Building HMM-SGMM Continuous Automatic Speech Recognition on Myanmar Web News

[†]Aye Nyein Mon, [†] Win Pa Pa, [Λλ] Ye Kyaw Thu

[†]*Natural Language Processing Lab., University of Computer Studies Yangon, Myanmar*

[Λ]*Artificial Intelligence Lab.,Okayama Prefectural University, Okayama, Japan*

[λ]*Language and Speech Science Research Lab., Waseda University, Tokyo, Japan*

{ayenyeinmon, winpapa}@ucsy.edu.mm, ye@c.oka-pu.ac.jp

Abstract

Myanmar language is a tonal and analytic language. It can be considered as an under-resourced language because of its linguistic resource availability. Therefore, speech data collection is a very challenging task in building Myanmar automatic speech recognition. Today a lot of speech data are freely available on the Internet and we can collect it easily. Therefore, in this system, we take the advantages of Internet and we use daily news from the Web in building our speech corpus. In this paper, we will present about the task of data collection, the effect of Automatic Speech Recognition (ASR) performance according to amount of training data, language model size and error analysis of the experimental result. The experiments will be developed using Hidden Markov Model (HMM) with Gaussian Mixture Model (GMM) and Subspace Gaussian Mixture Model (SGMM). As a result, using our developed 5 hours training data, this system achieves word error rate (WER) of 7.6% on close test data and 31.9% on open test data with HMM-SGMM.

Keywords: *Automatic Speech Recognition (ASR), speech corpus developing, News Domain, HMM-GMM, HMM-SGMM, Myanmar language*

1. Introduction

Speech recognition is the process by which a computer maps an acoustic speech signal to text. ASR is applied in many application areas. One major application area is in human-computer interaction. While many tasks are better solved with visual or pointing interfaces, speech has the potential to be a better interface than the keyboard for tasks where full natural language communication is useful, or for which keyboards are not appropriate. This includes hands-busy or eyes-busy applications, such as where the user has objects to manipulate or equipment to control. Another important application area is telephony, where speech recognition is already used for example in spoken dialogue systems for entering digits. Finally, ASR is applied to dictation, that is, transcription of extended monologue by a single specific speaker.

There are many variations in speech recognition task. One dimension of variation is vocabulary size. Speech recognition is easier if the vocabulary size is small. On the other hand, tasks with large vocabularies, like transcribing broadcast news are much harder. A second dimension of variation is how fluent, natural, or conversational the speech is. Isolated word recognition, in which each word is surrounded by some sort of pause, is much easier than recognizing continuous speech, in which words run into each other and have to be segmented. A third dimension of variation is channel and noise. Noise of any kind makes recognition harder. A final dimension of variation is accent or speaker-class characteristics. Speech is easier to recognize if the speaker is speaking a standard dialect, or in general one that matches the data the system trained on [1].

Myanmar language is one of the under-resourced languages and there are no pre-created speech corpora. Therefore, in building the speech corpora, if the speech is recorded by ourself, the professional recording devices are very expensive and it is very time-consuming. Therefore, in this system, we collect the recorded speech data from the Web for the news domain. We will propose speaker independent large-vocabulary continuous speech recognition. This paper is organized as follow. In Section 2, about Myanmar ASR will be reviewed. In Section 3, collection of online resources will be presented. The architecture of the ASR system will be showed in Section 4. In Section 5, experimental setup will be presented. In Section 6, evaluation of the experimental results will be discussed. Finally, in Section 7, we present our conclusions and indicate promising avenues for future research on Myanmar ASR.

2. Myanmar ASR

There are some Myanmar ASR recently found in publications.

Wunna Soe [11] presented syllable-based speech recognition system for Myanmar. In this system, language model was built by using syllable segmentation and syllable-based n-gram method. Acoustic model was built by using GMM-HMM. The domain area is for news domain and the speech is recorded by using recording software such as wave surfer.

