Exam number: B117474

Speech Processing

Assignment 2

– Speech Recognition: Digit Recogniser

Word count: 3008

B117474

Outline

1 Introduction

Automatic Speech Recognition(ASR) maps acoustic signals to strings of words. Its main steps(components) include: front-end(feature vectors), training(acoustic model and language model, Viterbi and Baum-Welch algorithms), recognising(decoder) and evaluation.

The front-end converts waveform into feature vectors, normally Mel-Frequency Cepstral Coefficient(MFCC), which are uncorrelated means of cepstral features.

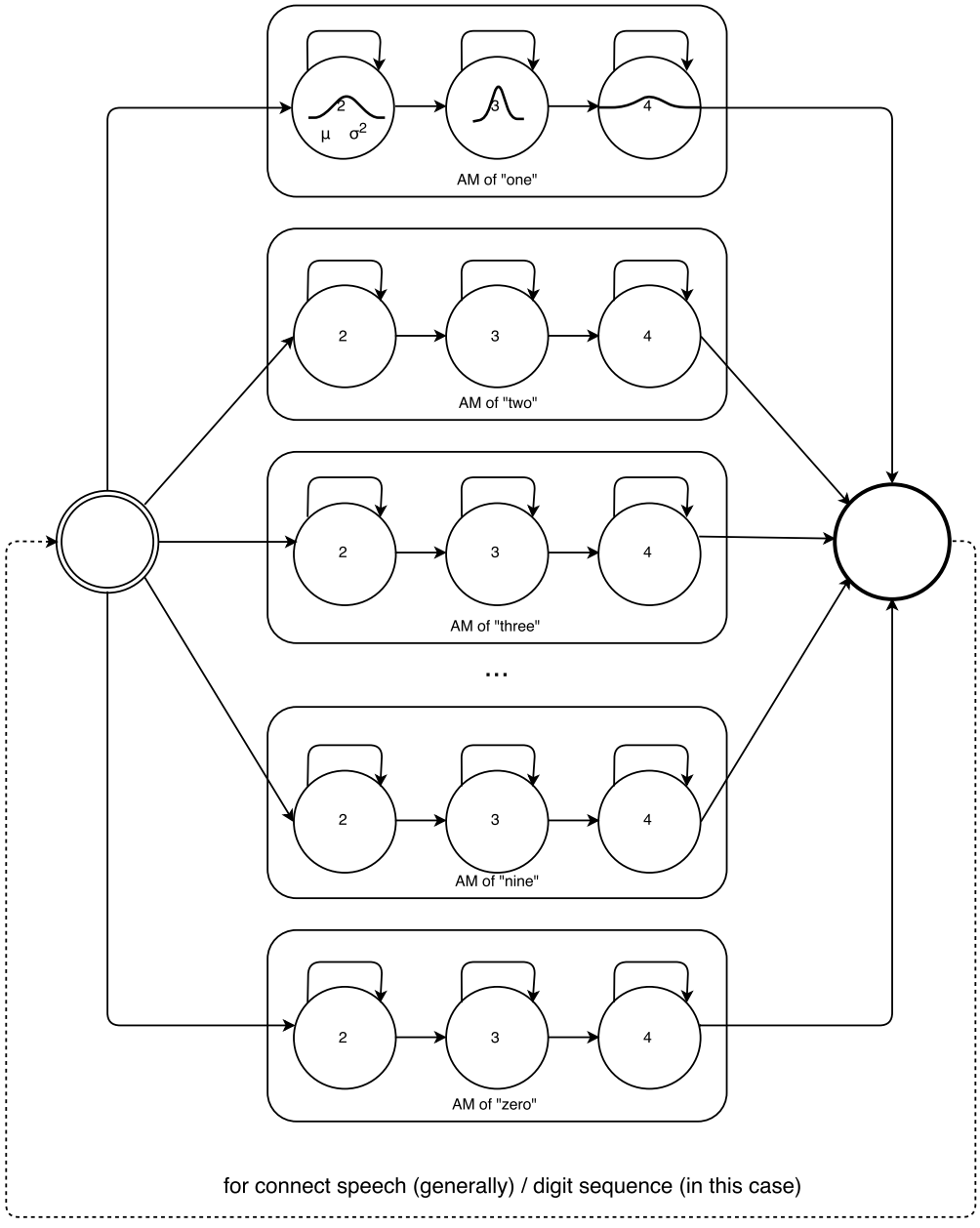
During training, the acoustic model(AM), basically Hidden-Markov Model(HMM), is learned from the observation of feature vectors and corresponding transcripts by creating statistical representation. AM applies Gaussian model(GM) probability density function(pdf)s to represent the mean and variance of MFCCs for each corresponding HMM state. Viterbi and Baum-Welch are the main algorithms for training. The language model(LM), including the dictionary mapping words to phones, provides probabilities of words occurring in some order. AM and LM are connected into a finite state automaton(FSA).

