Tonal phoneme based model for Vietnamese LVCSR

Van Huy Nguyen
Electronic faculty
Thai Nguyen University of Technology, Vietnam
huynguyen@tnut.edu.vn

Chi Mai Luong, Tat Thang Vu
Institute of Information Technology (IOIT)
Vietnam Academy of Science and Technology, Vietnam
{lcmai,vtthang}@ioit.ac.vn

***Abstract*—This paper proposes an algorithm that is first known as a grapheme-to-phoneme method to transform any Vietnamese word to a tonal phoneme-based pronunciation. The tonal phoneme set produced by this algorithm is further used to develop some acoustic models which integrated tone information and tonal feature. The processes using the Kaldi toolkit to develop a LVCSR system and extract a bottleneck feature which is calculated from a trained deep neural network for Vietnamese are also presented. The results showed that the use of tonal phoneme improved by 1.54% of word error rate (WER) compared to the system using the nontonal phoneme, the use of tonal feature information improved by 4.65% of WER, and of the bottleneck feature gave the best WER with about 10% improvement.**

Keywords—Vietnamese grapheme to phoneme, bottleneck feature, Vietnamese automatic speech recognition, pitch, VoiceTra.

I. INTRODUCTION

Tonal languages like Vietnamese, Mandarin and Cantonese generally use tones to represent phone level distinction, which are therefore essential to distinguish between words. Such tone information is generated by excursions in fundamental frequency, a feature that most recognition systems today discard as irrelevant for speech recognition. Vietnamese is a tonal monosyllable language in which each syllable has only one of six tones. Vietnamese automatic speech recognition (ASR) that integrates tone recognition for large vocabulary continuous speech is only at the beginning phase of development. The results in previous studies [1][2] showed that the tonal phoneme, a phoneme which integrated tone information, and pitch features improved the performance for Vietnamese ASR systems. But in study [2] the tonal phoneme set was created by concatenating the vowel with the tone symbol corresponding to a current syllable based on an available lexicon, and in study [1] the authors just gave some examples to depict the approach used for building it. Therefore it is quite difficult for a foreign researcher who has less knowledge about Vietnamese grammar, to make a tonal phoneme set or even a non-tonal phoneme set from an inputed Vietnamese text. An available grapheme-to-phoneme method could be applied, but generally a grapheme statistic model has to firstly be trained based on an available lexicon before it can be used for converting, and the output will not be obviously all correct. However, Vietnamese is a language that any Vietnamese word can be correctly pronounced even if the speaker does not know its meaning and has never seen it. In this paper an algorithm is proposed as a grapheme-to-phoneme method to convert any Vietnamese word from grapheme-based into a phoneme-based pronunciation that integrates tone information. The output of the algorithm can be both tonal and nontonal phoneme kinds. Similar to studies [1][2] we obtained a better WER on the system using the tonal phoneme set where the improvement is 1.54% compared to the system using the nontonal phoneme set. In this paper we also present a recipe using the Kaldi toolkit [3] to develop Vietnamese LVCSR systems with different kinds of features. The results showed that by using a combined feature of pitch and acoustic feature for the system using the tonal phoneme improved by 4.65% of WER. A process to extract a bottleneck feature which was calculated from a trained deep neural network that relatively improved 10% of WER is also presented. This word was done with the support of National Institute of Information and Communications Technology (NICT) – Japan as the collaboration between NICT and IOIT, and the final system will be incorporated into the VoiceTra [4] application.

This paper is organized as follows: Section II describes the training and testing datasets. The basics of Vietnamese and the algorithm are presented in Section III. Section 0 describes the processes for extracting acoustic features including the bottleneck feature. Section V presents detailed information and steps to train the acoustic models using the Kaldi toolkit. The language models for decoding are presented in Section VI. The experiment results and discussions are described in Section VII. We conclude the paper in Section VIII with the summary of this study.

II. COPORA

A. Training data

- Speech data: Approximately 212 hours of four datasets of 1267 speakers consisting of male and female which are named as IOIT2013, VOV, GlobalPhone, and VoiceTra, were used for estimating acoustic models. Where IOIT2013 was developed by Institute of Information and Technology (IOIT) – Vietnam Academic of Science and Technology and GlobalPhone was developed by Carnegie Mellon University are read speeches, VOV was developed by IOIT is a broadcast audio, and VoiceTra was a dataset that was recorded by the VoiceTra [4] software on various conditions.

