556
	42 Hrs 39 Mins	Web News + Conversational Data	219	88	307	31,114
TestSet1	31 Mins 55 Sec	Web News	5	3	8	193
TestSet2	32 Mins 40 Sec	Conversational Data	3	2	5	887

5.9.1.1 GMM and SGMM Acoustic Model

An open-source Kaldi toolkit is utilized to develop the acoustic model [41]. The standard Mel-Frequency Cepstral Coefficients (MFCC) features with its first and second derivatives without energy features are used for the baseline GMM-based

acoustic model training. After that, cepstral mean and variance normalization (CMVN) is used on MFCC features. Then 9 frames of MFCCs are spliced together and linear discriminant analysis (LDA) is applied to project down to 40 dimensions. A maximum likelihood linear transform (MLLT) is used to estimate on the LDA features and the LDA + MLLT model is generated. Then, speaker adaptive training (SAT) is done with feature-space Maximum Likelihood Linear Regression (fMLLR) on the top of LDA + MLLT model. The baseline GMM model has 2050 context dependent (CD) triphone states with an average of 44 Gaussian components per state. In the SGMM experiment, Universal Background Model (UBM) is initialized by clustering the diagonal Gaussians that derived the HMM set to I = 400 Gaussians, and phonetic subspace S = 40 dimensions. LDA features with 40 dimensions are applied for SGMM training.

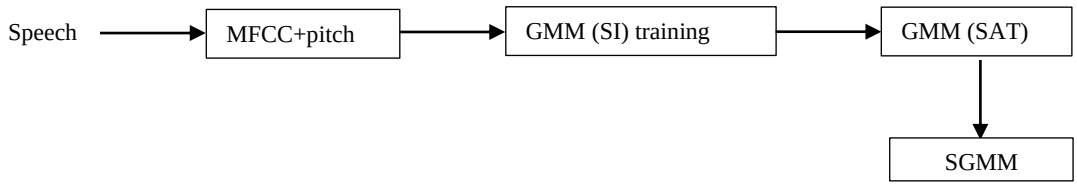


Figure 5.6 Flow Diagram of GMM and SGMM Acoustic Model Training

Figure 5.6 displays the process flow for GMM and SGMM acoustic models training. MFCC features with pitch features are extracted from speech frames. MFCC+LDA+MLLT is applied to the Speaker Independent (SI) GMM model training. And then, fMLLR are estimated on (SI) GMM model for SAT training and fMLLR transformed features are applied for SGMM training.

5.9.2 Evaluation with Number of Gaussian

Experiments are done to investigate the best number of Gaussian mixtures which corresponds to the best system's performance. The ASR performance on various numbers of Gaussians mixtures is displayed in Table 5.2.

Table 5.2 The WER % of the ASR Performance with the Number of Gaussian Mixtures at each HMM State

GMM-HMM	TestSet1	TestSet2
5-mixture Gaussian	28.62	37.06
9-mixture Gaussian	26.68	35.55
14-mixture Gaussian	25.38	34.76
19-mixture Gaussian	24.97	34.06
24-mixture Gaussian	24.45	33.56
29-mixture Gaussian	24.40	33.06
34-mixture Gaussian	**23.96**	**32.56**
39-mixture Gaussian	24.94	33.49
44-mixture Gaussian	24.18	33.29

According to Table 5.2, increasing the number of components in the GMM does not provide better accuracy. With 42 hrs training data set, the WERs on both open test sets decrease gradually using from 5-mixture Gaussian to the 34-mixture Gaussian. However, above the number of 34-mixture Gaussian, the WER rates on both test sets increase gradually starting from 39-mixture Gaussian. It shows that the lowest WERs, 23.96% on TestSet1 and 32.56% on TestSet2, are obtained with 34-mixture Gaussian and hence, it is the best number of Gaussian mixtures. The 34-mixture Gaussian is chosen for future GMM model building.

5.9.3 Evaluation on Training Data Size

The effect of varying the size of the training set on the error rate of the system depicts in a chart with word error rate as a function of training set in Figure 5.7.

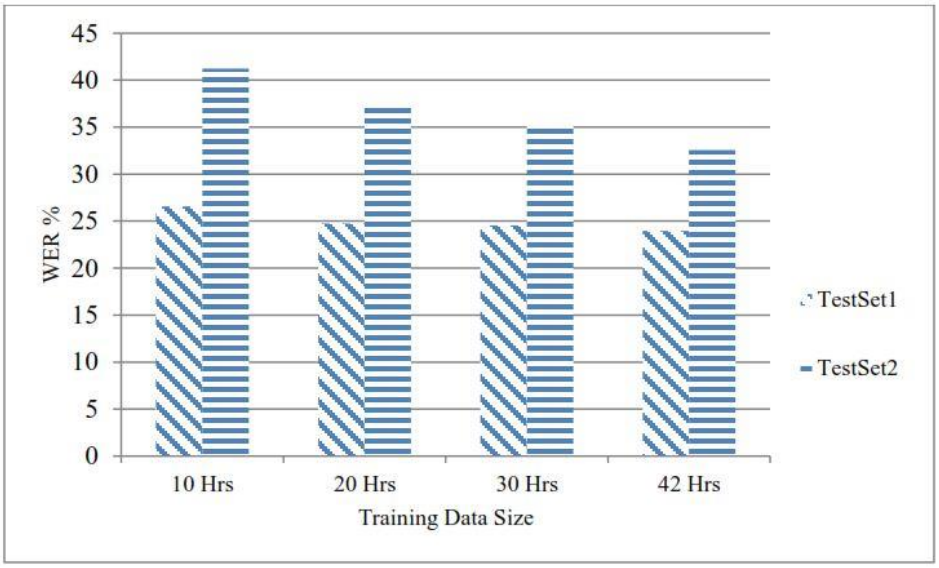


Figure 5.7 Chart Diagram of Word Error Rate % for Increasing Amount of Training Data

According to Figure 5.7, when the training data set size is increased from 10 hrs to 20 hrs, the WERs of TestSet1 decrease considerably because it is the same domain with the training sets. However, the error rates of TestSet1 are not reduced notably even when the training data size is increased from 30 hrs to 42 hrs because the augmented data is from a different domain. The word error rates of TestSet2 obviously decrease over the increasing training data size. This is because the augmented data of the training sets of 30 hrs and 42 hrs are the same domain with the TestSet2, which results in diminishing the word error rates of TestSet2.

It can be clearly seen that when the amount of training data is increased, WERs are decreased. The largest amount of training data, 42-hr-data set, has the lowest WERs on both test sets. Thus, the training data size has a great effect on the performance of ASR. Moreover, the error rates of TestSet1 are lower than those of TestSet2. This is because the news presenters have clear and sharp voices than the voices in the recorded conversational data. In addition, the total length of the web news data is longer than that of the recorded conversational data. As the result, using GMM leads to the lowest WERs of 23.96% on TestSet1 and 32.56% on TestSet2.

5.9.4 Evaluation with N-gram Language Model

Statistical language modeling has been used in different areas, consisting of speech recognition, optical character recognition, machine translation, and spelling

correction, etc. Today, n-gram language models dominate as the technology of choice for current speech recognizers. Thus, in this task, the ASR accuracy is investigated based on different n-gram language models (LMs). In addition, the performance is evaluated on two different types of acoustic models: baseline GMM and SGMM. Therefore, this work intends to investigate the performance of the acoustic model language model. The evaluation results of n-gram language models are depicted in Table 5.3.

Table 5.3 The WER % of the ASR Performance with N-gram Language Model

LM	**GMM**		**SGMM**	
N-gram	**TestSet1**	**TestSet2**	**TestSet1**	**TestSet2**
0-gram	56.84	65.56	51.37	60.54
1-gram	37.28	47.06	31.55	43.35
2-gram	24.15	33.05	23.97	29.77
3-gram	**23.96**	**32.56**	**21.56**	**27.50**
4-gram	25.13	34.06	23.95	30.44
5-gram	25.23	34.96	24.65	31.65

As presented in Table 5.3, the ASR performance is compared and evaluated based on different n-gram (0-gram to 5-gram) LMs. SRILM [51] language modeling toolkit is applied to create the language model by using the default smoothing technique, good-turing.

From the table, it is found that the highest WERs are reached by using 0-gram LM (without language model) on both open test sets. 3-gram based language model is the best and it has the lowest WERs among 0 to 5-gram. Therefore, for GMM-based acoustic model, the lowest WERs, 23.96% for TestSet1 and 32.56% for TestSet2 are obtained with 3-gram language model. There are 21.56% on TestSet1 and 27.50% on TestSet2 by using SGMM-based acoustic model. As the result, it can be clearly found that language model has an important role to improve ASR accuracy.

5.9.5 Evaluation on Different Smoothing Techniques

Smoothing techniques are usually used to better estimate probabilities when there is not enough data to estimate probabilities accurately. In this work, experiments

are performed with five different smoothing techniques (good-turing discounting, absolute discounting, kneser-ney, witten-bell, and natural discounting). Then, the impact of the smoothing methods is analyzed on the perplexity values and word error rates (WERs) for GMM and SGMM-based acoustic models.

Table 5.4 Comparison of Different Smoothing Techniques on Perplexity Values and Word Error Rate (WER)

Discounting Techniques	PPL	WER% onTestSet1		WER% on TestSet2	
		GMM	SGMM	GMM	SGMM
Good-Turing	115.94	23.96	21.56	32.56	27.50
Absolute	108.11	22.26	20.56	31.55	26.65
Kneser-Ney	**96.13**	**21.73**	**19.47**	**30.06**	**25.45**
Witten-Bell	109.31	23.32	21.01	31.65	26.95
Natural	114.54	23.43	21.33	32.32	27.30

Table 5.4 shows the perplexity values and WER% based on the different smoothing techniques. It is observed that smoothing techniques have a significant impact on reducing perplexity values and WERs for Myanmar language. The experimental results proved that kneser-ney discounting better than the other smoothing techniques in terms of perplexity and WERs. Therefore, the lowest perplexity values and the lowest error rates on both test sets are attained with kneser-ney discounting method.

CHAPTER 6

BUILDING CONVOLUTIONAL NEURAL NETWORK (CNN) - BASED ACOUSTIC MODEL

In this chapter, deep learning techniques, deep neural network (DNN) and convolutional neural network (CNN) for automatic speech recognition are described. Moreover, the optimized CNN architecture is investigated by varying the different CNN hyparameters such as number of the feature maps and pooling size. Furthermore, the effect of tones is explored at both syllable and word levels. The comparison of syllable-based ASR model and word-based ASR model is also discussed. The error analysis on the best hypothesis text of Myanmar ASR is presented.

6.1 Deep Neural Network (DNN)

Deep neural network (DNN) has been applied in many ASR tasks and it has gained significant performance than GMM [6] [15].

GMM that has a large number of components is inefficient as each parameter applies only to a small fraction of the data while each parameter of a product model is constrained by a large fraction of the data. The nonlinearity of the two models is different: DNN can exploit multiple frames of input coefficients (correlated data) while GMMs need uncorrelated data. Moreover, DNN learning uses stochastic gradient descent, GMM uses the Expectation-Maximization (EM) algorithm, and the GMM learning is much easier to parallelize.

A deep neural network (DNN) is a feed-forward, artificial neural network. There are more than one hidden layers between its inputs and outputs. Each hidden unit in hidden layers, j, normally applies the logistic function to map its total input from the layer below, x_j , to the scalar state, y_j that it sends to the layer above.

$$y_j = logistic(x_j) = \frac{1}{1+e^{-x_j}}, \quad x_j = b_j + \sum_i y_i w_{ij} \qquad \text{Equation (6.1)}$$

where b_j is the bias of unit j, i is an index over units in the layer below, and w_{ij} is a the weight on a connection to unit j from unit i in the layer below. For multiclass classification, output unit j transforms its total input, x_j, into a class probability, p_j, by applying the “softmax” non-linearity:

$$p_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)} \qquad \text{Equation (6.2)}$$

where *k* is an index over all classes.