In recognition, the Viterbi decoder, aligns observation to be recognised with states and finds the most probable state sequence given the observation. The recognition result is evaluated by word error rate(WER).

ASR faces challenges due to the inherent variability of speech. Speakers' genders, accents, recording equipment, etc. might differently affect ASR performance. To explore whether and how different factors may influence ASR performance, we implement various experiments on an HMM-based ASR system built by the Hidden Markov Model Toolkit(HTK)[13].

2 Background

Figure 1. A compiled model of digit recogniser: uniform-probability LM and 3-emission-state whole-word AMs
(rounded rectangles are AMs of digit, within which are states of GMs featuring mean and variance; all rounded rectangles constitute the FSA LM for digits; the dashed line makes digit sequence recogniser(DSR) possible)



LM

2.1 Data collection and acoustic features

Data is the pre-requisite for ASR. We need labelled training data to train models and labelled testing data to test models' performance. In an isolated digit recogniser(IDR), each digit is labelled its corresponding orthographic word; silence between digits is labelled "junk" and discarded. Besides, an orthographic transcription of test data, in the form of MLF(master label file) is created to measure the model performance.

Collected waveform data, is parameterised and represented by acoustic features. These acoustic features are feature vectors, each representing the information in a small time window(frame) of the signal[10, p.331]. For each signal in each frame, Discrete Fourier Transform extracts its spectral information. Then, for each spectrum, a cepstrum, which separates source and filter, is computed by taking log of the spectrum. We keep the first twelve cepstral coefficients as twelve features for a frame and take the energy in this frame as the thirteenth, and thirteen more velocity features and another thirteen acceleration features of energy changing. The above 39 features constitute a MFCC feature vector for a frame.

MFCC is important in ASR. First, due to the feature of cepstrum, it excludes features of source which are irrelevant to ASR. Second, as cepstral coefficients are uncorrelated[10, p.336], HMM becomes simpler; otherwise it has to capture feature covariance, which complicates the computation.

In HTK, `HCopy` makes waveform into MFCC feature vectors.

2.2 Training HMMs

HMM, a probabilistic generative model, is ASR's dominant paradigm. In ASR, for each basic unit to be recognised, there is an HMM. In IDR, if using whole-word models, for every word there is an HMM(Fig.1); if using phone models, for each phone or sub-phone there is an HMM. In each HMM, there are two non-emission states(start&end) with at least one emission state. Each emission state is a GM parameterised by its mean μ and variance σ^2 learned from training data. GM statistically captures speech variability.

Training includes initialisation and re-estimation. First, we uniformly segment MFCCs according to the number of states, and calculate each state's initial μ and σ^2. We then apply Viterbi to re-align the observation to states and update μ and σ^2 until they become steady. Besides Viterbi, a more fine-grained algorithm, Baum-Welch algorithm is also implemented in re-estimation. Different from Viterbi which gives the single most probable alignment, Baum-Welch considers the probabilities of all possible state sequences, and update model parameters by taking the mean of the sum of weighted probabilities of each state sequence given an observation. Thus it "softly" aligns a state sequence to an observation. However, it takes a longer time and more computation.

HTK provides `HInit` to initialise HMMs and use Viterbi algorithm to update the parameters. In `HRest`, the parameters are further re-estimated by Baum-Welch algorithm.

By default, we use 4-emission-state 39-dimension MFCC HMMs, but it is flexible to change the model topology in HTK. HMM in HTK includes dimensionality of observation vectors; type of observation vector MFCC, form of the covariance matrix diagonal, number of states, parameters of GMs in states; a transition matrix with zero probabilities indicating never-ever allowed transitions.

2.3 Language modelling

LM is a generative model, emitting a word sequence given a sentence. LM is important in ASR in the sense that it guides and constrains the search among alternative word hypotheses as LM determines the probability for a word given word history[3]. With LM, it is more likely to have lower WERs[9]. In Large-Vocabulary Continuous Speech Recognition(LVCSR), we usually use trigrams or 4-grams[10, p.348]. In IDRs, we apply a simple FSA LM assigned uniform probability to each digit, only allowing sentences containing one digit. The dashed line linking the end to the start indicates an LM for digit sequence of arbitrary length.

An FSA LM can be easily combined with HMM AMs as in Fig.1, because the LM and AM are both FSA. We simply substitute every state of the LM with the corresponding AM.

