AUTOMATIC SPEECH RECOGNITION SYSTEM FOR BURMESE SENTENCES USING KALDI

Presented by

Hlaing Myat Nwe (PhD-IT-1)

Khant Khant Win Tint (PhD-IT-3)

Khaing Hsu Wai

28. 1. 2021

Outlines

- ABSTRACT
- AUTOMATIC SPEECH RECOGNITION (ASR)
- DATA PREPARATION
- KALDI
- EXPERIMENT
- EVALUATION
  - EVALUATION ON TRAINING DATA SIZE
  - EVALUATION WITH N-GRAM LANGUAGE MODEL
  - EVALUATION ON TRAINING WITH DIFFERENT MODEL SIZE
- CONCLUSION
- REFERENCE

1. Abstract

- Automatic speech recognition (ASR) is a process that converts human speech to a sequence of words which is spoken by human
- The accuracy remains one of the most important research challenges based on the vocabulary size, noise and the variety of language and speaker
- The major difficulty in the research process of Myanmar language ASR is the lack of Myanmar speech corpus
- In this project, we aimed to built an accurate ASR system for small amount of data-set using **Kaldi**

1. Abstract (Cont’d)

- For training the acoustic part of the model, **Hidden Markov Model** and **Gaussian Mixture Model** is used
- The performance of the system is evaluated in terms of **Word Error Rate (WER)**
- To improve the recognition accuracy, we made three types of experiment
  1) We applied the ***incremental training concept*** based on the different approaches of training and adaptation techniques
  2) We used different ***n-gram language models*** to investigate the accuracy
  3) We evaluated the performance of the model depending on the different ***model size***

2. Automatic Speech Recognition (ASR)

- Automatic speech recognition (ASR) is the process of converting an unknown speech waveform into the corresponding orthographic transcription