The natural cost function C of softmax output is the cross-entropy between the target probabilities d and the softmax outputs, *p*:

$$C = -\sum_j d_j log p_j, \qquad \text{Equation (6.3)}$$

where the target probabilities, normally taking values of one or zero, are used for supervised training of the DNN classifier.

For large training sets, it is usually more effective to calculate the derivatives on a small, random "mini-batch" of training cases, rather than the whole training set, before updating the weights in proportion to the gradient [15]. This stochastic gradient descent method can be more enhanced by using a "momentum" coefficient, $0 < \alpha < 1$, that smooth the gradient calculated for mini-batch *t*, thus damping oscillations across ravines and speeding progress down ravines:

$$\Delta w_{ij}(t) = \alpha \Delta w_{ij}(t-1) - \epsilon \frac{\partial C}{\partial w_{ij}(t)} \qquad \text{Equation (6.4)}$$

6.2 Convolutional Neural Networks (CNN) for ASR

The convolutional neural network (CNN) is considered as a successful variant of the standard neural network. The CNN has a special network architecture, which involves convolution and pooling layers to get translation invariance and tolerance to small deformations in patterns. The CNN was initially motivated by image processing and it has yielded excellent results in a number of image recognition tasks [3].

6.2.1 Input Data Organization in CNN

In CNN, the data input are required to be formed in a certain way. Each different feature is consisted of in a different feature map. A feature map denotes the values of the same feature along different locations. For processing speech signals, features that are organized along frequency or time (or both) are applied in order to correctly use the convolution operation. In this sense, the well-known mel-frequency

cepstral coefficient (MFCC) features are not suitable for convolution over frequency because the decorrelating discrete cosine transform projects the data into a new basis that does not enjoy locality along the frequency axis. In other words, each MFC coefficient represents a feature extracted from the whole frequency spectrum. On the other hand, the log energy computed from a set of Mel filter banks (denoted as log mel-frequency spectral coefficients (MFSC) features) can be applied because each value denotes the energy in a different frequency band. Moreover, their first and second temporal derivatives can be appended. The features from a number of consecutive frames representing a context window of 11-15 frames are included as an input to the CNN. There are several different methods to organize these log-MFSC features as different maps for a CNN. In this work, 1-D convolution is conducted along frequency. A number of one-dimensional (1-D) feature maps are used. The different frames and different feature orders (static, first, and second derivatives) belong to different feature maps. Each feature map represents the values of the same feature along different frequency bands (filter bank indexes).

Once input feature maps are organized, convolution and pooling operations are used to generate the convolution and pooling layers activations in sequence as in Figure 6.1. Like in the input layer, each one of them has a number of feature maps as well. A convolution and pooling layer pairs in figure is usually called one CNN layer [13]. Obviously, more CNN layers can be added one by one to construct a deep CNN.

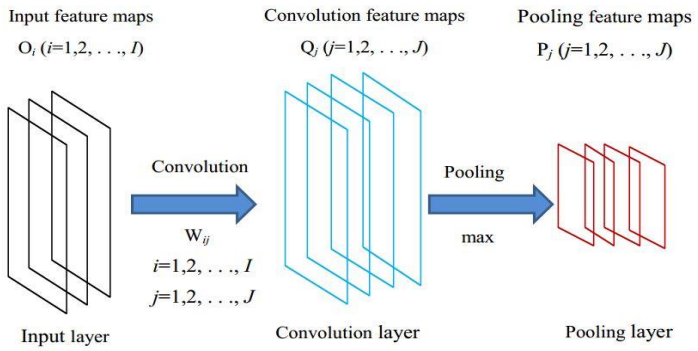


Figure 6.1 An Illustration of one CNN Layer

6.2.2 Convolution Layer

As displayed in Figure 6.1, all feature maps input (assume I in total), O_i(i =1, …, I) are mapped into feature maps numbers (assume J in total), Q_j(j = 1,…,J), in the convolution layers based on a number of local filters($I \times J$ in total), $w_{i,j}$(i = 1, . . ., I ; j

= 1, …, *J*). The mapping can be characterized as the recognized convolution operation in signal processing. Supposing input feature maps are all one dimensional, each unit of one feature map in the convolution layer can be calculated as:

$$q_{j,m} = \sigma\left(\sum_{i=1}^{I} \sum_{n=1}^{F} o_{i,n+m-1} w_{i,j,n} + w_{0,j}\right), \qquad (j = 1, \dots, J) \quad \text{Equation (6.5)}$$

where $o_{i,m}$ is the *m*-th unit of the *i*-th input feature map O_i , $q_{j,m}$ is the *m*-th unit of the *j*-th feature map Q_j of the convolution layer, $w_{i,j,n}$ is the n^{th} element of the weight vector, $w_{i,j}$, connecting the i^{th} feature map of the input to the j^{th} feature map of the convolution layer, and *F* is called the filter size which is the number of input bands that each unit of the convolution layer receives. The previous equation can be described as a more concise matrix form applying the convolution operator $*$ as:

$$Q_j = \sigma\left(\sum_{i=1}^{I} O_i * w_{i,j}\right) (j = 1, \dots, J), \qquad \text{Equation (6.6)}$$

where O_i represents the *i*-th input feature map and $w_{i,j}$ represents each local filter with the weights flipped to adhere to the convolution operation definition. Both O_i and $w_{i,j}$ are vectors if one dimensional feature maps are used. The number of feature maps in the convolution layer depends on how many sets of local filters are used in the convolutional mapping. Obviously, feature maps become smaller after each convolution operation, i.e., each dimension reduces by the filter size minus one due to convolution.

A convolution layer is different from a standard fully connected layer in a number of facts. Firstly, each unit takes input from a local area of the input, so the computed features have a locality property and thus each unit characterizes features of a local region of the input. Secondly, the units are organized in a number of feature maps, where all units that have in the same feature map share the same weights. But, they take input from different locations of the lower layer. Each feature map computes one feature of the input over all possible locations by applying the local filter defined by the map weights to the input through the convolution operation.

6.2.3 Pooling Layer

From Figure 6.1, a pooling operation is used in the convolution layer to generate the pooling layer above each convolution layer. The pooling layer is organized as a number of feature maps that is equal to the number of the feature maps in the convolution layer. The pooling layer serves two purposes.

Firstly, it reduces the resolution of feature maps to minimize the number of values to be fed into upper layers. Secondly, it adds invariance to small variations in location. This is attained by applying some pooling function at every location of the convolution feature map. The pooling function computes some overall property of a local region by using a simple function like maximization or averaging. The pooling function is applied to each convolution feature map independently so that each pooling unit applies the pooling function to a local range. If the max-pooling function is applied, the pooling layer is defined

$$p_{i,m} = max_{n=1}^{G}\ q_{i,(m-1)\times s+n} \qquad \text{Equation (6.7)}$$

where G is the pooling size, and s is a sub-sampling factor representing the shift between adjacent pooling regions. Likewise, if the average function is applied, the output is computed as:

$$p_{\mathrm{i,m}} = r \sum_{n=1}^{G} q_{i,(m-1)\times s+n} \qquad \text{Equation (6.8)}$$

where r is a scaling factor that can be learned. It has been shown that max-pooling outperforms than the average function in image recognition applications.

6.2.4 Learning Weights in the CNN

All weights in the convolution ply can be learned by applying the same error back-propagation algorithm but some special modifications are required to be careful of sparse connections and weight sharing. To demonstrate the learning algorithm for CNN layers, the convolution operation in Equation (6.6) is represented in the same mathematical form as the fully connected artificial neural network (ANN) layer.

When one-dimensional feature maps are applied, the convolution operations in Equation (6.6) can be denoted as a simple matrix multiplication by introducing a large sparse weight matrix $\widehat{W}$, which is organized by replicating a basic weight matrix W. The basic matrix W is created from all of the local weight matrices, $w_{i,j}$, as follows:

$$W = \begin{bmatrix} w_{1,1,1} & w_{1,2,1} & \dots & w_{1,J,1} \\ \vdots & \vdots & \vdots & \vdots \\ w_{I,1,1} & w_{I,2,1} & \dots & w_{I,J,1} \\ \vdots & \vdots & \vdots & \vdots \\ w_{I,1,2} & w_{I,2,2} & \dots & w_{I,J,2} \\ \vdots & \vdots & \vdots & \vdots \\ w_{I,1,F} & w_{I,2,F} & \dots & w_{I,J,F} \end{bmatrix}_{I \cdot F \times J} \qquad \text{Equation (6.9)}$$

where W is formed as $I \cdot F$ rows, where F means filter size, each band consists of I rows for I input feature maps, and W contains J columns denoting the weights of J feature maps in the convolution ply.

The input and the convolution feature maps are also vectorized as row vectors $\hat{o}$ and $\hat{q}$. One single row vector $\hat{o}$ is made from all of the input feature maps O_i = (i=1,. . ., I) as follows:

$$\hat{o} = [v_1 | v_2 | \dots | v_M] \qquad \text{Equation (6.10)}$$

where v_m is a row vector having the values of the m^{th} frequency band along all I feature maps, and M is the frequency bands numbers in the input layer. Thus, the convolution ply outputs calculated in Equation (6.6) can be equivalently stated as a weight vector:

$$\hat{q} = \sigma(\hat{o}\widehat{W}) \qquad \text{Equation (6.11)}$$

The convolution ply weights can be updated by applying the back-propagation algorithm. The updated for $\widehat{W}$ is similarly calculated as:

$$\Delta\widehat{W} = \epsilon \cdot \hat{o}' e. \qquad \text{Equation (6.12)}$$

where ϵ is the learning rate and e is the error signal vector.

The usage of shared weights in the convolution ply is a little different from the fully-connected DNN case. The difference is that for the shared weights, their updates are sum according to:

$$\Delta w_{i,j,n} = \sum_m \Delta\widehat{W}_{i+(m+n-2)\times I, j+(m-1)\times J} \qquad \text{Equation (6.13)}$$

The error signal reaching the lower convolution ply can be calculated as [13]:

$$e_{i,n}^{low} = \sum_m e_{i,m} \cdot \delta\left(u_{i,m} + (m-1) \times s - n\right) \qquad \text{Equation (6.14)}$$

where δ (x) is the delta function and it has the value of 1 if x is 0 and zero otherwise, and $u_{i,m}$ is the index of the unit with the maximum value among the pooled units and is described as:

$$u_{i,m} = argmax_{n=1}^{G} q_{i,(m-1)\times s+n} \qquad \text{Equation (6.15)}$$

6.2.5 Advantages of Using CNNs in ASR Tasks

CNN has three properties: locality, weight sharing, and pooling. Each of them has good potential to increase speech recognition accuracy. Locality property in convolution layer units permits more robustness in contrast to non-white noise in case some frequency bands are cleaner than the others. This is because features calculated in cleaner portions are less contaminated by noise. Noise only affects speech features in noisy frequency bands in the lower layers while noise can be better dealt with in the upper layers, which combine different frequency bands. In addition, locality decreases NN weights numbers to be learned and hence decreases overfitting. Weight sharing may increase the robustness of model and the reduction of overfitting as each weight is learned from all locations in the input rather than only one location. Both locality and weight sharing are required for pooling. In pooling, the same feature values calculated at different locations are pooled together and represented by one value. This leads to slight changes in the computed features when the input patterns are shifted, particularly when max-pooling is applied. This is very useful in management of shifting small frequency that is common in speech signals. Furthermore, it is hard to learn an operation like max-pooling in standard NNs. The same applies to temporal differences too. In the hybrid NN-HMM model, a number of frames within a context window are typically handled by the NNs. Varying speaking rate that causes temporary variability are hard to solve in standard NNs. CNNs can inherently handle this kind of variability when convolution is applied along the context window frames. However, since the CNN computes each frame output for decoding, pooling and sub-sampling may affect the fine temporal resolution seen by the higher layers of the CNN. A large pooling size may affect temporal localization of state labels. Therefore,

a suitable pooling size should be selected to balance temporal invariance and temporal localization of output labels [13].