In HTK, `HVite` compiles the LM with AMs to make a single finite-state recognition network, resulting in a generative model that emits a sequence of MFCCs given a sentence.

2.4 Recognition using HMMs

Recognition is to use Viterbi and forward algorithm to find the most probable sequence of states in HMM AM given the observation of MFCCs, and then find the most probable word sequence given observation of a sentence based on LM, thus mapping acoustic signals to strings of words.

In IDR, we find the most probable state sequence of each HMM given the observation of MFCCs to be recognised, and take the model that gives the highest probability as the recognition result because the LM in this case is one with uniform transition probabilities. And DSR is only different from IDR that it repeats the recognition process of a single digit to deal with digit sequence of arbitrary length.

Generally in ASR, LM gives the probability of emitting a certain word given the state. By combining the probabilities provided by LM and AM estimators, we finally get a string of words that maximises the combined probabilities.

In HTK, `HVite` does the recognition.

The performance of HMM-ASR is evaluated by word error rate(WER). WER tells how much the word string returned by the recogniser differs from a reference transcription[10, p.362]. WER is computed as:

$$WER = \frac{word\_insertions + word\_substitutions + word\_deletions}{total\_words\_in\_reference\_transcript} \times 100\%$$

In HTK, `HResults` computes insertions, substitutions and deletions, and reports accuracy. The following experiments are consistent with using WER as evaluation.

3 Experiments

3.1 Number of HMM emission states

3.1.1 Hypothesis

The number of emission states(No.ES) defines different HMM topology. So, the hypothesis is, No.ES affects ASR performance. If so, some relationship between No.ES and ASR performance is expected to be seen.

3.1.2 Experiment design

No.ES varies from 3 to 21, and the other variables(gender, accent and microphone type) are controlled the same or of the same distribution. Two groups of experiments of different amount of training data are designed below:

	Amount	Gender	Accent	Microphone type	Overlap?[1]
Training A	50	1:1[2]	NN[3],Mx[4]	Mx	No
Training B	20				
Test	20				

Table 1. The training sets include 50 or 20 non-native speakers with mixed accents and microphone types; the test data includes 20 non-native speakers with the same distribution of genders and microphone types as the training data

[1] Whether there is overlap between training and test data.
[2] If not stated, female:male.
[3] NN:Non-Native speakers.
[4] All experiments control variables not of concern. Those variables, marked by “Mx”, are either controlled the same or of the same distribution.

3.1.3 Results

The WER(%) and runtime(s) are shown below.

Training data amount	No.ES	WER(%)	WER ranking	Runtime(s)	Runtime ranking
50	3	25.33	19	14.483	1
	4	20.33	18	17.495	2
	5	16.00	17	20.847	3
	6	12.67	16	22.997	4
	7	10.00	15	27.606	5
	8	8.50	14	29.253	6
	9	6.83	13	30.874	7
	10	5.50	12	34.163	8
	11	4.67	8	37.697	9
	12	4.67	9	40.108	10
	13	4.17	2	42.910	11
	14	4.67	10	43.318	12
	15	4.67	11	47.355	13
	16	4.17	3	49.627	14
	17	4.50	6	53.301	15
	18	4.17	4	55.669	16
	19	4.17	5	58.087	17
	20	**3.67**	1	63.346	18
	21	4.50	7	63.866	19

Table 2. The WER(%) and runtime(s) with varying No.ES(from 3 to 21); training on 50 and testing 20 speakers of mixed genders, accents and microphone types

Training data amount	No.ES	WER(%)	WER ranking	Runtime(s)	Runtime ranking
20	3	26.00	19	9.830	1
	4	17.17	18	10.327	2
	5	12.33	17	11.894	3
	6	10.17	16	12.926	4
	7	8.50	15	14.480	5
	8	6.83	14	14.998	6
	9	6.50	13	16.221	7
	10	5.17	12	16.993	8
	11	4.83	11	17.090	9
	12	3.83	7	18.982	10
	13	**3.33**	1	19.797	11
	14	3.50	2	21.498	12
	15	3.67	4	25.686	13
	16	3.67	5	22.737	14
	17	4.33	9	24.001	15
	18	3.50	3	25.042	16
	19	3.67	6	26.665	17
	20	4.00	8	27.012	18
	21	4.33	10	28.469	19

Table 3. The WER(%) and runtime(s) with varying No.ES(from 3 to 21); training and testing on 20 speakers of mixed genders, accents and microphone types

Table 4 shows the Spearman correlation coefficient of No.ES and WER and runtime.

	r_s of No.ES and WER	r_s of No.ES and runtime
Training A	-0.8859649	1
Training B	-0.7701754	1

Table 4. The Spearman correlation coefficient r_s between No.ES and WER and runtime of two groups

We see an almost monotonic negative correlation between No.ES and WER, and a perfect monotonic positive correlation between No.ES and runtime in both groups.

