Research Report on ASR Development for Mongolian

A.Altangerel, B.Damdinsuren

Abstract

More than 6000 living languages are spoken in the world today. However, automatic speech recognition (ASR) systems have been built for a small number of major languages, such as English, Japanese, French, German, Chinese, Italian, and Arabic. Most other languages are resource-deficient, having no database or only sparse databases.[5.p11]

Every languages has own specific acoustic and linguistic characteristics that require special modeling techniques. This report presents our work on building ASR systems for Mongolian. Mongolian is one of the least studied languages for speech recognition. To build a Large Vocabulary Continuous Speech Recognition (LVCSR) system, high accurate acoustic models and large-scale language models are essential. There were no Mongolian speech database and text corpus for use in study. First, we collected text corpus. The text is selected from television programs, newspapers and web. Selection criterion was to cover as many different subjects as possible. In speech data, the most frequent words are selected from the text corpus. We are training the acoustic and language models based on Hidden Markov Models (HMMs). We evaluated the performance of isolated word recognition with context independent and context dependent models.

This report has following structure: the first section introduces Mongolian script and phonetics. In section two, introduction ASR, data preparation and acoustic modeling are described. Next section describes the experiments.

1. Introduction ASR

Automatic speech recognition(ASR) is one branch of the field of speech processing and related with a number of different fields of knowledge such as acoustic, linguistics, pattern recognition, and artificial intelligence. The complexity of an ASR system depends on its limitations, such as speaker dependence or independence, continuous or isolated speech, large, medium or small vocabulary. There are many speech recognition engines for speech recognition.
All Speech Recognition Engines (SRE)are made up of the following components:

- Language Model or Grammar

  Language Models contain a very large list of words and their probability of occurrence in a given sequence. Grammars are a much smaller file containing sets of predefined combinations of words. Grammars are used in **interactive voice response** or desktop command and control applications. Each word in a language model or grammar has an associated list of phonemes.
- **Acoustic Model** is a file that contains a statistical representation of each distinct sound that makes up a spoken word. It must contain the sounds for each word used in your grammar. The words in the grammar give the SRE the sequence of sounds it must listen for. The SRE then listens for the sequence of sounds that make up a particular word, and when it finds a particular sequence, returns the textual representation of the word The Grammar and the Acoustic Model work together.
- **Decoder** - Software program that takes the sounds spoken by a user and searches the Acoustic Model for the equivalent sounds. When a match is made, the Decoder determines the phoneme corresponding to the sound. It keeps track of the matching phonemes until it reaches a pause in the users speech. It then searches the Language Model or Grammar file for the equivalent series of phonemes.

We selected two toolkits, HTK and CMU Sphinx 4. These toolkits are both HMM based and support Windows OS and linux, and have modular design that is easily adaptable. We compared two speech recognizers that is HMM-based small vocabulary, built using HTK and CMU Sphinx 4 toolkits. HTK recognizer gave better result than CMU Sphinx 4. Therefore we selected HTK toolkit based on its recognition accuracy. Another reason why we selected HTK is it could be trained automatically and is simple and computationally feasible to use. [1]

2. Brief introduction of Mongolian language

2.1 Mongolian written system

There are three written systems in Mongolian language:

a) Classic Mongolian (Uygur Mongolian scripts or Hudomu Mongolian scripts)

b) Todo scripts

c) Cyrillic scripts

Classic Mongolian scripts and Todo script systems are used in inner Mongolia and Xinjiang of China. Classic Mongolian script and Cyrillic script systems are used in Mongolia. But official written system is Cyrillic. In the early 1930s, Classic Mongolian script was replaced with a short-lived Latin script that was used until about 1938. The Mongolian People's Republic, as it was called then, first started using a modified Russian Cyrillic alphabet in 1940 which is still used in Mongolia. Mongolian Cyrillic has 35 characters as shown in Table1.