Myanmar language speech recognition with hybrid artificial neural network and hidden markov model was demonstrated by Thin Thin Nwe [5]. This system used syllable-based segmentation. Mel Frequency Cepstral Cofficient (MFCC), Linear Predictive Cepstral Coding (LPCC) and Perceptual Linear Prediction (PLP) were used in feature extraction techniques. To recognize the words, hybrid ANN-HMM was used.

Hay Mar Soe Naing [4] presented a Myanmar large vocabulary continuous speech recognition system for Travel domain. In this system, deep neural network (DNN) was used for acoustic modeling. Tonal features were added to the acoustic model. Sequence discriminative criteria such as cross-entropy (CE) and state-level minimum Bayes risk (sMBR) were used for DNN training.

Myanmar continuous speech recognition based on Dynamic Time Wraping (DTW) and HMM was expressed by Ingyin Khaing [2]. In this system, combinations of LPC, MFCC and Gammatone Cepstral Coefficients (GTCC) techniques were used in feature extraction. Moreover, DTW was used in the feature clustering in order to solve the lack of discrimination in the Markov model. HMM was used for recognition process.

3. Collecting Online Resources

Speech corpora creation is a mandatory task for training and testing any automatic speech recognition system. It is also the first step in building ASR system. In order to develop ASR systems, many hours of speech data are needed for training and testing data. Moreover, the performance of the system is also depended on the speech data. Speech corpora have been developed for many languages. For example, speech resources for English, Chinese and Japanese are well known and widely available. It has abundant of collected speech data. For under-resourced languages including Myanmar language, it has to build speech data sets from the scratch since most of them do not have pre-created speech corpora.

3.1. Building a Speech Corpus

A Speech corpus can be developed mainly in two ways. One way is to collect existing speech data (speech that is already been recorded) and manually transcribe them into text. The second way is to design the text corpus first and record the speech by reading the collected text. We used the first approach to build our speech corpus for news domain because lots of recorded speech data for news domain can be found on Internet and used them freely. We collected the daily video news for our building speech corpus. We took them from Eleven news, MRTV, InfoNews and 7daysTV. Our speech corpus includes various types of news. They are about political, health, speech, crime, football, weather and local news.

The original format of the downloaded speech files is .FLV and .MP4 and we converted them into .WAV format for building a speech corpus. So, we convert these file to .WAV format and therefore, our speech corpus include wave files (.WAV). The collected speech wave files are set to single channel (mono) type and 16 KHz is used for sampling rate. One news was cut into several segmented sentences and thus, generally, one segmented file lasts at a range between 2 to 25 seconds.

3.2. Building a Text Corpus

Although there is lots of recorded speech on Internet, it is almost never transcribed into text. Therefore, we manually transcribe them into text as transcription of the speech. Moreover, Myanmar language needs to segment the text because it usually writes in no space between words. Therefore, the transcribed texts are segmented into words using [8]. In addition, we also manually check the segmented texts again to get more accurate segmentation. Finally, we also check manually the spelling of the words in the segmented sentences. Generally, one utterance contains 33 words and 54 syllables in average. We use Myanmar 3 Unicode font in building the text corpus.

4. ASR Architecture

There are three main stages of the 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 contains information besides the linguistic message and has a high dimensionality. These characteristics of the raw speech signal would be unfeasible for the classification and result in high WER [3], [6]. Commonly used feature extraction techniques are Mel-Frequency Cepstrum Coefficients (MFCC), Linear Predictive Coding (LPC), Linear Prediction Cepstral Coefficients (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 given linguistic units (words, phones, subparts of phones). Final stage of ASR, the decoding is the process to calculate which sequence of words is most likely to match to the input acoustic signal that represented by the feature vectors. In other words, it is searching 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.