Fig.2 and Fig.3 visualises Table 1 and Table 2 respectively.

Figure 2. The visualisation of WER(%) and runtime(s) with varying No.ES(from 3 to 21); training on 50 speakers(red dot at 20 states: the so-far lowest WER 5.17%)

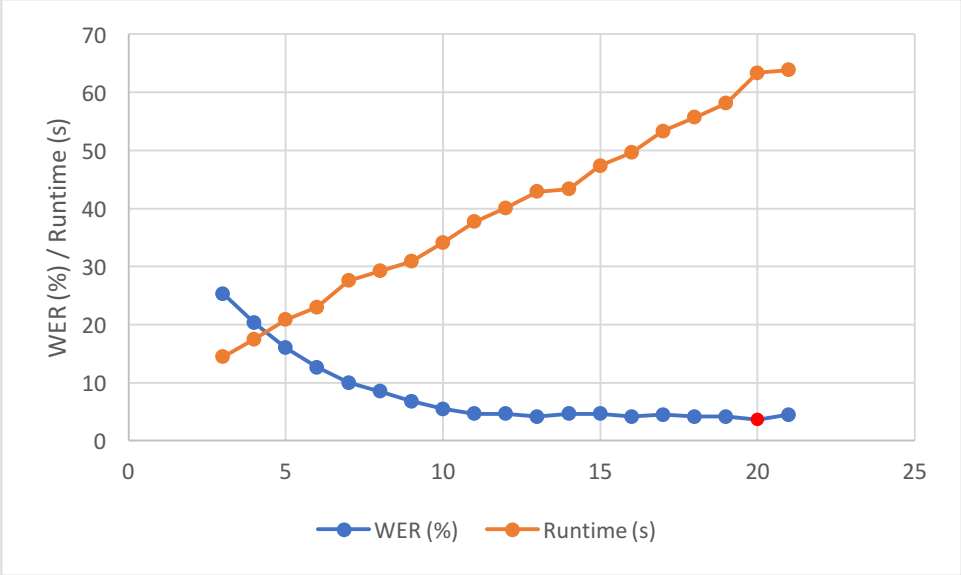
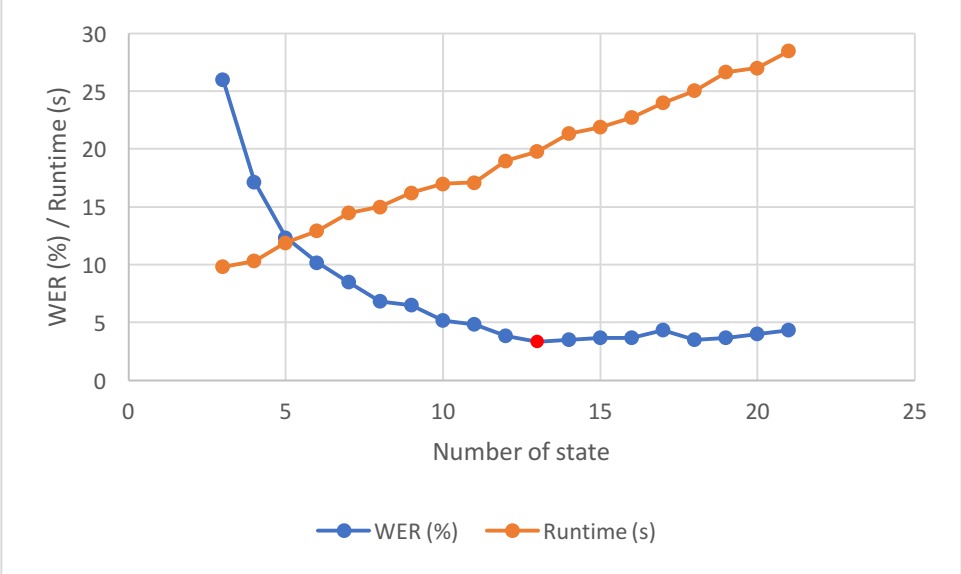


Figure 3. The visualisation of WER(%) and runtime(s) with varying No.ES(from 3 to 21); training on 20 speakers(red dot at 13 states: the so-far lowest WER 3.33%)



These two figures show that, for the group trained on 50 speakers, the optimal No.ES is 20; for the other, it is 13.

Furthermore, Fig.4 compares the WERs between two groups.