Letter	English phonetics	Example of sound	Example of word	English transcription
Аа	(a)	c**a**r	агаар [aga:r]	Air
Бб	(b)	**b**ear	Бар [bar]	Tiger
Вв	(v)	**v**ery	Тавь[tav’]	fifty
Гг	(g)	**g**o	Гар[gar]	
Дд	(d)	**d**o	Давс[davs]	salt
Ее	(ye)	**ye**s	Ес[yes]	nein
Ёё	(yo)	**yo**ur	Ёотон[yoton]	sygar
Жж	(j)	**j**ar	Жимс[jims]	fruits
Зз	(z)	**z**oo	Загас[zaGs]	fish
Ии	(i)	k**i**ng, h**e**	Илжиг[iljig]	monkey
Йй	(i)	k**i**ng	Хайр[hair]	love
Кк	(k)	**k**ing	Карл[karl]	
Лл	(l)	**l**amp	Луу[lu:]	dragon
Мм	(m)	**m**an	Машин[mashin]	car
Нн	(n)	**n**ice	Нохой[nohoi]	dog
Оо	(o)	M**o**re, **o**r	Ор[or]	bed
Өө	(oe)	h**er**	Өндөг[oendoeg]	egg

Пп	(p)	**p**ark	Пуужин[ju:jin]	rocket
Рр	(r)	**r**adio	Радио[radio]	Radio
Сс	(s)	**s**ay	Сар[sar]	Moon
Тт	(t)	**t**axi	Талх[talh]	bread
Уу	(u/ow)	h**ow**	Ус[us]	water
Үү	(ue)	cart**oo**n	Үнээ[uene:]	cow
Фф	(f)	**ph**one		
Хх	(h)	**h**ot	Хонь[hon’]	sheep
Цц	(ts)	bli**tz**	Цана[tsan]	ski
Чч	(ch)	**ch**air	Чи[chi]	you
Шш	(sh)	**sh**oes	Шал[shal]	floor
Щ	(shch)			
Ъъ	hard sign	No sound	Явъя[yavya]	Let’s go
Ыы	iy	same i		
Ьь	soft sign	No sound	Морь[mor’]	horse
Ээ	e	**e**gg	Эгч[egch]	sister
Юю	yu	h**u**ge	Юу[yu]	what
Яя	ya	**ya**k	Яс[yas]	bone

Table 1. Mongolian Cyrillic alphabet

2.2 Mongolian phonetics

Mongolian languages is used mainly in Mongolia, parts of China(Inner Mongolia, Uugar), Russia(Buryat, Khalmyc) and their neighboring areas. Today, more than 8 million people speak Mongolian.[4] Mongolian has several dialects. For example: Khalkha, Oirat,Buryat, Khalmyc and Inner Mongolia etc. The khalha is main dialect in Mongolia.

There are seven short vowels, seven long vowels, five diphthongs and five 'y'-vowels and 19-23 consonants in the Mongolian languages.

Short vowels, long vowels, diphthongs and "y" vowels are shown in Table 2.

short vowels	a, o, u, ü(ue), ö(oe), e, i
long vowels	a:, o:, u:, ü:(ue:), ö:(oe:), e:,i:
dipthongs -5	ai, oi, ui, üi(uei), ei
"y" vowels	ya, yo, ye, yu, yü

Table 2. Mongolian vowels

Mongolian short vowels are shorter than English short vowels. Mongolian long vowels are shorter than English long vowels. Mongolian diphthongs are pronounced more united than English diphthongs.

Articulation chart for Mongolian vowels is shown in figure 1. The practical results of the formant distribution of vowels (in the Khalka dialect) are shown in figure 2 [9]. HTK training code is shown in Appendix B.