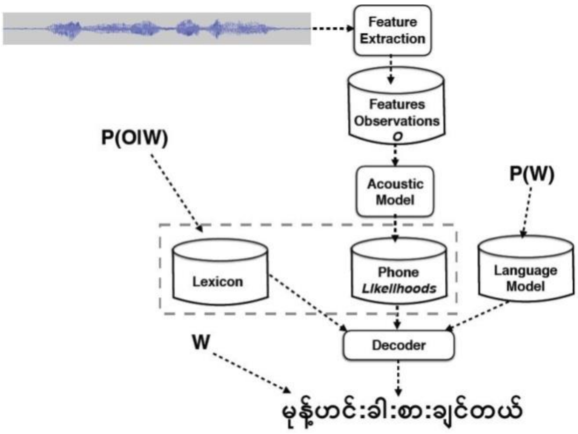


Figure1. The Automatic Speech Recognition System Overview

The most probable sentence W given some observation sequence O can be computed by taking the product of two probabilities for each sentence and choosing the sentence for which this product is greatest.

$$\hat{W} = \underset{W\epsilon L}{argmax} P(O|W)P(W) \qquad (1)$$

In the equation 1, the acoustic model can be computed the observation likelihood, P(O\W). The language model can get for computing the prior probability, P(W) [1].

4.1 Feature Extraction

Feature extraction is the one of the most important steps in speech recognition; each phoneme is identified by its unique characteristics features. In this system, we use MFCC.

The MFCC processor is less susceptible to speech signal variations and it mimics the behavior of human ear. The steps involved in MFCC feature extraction are pre-emphasis, framing, windowing, Fast Fourier Transform, Mel scale filter bank analysis, logarithmic compression and Discrete Cosine transform. The MFCC is based on short term analysis of speech; the speech waveform is divided into frames of f_N samples in time domain. In next step, Fast Fourier Transform (FFT) is applied to frames of signal which converts time domain samples into frequency domain samples. To implement Discrete Fourier Transform is defined on set of f_N samples. The resulting frequency based signal is also known as spectrum or periodogram.

The frequency spectrum of signal is very wide and it does not follow a linear scale, usually frequency is measured in Herz likewise subjective pitch is measured using mel scale. The set of triangular filters calculate sum of filtered spectral components and produces filter magnitude frequency as output. The collection of filters together called as Mel scale filter bank. As a result of mel scale filter bank analysis, mel frequency spectrum is obtained which is only linear below 1000 Hz and frequency spacing is nonlinear above 1000 Hz hence Logarithmic compression is used. Finally log mel spectrum is converted back to time domain using Discrete Cosine Transform (DCT), which results in MFCC [7].

4.2 HMM-GMM Acoustic Model

The acoustic model is used to model the statistics of speech features for each speech unit of the language such as a phone or a word. Hidden Markov Model (HMM) is the de facto standard used in the state-of-the-art acoustic models. It is a very powerful statistical method to model the observed data in a discrete-time series. An HMM is a structure formed by a group of states connected by transitions. Each transition is specified by its transition probability. The word hidden in HMM is used to indicate the fact that the state sequence generating the output symbols is hidden. In speech recognition, state transitions are usually constrained to be from left to right or self-repetition called the left-to-right model.

Each state of HMMs is usually represented by a Gaussian Mixture Model (GMM) to model the distribution of feature vectors for the given state. Figure 2 shows the states of HMM and the probability of each state can be computed by Gaussian Mixtures model.

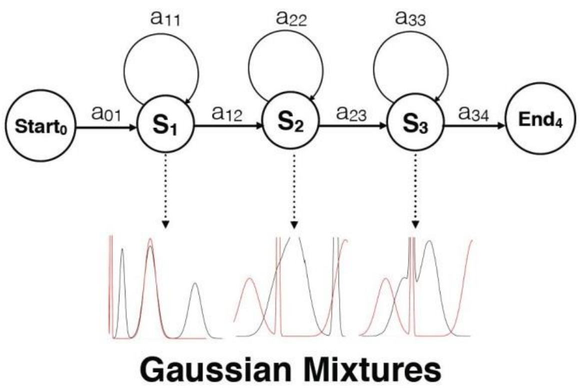


Figure2. How HMM-GMM Model a process

A GMM is a weighted sum of M component Gaussian densities and is described by Equation 2 [10].