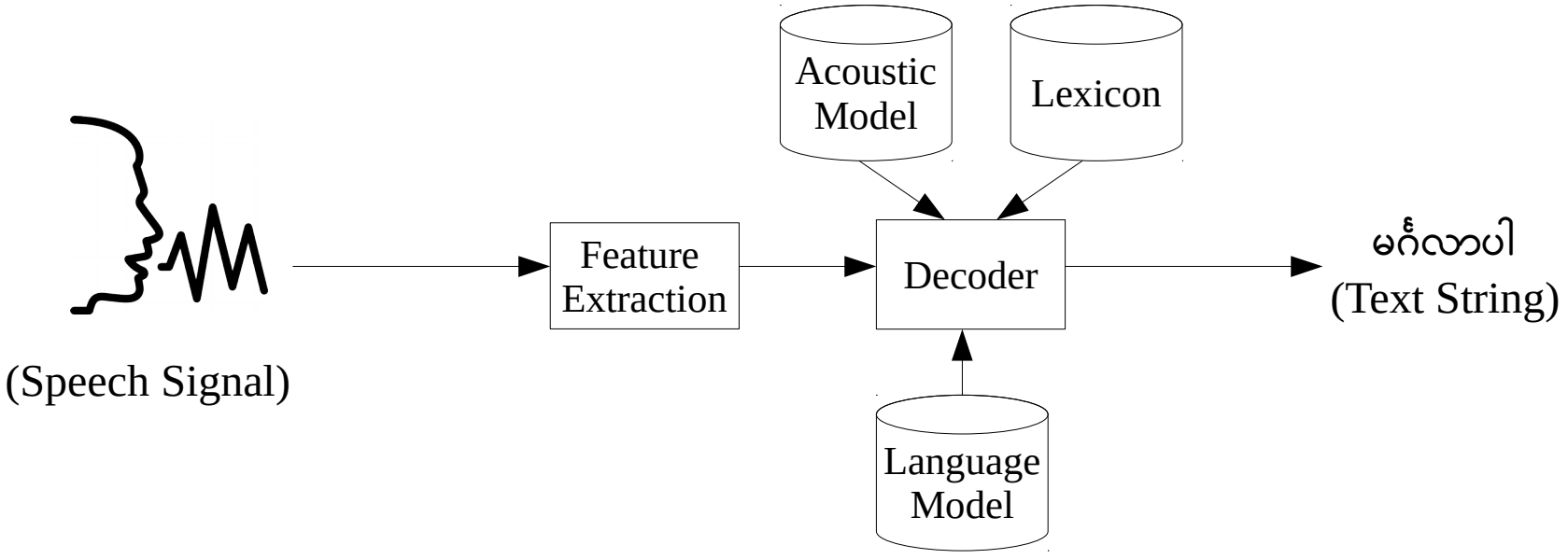

Figure 1. Automatic Speech Recognition System (ASR)

2.1. Feature Extraction

- Feature extraction is an essential first step in speech recognition applications
- The goal of the feature extraction is to extract sequences of acoustic observations that contain all the useful features and information for recognition operation
- Common signal processing techniques used in automatic speech recognition are based on Mel Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Prediction (PLP)
- In this project, we are going to make use of MFCC was used with the aim of reducing the spectral distortion by shrinking the beginning and end of each frame of the signal to zero

2.1. Feature Extraction (Cont’d)

- In addition to static features extracted from each frame of speech data, it is beneficial to use some transformations to improve the recognition
- There are transforms, projections and other feature operations:
  - Frame splicing and Delta feature computation
  - Linear Discriminant Analysis (LDA) transform
  - Heteroscedastic Linear Discriminant Analysis (HLDA)
  - Maximum Likelihood Linear Transform (MLLT) estimation
- In this project, we used and compared both Delta feature computation and LDA+MLLT

2.1. Feature Extraction (Cont’d)

- ***Delta feature computation (Δ+ΔΔ)***
  - Delta feature is the Fourier Transform of the time order of the phonetic frames order
  - For instance: If we have 13 MFCC coefficients, with the Δ+ΔΔ transformation we also get 13+13 delta coefficients, which would combine to give a feature vector of length 39 (13+13+13)
  - Then, the original vector is reduced to vector of 39 MFCC Δ+ΔΔ acoustic features.
- ***LDA+MLLT***
  - LDA: Is a linear transform that reduce dimensionality of our input features. The idea of LDA is to find a linear transformation of feature vectors from an n-dimensional space to vectors in an m-dimensional space (m<n) such that the class separability is maximum
  - MLLT: Estimates the parameters of a linear transform in order to maximize the likelihood of the training data given a diagonal-covariance Gaussian mixture models; the transformed features are better represented by the model than the original features

2.2. Acoustic Model

- The acoustic model (AM) is a very important component of the recognition process
- The main goal for the acoustic model is to enhance the speech recognition accuracy by specifying the modeling units and computing the likelihood of the acoustic features' components for the phonetic units that need to be recognized
- Acoustic models are trained by taking audio recordings of speech, and their text transcriptions, and creating statistical representations of the sounds that make up each word
- In this project, the Hidden Markov Model (HMM) with the Gaussian Mixture Model (GMM) are used to extract the acoustic features vectors

2.2. Acoustic Model (Cont'd)

Hidden Markov Model (HMM)

- HMM is a popular choice in speech recognition, being able to model this uncertainty between acoustic features and corresponding transcription.
- HMM is a stochastic finite state automaton that models the variation in the acoustic signal via a two-stage stochastic process
- The automaton is defined through a set of states with transitions connecting the states
- The probability $P(x^T_1 \mid w_1^N)$ is extended by an unobservable (hidden) variables representing the states as in the following equation (1)

$$P(x^T_1 \mid w_1^N) = \Sigma P(x^T_1, s^T_1 \mid w_1^N) \quad \ldots\ldots\ldots\ldots \quad \text{eq (1)}$$

2.2. Acoustic Model (Cont'd)

Gaussian Mixture Model (GMM)