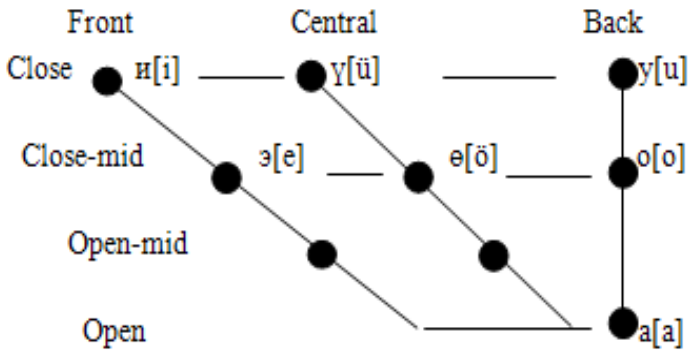

Figure 1 Articulation chart for Mongolian vowels

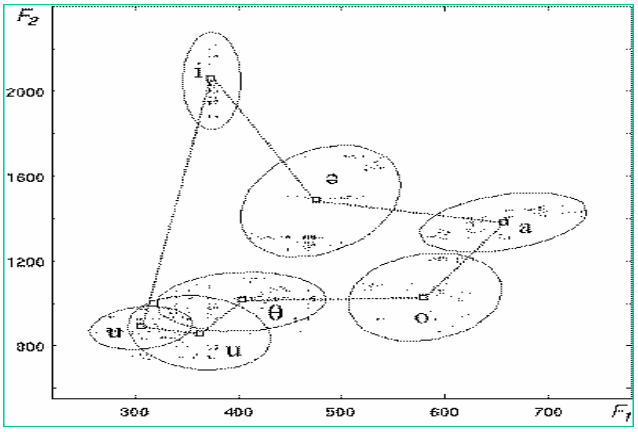

Figure 2. F1-F2 formant distribution of seven vowels

The consonants on the International Phonetic Alphabet(IPA) table is shown in Figure 3.

THE INTERNATIONAL PHONETIC ALPHABET (revised to 2005)

CONSONANTS (PULMONIC) © 2005 IPA

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	B			r					R		
Tap or Flap		ⱱ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

Figure 3.

Mongolian consonants are divided into voiced and voiceless and into hard and soft consonants. When pronouncing hard consonants the back part of the tongue is drawn back and raised. Soft consonants differ in the pronunciation from hard ones by some additional articulation: the middle part of the tongue is raised to the palate and the whole tongue is a bit protruded. The consonants **б** /b, b’/, **г** /g, g’/, **д** /d, d’/, **ж** /j, j’/, **з** /z z’/, **к** /k, k’/, **л** /l,l’/, **м** /m, m’/, **н** /n, n’/, **п** /p, p’/, **р** /r, r’/, **с** /s, s’/, **т**/t, t’/, **х** /h, h’/ have both hard and soft pronunciation. Their softness is indicated in by a soft sign **ь**. For example: бар/bar/- барь/bar’/, ам/am/ - амь/am’/

Mongolian vowels can be stressed and unstressed. Stress is the pronunciation of one vowel in a word with a greater force accompanied by an increase in the length of the sound compared with

other ones; such vowel is called stressed. Stress is marked by / ' / which is placed above the stressed vowel.

2. Methodology

The development of an ASR system requires the collection of a huge amount of annotated and transcribed speech data. The purpose of our work is to develop Mongolian speech recognition system using Hidden Markov Model. We use HTK Toolkit for implementing our system. The Hidden Markov Model Toolkit (HTK) is a toolkit for building and manipulating hidden Markov models. It is developed by the Cambridge University Speech Group. HTK consists of a set of library modules and tools available in C source form. The tools provide sophisticated facilities for speech analysis, HMM training, testing and results analysis. The software supports HMMs using both continuous density mixture Gaussians and discrete distributions and can be used to build complex HMM systems.