6.3 Experiments on Optimization of CNN Parameters

In this experiment, the CNN hyperparameters are optimized to investigate on Myanmar ASR performance.

6.3.1 Experimental Setup for Optimization of CNN Parameters

Experiments are done by using the developed speech corpus that has 42-hr-data size. Two open test sets are used and the detailed information on the training and testing sets are depicted as in Table 5.1. 3-gram language model that applied Kneser-Ney discounting is built. The input features are 40-dimensional log mel-filter bank features padded with 11 frames (5 frames left and right frame contexts). For this work, one dimensional convolution across frequency domain is applied and there are two convolutional layers, one pooling layer and two fully connected hidden layers with 300 units per layer.

A CNN training flow in ASR is depicted in Figure 6.2. The baseline hidden Markov model (HMM)-Gaussian mixture model (GMM) ASR system is trained by applying 39 dimensional MFCC features. And then, CNN training is performed by using reliable alignments from the GMM-HMM system. In CNN training, 40 dimensional filter bank features are used. The final acoustic model is consisted of the HMM from the baseline HMM-GMM system and the new CNN. The targets are 2,052 context dependent (CD) triphone states that are generated from GMM-HMM model. For CNN training, 0.008 of constant learning rate is diminished by half based on cross-validation error decreasing. When the error rate has no more noticeably reducing or the error rate started to increase, training process stopped. Stochastic gradient descent with a mini-batch of 256 training examples is utilized for backpropagation. TESLA K80 GPU is used for all the neural networks training.

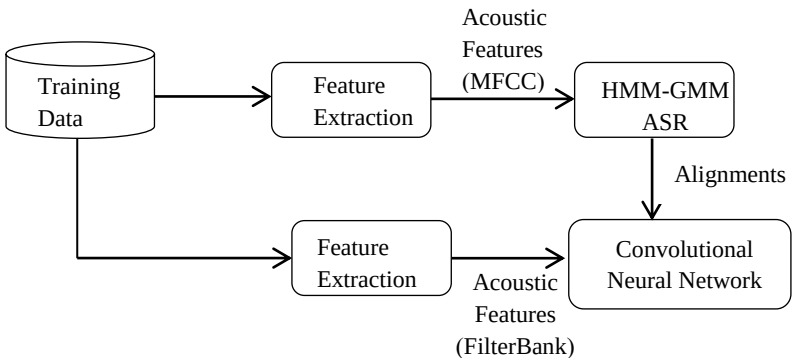


Figure 6.2 Flow Diagram of CNN Training for ASR

6.3.2 Number of Feature Maps of First Convolutional Layer

Firstly, different numbers of feature maps in the first convolutional layer are changed and compared on the evaluation results. The filter size of the first layer is set to 8. The number of feature maps is altered to 32, 64, 128, 256, 512, and 1,024 respectively. Table 2 depicts the WER% on two open test sets based on changing the feature map numbers of first convolutional layer.

Table 6.1 Evaluation Results on Number of Feature Maps of the First Convolutional Layer

Number of Feature Maps	WER%	
	TestSet1	TestSet2
32	18.97	24.01
64	18.59	23.80
128	**18.18**	**23.50**
256	18.52	23.75
512	18.31	23.65
1,024	18.53	23.85

It is observed that the lowest WERs on both test sets are obtained with feature map numbers of 128. According to Table 6.1, the WERs reduce slightly when the number of feature maps is increased up to 128. But, if the number of feature maps exceeds 128, it did not get the lower WERs and there is a small increment of WER. As the result, the lowest WERs are attained with 128 feature maps.

6.3.3 Pooling Size

Pooling has the ability in handling the small frequency shifts in speech signal. Max pooling has achieved a greater accuracy than the other pooling types in ASR tasks [17]. Thus, for this experiment, max pooling is used and the best pooling size is investigated. The pooling layer is added above the first convolution layer. The experiments are done by varying max pooling size with a shift size of 1. The feature maps of first convolution layer are fixed at 128. Table 6.2 gives the evaluation results based on the different pooling sizes.

Table 6.2 Evaluation Results on Pooling Size

	WER%	
	TestSet1	**TestSet2**
pool size=0	18.18	23.50
pool size=2	18.05	23.00
pool size=3	**17.22**	**22.56**
pool size=4	18.06	23.33

From the above table, it can be found that the first convolution layer using 128 feature maps, followed by the max pooling size 3 has the lowest WERs, 17.22% on TestSet1 and 22.56% on TestSet2. It indicates that after the pooling layer is appended, the WER diminishes significantly than without using it. Hence, pooling has an impact on the ASR performance and the lowest error rates are achieved with the pooling size of 3.

6.3.4 Number of Feature Maps of Second Convolutional Layer

The second convolutional layer is added on top of the pooling layer and the experiments are further done to investigate the best number of feature maps for the second convolutional layer. The filter size of this layer is set to 4. The best result of 128 feature maps is fixed in the first convolution layer. The max pooling layer with pool size 3 is also fixed in this experiment.

From Table 6.3, when the number of feature maps in the second convolutional layer is altered to 32, 64 and 128 respectively, it did not get the lower error rates on test sets. However, when the feature map numbers are set above 128, it slightly decreases the WERs on both test sets. Hence, the more number of filters the

convolutional layers have, the more acoustic features get extracted and the better the network becomes at recognizing acoustic patterns of unseen speakers.

Table 6.3 Evaluation Results on Number of Feature Maps in the Second Convolutional Layer

Number of Feature Maps	WER%	
	TestSet1	TestSet2
32	18.14	23.40
64	18.40	23.64
128	17.86	22.65
256	17.03	22.20
512	17.02	21.89
1,024	**16.70**	**21.83**

Therefore, it is analyzed that 128/1,024 feature maps in first and second convolutional layers with max pooling size 3 gives the best accuracy for Myanmar ASR.

6.4 Exploring the Effect of Tones on both Syllable and Word-Based ASR

In this section, the effect of tones is explored at both syllable and word-based ASR models using state-of-the-art acoustic models, DNN and CNN. The experimental setup for lexicon, tones clustering using a phonetic decision tree, building word-based and syllable-based ASRs, feature extraction and acoustic models are described.

6.4.1 Pronunciation Lexicon

Lexicon consists of a list of words, with pronunciation for each word described as a phone sequence. Grapheme-to-phoneme (g2p) converter [53] is applied to produce the pronunciation of the new words. In this work, two kinds of dictionary are utilized: dictionary with tone and dictionary without tone.

6.4.1.1 Tonal Pronunciation Lexicon

Myanmar language commission (MLC) dictionary is a standard dictionary and it is used as baseline [31]. This dictionary is increased with the vocabularies from the speech corpus. The lexicon has 36,700 words in total.

Table 6.4 depicts some words of Myanmar lexicon including tone informtion. Tones in Myanmar language are expressed as tone markers.

Table 6.4 Example of Myanmar Phonetic Dictionary with Tone

Myanmar Words	Phonetic
အ	a.
အားကစား	a: g a- z a:
အာကာသ	a k a th a.
အပ်နှင်း	a' n h in:

6.4.1.2 Non-Tonal Pronunciation Lexicon

Tone information does not contain in this dictionary. Some examples of Myanmar phonetic lexicon with no tone is described in Table 6.5.

Table 6.5 Example of Myanmar Phonetic Dictionary without Tone

Myanmar Words	Phonetic
အ	a
အားကစား	a g a z a
အာကာသ	a k a th a
အပ်နှင်း	a n h in

6.4.2 Tones Clustering Using Phonetic Decision Trees

The decision tree is used to search similar triphones, and share the parameters between them. The main advantage of the decision trees is that they do not restrict the questions scope in any way. This makes it possible to combine different information sources in the decision process [16]. To create the phonetic decision tree, the question set is considered and designed as follows. It is generated based on the phonological nature of vowels and consonants.

6.4.2.1 Same Base Vowel

The phonemes with the same base vowels are grouped together. The two examples of the same base vowels are shown in Table 6.6.

Table 6.6 Some Examples Phoneme Groups with the Same Base Vowels

Same base vowel
/a./ /a/ /a:/ /a'/
/ei./ /ei/ /ei:/ /ei'/

6.4.2.2 Same Tone

The phonemes that have the same tone are clustered together and the Table 6.7 describes the two examples of the phonemes with the same tone.

Table 6.7 Some Examples Phoneme Groups with the Same Tone

Same tone
/a./ /i./ /ei./ /an./ /e./ /in./ /o./ /ou./ /u./ /un./
/a/ /i/ /ei/ /an/ /e/ /in/ /o/ /ou/ /u/ /un/

6.4.3 Word-based and Syllable-based ASR Models

Two types of ASR Models: word-based and syllable-based ASR Models are developed.

6.4.3.1 Word-based ASR Model

In developing the word-based ASR model, word segmentation is done on the training and testing data. Language model is built by using the training data and lexicon is created with the words from the training data. 3-gram language model is utilized and there are 11,595 1-gram, 93,264 2-gram, and 51,035 3-gram. The vocabulary size of the training data is 12,875 unique words. For this word-based ASR model, WER is used in evaluating the ASR performance.

An example of word-segmented Myanmar sentence is as follows:

ဒီ သတင်း ကို တော့ ဒီကနေ့ ထုတ် မြန်မာ့ အလင်း သတင်းစာ က နေ ပြီးတော့ ရွေးချယ် ခဲ့ တာ ပါ ရှင်

6.4.3.2 Syllable-based ASR Model

The basic unit of Myanmar language is a syllable and therefore, the ASR performance is evaluated based on syllable. To develop the syllable-based ASR model, syllable segmentation is performed on training and testing data using the

syllable breaking tool[1]. It is a syllable segmentation tool for Myanmar language text encoded with Unicode (e.g. Myanmar3, Padauk). It is the regular expression rule based syllable breaking tool. Language model is constructed with the syllable units of the training data and syllable-based lexicon is developed. In the syllable-based 3-gram language model, there are 2,224 1-gram, 61,771 2-gram, and 73,009 3-gram. Lexicon has 4,398 syllables. For the syllable-based ASR model, syllable error rate (SER) is used in assessing the ASR performance.

An example of syllable-segmented Myanmar sentence is as follows:

ဒီ သ တင်း ကို တော့ ဒီ က နေ့ ထုတ် မြန် မာ့ အ လင်း သ တင်း စာ က နေ ပြီး တော့ ရွေး ချယ် ခဲ့ တာ ပါ ရှင်

6.4.4 Feature Extraction and Acoustic Models

The three different kinds of (GMM, DNN and CNN)-based acoustic models are developed for this experiment.

6.4.4.1 GMM

The training procedures for GMM are the same as in section 5.6.1 of Chapter 5. However, the best number of Gaussian mixtures, 34-mixture Gaussian, is used in this experiment.

6.4.4.2 DNN

In the DNN-based acoustic model, 4 layers that have 300 units per hidden layers are set. The DNN input features contains 40-dimensional log mel-filter bank features. Pre-training is not used in DNN training.