978-1-4673-8279-3/15/$31.00 ©2015 IEEE

All of the datas were in the wave format with 16 kHz sampling rate and analog/digital conversion precision of 16 bits.

- Text data: The transcription of speech datasets and a text dataset which is a basic travel expression corpus (BTEC) developed by NICT, were used for training language models with about 291k utterances of 9000 unique words. The detailed information of these datasets is described in TABLE 1.

B. Test data

Two datasets were used for testing, these were BTECTest and VoiceTraTest. BTECTest has 510 utterances with the length is about 19 minutes, and VoiceTraTest has 803 utterances with the length is about 36 minutes.

III. GRAPHEME TO PHONEME

Vietnamese is a monosyllable language. Each word is pronounced as a unique syllable or each syllable is represented by only one word. The total of pronounceable distinct syllables in Vietnamese is 18958, but the used syllables in practice are only around 7000 different syllables [5].

In addition, Vietnamese is a tonal language. Generally there are six tones which would make six different meanings when combining with an individual word as described in TABLE 3. However, there is a national rule to spell each Vietnamese syllable that is considered as a combination of Initial, Final and Tone components. The Initial component is always a consonant, or it may be omitted in some syllables (or seen as zero Initial). There are 21 Initials and 155 Final components in Vietnamese. The Final can be decomposed into Onset, Nucleus and Coda. The Onset and Coda are optional and may not exist in some syllables. The Nucleus consists of a vowel or a diphthong, and the Coda is a consonant or a semi vowel. There are 1 Onset, 16 Nuclei and 8 Codas in Vietnamese. All of these components are listed out in TABLE 4. In order to make a phoneme-based pronunciation dictionary for Vietnamese, we proposed an algorithm to convert any Vietnamese word to tonal phoneme-based pronunciation. The algorithm is presented as below:

TABLE 1. TRAINING DATABASE

Name	Length	Speaker	Utterance	Topic
IOIT2013	170h	206	86k	Open
VOV	20h	91	23k	Stories, news, conversations
GlobalPhone	19.7h	129	19k	Open
VoiceTra	2.6h	841	3k	Open
BTEC	-	-	160k	Open

TABLE 2. EXAMPLE FOR CONVERTING A SYLLABLE "CHUYỄN"

Input syllable	**chuyễn** (movement)			
Structure of a Vietnamese syllable	***Initial***	***Tone***		
		Onset	***Nucleus***	***Coda***
Components	**ch**	**u**	**yê**	**n**
Phoneme output without tone	/c/	/w/	/i̯e/	/-n/
Phoneme output integrated tone	/c/	/w/_3	/i̯e/_3	/-n/_3

Input: Vietnamese text

Output: Vietnamese tonal phoneme-based pronunciation dictionary

Note: using the writing copies of initial, Onset, Nucleus, and coda components in TABLE 4.

Algorithm:

Step 1: Extract the vocabulary from the input text

Step 2: For each word W in the vocabulary do:

2.1. W'=W

2.2: If (Found an initial component from the beginning of W) then {Initial I = "the found initial component"; Remove I out of W} Else {I = "NULL"}

2.3: If (((Found any coda component except "o", "u", "i", and "y" in W) or (Found any coda component that is "o", "u", "i", or "y" and the string length of W is more than 1)) and this found component is the ending of W) then {Coda C = "the found coda component"; Remove C out of W;} Else {C = "NULL"}

2.4: Extract the Tone T of W by finding the code number of W based on the Unicode Table, and remove its symbol out of W; If (The number of letters of "W" is more than one) then:

- If ((The first letter of W is "o" and the second letter of W is not "o") or (The first letter of W is "u" and the second letter of W is not "ô" or "a")) then {Onset O = "first letter of W"; Remove O out of W;}
- If (W is exactly one of Nucleus) then {Nuclei N = W} else {W' is not a Vietnamese word; Go back step 2;}

Else {O = "NULL"; N = W ;}

2.5: G = "I O_T N_T C_T"; Map I, O, N, C and T in G to the phoneme-based marker based on the TABLE 2;

TABLE 2 presents an example for converting a syllable "chuyễn" using the algorithm above. Vietnamese is a tonal language, therefore integrating tonal information into the syllable's phonemes should help to improve the performance for the acoustic models as shown in study [1]. In this work we want to carry on evaluating the performance of a tonal phoneme set which integrated tone symbol and a nontonal phoneme set. For this purpose two dictionaries were created. The first one, a tonal dictionary, was constructed by applying the algorithm on the whole text data. The second one, a nontonal dictionary, was constructed by removing all tone symbols out from the tonal dictionary.