$$P(x|\lambda) = \sum_{i=1}^{M} w_i g(x\Sigma_i) \qquad (2)$$

where P (x|λ) is the likelihood of a D-dimensional continuous-valued feature vector x, given the model parameters $\lambda = \{w_i , \mu_i , \Sigma_i \}$, where w_i is the mixture weight which satisfies the constraint ΣM $w_i = 1$, μ_i is the mean vector, and Σ_i is the covariance matrix i of the ith Gaussian function $g(x|\mu_i , \Sigma_i)$ which is defined by,

$$g(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\right\} \qquad (3)$$

SGMM Acoustic Model

SGMM has been proposed by exploiting the idea of that the mean of Gaussian components lie in a subspace. In this method the state dependent parameters (mean, covariance and mixture weights) are not estimated independently. Instead, we train a globally shared low dimensional subspace S from which the GMM models are trained. This captures the correlations between tied-states. M_i is the mean subspace from which the mean of the GMM μ_{ji} can be obtained using the state specific vector v_j. The mixture weights w_{ji} can be obtained from the weight vector w_i using v_j. Each GMM has the same number of mixture components I for all states. The parameters M_i, Σ_i, w_i are shared across all the states ($\Sigma_{ij}= \Sigma_i$) [12].

4. 4 Language Model

The language model describes what is likely to be spoken in a particular context. There are two types of language models: that are used in speech recognition systems - grammars and statistical language models. The grammar-type of language model describe very simple types of languages for command and control, and they are usually written by hand or generated automatically with plain code. The statistical language model uses stochastic approach called n-gram language model. An N-gram is an N-token sequence of words: a b2-gram (bigram) is a two-word sequence; a 3-gram (trigram) is a three-word sequence.

4.3 Decoding

Decoding is the process to calculate which sequence of words is most likely to match to the acoustic signal represented by the feature vectors. Weighted Finite State Transducer (WFST)-based decoder is used in this system. It Integrate different models into a single model via composition operations (lexicon, grammar, phonetics).

5. Experimental Setup

5.1 Data Preparation

In this experiment, the testing data set size is about 3 minutes 29 seconds. There are 3 females and 1 male in the test set. The development data set size is 3 Minutes 27 seconds. It includes 2 females and 2 males in the development data set. The total hour of training set is about 5 hours and 6 minutes. It includes 58 females and 14 males. As a total, there are 72 speakers in the training set. There are

1632 sentences in the training set. Test set is composed of 19 sentences. Development set has 22 sentences.

5.2 GMM and SGMM Acoustic Model

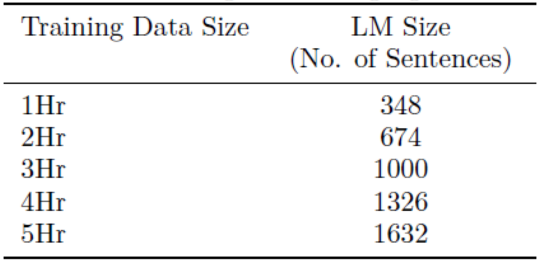
We use the open-source Kaldi toolkit [9]. Adopting the Kaldi's standard scripts, we used MFCC+Δ+ΔΔ features with standard cepstral mean and variance normalization (CMVN) to train the acoustic model. Then 9 frames of MFCCs are spliced together and projected down to 40 dimensions with linear discriminant analysis (LDA). A maximum likelihood linear transform (MLLT) is estimated on the LDA features and generates the LDA+MLLT model. Then, speaker adaptive training is performed on the top of LDA+MLLT model. Our GMM model has 2050 context dependent (CD) triphone states and an average of 14 Gaussian components per state.

In the SGMM experiment, we initialized Universal Background Model (UBM) by clustering the diagonal Gaussians that derived the HMM set to I=400 Gaussians, and phonetic subspace S=40 dimension for 5 hour training data case. LDA features with 40 dimensions were used for training the SGMM.