6.4.4.3 CNN

To build the CNN-based acoustic model, as in DNN input features, FBank, are also used in CNN. It consists of two convolution (conv) layers, one pooling (pool) layer and two fully connected (fc) layers with 300 hidden units. The structure of the CNN can be written as conv1-pool-conv2-fc1-fc2-softmax. The filter size is 8×4 and pre-training is not done. The optimized CNN archieture 128/1,024 feature maps in first and second convolutional layers with max pooling size 3 is applied.

[1] https://github.com/ye-kyaw-thu/sylbreak

6.4.5 Experimental Result

In this work, three experiments are done to assess the ASR performance on syllable and word-based ASRs with tone information.

- **Experiment1 (Exp1)**

  In the Exp1, using dictionary without tone, MFCC features are applied for GMM model and FBank features are utilized for DNN and CNN. Therefore, tone and pitch features are not included in this work.

- **Experiment2 (Exp2)**

  The pitch features are augmented into the acoustic models and tone information is not added in phonetic dictionary. Hence, only pitch features are applied in Exp2.

- **Experiment3 (Exp3)**

  Exp3 is done using pitch features and dictionary with tone. In addition, tonal extra questions (same base vowel and same tone question sets) are also applied in building the phonetic decision tree.

In this test, the effect of pitch and tone features is analyzed on word-based ASR model performance. Table 6.8 displays that the evaluation results over tone and pitch features for word-based ASR models. The lowest WERs are displayed in highlighted.

Table 6.8 Word-based ASR Model Performance Evaluation based on Tone and Pitch Features

Models	WER%					
	Exp1		Exp2		Exp3	
	TestSet1	TestSet2	TestSet1	TestSet2	TestSet1	TestSet2
GMM-HMM	32.47	35.55	32.10	35.33	21.73	30.06
DNN	30.95	32.92	29.42	31.52	19.23	27.76
CNN	**27.70**	**30.45**	**27.66**	**29.92**	**16.70**	**21.83**

Without using tones and pitch features, there are WERs of 32.47% for GMM model, 30.95% for DNN model and 27.70% for CNN model on web news. There are error rates of 35.55% for GMM, 32.92% for DNN and 30.45% for CNN on recorded conversational data. As a result, CNN outperformed over GMM and DNN. After augmenting the pitch features, for the web news, it obtained lower WERs of 0.37%, 1.53% and 0.04% and for the conversational data, 0.22%, 1.40%, and 0.53% over GMM, DNN and CNN than without using pitch. These results prove that pitch features give better performance on all the three models than without using it. When both pitch and tone features are added, it notably reduced the error rates on both test sets with the CNN model. The lowest WERs of 16.70% on the web news and 21.83% on the conversational data by augmenting the tone and pitch features.

Table 6.9 shows the syllable-based ASR model performance over tones and pitch features. The lowest SERs are displayed in bold. With no tones and pitch information, there are 27.67% SER on GMM, 25.18% SER on DNN, and 22.14% SER on CNN for TestSet1, web news. There are 29.95% SER on GMM, 26.55% SER on DNN, and 23.35% SER on CNN for TestSet2, conversational data. For both test sets, when the pitch features are added in acoustic models, the syllable error rates decreased slightly.

Table 6.9 Syllable-based ASR Model Performance Evaluation based on Tone and Pitch Features

Models	SER%					
	Exp1		Exp2		Exp3	
	TestSet1	TestSet2	TestSet1	TestSet2	TestSet1	TestSet2
GMM	27.67	29.95	26.64	29.55	18.70	24.49
DNN	25.18	26.55	24.84	25.77	16.75	21.19
CNN	**22.14**	**23.35**	**21.35**	**22.86**	**15.00**	**18.33**

For all experiments, it is found that SER decreases largely after the tones and pitch information is applied. It showed that CNN-based model with tones and pitch information yields lower 7.14% of SER for web news and 5.02% of SER for recorded conversational data than that of without tones and pitch. And, with CNN-based model

using tones and pitch features, the lower SERs of 6.35% for web news and 4.53% for conversational data is obtained than CNN with pitch information. Among the three different models, the lowest syllable error rates, 15.00% for TestSet1 and 18.33% for TestSet2, are achieved by using the CNN with tones and pitch information.

Hence, it can be obviously observed that tone and pitch information are crucial to improve the ASR accuracy for Myanmar language.

6.5 Comparison of Syllable-based vs. Word-based ASR Models

In this experiment, the evaluation results of the syllable-based ASR and word-based ASR models are compared.

Table 6.10 describes the evaluation results of word-based and syllable-based ASR Models in terms of word error rate (WER%) and syllable error rate (SER%). For word-based ASR model, there are 16.70% WER on TestSet1, web news and 21.83% WER on TestSet2, conversational data. For syllable-based model, there are 15.00% SER on TestSet1 and 18.33% SER on TestSet2.

Table 6.10 Evaluation Results of Word-based Model and Syllable-based Model

Model	TestSet1	TestSet2
Word-based ASR Model (WER%)	16.70	21.83
Syllable-based ASR Model (SER%)	15.00	18.33

However, it cannot make direct comparison between syllable-based and word-based models because they are not the same in units (syllable vs. word). The segmentation of hypothesis of word-based model is needed to adjust to the syllable units. Therefore, syllable segmentation is done on the hypothesis texts of the word-based model. Then, the word-based and syllable-based models are compared again.

Table 6.11 describes the evaluation results of word-based and syllable-based model on syllable units. According to the results, it is found that the word-based model has lower error rates than the syllable-based model. Hence, it can be said that the accuracy of word-based model is better than that of syllable-based model.

Table 6.11 Evaluation Results of Word-based Model and Syllable-based Model on Syllable Units

Model	TestSet1	TestSet2
Word-based ASR Model (SER%)	9.70	16.80
Syllable-based ASR Model (SER%)	15.00	18.33

In addition, the hypothesis texts of word-based and syllable-based ASR models are manually evaluated by a native speaker. It also analyzed on the two open test sets. There are 193 utterances in web news and 887 utterances in conversational data. The hypothesis texts are evaluated on three different criteria: (1) the hypothesis text of word-based model is more meaningful than that of syllable-based model, (2) the hypothesis text of syllable-based model is more meaningful than that of word-based model, and (3) neutral which means the hypothesis texts of both word-based and syllable-based models are the same. Otherwise, it is difficult to analyze which model is better.

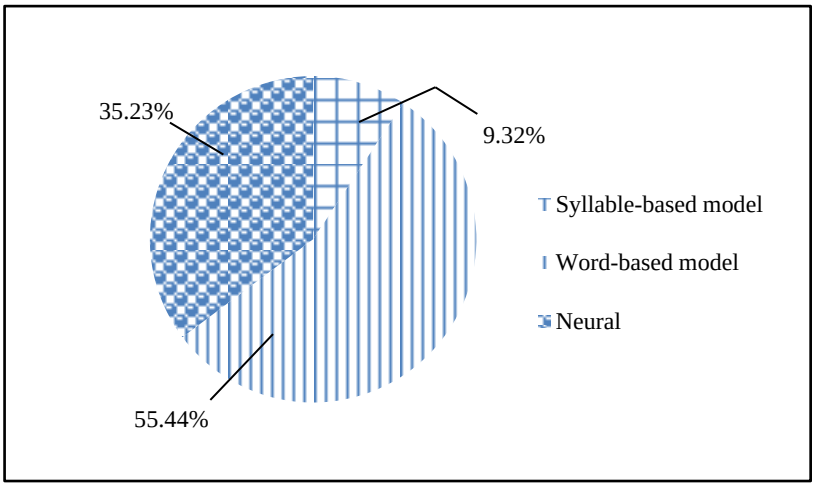


Figure 6.3 Evaluation on Hypothesis Text of TestSet1, Web News

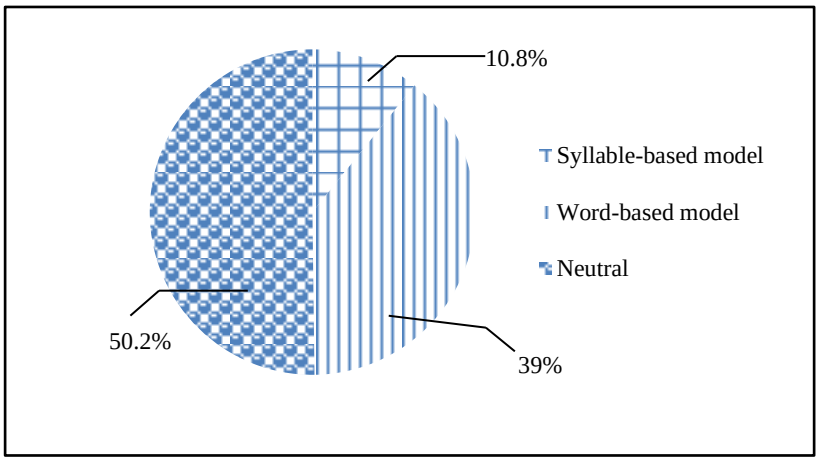


Figure 6.4 Evaluation on Hypothesis Text of TestSet2, Recorded Conversational Data

Figure 6.3 shows the evaluation results of hypothesis texts on web news. For syllable-based model, 9.32% of hypothesis texts are more meaningful than word-based model. The 55.44% hypothesis texts of word-based model give more meaningful results than that of syllable-based model. The rest output texts of both models are the same and it is difficult to decide which model is better to understand.

Figure 6.4 depicts the evaluation results of hypothesis texts on recorded conversational data. For syllable-based model, 10.8% of output texts are more meaningful than that of word-based model. And, for word-based model, 39% of hypothesis texts are more reasonable than syllable-based model. The rest hypothesis texts of two models are difficult to analyze for which model is better.

Therefore, it can be concluded that for both test sets, the output texts of word-based ASR model are more meaningful than that of the syllable-based ASR model. Furthermore, according to the evaluation results of word-based and syllable-based models on syllable units, the word-based model has better accuracy than the syllable-based model. Some output of the comparison of that two models results are shown in Appendix B.

6.6 Error Analysis

Error analysis is done based on the recognition outputs, hypothesis text. SCLITE[2] (score speech recognition system output) program from the NIST scoring toolkit SCTK version 2.4.10 is used. SCLITE has the ability to align a hypothesized text (HYP) of recognizer with a correct or reference text (REF) of human transcription.

An example evaluation of one of the referential Myanmar sentences for word-based ASR is:

REF: နှစ် ထောင့် ဆယ့် လေး ခုနှစ် မှာ ဓာတ်အား ငါး ဆယ့် နှစ် ********** မဂ္ဂါဝပ် စတင် ထုတ်လုပ် ခဲ့ ပါတယ်
HYP: နှစ် ထောင့် ဆယ့် လေး ခုနှစ် **မှ** ဓာတ်အား ငါး ဆယ့် နှစ် **မ ကောင်း** **ပွဲ** စတင် ထုတ်လုပ် ခဲ့ ပါတယ်
Eval: S I I S

Scores: (#C #S #D #I) 13 2 0 2

In this example, there are 13 Correct (C) words, 2 Substitution (S) words, 0 Deletion (D) words and 2 Insertion (I) words in the sentence. The WER of that sentence is 26.67%. According to the evaluation results, there are 4 significant types of errors found in the experiment with 42 hrs training set.

6.6.1 Similar Pronunciation Error

This system falsely recognized the words that have similar pronunciations. For instance, Myanmar word "သြဂုတ်" ("o: goú") was incorrectly recognized as "အုတ်ဖုတ်" ("oú hpoú"). Another example is the word "နေပြီးတော့" ("nei pji: do.") is wrongly output as "နေပြည်တော်" ("nei pji to") . In example cases, since both reference and hypothesis words have the similar pronunciation, this can cause the error and it leads to increase the WER. There are 8.48% of similar pronunciation errors.