TABLE 3. VIETNAMESE TONES

Name	Tone	Marker	Example	English
Middle level	1	None	Ma	Ghost
Falling	2	`	Mà	Which
Rising	3	´	Má	Mother
Falling rising	4	’	Mả	Tomb
High rising	5	~	Mã	Horse
Low constricted	6	.	Mạ	Plate

TABLE 4. ALL INITIAL, ONSET, NUCLEUS AND CODA COMPONENTS IN VIETNAMSESE

Initials				Onset		Nuclei				Codas	
IPA	*Writing*	*IPA*	*Writing*	*IPA*	*Writing*	*IPA*	*Writing*	*IPA*	*Writing*	*IPA*	*Writing*
/b/	b	/ʂ/	s	/w/	o, u	/i/	i, y	/u/	u	/-p/	p
/m/	m	/c/	ch			/e/	ê	/o/	ô, ôô	/-t/	t
/f/	ph	/ʈ/	tr			/ɛ/	e	/ɔ/	o, oo	/-k/	c, ch
/v/	v	/ɲ/	nh			/ɛˇ/	a (followed by /-k/, /-ŋ/)	/ɔˇ/	o (followed by /-k/, /-ŋ/)	/-m/	m
/t/	t	/l/	l			/i̯e/	iê, ia, yê, ya	/u̯o/	uô, ua	/-n/	n
/tʻ/	th	/k/	c, k, q			/ɯ/	u			/-ŋ/	ng, nh
/d/	đ	/χ/	kh			/ɣ/	o			/-u̯/	u, o
/n/	n	/ŋ/	ng, ngh			/a/	a			/-i̯/	i, y
/z/	d, gi	/ɣ/	g, gh			/ɣˇ/	â				
/ʐ/	r	/h/	h			ă	ă, a (followed by /-u̯/,/-i̯/)				
/s/	x					/ɯ̯ɤ/	ươ, ưa				

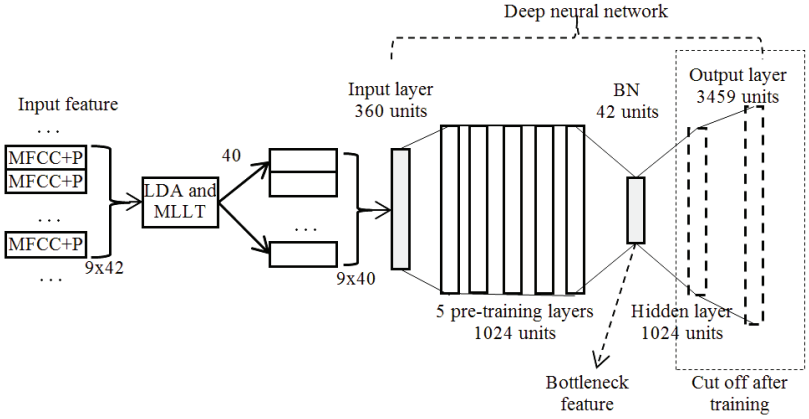

Fig. 1. Process of calculating the Botleneck feature

IV. FEATURE EXTRACTION

In this work, three kinds of acoustic features are used as the input features for estimating acoustic models. The first one was Mel-frequency Cepstral Coefficients (MFCC) which was extracted with a Hamming window of 25ms that was shifted at the interval of 10ms. Each MFCC vector consists of 39 coefficients which are 13 MFCCs, the first and the second order derivatives. The second kind was a combined feature of MFCC and pitch feature (MFCC+P). The pitch feature was a vector consisting of 3 coefficients which are the pitch calculated by Normalized Cross-Correlation [6] method, its first derivative, and the probability of voice for the current frame. The dimension of an MFCC+P vector is 42 (39 for MFCC and 3 for Pitch). Once the MFCC and MFCC+P were adjusted by concatenating 9 neighbor vectors (4 ones for each left and right side of a current MFCC vector) to make the context independent feature, afterward the dimension of the concatenated vector was reduced to 40 by applying a linear discriminate analysis (LDA) and decorrelated with a maximum likelihood linear transformation (MLLT). The third kind was a bottleneck feature (BNF) [7][8] which was calculated through a trained deep neural network. The deep neural network used in this work is a multilayer perceptron network that has 9 layers which are an input layer, 5 restricted Boltzmann machines (RBM) pre-training layers [9], a bottleneck layer (BN), a hidden layer and an output layer. Where the sizes of the RBM, BN, and hidden layers were chosen to be 1024, 40, and 1024 respectively, the size of the input layer was 360 since the input feature was a concatenation vector of 9 neighbor LDA+MLLT outputs as was proposed in [10] to improve the performance of the network, and the output layer's size was 3459 which was equal to the amount of labels of the training data. The training data was forced alignment based on a basic acoustic model of 3459 tri-phone states which was early trained using LDA+MLLT feature on top of MFCC+P feature.