5.3 Language Model

Our language model was trained using the SRI Language Modeling (SRILM) language modeling toolkit. SRILM is a toolkit for building and applying statistical language models (LMs), primarily for use in speech recognition, statistical tagging and segmentation, and machine translation [13].

Table 1: Training set and language model size



Training Data Size	LM Size (No. of Sentences)
1Hr	348
2Hr	674
3Hr	1000
4Hr	1326
5Hr	1632

We also explored the effect of the size of the language model on ASR performance. For the test set contains 19 sentences and 547 words.

For evaluation based on the amount of training data, the same language model was used that built from 5hr training set for all 1hr, 2hrs, 3hrs, 4hrs and 5hrs training set. Table 1 shows size of training data and language model that we used for experiments.

5.4 Evaluation

To evaluate the performance of ASR models, we used automatic evaluation of word error rate (WER). The WER is based on how much the word string returned by the recognizer differs from a correct or reference transcription.

$$WER = \frac{Insertions+Substitutions+Deletions}{TotalWords} \times 100 \quad (4)$$

The result of this computation will be the minimum number of word substitutions, word insertions, and word deletions necessary to map between the correct string and the string returned by the recognizer.

Moreover, we used SCLITE (score speech recognition system output) program from the NIST scoring toolkit SCTK version 2.4.10[1] to do error analysis based on WER.

In this system, we used perplexity for the language model evaluation. The perplexity of a language model on a test set is a function of the probability that the language model assigns to that test set. For a test set, $W = w_1 w_2 \ldots w_N$ (a sequence of words from a vocabulary set, W) and the probability of a symbol w_i is dependent upon the previous symbol $w_1, \ldots, w_{i-1}$. The perplexity W can be computed as the following equation:

$$pp(W) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i|w_1 .. w_{i-1})}} \quad (5)$$

6. Evaluation Results

In this section, we will present evaluation results based on the training data, language model and the number of Gaussians in GMM and SGMM approaches. Moreover, we will make some discussions on ASR error types.

6.1 Evaluation with Training Data Size

We used WER to evaluate the performance of ASR. The effect of varying the size of the training set on the error rate of the system depict in a chart with word error rate as a function of training set in Figure 3.

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

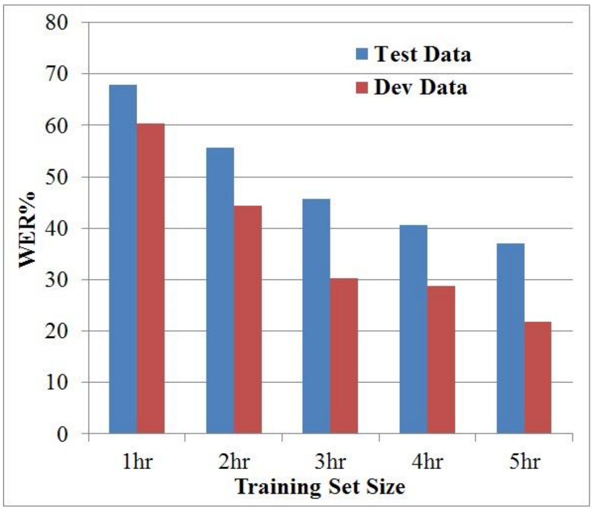


Figure 3: Chart diagram of Word Error Rate % for increasing amount of training data

With 1hr training data, the system achieves 60.5% of WER in close test data and 67.8% of WER in open test data. The error rate reduces gradually when the training data size is increased to 2 hrs, 3hrs, 4hrs and 5hrs respectively. Finally, with 5hrs training set, we achieve 21.1% of WER with close test data and 37.7% of WER with open test data. The results demonstrate that increasing training database size resulted in decreasing word error rates. Both close and open test data reduce error rates when the training set size increase continuously.

6.2 Evaluation with Language Model Size

Table 2 shows that 5hrs training data and language model that built from that data obtained lowest perplexity and Out of Vocabulary (OOV) rate.