6.6.2 Tone Error

Tone mistakes were also occurred in this experiment. It can be that Tone1 is misrecognized as Tone3, Tone2 is erroneously produced as Tone1, etc. For example, the Myanmar word "မ့" (mha.) gave incorrect result "မှာ" ("mha"). It is the misrecognition of Tone3 with Tone1. Another example of tone error is that the word "အကောင်းစား" ("a- kaun: za:") is wrongly recognized as "အကောင့်စ" ("a- kaun. za.")

[2] http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sclite.htm

and it is the error of tone recognition, Tone2 is mistakenly output as Tone3. This type of tone errors rate is 7.09% on the other types of errors.

6.6.3 Vowel Error

Some vowels were misrecognized in the ASR output. As an example, the Myanmar word "သံတွဲ" ("than dwe:") was falsely recognized as "သံတွေ" ("than dwei"). In this case, the vowel 'e:' is incorrectly recognized as 'ei'. The next example is that the word "ကရင်" ("ka- jin") is produced as "ကရဲ့" ("ka- je."). The vowel 'e.' is not correct in output text. There are 7.89% of vowel errors.

6.6.4 Ambiguous Error

Some ambiguous cases were not clearly defined in this work. As an example, the Myanmar word "စက်မှုကျောင်း" ("sé mhu. kyaun:") was confused as "ဆက်မှုကြောင်း" ("hsé mhu. kyaun:"). The other example is that the word "ဆီ" ("hsi") is ambiguous as "စီ" ("si"). Both words appear to have the same pronunciation but, they are different words and have different meanings. Only a few percentages of errors are found among the other types of errors.

CHAPTER 7

CONCLUSION AND FURTHER EXTENSION

This chapter describes summarization of the research work. The advantages and limitations of the system are presented. It also indicates promising avenues for future research on Myanmar ASR.

7.1 Thesis Summary

In this research, a large vocabulary continuous speech recognition for Myanmar language is proposed by using state-of-the-art acoustic model, convolutional neural network (CNN). Myanmar is being regarded as a low-resourced language because there is no freely and commercially available Myanmar speech corpus. Therefore, in this work, speech corpus is built by using two types of domain: web news and daily conversations for Myanmar ASR. The news is collected from the Internet and the conversational data is recorded by ourselves. There are over 42 hrs data size in the speech corpus which is collected from 219 males and 88 females.

The baseline GMM-HMM acoustic model is built by using the speech corpus as training data. It was evaluated on two open test sets: web news and recorded data. The experiments are done according to the training data, language model and number of Gaussian in HMM with GMM and SGMM approaches. It can be concluded that training data and language model are crucial to improve the ASR performance. Moreover, DNN-based and CNN-based acoustic models are developed and compared their results. It showed that CNN outperformed over DNN and GMM, and the best accuracy is achieved with CNN-based model in Myanmar ASR. The better accuracy of automatic speech recognition for Myanmar language is investigated by optimizing the hyperparameters of CNN. It is found that feature map numbers and pooling sizes of CNN have a great impact on ASR performance.

Myanmar language is one of the tonal languages and different types of tones convey the difference in meanings. In addition, syllable is the basic unit of Myanmar language. Hence, in this work, the effect of tones is explored on both syllable and word-based ASR models. Phonetic decision tree is built by using tonal questions and it is applied in tone modeling. It is proved that CNN with tone information achieves the better accuracy than those of without using tones. It can be said that tone

information plays an important role in increasing the accuracy of ASR for Myanmar language. The performance of word-based and syllable-based ASR models is compared on two open test sets. It can be found that the output texts of word-based ASR model are more meaningful than that of the syllable-based ASR model. Furthermore, according to the evaluation results of word-based and syllable-based models on syllable units, the word-based model has better accuracy than the syllable-based model.

Finally, error analysis is performed on the recognition outputs, hypothesis texts. It is observed that 4 significant types of errors are found. They are similar pronunciation error, tone error, vowel error, and ambiguous error. The statistics of these types of errors are also described. This error analysis will be taken into account for future ASR performance improvement.

7.2 Advantages and Limitations of the System

This Myanmar automatic speech recognition achieved good quality on read speech as it has been proved in the previous chapters. It can recognize both broadcast news and daily conversations using the training data that consists of both types of data. The recognition accuracy of web news data has got better than that of daily conversations from our research. The system is able to perform recognition for both speaker dependent and speaker independent, that is, it can recognize the voices of any speakers since it is trained by using a large number of different speakers. The recognition accuracy of female speakers is better than that of male speakers. It is because females outnumber males in training data. This continuous speech recognition system can recognize long utterances with very large vocabularies. It can help individuals in the disability community. This can assist for hearing impaired persons in reading online news. Moreover, it is also convenient and useful to news reporter in automatic transcribing the recorded audios. It can be integrated into Myanmar speech processing systems such as speech to speech translation, dictation, voice commanding, automatic question and answering, and some robotic applications.

Meanwhile, there are limitations and weaknesses in the system. The recognition accuracy of spontaneous speech is rather lower than that of read speeches. It can degrade the ASR performance and quality of the recognition process at noisy environment since the data are recorded in clean environment. Furthermore, it cannot

perform the recognition process for English words directly, instead transliteration to Myanmar words are produced because training was done on Myanmar language. In addition, syllable-based ASR model has better performance in the recognition of the proper names, for example, names of people, locations, etc., than word-based ASR model.

7.3 Future Work

Traditional ASR systems are involved of an acoustic model (AM), a lexicon and a language model (LM), all of which are needed to train independently on different data sets. Training the independent components makes extra complexities. Nowadays, there has been increasing popularity in building end-to-end systems, which try to learn these components together as a single system. This simplifies the training and deployment processes. Moreover, it allows handling a different variety of speech including noisy environments, accents and different languages. Therefore, in the future, end-to-end learning approach will be used for building Myanmar automatic speech recognition system.

AUTHOR'S PUBLICATIONS

[p1] A.N.Mon, W.P.Pa and Y.K.Thu, "Building HMM-SGMM Continuous Automatic Speech Recognition on Myanmar Web News", In Proceedings of the 15th International Conference on Computer Applications (ICCA 2017), Yangon, Myanmar, pp. 446-453, February 16-17, 2017.

[p2] A.N.Mon, W.P.Pa and Y.K.Thu, "Exploring the Effect of Tones for Myanmar Language Speech Recognition Using Convolutional Neural Network (CNN)", In Proceedings of the 15th International Conference of the Pacific Association for Computational Linguistics (PACLING), Yangon, Myanmar, pp. 314-326, August 16-18, 2017.

[p3] A.N.Mon, W.P.Pa, Y.K.Thu and Y. Sagisaka, "Developing A Speech Corpus From Web News for Myanmar(Burmese) Language", In Proceedings of the 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA 2017), Seoul, R.O.Korea, pp. 1-6, November 1-3, 2017.

[p4] A.N.Mon, W.P.Pa and Y.K.Thu, "Improving Myanmar Automatic Speech Recognition with Optimization of Convolutional Neural Network Parameters", International Journal on Natural Language Computing (IJNLC), Vol.7, No.6, December 2018.

BIBLIOGRAPHY

[1] D.Bahdanau, J.Chorowski, D.Serdyuk, P.Brakel, and Yoshua Bengio, "End-to-End Attention-based Large Vocabulary Speech Recognition", 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, pp. 4945-4949, March 20-25, 2016.

[2] F.Barbieri and H.Saggion, "Modelling Irony in Twitter", in Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, Gothenburg, Sweden, pp. 56-64, April 26-30, 2014.

[3] M.Browne and S.S.Ghidary, "Convolutional Neural Networks for Image Processing: An Application in Robot Vision", AI 2003: Advances in Artificial Intelligence, pp. 641-652, 2003.

[4] W.Chan, N.Jaitly, Q.V.Le, and O.Vinyals, "Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2016, Shanghai, China, pp. 4960-4964, March 20-25, 2016.

[5] P.Clarkson and R.Rosenfeld, "Statistical Language Modeling Using the CMU-Cambridge Toolkit", Fifth European Conference on Speech Communication and Technology, EUROSPEECH 1997, Rhodes, Greece, September 22-25, 1997.

[6] G.E.Dahl, D.Yu, L.Deng, and A.Acero, "Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition", in IEEE Transactions on Audio, Speech, and Signal Processing, Vol. 20, No. 1, pp. 30–42, January 2012.

[7] K.H.Davis, R.Biddulph, and S.Balashek, "Automatic Recognition of Spoken Digits", The Journal of the Acoustical Society of America, Vol. 24, No. 6, pp. 637–642, 1952.

[8] L.Deng, J. Li, J.Huang, K. Yao, D. Yu, F.Seide, et al., "Recent Advances in Deep Learning for Speech Research at Microsoft", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), May 2013.

[9] L.Deng and X.Li, "Machine Learning Paradigms for Speech Recognition: An Overview", IEEE Transactions on Audio, Speech, and Language Processing, Vol. 21, No. 5, pp. 1060-1089, 2013.

[10] P.Ghahremani, B.BabaAli, D.Povey, K.Riedhammer, J.Trmal, and S.Khudanpur, "A Pitch Extraction Algorithm Tuned for Automatic Speech Recognition", in IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, pp. 2494-2498, May 4-9, 2014.

[11] A.Graves and N.Jaitly, "Towards End-To-End Speech Recognition with Recurrent Neural Networks", Proceedings of the 31th International Conference on Machine Learning (ICML), Beijing, China, pp. 1764-1772, June 21-26, 2014.

[12] J.F.Gruber, M.S., "An Articulatory, Acoustic, and Auditory Study of Burmese Tone", A Dissertation Submitted to the Faculty of the Graduate School of Arts and Sciences of Georgetown University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Linguistics, Washington, D.C. June 15, 2011.

[13] O.A.Hamid, A.R.Mohamed, H.Jiang, L.Deng, G.Penn, and D.Yu, "Convolutional Neural Networks for Speech Recognition", IEEE/ACM Transactions on Audio, Speech, And Language Processing, Vol. 22, No. 10, October 2014.

[14] N.Hateva, P.Mitankin, and S.Mihov, "Bulphonc: Bulgarian Speech Corpus for the Development of ASR Technology", in Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portoroz, Slovenia, May 23-28, 2016.

[15] G.Hinton, L.Deng, D.Yu, G.Dahl, A.Mohamed, N.Jaitly, A.Senior, V.Vanhoucke, P.Nguyen, T.Sainath, and B.Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition", in IEEE Signal Processing Magazine, Vo. 29, No. 6, pp. 82-97, November 2012.

[16] T.Hirsimäki, "A Review: Decision Trees in Speech Recognition", May 27, 2003.

[17] X.Hu, M.Saiko, and C.Hori, "Incorporating Tone Features to Convolutional Neural Network to Improve Mandarin/Thai Speech Recognition", Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2014, Chiang Mai, Thailand, pp.1-5, December 9-12, 2014.

[18] X.Hu, X.Lu, and C.Hori, "Mandarin Speech Recognition Using Convolution Neural Network with Augmented Tone Features", The 9th International Symposium on Chinese Spoken Language Processing, Singapore, pp.15-18, September 12-14, 2014.

[19] B.H.Juang and L.R.Rabiner, "Automatic Speech Recognition–A Brief History of the Technology Development", Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara, January, 2005.

[20] D. Jurafsky, J.H.Martin, "Speech and Language Processing", 2nd edition, Pearson Prentice Hall Series in Artificial Intelligence, 2008.

[21] D.Jurafsky and J.H.Martin, "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics, and Speech Recognition, Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2000.

[22] I.Khaing and K.Z.Lin, "Design and Implementation of Speech Recognition System for Myanmar", Proceeding of International Conference on Computer Science & Human Computer Interaction (ICSSHCI 2014), 2014.