- In this project, Gaussian Mixture Model (GMM) is used for estimating the output distribution $\{b_j()\}$
- The expression for the output observation becomes,

$$b_j(x) = \sum_{m=1}^{M} c_{jm} N(x; \mu_{(jm)}, \Sigma_{(jm)})$$

- where c_{jm} is the prior probability for component m of state s_j and $N(x; \mu_{(jm)}, \Sigma_{(jm)})$ is the Gaussian (or normal) disitribution with parameters μ (jm) & Σ (jm) corresponding to the mean and co-variance of state s_j respectively
- The prior probabilities satisfy the probability mass function constraints

$$\sum_{m=1}^{M} c_{jm} = 1, \; c_{jm} \geq 0.$$

- If M is set to equal one, a single Gaussian distribution is obtained. In the training of a GMM the aim is to update the mean $\mu_{(jm)}$ and co-variance $\Sigma_{(jm)}$.

2.3. Language Model

- The language model P(w_1^N)provides a prior probability for the word sequence w_1^N = w_1,..., w_N
- For large vocabulary speech recognition, n-gram language models have become widely accepted
- An n-gram language model is based on the assumption that a sequence of words follows an (n-1)-th order Markov process, that is, the probability of a word w_n is supposed to depend only on w_{n-1} predecessor words as follows:

$$P(w_{1^N}) = \prod_{n=1}^{N} P\left(w_n \middle| w_1^{n-1}\right) \qquad \text{…….. eq (2)}$$

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

2.4. Weighted Finite-state Transducer (WFST)

- 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 A FST is a finite automaton whose state transitions are labeled with both input and output symbols
- Therefore, a path through the transducer encodes a mapping from an input symbol sequence, or string, to an output string
- A weighted transducer puts weights on transitions in addition to the input and output symbols
- Weighted transducers are thus a natural choice to represent the probabilistic finite-state models prevalent in speech processing
- In this project, we used WFST for all the training and decoding algorithms

2.5. Lexicon

- Lexicon file that contains every word from dictionary with its phone transcription is vital for improving the ASR accuracy
- For our experiment, we built myG2P model using Ripple Down Rules (RDR) to create a lexicon file (https://github.com/ye-kyawthu/myG2P)
- Grapheme-to-Phoneme (G2P) conversion is about predicting the pronunciation of words given only the spelling.

Table 1. Example of Myanmar Lexicon

Myanmar Word	Phoneme
ကား	ka:
ကူ	ku
ကူညီ	ku nj i
ကော်ဖီ	ko hpi
ကို	kou
ကိုး	kou:
ကောင်မလေး	kaun ma. lei:

3. Data Preparation

- Data preparation is a necessary step to set up ASR system with our own data
- There are three parts to prepare: audio data, acoustic data and language data

1) **Audio Data**
   - We recorded the audio files with the students of NLP class in the E-learning studio room of University of Technology (Yatanarpon Cyber City) and changed the file format to wav
   - In our experiments we used 16KHz sampling frequency and 16 bit samples
   - Each of these audio files is named in a recognizable way and placed in the recognizable folder representing particular speaker

3. Data Preparation

- In this project, we prepared four types of dataset for the incremental training such as we used 6 speakers for first experiment, 10 speakers for second experiment, 16 speakers for third experiment and 20 speakers for fourth experiment

Table 2. Data used for Incremental Training

Experiment	No. of Speakers	No. of Sentences for Training	No. of Sentences for Testing
1st Experiment	6	2100	900
2nd Experiment	10	3500	1500
3rd Experiment	16	5600	2400
4th Experiment	20	7000	3000

3. Data Preparation (Cont’d)

2) <u>Acoustic Data</u>

- For this project, we collected 500 general sentences for each speaker
- Some text files are needed to create to communicate with audio data:

  1) ***text***: It contains the information about every utterance ID matched with its text transcription

  2) ***spk2gender***: It informs about speakers gender

  3) ***utt2spk***: It tells the ASR system which utterance belongs to particular speaker

  4) ***wav.scp***: It connects every utterance (sentence said by one person during particular recording session) with an audio file related to this utterance