Table 2: Language model perplexity and its OOV rate

LM Size (No. of Sentence)	Perplexity	OOV Rate
348	308.5	20.5%
674	363.8	14.1%
1000	275.8	9.5%
1326	155.4	6.9%
1632	129.1	6.0%

Figure 4 shows the effect of ASR performance on both the amount of training data, and the size of the language model estimated from such data.

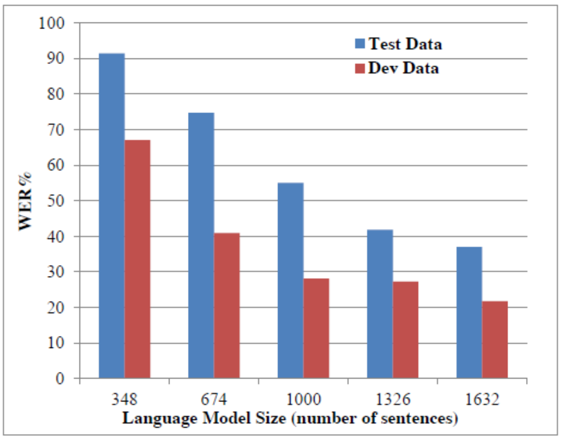


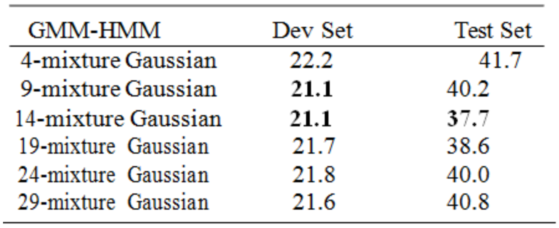
Figure 4: The ASR performance based on the different language model sizes

According to the Table 2 and Figure 4, the perplexity is strongly correlated with WER. In table 1, the largest LM size (1632 sentences) has the lowest perplexity value 129.1 and its WER (37.7) % with open test data and (21.1) % with close test data are also the lowest in Figure 4. Therefore, low perplexity means high recognition accuracy. It is clearly seen that the performance of the ASR system can be improved by enhancing the language model size and increasing the training set.

6.3 Evaluation with Number of Gaussian

According to table 3, increasing the number of components in the GMM doesn't provide better accuracy. With 5hrs training data set, the WER in open test set decreases gradually from 41.7% with 4-mixture Gaussian to 37.7% with the 14-mixture Gaussian. The WER in close test set also decreases slightly from 22.2% to 21.1% in 14-mixture Gaussian. However, above the number of 14-mixture Gaussian, the WER rates in open and close test data increases gradually starting from 19-mixture Gaussian. Therefore, with 14-mixture components at each state, the WER of the GMM-HMM reaches 21.1% in close test data and 37.7% in open test data.

Table 3: The WER % of the ASR performance with the number of Gaussian mixtures at each HMM state



GMM-HMM	Dev Set	Test Set
4-mixture Gaussian	22.2	41.7
9-mixture Gaussian	**21.1**	40.2
14-mixture Gaussian	**21.1**	**37.7**
19-mixture Gaussian	21.7	38.6
24-mixture Gaussian	21.8	40.0
29-mixture Gaussian	21.6	40.8

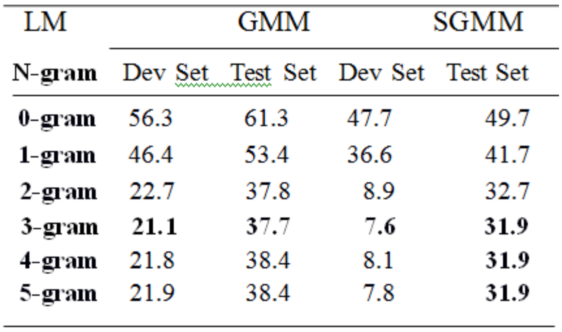
6.4 Evaluation with N-grams language model using GMM and SGMM

In the table 4, we compared the evaluation of ASR performance based on n-grams (1-gram, 2-gram, 3-grams, 4-grams and 5-grams) language model using GMM and SGMM. We used SRILM [11] language modeling toolkit to build the language model. According to the experiment in table 4, we found the accuracy given by 1-gram, 2-gram, 4-grams and 5-grams didn't improve the ASR performance. The 3-grams based language model is the best among 1 to 5-grams.