[23] P.K.Kurzekar, R.R.Deshmukh, V.B.Waghmare and P.P.Shrishrimal, "Continuous Speech Recognition System A Review", Asian Journal of Computer Science and Information Technology, Vol. 4, No. 6, pp. 62-66, 2014.

[24] P.K.Kurzekar, R.R.Deshmukh, V.B.Waghmare, P.P.Shrishrimal, "A Comparative Study of Feature Extraction Techniques for Speech Recognition System", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 3, Issue 12, pp. 18006-18016, December 2014.

[25] B.S.Lee and D.P.W.Ellis, "Noise Robust Pitch Tracking by Subband Autocorrelation Classification", in INTERSPEECH 2012, 13th Annual

Conference of the International Speech Communication Association, Portland, Oregon, USA, pp. 707-710, September 9-13, 2012.

[26] L.Lu, "Subspace Gaussian Mixture Models for Automatic Speech Recognition", Doctor of Philosophy Institute for Language, Cognition and Computation, School of Informatics, University of Edinburgh, 2013.

[27] S.Mandal, B.Das, P.Mitra, and A.Basu, "Developing Bengali Speech Corpus for Phone Recognizer Using Optimum Text Selection Technique", in International Conference on Asian Language Processing, IALP 2011, Penang, Malaysia, pp. 268-271, November 15-17, 2011.

[28] C.Martins, A.Teixeira, J.Neto, "Language Models in Automatic Speech Recognition", REVISTA DO DETUA, Vol. 4, No. 2, 2004.

[29] N.A.Meseguer, "Speech Analysis for Automatic Speech Recognition", Norwegian University of Science and Technology, Noelia, Vol. 3, Issue 12, December 2014.

[30] F.Metze, Z.A.W.Sheikh, A.Waibel, J.Gehring, K.Kilgour, Q.B.Nguyen, and V. H.Nguyen, "Models of Tone for Tonal and Non-Tonal Languages", in 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, pp. 261-266, December 8-12, 2013.

[31] MLC, Myanmar Language Commission, "Myanmar-English Dictionary", Department of the Myanmar Language Commission, Yangon, Ministry of Education, Myanmar, 1993.

[32] MLC, Myanmar Language Commission, "Myanmar Grammar", 30th Year Special Edition, University Press, Yangon, Myanmar, 2005.

[33] M.Mohri, F.Pereira and M.Riley, "Weighted Finite-State Transducers in Speech Recognition", International Journal of Computer Speech & Language, Vol. 16, No. 1, pp. 69-88, 2002.

[34] T.Nadungodage, V.Welgama, and R.Weerasinghe, "Developing a Speech Corpus for Sinhala Speech Recognition" In: ICON-2013: 10th International Conference on Natural Language Processing, CDAC Noida, India, 2013.

[35] H.M.S.Naing, A.M.Hlaing, W.P.Pa, X.Hu, Y.K.Thu, C.Hori, and H.Kawai, "A Myanmar Large Vocabulary Continuous Speech Recognition System", In

Asia Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2015, Hong Kong, pp. 320–327, December 16-19, 2015.

[36] H.M.S.Naing and W.P.Pa, "Automatic Speech Recognition on Spontaneous Interview Speech", Proceedings of 16th International Conference on Computer Applications (ICCA), pp.446-453, 2018.

[37] T.Neuberger, D.Gyarmathy, T.E.Gráczi, V.Horváth, M.Gósy, and A.Beke, "Development of a Large Spontaneous Speech Database of Agglutinative Hungarian Language", in Text, Speech and Dialogue - 17th International Conference, TSD 2014, Brno, Czech Republic, pp. 424-431, September 8-12, 2014.

[38] V.H.Nguyen, C.M.Luong, and T.T.Vu, "Tonal Phoneme Based Model for Vietnamese LVCSR", 2015 International Conference Oriental COCOSDA held jointly with 2015 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), Shanghai, China, pp.118-122, October 28-30, 2015.

[39] T.T.Nwe and T.Myint, "Myanmar Language Speech Recognition with Hybrid Artificial Neural Network and Hidden Markov Model", In Proceedings of 2015 International Conference on Future Computational Technologies (ICFCT'2015), Singapore, pp. 116–122, March 29-30, 2015.

[40] W.P.Pa, Y.K.Thu, A.M.Finch, and E.Sumita, "Word Boundary Identification for Myanmar Text Using Conditional Random Fields", in Genetic and Evolutionary Computing -Proceedings of the Ninth International Conference on Genetic and Evolutionary Computing, ICGEC 2015, Yangon, Myanmar, pp. 447-456, August 26-28, 2015.

[41] D.Povey, A.Ghoshal, G.Boulianne, L.Burget, O.Glembek, N.Goel, M.Hannemann, P.Motlicek, Y.Qian, P.Schwarz, J.Silovsky, G.Stemmer, and K.Vesely, "The Kaldi Speech Recognition Toolkit", in IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, December 2011.

[42] T.N.Sainath, A.Mohamed, B.Kingsbury, and B.Ramabhadran, "Deep Convolutional Neural Networks for LVCSR", IEEE International Conference

on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, pp. 8614-8618, May 26-31, 2013.

[43] T.N.Sainath, B.Kingsbury, A.Mohamed, G.E.Dahl, G.Saon, H.Soltau, T.Beran, A.Y.Aravkin, and B.Ramabhadran, "Improvements to Deep Convolutional Neural Networks for LVCSR", 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, pp. 315-320, December 8-12, 2013.

[44] T.N.Sainath, B.Kingsbury, G.Saon, H.Soltau, A.Mohamed, G.E.Dahl, and B.Ramabhadran, "Deep Convolutional Neural Networks for Large-Scale Speech Tasks, Neural Networks", Vol. 64, pp. 39-48, 2015.

[45] S.K.Saksamudre, P.P.Shrishrimal, and R.R.Deshmukh, "A Review on Different Approaches for Speech Recognition System", International Journal of Computer Applications (0975 – 8887), Vol. 115, No. 22, April 2015.

[46] F.Santos and T.Freitas, "CORP-ORAL: Spontaneous Speech Corpus for European Portuguese", in Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, Marrakech, Morocco, May 26 – June 1, 2008.

[47] K.P.Scannell, "The crubadan project: Corpus Building for Under-Resourced Languages", 2007.

[48] T.Sercu, C.Puhrsch, B.Kingsbury, and Y.LeCun, "Very Deep Multilingual Convolutional Neural Networks for LVCSR", 2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, pp. 4955-4959, March 20-25, 2016.

[49] T.Soe, S.S.Maung, N.N.Oo, "Combination of Multiple Acoustic Models with Multi-scale Features for Myanmar Speech Recognition", International Journal of Computer (IJC), Vol. 28, No. 01, pp. 112-121, February 2018.

[50] W.Soe and Y.Thein, "Syllable-based Myanmar Language Model for Speech Recognition", in Proceedings of IEEE/ACIS 14th International Conference on Computer and Information Science(ICIS)-2015, pp. 291-296, 2015.

[51] A.Stolcke, “Srilm - An Extensible Language Modeling Toolkit”, pp. 901–904, 2002.

[52] M.N.Stuttle, “A Gaussian Mixture Model Spectral Representation for Speech Recognition”, Dissertation submitted to the University of Cambridge for the degree of Doctor of Philosophy, July 2003.

[53] Y.K.Thu, W.P.Pa, A.Finch, J.Ni, E.Sumita, and C.Hori, “The Application of Phrase Based Statistical Machine Translation Techniques to Myanmar Grapheme to Phoneme Conversion, Computational Linguistics”, 14th International Conference of the Pacific Association for Computational Linguistics, PACLING 2015, Bali, Indonesia, pp. 238-250, May 19-21, 2015.

[54] Dr.T.Tun, “Acoustic Phonetics and the Phonology of the Myanmar Language”, First Edition, The Emperor Press, Yangon, Myanmar, 2007.

[55] Dr.T.Tun, “The Subtleties of the Myanmar Language (Grammar, Segments and Prosody in the Sound System of the Language and Spelling)”, School of Human Communication Sciences, La Trobe University, Melbourne, Australia, 2012.

[56] N.S.Uchat, “Hidden Markov Model and Speech Recognition”, Department of Computer Science and Engineering Indian Institute of Technology, Bombay Mumbai.

[57] A.Waibel, T.Hanazawa, G.Hinton, K.Shikano, K.J.Lang, “Phoneme Recognition using Time-Delay Neural Networks”, in IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 37, No. 3, pp. 328–339, March 1989.

[58] S.Watanabe, T.Hori, S.Kim, J.R.Hershey and T.Hayashi, “Hybrid CTC/ Attention Architecture for End-to-End Speech Recognition”, in IEEE Journal of Selected Topics in Signal Processing, Vol. 11, No. 8, pp. 1240-1253, December 2017.

[59] J.Wu, and C.Chan, “Isolated Word Recognition by Neural Network Models with Cross-Correlation Coefficients for Speech Dynamics”, IEEE Transactions on Pattern Analysis & Machine Intelligence, Vol. 15, No. 11, pp.

1174–1185, November 1993.

[60] Yu, Dong and Deng, Li, “Automatic Speech Recognition: A Deep Learning Approach”, ISBN-1447157788, 9781447157786, Springer Publishing Company, Incorporated, 2014.

[61] D.Yu, L.Deng, G.E.Dahl, “Roles of Pre-Training and Fine-Tuning in Context-Dependent DBN-HMMs for Real-World Speech Recognition”, NIPS Workshop on Deep Learning and Unsupervised Feature Learning, December 1, 2010.

[62] N.Zeghidour, N.Usunier, G.Synnaeve, R.Collobert, and E.Dupoux, “End-to-End Speech Recognition from the Raw Waveform”, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, September 2-6, 2018.

[63] P.Zelasko, B.Ziólko, T.Jadczyk, and D.Skurzok, “AGH Corpus of Polish Speech”, Language Resources and Evaluation, Vol. 50, No. 3, pp. 585-601, 2016.

[64] Y.Zhang, W.Chan, and N.Jaitly, “Very Deep Convolutional Networks for End-to-End Speech Recognition”, 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, pp. 4845-4849, March 5-9, 2017.

[65] S.Zhou, S.Xu, and B.Xu, “Multilingual End-to-End Speech Recognition with A Single Transformer on Low-Resource Languages”, CoRR, Vol. abs/1806.05059, August 13, 2018.

[66] M.Ziolko, J.Galka, B.Ziolko, T.Jadczyk, D.Skurzok, and M.Masior, “Automatic Speech Recognition System Dedicated for Polish”, in INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association, Florence, Italy, pp. 3315-3316, August 27-31, 2011.

APPENDICES

Appendix A: Developing of Myanmar ASR

Kaldi automatic speech recognition toolkit is used to implement the Myanmar ASR. In this appendix, we will present data preparation steps, feature extraction and language model preparation and building acoustic models.

1. Data Preparation

Data preparation is a necessary step to set up ASR system with our own corpus. This section give details how to prepare the data for training, development and testing sets for Myanmar language.

The stage has two things to prepare. They are "the data" directory and “the language" directory. The "data" contains information regarding the specific of the audio files, and the "lang" part contains language data specific, such as the lexicon, etc.

1.1 Data Preparation (“data” directory)

In the “data” part, for training, development and test data, the transcription of each utterance (text), audio files related to utterances (wav.scp), speaker and utterance mappings (utt2spk) are necessary to prepare manually.

(a) text

The file "text" includes the transcriptions of each utterance.