The network was further trained on this labeled data using the Kaldi toolkit. The 5 RBM pre-training layers were firstly trained for 4 epochs and the whole network was then fine-tuned by Stochastic Gradient Descent [11] approach with a learning rate of 0.05, once the two last layers of the network were cut off and the rest of network (input, 5 RBM and BN layers) was used to calculate the BNF by forwarding each input feature vector from the input layer to the bottleneck layer, by the time the output function was the linear function instead of sigmoid function used when training. The topology of the network and the process of calculating the BNF are depicted as Fig. 1. The BNF was also adjusted the same as the strategy for MFCC and MFCC+P before it was used as an input feature for acoustic models.

V. ACOUSTIC MODEL

Five acoustic models (AM) using Gaussian hidden Markov models (GMM-HMMs) were built in order to ascertain the best approach to model Vietnamese acoustic models. All of the models were trained by the same strategy using the Kaldi toolkit. Firstly they were trained as the basic context dependent tri-phone model, subsequently a discriminative training was applied using the maximum mutual information (MMI) to

TABLE 5. EXPERIMENT RESULTS

ID	AM	LM	Feature	Data	Dict	VoiceTraTest			BTECTest		
						OOV	*PPL*	*WER*	*OOV*	*PPL*	*WER*
#1	AM1	LM1	MFCC	IOIT2013+VOV+GlobalPhone	Tonal	31	141	57.48	-	-	-
#2	AM2	LM1	MFCC+P	IOIT2013+VOV+GlobalPhone	Tonal	31	141	52.83	-	-	-
#3	AM3	LM1	MFCC+P	IOIT2013+VOV+GlobalPhone	NonTonal	31	141	54.37	-	-	-
#4	AM4	LM2	MFCC+P	IOIT2013+VOV+GlobalPhone+ VoiceTra	Tonal	0	61	33.75	0	147	22.50
#5	AM4	LM3	MFCC+P	IOIT2013+VOV+GlobalPhone+ VoiceTra+ BTEC	Tonal	0	51	32.85	0	28	10.19
#6	AM5	LM3	BNF	IOIT2013+VOV+GlobalPhone+ VoiceTra+ BTEC	Tonal	0	51	27.73	0	28	9.14

improve the performance. Each of the final models had about 60k Gaussian components for about 4k tied states. The first acoustic model using the MFCC feature, denoted as AM1, was built on the datasets including IOIT2013, VOV and GlobalPhone. Its vocabulary is 5378 which was extracted from the training data's transcription. The dictionary is the tonal dictionary that was created by applying the above algorithm on the vocabulary. The foreign words in the vocabulary set which cannot be transcribed by the algorithm were converted to phoneme-based pronunciation by applying a grapheme-to-phoneme conversion Phonetisaurus [12]. To evaluate how the pitch feature affected the performance of an AM, the second acoustic model (AM2) was trained by using the same training data and dictionary but the feature was MFCC+P. In addition to the effect of the pitch feature we also want to evaluate the effect of the tonal phoneme set to make it comparable to the nontonal one. Therefore another AM, denoted as AM3, was trained the same as AM2 but used the nontonal dictionary. The fourth AM, denoted as AM4, was built by complementing about 2 more hours of data including audio and transcription in the same domain to the VoiceTraTest test set that is the VoiceTra. Since that test set was recorded in poor and different conditions, it contains many kinds of noise, nonhuman sound, vehicle's sound, etc. Based on AM4 we want to find out the gain that a same domain data could help to improve the performance for an AM and a language models. Finally, the last AM (AM5) was trained on all of the training datasets using the tonal dictionary and BNF.