3. Data Preparation

- Example Sentences that used in this project are as follows:
  - ကား ရဲ့ အရှိန် ကို လျှော့ ပါ
  - ကူ ကြ ပါ ဦး ရှင့်
  - ကူညီ ကြ ပါ ခင်ဗျာ
  - ကူညီ ကြ ပါ ဦး
  - ကော်ဖီ ထပ် ရ နိုင် မလား
  - ကော်ဖီ နောက် တစ်ခွက် ရ နိုင် ပါ မလား

3. Data Preparation (Cont’d)

3) <u>Language Data</u>

- Some text files that relate to language modelling files are created:

  1) ***lexicon.txt*:** This file contains every word from your dictionary with its phone transcription

  2) ***nonsilence_phones.txt***: This file lists nonsilence phones of our data. We extracted phones from lexicon.txt and collected unique phones in this file

  3) ***silence_phones***: This file lists silence phones

  4) ***optional_silence.txt***: This file lists optional_silence phones

4. Kaldi

- Kaldi is an open-source toolkit for speech recognition written in C++ and licensed under the Apache License v2.0
- Access to the library functionalities is provided through command-line tools written in C++, which are then called from a scripting language for building and running a speech recognizer
- The goal of Kaldi is to have modern and flexible code that is easy to understand, modify and extend

5. Experiment

- In each experiments, each speech signal was parameterized using 13 MFCC and the analysis windows size was 25ms with 10 ms overlap
- Initially, we use **3-gram** language model which is estimated from the training data transcription
- The decoding of the test utterances is performed with the same parameters as follows:
  - max-active=7000
  - beam=13.0
  - lattice-beam=6.0
  - acoustic scale= 0.083333
  - model size= #num-leaves #tot-gauss = 200 1100

5. Experiment (Cont’d)

- In our experiment, HMM-GMM system is trained on top of MFCC features
- Firstly, we trained a mono-phone system (mono) using the MFCC’s and Tri-phone model with Δ+ΔΔ features on the amount of different data
- Finally, we retrain the triphone acoustic model on Linear Discriminant Analysis - Maximum Likelihood Linear Transform (LDA+MLLT) features and made speaker adaptive training (SAT) for speaker and noise normalization by adapting to each specific speaker with a particular data transform
- We aligned the feature vectors to HMM states using utterances’ transcription

6. Evaluation

- There are different methods to evaluate the quality of an ASR system
- In this project, we used Word Error Rate (WER), a common metric of the performance of a speech recognition
- The formula for WER, summing up the three types of errors (substitution, deletion, and insertion), over the length of the string as follows:

$$WER = \frac{100 * (S + I + D)}{N}$$

where;

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

6. Evaluation (Cont’d)

A basic alignment example:

Example 1:

Reference: စဉ်းစား ပေး ပါ *** ရှင်

Hypothesis: စဉ်းစား ပေး ပါ တယ် ရှင်

Eval: I

$$WER = \frac{100 * (0+1+0)}{4}$$

Example 2:

Reference: ကား ရဲ့ အရှိန် ကို လျှော့ ပါ

Hypothesis: ကား ရဲ့ အရှိန် ** လျှော့ ပါ

Eval: D

$$WER = \frac{100 * (0+0+1)}{6}$$

6.1. Evaluation on Training Data Size

- In our experiment,we used four different data sizes – 2 hrs, 3 hrs, 5 hrs, and 6 hrs for incremental training
- First of all, we analyzed the amount of data needed to train an mono-phone system (mono) and tri-phone systems such as Tri1 (Δ+ΔΔ), Tri2 (LDA + MLLT) and Tri3 (SAT) by using number of leaves, 200 and total Gaussian number, 1100

Table 3. The Performance Result (WER%) of Different Acoustic Training Methods with Incremental Data

Experiment	Mono	Tri1 (Δ+ΔΔ)	Tri2 (LDA+MLLT)	Tri3 (SAT)
1st Experiment (2hr)	29.65	31.58	28.62	23.22
2nd Experiment (3hr)	22.32	23.46	20.4	19.46
3rd Experiment (5hr)	18.46	20.1	17.97	17.69
4th Experiment (6hr)	**17.62**	**19.54**	**16.35**	**16.44**