Pattern: <uterranceID> <text_transcription>

s5# **head -2 data/train/text**

ucsy-eleven-chosetpaing_2104 ခန္ဓာကိုယ် မှာ ရှဉ့် များ လူး ကာ အိမ်မွေး တိရစ္ဆာန် တွေ သယ်ဆောင် ပြီး သရုပ်ပြ မှု တွေ နဲ့ မြေပြိုကန် အရေး အတွက် လမ်းလျှောက် ဆန္ဒပြ ပွဲ ကလေး မြို့ မှာ ပြုလုပ် ခဲ့ တယ် ဆိုတဲ့ သတင်း

ucsy-eleven-chosetpaing_2105 ရွာ သုံး ရေ စမ်းချောင်း လယ်ကွင်း တွေ မှာ ကျောက်မှုန့် သဲမှုန့် တို့ ရောက်ရှိ ခဲ့ ပြီး လယ်ယာမြေ ပျက်စီး ခဲ့ တဲ့ အတွက် ကြောင့် ပေါင် မြို့နယ် အုတ်တား ကျေးရွာ သား ငါး ရာ ဝန်းကျင် ခန၀်ဝ် က ကျောက်မိုင်း ကုမ္ပဏီ ရုပ်တန့် ပေး ရန် ဆန္ဒ ထုတ်ဖော် တယ် ဆိုတဲ့ သတင်း

(b) wav.scp

This file links every utterance with an audio file related to this utterance.

Pattern: <uterranceID> <full_path_to_audio_file>

```
s5# head -3 data/train/wav.scp
ucsy-eleven-yunwintwintkyaw_4072 waves_data/4072.wav
ucsy-eleven-yunwintwintkyaw_4073 waves_data/4073.wav
ucsy-eleven-yunwintwintkyaw_4074 waves_data/4074.wav
```

(c) **utt2spk**

This file shows which utterance belongs to a particular speaker.

Pattern:<uterranceID><speakerID>

```
s5# head -3 data/train/utt2spk
ucsy-eleven-yunwintwintkyaw_4072 ucsy-eleven-yunwintwintkyaw
ucsy-eleven-yunwintwintkyaw_4073 ucsy-eleven-yunwintwintkyaw
ucsy-eleven-yunwintwintkyaw_4074 ucsy-eleven-yunwintwintkyaw
```

The "spk2utt" file can be produced by a command like the following using the “utt2spk” file.

Pattern: <speakerID> <uterranceID>

Command:

utils/utt2spk_to_spk2utt.pl data/train/utt2spk > data/train/spk2utt

```
s5# head -1 data/train/spk2utt
ucsy-eleven-yunwintwintkyaw  ucsy-eleven-yunwintwintkyaw_4072 ucsy-eleven-yunwintwintkyaw_4073 ucsy-eleven-yunwintwintkyaw_4074
```

1.2 Data Preparation (“lang” directory)

In this part, language data specific, such as the lexicon, phones, etc., are needed to create in kaldi acceptable format. Firstly, it needs to prepare dictionary directory “data/local/dict/” for input. The directory has the following contents. The dictionary directory is prepared by the script ‘*./local/prepare_dict_myanmar.sh*’.

```
s5# ls data/local/dict
extra_questions.txt  lexicon.txt  nonsilence_phones.txt  optional_silence.txt
silence_phones.txt
```

The output directory “data/lang/” contains the following files:

```
s5# ls data/lang
L.fst  L_disambig.fst  oov.int oov.txt  phones phones.txt  topo  words.txt
```

There is a directory "data/lang_test" that contains the same information but also a file “G.fst” that is a Finite State Transducer form of the language model:

```
s5# ls data/lang_test
G.fst  L.fst  L_disambig.fst  oov.int  oov.txt phones phones.txt  topo  words.txt
```

The directory “data/lang/phones/” consists of the following files:

context_indep.txt context_indep.int context_indep.csl silence.txt nonsilence.txt disambig.txt optional_silence.txt sets.txt extra_questions.txt word_boundary.txt roots.txt

2. Preparing the Language Model or Grammar in Kaldi

The file G.fst is the language model in a finite state transducer form. This arpa2fst command converts the ARPA-format language model into a weighted finite state transducer. In this work, SRILM is used to build ARPA-format language model. The following statement is applied to change the ARPA-format language models into an OpenFst format.

```
cat input/myanmar.arpa | arpa2fst - | fstprint | utils/eps2disambig.pl | utils/s2eps.pl | fstcompile --isymbols=$test/words.txt --osymbols=$test/words.txt --keep_isymbols=false --keep_osymbols=false | fstrmepsilon | fstarcsort --sort_type=ilabel > $test/G.fst
```

3. Feature Extraction

After data preparation is finished, feature extraction steps are performed. In this task, we used MFCC and FilterBank feature extraction methods. The example the output of feature extraction in Kaldi format is as follows.

Pattern: <utterance-id> <extended-filename-of-features>

```
s5# head -3 data/train/feats.scp

ucsy-eleven-yunwintwintkyaw_4072 /root/kaldi- master/egs/Myanmar42Hr_syllable/
s5/mfcc/raw_mfcc_pitch_train.2.ark:1391263

ucsy-eleven-yunwintwintkyaw_4073 /root/kaldi-master/egs/Myanmar42Hr_syllable/
s5/mfcc/raw_mfcc_pitch_train.2.ark:1409189

ucsy-eleven-yunwintwintkyaw_4074 /root/kaldi-master/egs/Myanmar42Hr_syllable/
s5/mfcc/raw_mfcc_pitch_train.2.ark:1422971
```

This feats.scp file is made by the following command. This command is for MFCC feature extraction method. This will create feats.scp with corresponding archives in a folder called mfcc and written log files to exp/make_mfcc.

```
steps/make_mfcc_pitch.sh data/train exp/make_mfcc/ mfcc
```

After that, we do cepstral mean and variance normalization for each speaker on MFCC features. This scp file is indexed by speaker-id, not utterance-id. This file is made by the following shell script such as:

```
steps/compute_cmvn_stats.sh data/train exp/make_mfcc/ mfcc
```

4. Training Baseline HMM-GMM Acoustic Model

In this work, Hidden Markov Model-Gaussian Mixture Model (HMM-GMM) system is trained on top of MFCC features as the baseline. It includes the following steps.

4.1 Training Monophone Models

A monophone model is trained by using a script called '*steps/train_mono.sh'*. It needs a data directory and a language directory, and will store the model in the experiment directory. It does not contain any contextual information about the preceding or following phone.

```
steps/train_mono.sh --nj 4 data/train data/lang exp/mono
```

--nj 4 instructs Kaldi to split computation into four parallel jobs.

4.2 Aligning Audio with the Acoustic Models

The audio file is aligned to the reference transcript with the most current acoustic model to improve or refine the parameters of the model. Therefore, each training step will be followed by an alignment step where the audio and text can be realigned.

This can be aligned by using the following script.

```
steps/aign_si.sh --nj 4 data/train data/lang exp/mono exp/mono_ali
```

4.3 Train Triphone Models

Context-dependent triphones are created by applying monophone model and re-estimating using triphone transcriptions. To train a triphone system, the following command is used and two numbers, 2500 and 15000 are passed. These are respectively the number of leaves in the decision tree and the total number of Gaussians across all states in our model.

```
steps/train_deltas.sh 2500 15000 data/train_words \ data/lang_wsj exp/word/mono_ali
exp/tri1
```

Delta+delta-delta training computes delta and double-delta features, or dynamic coefficients, to supplement the MFCC features. Delta and delta-delta features are numerical estimates of the first and second order derivatives of the signal (features). Delta features are computed on the window of the original features; the delta-delta is then computed on the window of the delta-features.

And then, align the system by using the command:

```
steps/aign_si.sh --nj 4 data/train data/lang exp/tri1 exp/tri1_ali
```

Train a system on top of LDA+MLLT features, using the tri1_ali alignments:

```
steps/train_lda_mllt.sh \ --splice-opts "--left-context=3 --right-context=3" \ 2500
15000 data/train data/lang \ exp/tri1_ali exp/tri2
```

7 frames are spliced together (left and right-context=3 above) of the MFCC features. LDA-MLLT stands for Linear Discriminant Analysis - Maximum Likelihood Linear Transform. The LDA takes the feature vectors and builds HMM states. However, it has a reduced feature space for all data. The MLLT receives the reduced feature space from the LDA and derives a unique transformation for each speaker. Therefore, MLLT is a step towards speaker normalization because it minimizes differences among speakers.

Realign the triphone system again by using the command.

```
steps/aign_si.sh --nj 4 data/train data/lang exp/tri2 exp/tri2_ali
```

And then, speaker adaptive training (SAT) is done for speaker and noise normalization by adapting to each specific speaker with a particular data transform. This training is done by using the command:

```
steps/train_sat.sh \ --splice-opts "--left-context=3 --right-context=3" \ 2500
15000 data/train data/lang \ exp/tri2_ali exp/tri3
```

4.4 Decoding the Triphone System

The first command *(utils/mkgraph.sh*) combines the HMM structure in the trained model, any Context dependency, the Lexicon and the Grammar collectively termed HCLG - and creates a decoding graph in the form of a Finite State Transducer (FST). The second script generates lattices of word (phone) sequences for the data given the model.

```
utils/mkgraph.sh data/lang_test_tg exp/tri3 exp/tri3/graph
```

```
steps/decode.sh --nj 4 exp/tri3/graph data/test exp/tri3/decode_test
```

5. Training SGMM-based Acoustic Model

In this work, SGMM is built on top of LDA+MLLT+SAT features. Firstly, align the LDA+MLLT+SAT system to have the latest possible alignments for the SGMM training using the following command:

```
steps/align_si.sh --nj $nj --cmd "$train_cmd" data/train data/lang exp/tri3 exp/tri3_ali
```

In this work, Universal Background Model (UBM) is initialized by clustering the diagonal Gaussians that derived the HMM set to I = 400 Gaussians, and phonetic subspace S = 40 dimension. By using the UBM model, SGMM is trained.

The command used for UBM and SGMM training is as follows.

```
steps/train_ubm.sh --cmd "$train_cmd" 400 data/train data/lang exp/tri3_ali exp/ubm4a
|| exit 1;

steps/train_sgmm.sh --cmd "$train_cmd" 2500 30000 data/train data/lang exp/tri3_ali
exp/ubm4a/final.ubm exp/sgmm4a || exit 1;
```

6. Training Neural Network-based Acoustic Models

The neural network acoustic models are trained using a frame-level cross-entropy loss. The latest possible alignments from LDA+MLLT+SAT system are used for supervised training of neural network.

6.1 Training Deep Neural Network (DNN)

For training the deep neural network, Filter Bank (FBank) feature extraction technique is used and it is extracted using the following command.

```
steps/make_fbank_pitch.sh data-fbank/train exp/make_fbank/ fbank
```

And then, deep neural network is trained using the following command.

```
steps/nnet/train.sh  --hid-layers 2 --learn-rate 0.008 data-fbank/train data-fbank/dev
data/lang exp/tri3_ali exp/tri3_ali_dt exp/tri4_dnn
```

In this script, target labels that are generated using the alignments are got by running the command '*ali-to-pdf'*. The counts of the PDFs corresponding to the phones in the alignments are produced using the command '*ali-to-phones'*. The command "*nnet-forward*" is run to do a forward pass that will be used to provide the class probabilities for decoding. The network is trained by using stochastic gradient descent (SGD) to minimize the cross-entropy between the labels and network output. The SGD uses minibatches of 256 frames, and an exponentially decaying schedule that starts with an initial learning rate of 0.008. And then, it halves the rate when the improvement in frame accuracy on a cross-validation set between two successive epochs falls below 0.5%. The optimization stops when the frame accuracy increases by less than 0.1%. Cross-validation is done on a set of 4000 utterances that are held out from the training data.