Figure 4. The WER(%) of two groups with varying No.ES(from 3 to 21) (red points: the so-far lowest WERs)

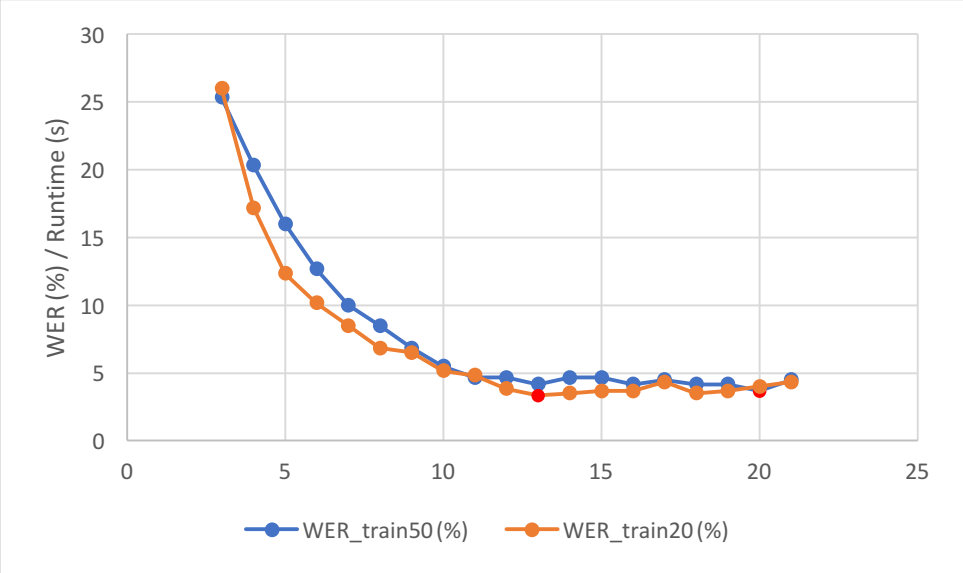


Fig.4 shows a similar trend of WER between two groups. Generally, WER decreases with No.ES increasing. In both groups, WER decreases considerably from 3 states to 8 states; after that, WER shrinks slowly, gradually becomes stable and fluctuates in a small range. Besides, the bigger the amount of training data is, the larger WER is, regardless of No.ES.

However, No.ES might vary, for each digit AM, the number of state is fixed. This is called fixed length modelling[14], which lacks flexibility, as the length/duration of digit actually differs. To assign a unique No.ES to each model might help, such as Bakis length modelling[2] finding the best alignment between each digit and states, or some more flexible method[7] adding a constant additive to Bakis formula.

Furthermore, with more states, models risk of overfitting to training data. Thus WER is expected to go up again after a certain No.ES. Specifically, when No.ES exceeds the biggest number of frames of digit, it will probably break down. What's more, more

states are penalised by accelerated increase in runtime and do not always guarantee better recognition. These are all important cues when deciding the optimal No.ES.

In the following each experiment, the No.ES is default to 4, and there is no overlap between any training and test sets, if not stated.

3.2 Language family

3.2.1 Hypothesis

Languages developed from the same historic language belong to the same language family. Languages in the same language family usually share some features in phonology, morphology and syntax. So, a hypothesis is that ASR systems perform better when tested on data of accents from the same or close language families as the training data, than those tested on data of accents different or far from the language family of the training data.

3.2.2 Experiment design

According to the language families categorised by Ethnologue[5], we choose Indo-European and Sino-Tibetan as cross-language-family comparison. Inside the Indo-European language family, there is cross-branch comparison between Germanic and Romance. The training data uses UK English(Indo-European: Germanic) and the same for Test A; Test B contains some other Indo-European Germanic accents except English; Test C contains Indo-European Romance accents; Test D contains accents from another language family: Sino-Tibetan.

	Amount	Gender	**Language family**		Accent	Microphone
Training	30	1:1	Indo-European	Germanic	UK	Mx
Test A					UK	
Test B					German,Dutch, Danish,Swedish, Norwegian	
Test C				Romance	French,Greece, Italian, Romanian	
Test D			Sino-Tibetan		Chinese,Thai	

Table 5. The training and data sets of testing how language families or branches influence ASR performance

3.2.3 Results

	WER(%)
Test A	7.63
Test B	9.22
Test C	15.24
Test D	16.70

Table 6. WER(%) of different language families or branches

In Table 6, WER ascends from Test A to Test D. The closer the accents of the test data are to the training data, the lower WER is. The WER of crossing-branch accents is lower than that of crossing-language-family accents. Languages belonging to the Germanic branch in the Indo-European family tend to have similar linguistic features with English, and even share common vocabulary in similar orthographic form and pronunciation. Languages in the Romance branch still share something in common with Germanic languages, but the differences influence ASR performance even more. The WER of testing on Romance speakers is considerably higher than that of Germanic speakers. Besides, testing on accents from a different language family results the highest WER. These suffice to conclude that accents from different language families and branches considerably affect ASR performance.