0-gram language model was used to investigate the performance of the acoustic model and the influence of language model. In comparison of 0-gram and the 3-grams language model, the WER of ASR using 0-gram (without language model) rapidly increased at both close and open test set than using 3-grams language model.

Therefore, it is clearly seen that the role of language model is significantly played to be improved ASR performance. We conclude that SGMM is significantly better than GMM according to the Table 4 when using limited amount of training data. SGMM showed relative improvement of 14% on close test data and 6% on open test data over GMM.

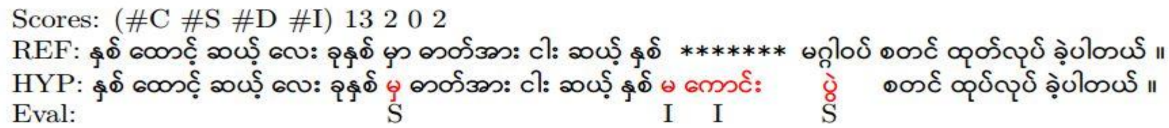
Table 4: The WER % of the ASR performance with N-grams language model using GMM and SGMM



LM	GMM		SGMM	
N-gram	Dev Set	Test Set	Dev Set	Test Set
0-gram	56.3	61.3	47.7	49.7
1-gram	46.4	53.4	36.6	41.7
2-gram	22.7	37.8	8.9	32.7
3-gram	**21.1**	**37.7**	**7.6**	**31.9**
4-gram	21.8	38.4	8.1	**31.9**
5-gram	21.9	38.4	7.8	**31.9**

6.5 Error Analysis

For error analysis, we used SCLITE ASR evaluation toolkit. An example evaluation of one of reference Myanmar sentence **“နှစ်ထောင့်ဆယ့်လေးခုနှစ်မှာ ဓာတ်အား ငါးဆယ့်နှစ်မဂ္ဂါဝပ် စတင်ထုတ်လုပ်ခဲ့ပါတယ်။”** (It started producing of 52 megawatts of electricity in 2014) is as follow:



```
Scores: (#C #S #D #I) 13 2 0 2
REF: နှစ် ထောင့် ဆယ့် လေး ခုနှစ် မှာ ဓာတ်အား ငါး ဆယ့် နှစ်  *******  မဂ္ဂါဝပ် စတင် ထုတ်လုပ် ခဲ့ပါတယ် ။
HYP: နှစ် ထောင့် ဆယ့် လေး ခုနှစ် မှ ဓာတ်အား ငါး ဆယ့် နှစ် မ ကောင်း      ပွဲ      စတင် ထုပ်လုပ် ခဲ့ပါတယ် ။
Eval:                          S                       I   I         S
```

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. In the sentence, the Myanmar word “**မဂ် ဂါ ဝပ်**” (**me' ga wa'**)[‘megawatt’ in English] was falsely recognized as “**မ ကောင်း ပွဲ**” (**ma-kaun: pwe:)[**‘bad festival’ in English**]**. There are 3 syllables and all vowels are wrong in that words. In the first words, the consonants are correct. In the second words, the consonants are not correct and the pronunciation commonly changes from “ဂ” [**g**] to “က” [**k**]. The whole syllable is not correct in the last words. The WER of that sentence is 26.67%.

According to the evaluation result, there are 4 significant types of errors that found in the experiment with 5hrs training set.

6.5.1 Similar pronunciation error

This system was falsely recognized the words that have similar pronunciations. For instance, Myanmar word **“ဩဂုတ်”(o: gou')** (‘August’ in English) was incorrectly recognized as **“အုတ်ဖုတ်” (ou' hpou')** (‘making brick’ in English). There are 10.48% of similar pronunciation errors.