There are four main phases when constructing a speech recognition system: data preparation, training, testing and analysis. The commands to use HTK and auxiliary software modules are related to these main phases. They have an abbreviated form of typing. These forms represent the commands executed in operating system environment, while some command-like abbreviations represent the module programs that are used by these commands. The commands (tools) use the modules when performing a user defined task. In this respect, the modules are internally embedded. However, the user can control these modules either defining an option in the command line or defining the desired parameters in a text file.

HTK deals with two types of files: text files and speech data files. Text files can hold some editing commands, configuration parameters, transcription of speech files or the list of the files that will be used when performing the task. [ ] .

HTK recognizer flowchart is illustrated in Figure 3.

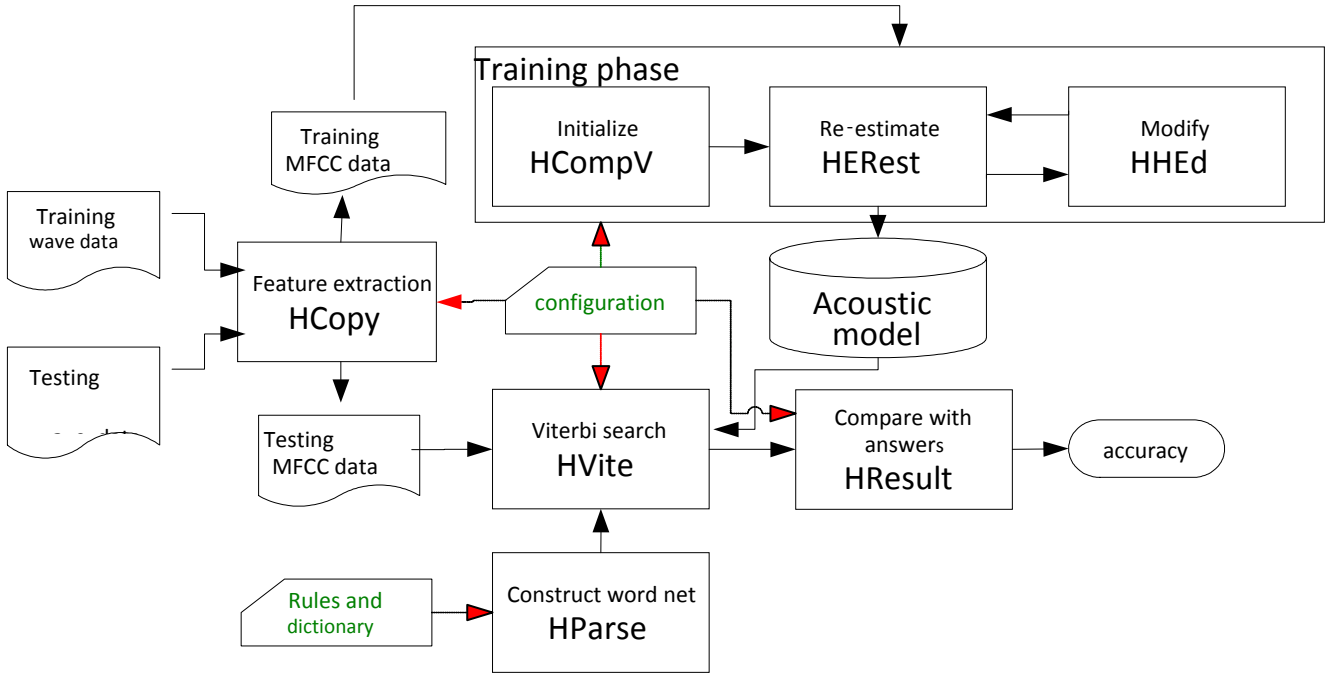

Figure 3. HTK recognizer flowchart.

2.1 **Speech data pre-processing and feature extraction**

The first stage of any recognizer development project is data preparation. Speech data is needed both for training and for testing. There were no Mongolian speech database and text corpus for use in study. First, we have collected text corpus. The text is selected from television programs, newspapers and websites. Selection of criteria was to cover as many different subjects as possible. In speech data, the most frequent 6000 words are selected from the text corpus. A total number of 80 speakers were selected to speak the selected words. The words were recorded using cardioids microphones. The age of speakers were selected from 19 to 25.