3.3 Gender and Accent

3.3.1 Hypothesis

Accents reflect the differences in shapes of vocal tracts, which are important features in MFCCs. Female and male may also have different shapes of vocal tracts. So, the hypothesis is that, accents and genders both affect ASR performance. And further comparison between these two factors will also be carried out.

3.3.2 Experiment design

From Table 7 to Table 10, all factors except genders are controlled. From Table 11 to Table 14, all factors except accents are controlled.

For the sake of convenience, we call from Test A to Test H the gender experiments, and the rest, the accent experiments.

Gender

	Amount	**Gender**	Accent	Microphone
Training	20	Female	UK	Mx
Test A	10	Female		
Test B		Male		

Table 7. Training on UK female; testing UK female and male

	Amount	**Gender**	Accent	Microphone
Training	20	Male	UK	Mx
Test C	10	Male		
Test D		Female		

Table 8. Training on UK male; testing UK male and female

	Amount	**Gender**	Accent	Microphone
Training	20	Female	NA[5]	Mx
Test E	10	Female		
Test F		Male		

Table 9. Training on NA female; testing NA female and male

	Amount	**Gender**	Accent	Microphone
Training	15	Male	NA	Mx
Test G	10	Male		
Test H		Female		

Table 10. Training on NA male; testing NA male and female

[5] North America

Accent

	Amount	Gender	**Accent**	Microphone
Training	20	Female	UK	Mx
Test I	10		UK	
Test J			NA	

Table 11. Training on UK female; testing UK and NA female

	Amount	Gender	**Accent**	Microphone
Training	20	Male	UK	Mx
Test K	10		UK	
Test L			NA	

Table 12. Training on UK male; testing UK and NA male

	Amount	Gender	**Accent**	Microphone
Training	20	Female	NA	Mx
Test M	10		NA	
Test N			UK	

Table 13. Training on NA female; testing NA and UK female

	Amount	Gender	**Accent**	Microphone
Training	15	Male	NA	Mx
Test O	10		NA	
Test P			UK	

Table 14. Training on NA male; testing NA and UK male

3.3.3 Results

	Test	WER(%)	Delta(%)	Mean(%)	Variance
Gender	A	3.44	-13.01	-16.6925	66.4474917
	B	16.45			
	C	2.26	-16.18		
	D	18.44			
	E	9.00	-9.39		
	F	18.39			
	G	5.48	-28.19		
	H	33.67			
Accent	I	3.44	-7.56	-4.22	11.2243333
	J	11.00			
	K	2.26	-6.13		
	L	8.39			
	M	9.00	-3.19		
	N	12.19			
	O	5.48	0		
	P	5.48			

Table 15. The WER(%), delta(%), mean(%) and variance of WER of gender and accent comparison test groups

Figure 5. The visualisation of results of the gender experiments (blue:WER(%), orange:WER delta(%))