6.2. Evaluation with n-gram Language Model

- In this task, the ASR accuracy is investigated based on different n-gram language models
- SRILM language modeling toolkit is applied to create the language model by using the default smoothing technique, good-tuning

Table 4. The WER % of the ASR Performance with N-gram Language Model

Language Model (n-gram)	Mono	Tri1 (Δ+ΔΔ)	Tri2 (LDA+ MLLT)	Tri3 (SAT)
0-gram	58.35	61.62	58.55	57.97
1-gram	47.68	50.84	47.09	48.09
2-gram	20.69	22.33	18.79	19.33
3-gram	**17.62**	**19.54**	**16.35**	**16.44**
4-gram	17.73	19.62	16.63	16.51
5-gram	17.72	19.58	16.46	16.65

6.3. Evaluation on Training with Different Model Size

- We re-trained the tri-phone model with two different transformation
- Here, we used all amount of data available that is we used 6 hr audio data and 3-gram LM to train the acoustic models. Moreover, we evaluated the performance of the model depending on the model size.

Table 5. The WER % of the ASR Performance using Different Model Size

Model Size	**No. of leaves**	200	200	200	500	500	500	700	700	700	1000
	Total No. of Gaussian	1100	3000	6000	1100	3000	6000	1100	3000	6000	6000
Tri1 (Δ+ΔΔ)		19.54	15.63	15.82	17.59	12.9	12.68	17.36	12.80	12.17	**11.95**
Tri2(LDA+MLLT)		16.35	14.02	14.38	14.7	11.71	11.66	14.65	11.16	10.82	**10.63**
Tri3 (SAT)		16.44	13.91	13.54	14.42	11.02	11.18	14.25	11.04	10.4	**10.21**

7. Conclusion

- In this project, different approaches to train and adapt acoustic models have been studied to be able to built an accurate automatic speech recognition system
- The training part is defined as the most important step since it determines mainly the accuracy of our system
- In our experiments, we observed a reduction of 11.78% on Mono-phone, 12.04% on Tri-phone (Δ+ΔΔ), 12.27% on Tri-phone (LDA+MLLT) and 6.57% on Tri-phone (SAT) in terms of word error rate during the incremental training
- Moreover, the ASR performance is compared and evaluated based on different n-gram (0-gram to 5-gram) LMs and also on the different model size
- From our experiment results, 3-gram based language model and total number of Gaussian, 6000 are the best for our system with the small amount of data.

References

[1] Ye Kyaw Thu, et al. “Comparison of Grapheme-to-Phoneme Con-version Methods on a Myanmar Pronunciation Dictionary”. Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing, pages 11-22, Osaka, Japan, December 11-17 2016.

[2] Yoav Ramon, “How to start with Kaldi and Speech Recognition”, https://towardsdatascience.co m/how-to-start-with-kaldi-and-speech-recognition-a9b7670ffff6, November 2018.

[3] Virendra Chauhan, Shobhana Dwivedi, Pooja Karale and Prof. S.M. Potdar, “Speech to Text Converter using Gaussian Mixture Model (GMM)”, Paper, International Research Journal of Engineering and Technology (IRJET), India, February 2016, pp. 160-164.

[4] Emelie Kullmann, “Speech to Text for Swedish using KALDI”, Master Thesis Book, KTH Royal Institute of Technology School of Engineering Sciences, Sweden, 2016.

[5] Madeline Briere, Automatic Speech Recognition using the Kaldi Toolkit, madelinebriere.com, Duke University, February 2018.

[6] Wolfgang Macherey, Discriminative Training and Acoustic Modeling for Automatic Speech Recognition, 2010.

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

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

[9] Mohri, Pereira and Riley, Speech Recognition with Weighted Finite-State Transducers, in Springer Handbook on Speech Pro-

cessing and Speech Communication, 2008.

[10] Shaghayegh Esmaeili, “Word Error Rate (WER) for Recognition of Natural Interactions”, Intelligent Natural Interaction Technology, April 2018.

Thank You