A sampling frequency of 16 kHz, a frame length of 25 ms with a Hamming window, a frame shift of 10 ms, and 39 dimensional feature parameters comprising 12 MFCC+1E with delta&delta^2 MFCC were used as the feature parameters. Three-state context-independent HMM AMs were used for each phoneme and Gaussian 4 mixture components per state were considered.

Our work has 4 phases :

- Selecting toolkit
- Training of selected toolkit
- Developing wrapping application
- Integration and testing

Each sub phase has following tasks:

- Preparing recording material (500 words)
- Recording 10 speakers 10 repetitions each word
- Transcription of recorded material
- Testing and integration

In each phase, all utterances were spoken by 10 Mongolian speakers using the Mongolian main dialect (halkh-95%).

3. Evaluation

Over 100 isolated words were used as a testing set. To record testing data, 20 native speakers who didn't attend to prepare the training data are selected. Word recognition accuracy shows over 90%.

References

[1] Report on Comparative Evaluation of Two Toolkits for Mongolian Speech

[2] Lawrence Rabiner, Biing-Hwang Juang, “Fundamental of Speech Recognition”, Pearson Education, 2005.

[3] Xueoong Huang, Alex Acero, and Hsiao-Wuen Hon, “Spoken Language Processing”, Prentice-Hall, 2001.

[4] I.Dawa, T.Shimizu, S.Nakamura, Y.Sagisaka A design and implementation of ATR Mongolian Speech recognition system

[5] Oriental-COCOSDA 2008, proceeding

[6] Ц. Дамдинсүрэн, “Монгол үсгийн дүрмийн толь” , 1983”.

[7] Ц. Дамдинсүрэн, Я. Цэвэл, “Үсгийн дүрмийн зөв бичих толь” , 1951”.

[8] А. Лувсандэндэв, “Монгол орос толь” , 1957”.

[9] Mark Aronoff, Kirsten Fudeman, “What is Morphology”, Blackwell Publishing, 2005.

[10] Peter Ladefoged, “A Course in Phonetics”, Thomson Learning, 2001

[11] Stephen Haag, Maeve Cummings, James Dawkins, “Management Information Systems for the Information Age”, McGraw-Hill, 1998.

[12] Я. Цэвэл, “Монгол хэлний товч тайлбар толь” , 1966”.

[13] Х. Далхжав, Ц. Цэрэнчимэд, “Зөв бичих зүйн толь бичиг” , 1974”.

[14] С. Шагж, “ Монгол үсгийн дүрмийн толь бичиг”

HParse lm/dict.gram lm/dict.wdnet

Creating the Transcription Files

HLEd -l * -d config\dict.dict -i config\monophn.mlf config\mkphones0.led config\word.mlf

Create Monophone HMMs

rmdir am

mkdir am

cd am

mkdir hmm_0

cd..

HCopy -T 1 -C config\code.config -S config\trcode.scp

HCompV -C config\train.config -f 0.01 -m -S config\train.scp -M am\hmm_0 config\proto5s

perl script\createmono.pl config\monophn.list am\hmm_0\proto5s am\hmm_0\vFloors

am\hmm_0\newMacros

cd am

mkdir hmm_1

mkdir hmm_2

mkdir hmm_3

mkdir hmm_4

cd..

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H

am\hmm_0\newMacros -M am\hmm_1 config\monophn.list

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H

am\hmm_1\newMacros -M am\hmm_2 config\monophn.list

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H

am\hmm_2\newMacros -M am\hmm_3 config\monophn.list

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H

am\hmm_3\newMacros -M am\hmm_4 config\monophn.list

cd am

mkdir hmm_5

mkdir hmm_6

mkdir hmm_7

cd..