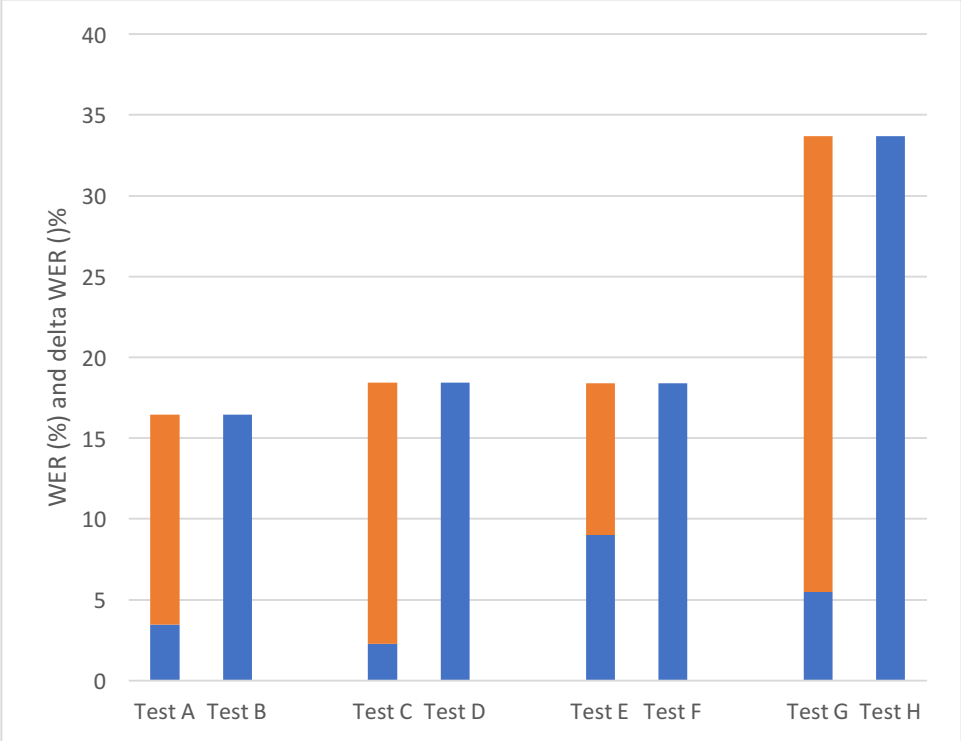


Figure 6. The visualisation of results of the accent experiments (blue:WER(%), orange:WER delta(%))

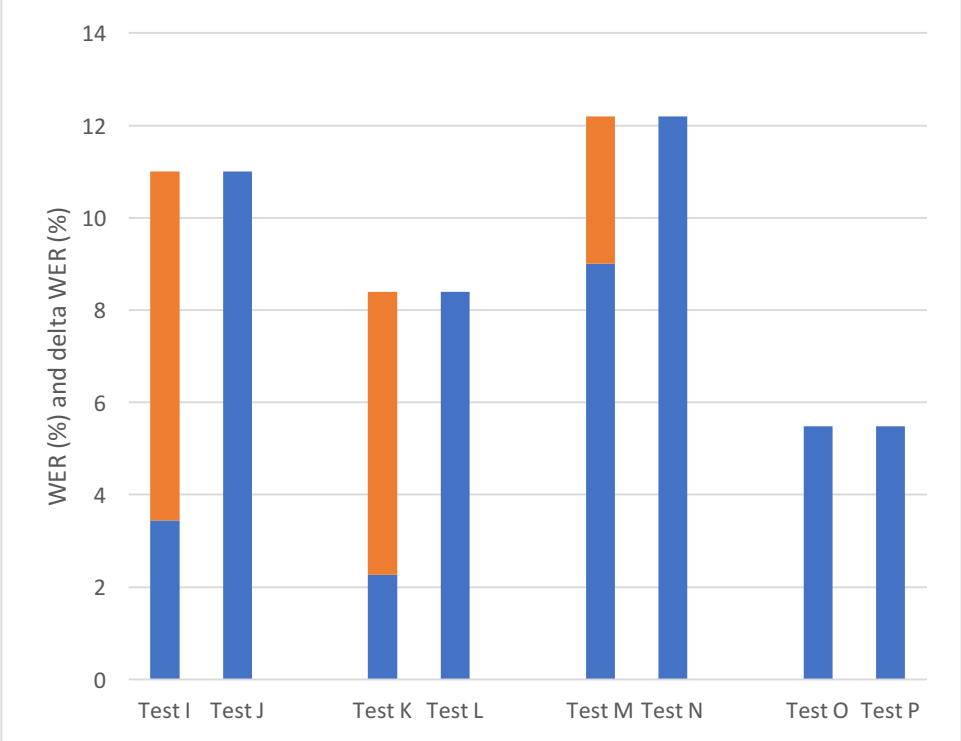


Table 15, Fig.5 and Fig.6 show that, mismatch between genders or accents both raise WERs. Besides, the mean and variance of WERs in the gender experiments are much larger than those in the accent experiments. Genders seem to influence ASR more than accents do. However, genders and accents both represent differences in filter frequencies indicating different shapes of vocal tracts, which should not result in such big gaps. We thus make two possible conjectures: (1) the differences in accents between native English speakers is not obvious enough to adversely impact ASR, and (2) features of tones might still be remained in cepstra and MFCCs, as genders largely determine tone features (male's tones being usually lower than female's). Although pitch information as irrelevant to recognition, is often said to be discarded by ASR[8], tone, which is equivalent to pitch in the sense of perception of fundamental frequency, might still be useful in ASR as it helps differentiate genders and even emotions in advanced ASR tasks.

3.4 Viterbi and Baum-Welch

3.4.1 Hypothesis

The training of ASR has two main steps: initialisation and re-estimation. In HTK, `HInit` initialises HMMs by uniform segmentation and Viterbi; `HRest` further updates model parameters by Baum-Welch. The hypothesis is that, omitting any one of the training steps, affects ASR performance.

3.4.2 Experiment design

This experiment includes two groups of different genders. All the other factors are mixed.