6.2 Training Convolutional Neural Network

CNN with convolution along the frequency axis is applied. The input to the network is an 11 frame (5 frames on each side of the current frame) context window of the 40 dimensional features. The best CNN architecture is used with 128/1024 feature maps in first and second convolutional layers. The filter sizes of the

convolutional layers are 8 and 4. The pooling size is set to 3 with pool step 1. The fully connected network has 2 hidden layers and there are 300 units per hidden layers.

CNN and DNN acoustic models are trained by applying cross-entropy on the alignments from the GMM-HMM system. It can be trained by using the following command.

```
steps/nnet/train.sh --cmvn-opts "--norm-means=true --norm-vars=true" --delta-opts "--delta-order=2" --splice 5 --network-type cnn1d --cnn-proto-opts "--patch-dim1 8" --hid-layers 2 --learn-rate 0.008 $train $dev data/lang $ali $ali_dev $dir
```

6.3 Decoding and Scoring

When the neural network training is finished, decoding is performed using the trained model. Before decoding, the decoding graph has to be created. This can be made by using the following command.

```
utils/mkgraph.sh data/lang_test_tg exp/tri4_dnn exp/tri4_dnn/graph
```

It made a fully expanded decoding graph (HCLG) that contains all the language-model, pronunciation dictionary, context-dependency, and HMM structure in the model. The output is a Finite State Transducer that has word-ids on the output, and pdf-ids on the input (these are indexes that resolve to Gaussians Mixture Models).

Then, the system is decoded by applying the decoding graph with the script as follows.

```
steps/nnet/decode.sh --nj 4 --num-threads 3 --cmd "$decode_cmd" --acwt 0.10 --config conf/decode_dnn.config exp/tri4_dnn/graph data/test exp/tri4_dnn/decode_test
```

During decoding, the test set can be split up and each different process (Job), decodes different subset of utterances, into lattices. A lattice is a representation of the alternative word-sequences that are "sufficiently likely" for a particular utterance. The Lattices are output during the decoding <decode-dir> into a numbered gzipped file. Each contains a single binary file. Each of these archives contains many lattices - one for each utterance.

When the decoding has finished, score the directory by running:

```
local/score.sh data/test exp/tri4_dnn/graph exp/tri4_dnn/decode_test
```

Scoring is done by '*local /score.sh*' and this program takes the minimal and maximum language model weights. It outputs the wer_N files for each of this different weight.

The scoring program works by opening all the lattice files, and getting them to output a transcription of the best guess at the words in all of the utterances they contain. The best guess is done with '*lattice-best-path*' command and the language model weight is passed to it, as -lm-scale.

The WER is calculated using '*compute-wer*' command, which takes two transcription files – the best guess output in the previous step, and the correct label s. The program outputs the portion that match.

Appendix B: Some Examples Output of Word-based and Syllable-based ASR Models

Reference	Hypothesis Text of Word-based Model	Hypothesis Text of Syllable-based Model
ရေမွေး သို့မဟုတ် လိုးရှင်း ကို အကြိေပး ချင် တယ်	ရေမွေး သို့မဟုတ် **လို့ ရှင်း** ကို အကြိေပး ချင် တယ်	ရေ မွေး **သုံး** မ ဟုတ် **လုပ် ရှင်** ကို အ ကြိ ေပး ချင် တယ်
နှစ် ပတ် ပရိဂရမ် အတို တက် ချင် ပါတယ်	နှစ် ပတ် ပရိဂရမ် **အတူ ကို** တက် ချင် ပါတယ်	နှစ် ပတ် ပ ရိ ဂ ရမ် **အ တို့ ကို** တက် ချင် ပါ တယ်
ဆိုးရွား သော ရာသီဥတု ဖြစ် ပြီး စပါး စိုက်ပျိုးရေး မှာ လိုအပ် ပါတယ်	**ဆိုးဝါး သုံး** ရာသီဥတု ဖြစ် ပြီး စပါး စိုက်ပျိုးရေး မှာ လိုအပ် ပါတယ်	**အ စိုး ရ သုံး** ရာ သီ ဥ တု ဖြစ် ပြီး စ ပါး စိုက် ပျိုး ရေး မှာ လို အပ် ပါ တယ်
သူမ အရမ်း ကို ဒေါသ အိုး ဆူ ေ နေ တယ်	သူမ က အရမ်း ကို ဒေါသ **ဦး စု ေး** နေ တယ်	သူ မ က အ ရမ်း ကို **တော် သ ဦး စု** ေ နေ တယ်
ဒီ တစ်ခေါက် လိုင်း မ ပါ တဲ့ အနိုး ကြည့် အ ေး ကြည့် ပါ ဝါ နှစ် မျိုး ပါ တာ ကို လိုချင် ပါ သလား	ဒီ တစ်ခေါက် လိုင်း မ ပါတဲ့ အနိုး ကြည့် အ ေး ကြည့် ပါ ဝါ နှစ် မျိုး ပါ **လာ** ကို **ယူ** ချင် ပါ သလား	ဒီ တစ် ခေါက် **တိုင်း** မ ပါ တဲ့ အ နိုး ကြည့် အ ေး ကြည့် ပါ ဝါ နှစ် မျိုး ပါ တာ ကို **လုပ်** ချင် ပါ သ လား
ဇူလိုင် လ နှစ် ရက်နေ့ နေ့ခင်း နှစ် နာရီ မှာ ဒွန် နဲ့ တွေ့ ဖို့ ရက်ချိန်း လိုချင် ပါတယ်	ဇူလိုင် လ နှစ် ရက်နေ့ **နေ့လည်** နှစ် နာရီ မှာ **ဒေါ်နယ်** တွေ့ ဖို့ ရက်ချိန်း **လို့ ထင်** ပါတယ်	ဇူ လိုင် လ နှစ် ရက် နေ့ **နေ့ ကင်း** နှစ် နာ ရီ မှာ **တော် နည်း တွေ** ကို ရက် ချိန်း လို ချင် ပါ တယ်
ဟင်း က အေး နေ တယ် နွေးပေး ပါ	ဟင်း က အေး နေ တယ် **နွေပေး** ပါ	ဟင်း က အေး နေ တယ် **ရှ** ပေး ပါ
အထူးသဖြင့် စိုက် တဲ့ စိုက်ကွက် ရဲ့ မျက်နှာပြင် ဟာ ညီညာ ဖို့ လိုအပ် ပါတယ်	အထူးသဖြင့် စိုက် တဲ့ စိုက်ကွက် ရဲ့ မျက်နှာပြင် ဟာ **ညီမျှ** ဖို့ လိုအပ် ပါတယ်	အ ထူး သ ဖြင့် စိုက် တဲ့ စိုက် ကွက် ရဲ့ မျက် နှာ ပြင် ဟာ **နည်း ပြ** ဖို့ လို အပ် ပါ တယ်
ထန်း ကုလားထိုင် နဲ့ ဝါး ဦးထုပ် တွေ ကို အမေရိကား ကို စ တင်ပို့ တဲ့ သတင်း ဖြစ် ပါတယ်	**ဖမ်း** ကုလားထိုင် နဲ့ ဝါး ဦးထုပ် တွေ ကို **အမေရိက** စ တင်ပို့ တဲ့ သတင်း ဖြစ် ပါတယ်	**ဖမ်း က လက်** ထိုင် နဲ့ **ဝ** ဦး ထုပ် တွေ ကို အ မေ ရိ ကား ကို စ တင် ပို့ တဲ့ သ တင်း ဖြစ် ပါ တယ်
မနှစ် က နှစ်လည် က စ ပြီးတော့ တာဝန်ယူ စဉ် က စ လို့ မူးယစ်ဆေးဝါး တားဆီး နှိမ်နင်း ရေး ကို ပြတ်ပြတ်သားသား လုပ်ဆောင် ခဲ့ တာ တွေ့ ရပါတယ်	မ နှစ် က နှစ်လည် က စ ပြီးတော့ တာဝန်ယူ **ဆယ်** က စ လို့ မူးယစ်ဆေးဝါး တားဆီး နှိမ်နင်း ရေး ကို **လည်း တဲ့ သာသာ** လုပ်ဆောင် **နေ** တာ တွေ့ ရပါတယ်	မ နှစ် က နှစ် လည် က စ ပြီး တော့ တာ ဝန် ယူ **ချိန်** က စ လို့ **မှေ** ဆေး ဝါး တား ဆီး **လေး** ကို **ပြီး ခဲ့ တဲ့ သ ရုပ်** ဆောင် ခဲ့ တာ တွေ့ ရ ပါ တယ်

Appendix C: Word Error Rate (WER) and Syllable Error Rate (SER)

The word error rate (WER) or syllable error rate (SER) can be calculated by dividing the total number of insertion, substitution and deletion words or syllables in the hypothesis text by the total number of words or syllables in the reference text. The result is multiplied by 100.

The example reference and hypothesis sentences of word-based ASR model are expressed as follows. The reference text consists of 33 words. In the hypothesis text, the total count of insertion, substitution and deletion words is 6. So, the WER of the sentence obtains 18.18% and the error words in the hypothesis text are highlighted in bold.

Reference Text of Word-based ASR Model:

တောင်သူ တွေ အတွက် ကုန်ကျစရိတ် သက်သက်သာသာ နဲ့ အထွက်နှုန်း အများဆုံး ရ နိုင် တဲ့ နည်းစနစ် တွေ ထဲ မှာ အကျိုး အာ အိုင် စနစ် ဖြစ် တဲ့ ပျိုးသန့်နု နဲ့ စိုက်ပျိုး ခြင်း စနစ် ဟာ လည်း တစ် ခု အပါအဝင် ဖြစ် ပါတယ်

Hypothesis Text of Word-based ASR Model:

တောင်သူ တွေ အတွက် ကုန်ကျစရိတ် **သက်သာ** နဲ့ အထွက်နှုန်း အများဆုံး **အရ** နိုင် တဲ့ နည်းစနစ် တွေ ထဲ မှာ အကျိုး အာ **ရေး** စနစ် ဖြစ် တဲ့ **ပြော သံ တို့** နဲ့ စိုက်ပျိုး ခြင်း စနစ် ဟာ လည်း တစ် ခု အပါအဝင် ဖြစ် ပါတယ်

The example reference and hypothesis sentences of syllable-based ASR model are described as follows. The total number of syllables in the reference texts is 56. There is 1 deletion syllable ‘သာ’ in the hypothesis text. Therefore, the total insertion, substitution and deletion syllables in the hypothesis text are 7 and so, 12.50% SER is attained.

Reference Text of Syllable-based ASR Model:

တောင် သူ တွေ အ တွက် ကုန် ကျ စ ရိတ် သက် သက် သာ သာ နဲ့ အ ထွက် နှုန်း အ များ ဆုံး ရ နိုင် တဲ့ နည်း စ နစ် တွေ ထဲ မှာ အကျိုး အာ အိုင် စ နစ် ဖြစ် တဲ့ ပျိုး သန့် နု နဲ့ စိုက် ပျိုး ခြင်း စ နစ် ဟာ လည်း တစ် ခု အ ပါ အ ဝင် ဖြစ် ပါ တယ်

Hypothesis Text of Syllable-based ASR Model:

တောင် သူ တွေ အ တွက် ကုန် **ကြား စေ တ** သက် သက် သာ နဲ့ အ ထွက် နှုန်း အ များ ဆုံး ရ နိုင် တဲ့ နည်း စ နစ် တွေ ထဲ မှာ အကျိုး **ဆာ ရေး** စ နစ် ဖြစ် တဲ့ ပျိုး သန့် **တို့** နဲ့ စိုက် ပျိုး ခြင်း စ နစ် ဟာ လည်း တစ် ခု အ ပါ အ ဝင် ဖြစ် ပါ တယ်