6.5.2 Tone error

Tone mistakes were also found in this experiment. For example, the Myanmar word **“မှ” (mha.)** (‘from’ in English) gave incorrect result **“မှာ” (mha)** (‘in’ in English) in figure 5. Tone errors rate is 9.09% on the other types of errors.

6.5.3 Vowel error

Some vowels were misrecognized in the ASR output. As an example, the Myanmar word **“သံတွဲ” (than dwe:)** (the name of a city in Rakhine State) was falsely recognized as **“သံတွေ” (than dwei)** (‘iron nails’ in English). There are 4.89% of vowel errors.

6.5.4 Ambiguous error

Some ambiguous cases were not clearly defined in this system. As an example, the Myanmar word **“စက်မှုကျောင်း”** (**sé mhu. kyaun:**) was confused as **“ဆက်မှုကြောင်း”** (“**hsé mhu. kyaun:**). These errors are less than any other types of errors.

7. Conclusion

In this paper, we evaluated the ASR performance according to the size of the training data, language model and number of Gaussian in HMM with GMM and SGMM approaches. Moreover, we analyzed the errors that found in the experiments. As a result, when the amount of the training data and language model size are increased, the WER is also decreased and the accuracy is improved. In the future, the errors that found in the error analysis are to be covered. Moreover, neural network such as Convolutional Neural Network will be used in acoustic model building.

References

[1] 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.

[2] I. Khaing. Myanmar continuous speech recognition system based on dtw and hmm. In *International Journal of Innovations in Engineering and Technology (IJIET), Vol.2 Issue 1, February 2013, ISSN: 2319-1058*, pages 78–83. IJIET, 2015.

[3] J. Mariani, J. Gauvain, and L. Lamel. *Comments on “*towards increasing speech recognition error rates*” by h. bourlard, h. hermansky, and n. morgan. Speech Communication, 18(3):249 – 252, 1996.*

[4] H. M. S. Naing, A. M. Hlaing, W. P. Pa, X. Hu, Y. K.Thu, C. Hori, and H. Kawai. A myanmar large vocab- ulary continuous speech recognition system. In *Asia- Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2015, Hong Kong, December 16-19, 2015,* pages 320–327, 2015.

[5] T. T. Nwe and T. Myint. Myanmar language speech recognition with hybrid artificial neural network and hidden markov model. In *Proceedings of 2015 Interna- tional Conference on Future Computational Technologies (ICFCT’2015), Singapore, March 29-30, 2015*, pages 116–122. ICFCT, 2015.

[6] D. O’Shaughnessy. Invited paper: Automatic speech recognition: History, methods and challenges. Pattern Recognition, 41(10):2965 – 2979, 2008.

[7] I. P and R. V. Comparative analysis of feature extraction techniques for tamil speech recognition. In *ERCICA 2013, Proceedings of International Conference on Emerging Research in Computing, Information, Communication and Applications*, pages 755–761. Elsevier Publication, 2013.

[8] 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, August 26-28, 2015, Yangon, Myanmar- Volume II*, pages 447–456, 2015.

[9] 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*. IEEE Signal Processing Society, Dec. 2011. IEEE Catalog No.: CFP11SRW-USB.

[10] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *PROCEEDINGS OF THE IEEE*, pages 257–286, 1989.

[11] W. Soe and Y. Thein. Syllable-based speech recognition system for myanmar. In *International Journal of Computer Science Engineering and Information Tech- nology (IJCSEIT)*, pages 1–13. IJCSEIT, 2015.

[12] R. Sriranjani, S. Umesh, et al. Investigation of different acoustic modeling techniques for low resource indian language data. In *Communications (NCC), 2015 Twenty First National Conference on*, pages 1–5. IEEE, 2015.

[13] A. Stolcke. Srilm - an extensible language modeling toolkit. pages 901–904, 2002.