Group_F	Amount	Gender	Accent	Microphone
Training_F	30	Female	Mx	Mx
Test_F	10	Female	Mx	Mx

Table 16. The female group of the omitting training algorithm experiment: training on 30 female and testing on 10 female, both with mixed accents and microphone types

Group_M	Amount	Gender	Accent	Microphone
Training_M	30	Male	Mx	Mx
Test_M	10	Male	Mx	Mx

Table 17. The male group of the omitting training algorithm experiment: training on 30 male and test on 10 male, both with mixed accents and microphone types

3.4.3 Results

	WER(%) Default	Time(s)	WER(%) Without `HInit`	Time(s)	WER(%) Without `HRest`	Time(s)
Group_F	6.12	18.021	6.12	13.738	9.78	11.503
Group_M	5.71	17.722	5.71	12.882	16.71	11.813

Table 18. The WER(%) and runtime(s) of the omitting training algorithm experiment

From Table 18, in both groups, the WERs are the highest when omitting `HRest`, i.e. Baum-Welch. The WERs do not change even omitting `HInit`, i.e. initialization and Viterbi.

The initialisation based on uniform segmentation is crude, but it provides initialised parameters for the following re-estimation algorithms. These initialised parameters greatly decrease the runtime and computation of Viterbi and Baum-Welch; otherwise the two algorithms would take a longer time to have the re-estimated parameters respectively converge.

Baum-Welch is more fine-grained than Viterbi that, Baum-Welch considers all possible state sequences generated by the model while training, but Viterbi only gives one single best alignment. This makes sense when we observe a higher WER when omitting Baum-Welch.

Therefore, Baum-Welch plays a bigger role in influencing ASR performance.

4 Discussion and conclusion

ASR compiles AM and LM together to recognise speech into words. The implementation of Viterbi and Baum-Welch, two effective algorithms in training and decoding, ensures good performance of ASR. However, still there are multiple factors that could influence ASR performance such as: speakers' gender, accent, recording equipment; the amount of data, the number of hidden states; and the actual implementation of algorithms. All are proved to affect ASR performance in different degrees, which inspires the consideration for building better ASR systems such as: choosing better-balanced training data of a reasonable amount, selecting optimal number of states or varying number of states for observations of different length/duration; effectively capturing the biggest diversity of training data regarding genders, accents, etc. Further experiments will cover some other factors, for instance, the quality of data, models implemented in AMs, etc.

Bibliography (in alphabetical order)

[1] Averbuch, A., Bahl, L., Bakis, R., Brown, P., Daggett, G., Das, S., ... & Fraleigh, D. (1987). Experiments with the TANGORA 20,000 word speech recognizer. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'87. (Vol. 12, pp. 701-704). IEEE.

[2] Bakis, R. (1976). Continuous speech recognition via centisecond acoustic states. The Journal of the Acoustical Society of America, 59(S1), S97-S97.

[3] Broman, S., & Kurimo, M. (2005). Methods for combining language models in speech recognition. In Ninth European Conference on Speech Communication and Technology.

[4] Chow, Y., Dunham, M., Kimball, O., Krasner, M., Kubala, G., Makhoul, J., ... & Schwartz, R. (1987). BYBLOS: The BBN continuous speech recognition system. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'87. (Vol. 12, pp. 89-92). IEEE.

[5] Ethnologue. Language families. https://www.ethnologue.com/browse/families (Retrieved on 23/11/2017)

[6] Fromkin, V. A. (Ed.). (2014). Tone: A linguistic survey. Academic Press.

[7] Geiger, J., Schenk, J., Wallhoff, F., & Rigoll, G. (2010). Optimizing the number of states for HMM-based on-line handwritten whiteboard recognition. In Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on (pp. 107-112). IEEE.

[8] Gerhard, D. (2003). Pitch extraction and fundamental frequency: History and current techniques (pp. 0-22). Regina: Department of Computer Science, University of Regina.

[9] K. F. Lee. (1988). The CMU SPHINX System, Ph.D. Thesis, CMU.

[10] Martin, J. H., & Jurafsky, D. (2009). Speech and language processing. Second Edition. New Jersey: Pearson Education.

[11] Schwartz, R., Chow, Y. L., Kimball, O., Roucos, S., Krasner, M., & Makhoul, J. (1985). Context-dependent modeling for acoustic-phonetic recognition of continuous speech. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'85. (Vol. 10, pp. 1205-1208). IEEE.

[12] Stuttle, M. N. (2003). A Gaussian mixture model spectral representation for speech recognition (Doctoral dissertation, University of Cambridge).
[13] The Hidden Markov Model Toolkit (HTK). Cambridge University. http://htk.eng.cam.ac.uk (Retrieved on 23/11/2017)
[14] Zimmermann, M., & Bunke, H. (2002). Hidden Markov model length optimization for handwriting recognition systems. In Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop on (pp. 369-374). IEEE.