HHEd -H am\hmm_4\newMacros -M am\hmm_5 config\sil.hed config\monophn.list

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H am\hmm_5\newMacros -M am\hmm_6 config\monophn.list

HERest -T 1 -C config\train.config -I config\monophn.mlf -S config\train.scp -H am\hmm_6\newMacros -M am\hmm_7 config\monophn.list

Create Tied-State Triphones

cd am

mkdir trihmm_0

cd..

HLED -T 1 -n config\triphn.list -l * -i config\triphn.mlf config\mktri.led config\monophn.mlf

perl script/maketrihed.pl config/monophn.list config/triphn.list config/mktri.hed

HHEd -T 1 -H am/hmm_7/newMacros -M am/trihmm_0 config/mktri.hed config/monophn.list

cd am

mkdir trihmm_1

mkdir trihmm_2

mkdir trihmm_3

cd..

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -s am/trihmm_1/stats -S config/train.scp -H am/trihmm_0/newMacros -M am/trihmm_1 config/triphn.list

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -s am/trihmm_2/stats -S config/train.scp -H am/trihmm_1/newMacros -M am/trihmm_2 config/triphn.list

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -s am/trihmm_3/stats -S config/train.scp -H am/trihmm_2/newMacros -M am/trihmm_3 config/triphn.list

cd am

mkdir tiehmm_0

cd..

HHed -T 1 -H am/trihmm_3/newMacros -M am/tiehmm_0 config/tie.hed config/triphn.list

cd am

```
mkdir tiehmm_1
mkdir tiehmm_2
mkdir tiehmm_3
cd..
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H
am/tiehmm_0/newMacros -M am/tiehmm_1 config/tie.list
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H
am/tiehmm_1/newMacros -M am/tiehmm_2 config/tie.list
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H
am/tiehmm_2/newMacros -M am/tiehmm_3 config/tie.list

cd am
mkdir hmm4m_0
mkdir hmm4m_1
mkdir hmm4m_2
mkdir hmm4m_3
cd..
HHED -T 1 -H am/hmm_3/newMacros -M am/hmm4m_0 config/mix1to4.hed config/monophn.list
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/monophn.mlf -S config/train.scp -H
am/hmm4m_0/newmacros -M am/hmm4m_1 config/monophn.list
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/monophn.mlf -S config/train.scp -H
am/hmm4m_1/newmacros -M am/hmm4m_2 config/monophn.list
HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/monophn.mlf -S config/train.scp -H
am/hmm4m_2/newmacros -M am/hmm4m_3 config/monophn.list

cd am
mkdir tiehmm4m_0
mkdir tiehmm4m_1
mkdir tiehmm4m_2
mkdir tiehmm4m_3
cd..
```

HHEd -A -D -T 1 -H am/tiehmm_3/newMacros -M am/tiehmm4m_0 config/mix1to4.hed config/tie.list

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H am/tiehmm4m_0/newmacros -M am/tiehmm4m_1 config/tie.list

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H am/tiehmm4m_1/newmacros -M am/tiehmm4m_2 config/tie.list

HERest -T 1 -C config/train.config -v 1.0e-8 -m 1 -I config/triphn.mlf -S config/train.scp -H am/tiehmm4m_2/newmacros -M am/tiehmm4m_3 config/tie.list

HCopy -T 1 -C config/code.config -S config/tscode.scp

HCopy -T 1 -C config/code.config -S config/tscode.scp

HVite -H am/hmm4m_3/newMacros -S config/test.scp -l '*' -w lm/dict.wdnet -i result/resmono.mlf config/dict.dict config/monophn.list

HVite -H am/tiehmm4m_3/newMacros -S config/test.scp -l '*' -w lm/dict.wdnet -i result/restie.mlf config/dict.dict config/tie.list

evalution

HResults -I config/test.mlf config/monophn.list result/resmono.mlf

HResults -I config/test.mlf config/tie.list result/restie.mlf