VI. LANGUAGE MODEL

Since the text data BTEC and the transcription of training datasets were drawn and transcribed from different sources and transcribers, these texts are needed to normalize before being used to build language models. All uppercase letters were converted to lowercase. Symbols such as underline, quotes, etc, were removed. Numbers were transcribed into written words corresponding to their spoken syllables. Afterwards three language models were created from those normalized texts. The first one, denoted as LM1, was created from normalized transcripts of VOV, IOIT2013, and GlobalPhone datasets with about 128k utterances and 5300 unique words. The second one, denoted as LM2, was created from transcripts as same as which was used for the LM1 and the VoiceTra's transcripts with about 131k utterances and 5400 unique words. All transcripts and the text data BTEC which had about 291 utterances and 8700 unique words were used to build the third language model (LM3). All of the language models were trigram models and simply trained by using The SRI Language Modeling Toolkit (SRILM) [13] with Kneser-Ney smoothing.

VII. EXPERIMENT RESULTS AND DISCUSSIONS

TABLE 5 shows the performance of experiments for different acoustic models, language models, dictionaries and features as described in the previous sections on two test sets, VoiceTraTest and BTECTest. Three experiments (as shown in line #1, #2 and #3) were implemented using different features and phoneme sets but the same training data and language model to evaluate the effect after applying the tonal phoneme and feature. The results in the line #1 and #2 for VoiceTraTest showed that by combining pitch feature to MFCC improved by 4.65% of WER compared to the system using only MFCC. The tonal phoneme set that was used for the experiment shown in line #2 improved by 1.54% of WER compared to the non-tonal phoneme set shown in line #3. This is similar to the results from previous studies [1][6][14][15] which showed that the tonal feature is useful for improving the performance for not only tonal language but also nontonal language recognition system, and the tonal phonemes which were produced by our proposed algorithm are much effective to improve the performance for Vietnamese speech recognition. In order to improve the performance on the VoiceTraTest set, two more hours of training data, which was produced in the same way as the VoiceTraTest, were complemented to estimate new acoustic and language models. Afterward we decoded for both VoiceTraTest and BTECTest. We obtained an improvement of 19.08% of WER on VoiceTraTest set, and the WER on BTECTest is 22.50%. To improve the performance on the BTECTest set we did another experiment as shown in line #5. The language model LM3 was improved by adding about 160k utterances from the text data BTEC, and consequently the perplexity (PPL) reduced from 61 of LM2 to 51 on VoiceTratest and from 148 of LM2 to 28 on BTECTest. That made improvements by 0.9% of WER on VoiceTraTest and by 12.31% of WER on BTECTest. The last experiment shown in line #6 was to apply the BNF feature on the best language model LM3. The BNF helped to improve by 5.12% of WER and by 1.05% of WER on VoiceTraTest and BTECTest respectively.

VIII. CONCLUSION AND SUMMARY

In this paper we proposed an algorithm for transcribing any Vietnamese word to a tonal phoneme-based pronunciation that improved by 1.54% of WER. A recipe using Kaldi toolkit to build a Vietnamese LVCSR system integrated the tonal model and feature was presented. By applying different features and dictionaries we pointed out that the tonal feature and phoneme are the important components to improve the performance for Vietnamese speech recognition, especially the tonal feature

reduced by 4.65% of WER in this work. The second thing that is necessary to adapt the language model to the test domain, since it is easier to achieve a big gain than improving acoustic models or features. It would be done just by collecting more text corpus in the same domain with test set. We also described a process to calculate the bottleneck feature for Vietnamese which is a state-of-art feature and extracted from a trained deep neural network. In this work it relatively improved about 10% of WER. For the future work we will continue to apply some adaptation training methods on speaker adaptation and feature space, and will draw more text data from online resources to improve the language model, and probably do more experiments using the deep neural network (DNN) or the hybrid of HMM and DNN models to make comparison to the model using bottleneck feature.

REFERENCES

[1]. Van Huy Nguyen, Chi Mai Luong, Tat Thang Vu, Quoc Truong Do, "Vietnamese recognition using tonal phoneme based on multi space distribution," Journal of Computer Science and Cybernetics, IOIT, ISSN 1813-9663, vol 30, No 1, Jan 2014, pp.28-38.

[2]. Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, Van Huy Nguyen, Florian Metze, Zaid A. W. Sheikh, Alex Waibel, "Models of tone for tonal and non-tonal languages," Automatic Speech Recognition and Understanding (ASRU), Czech Republic, Dec 2013.

[3]. Kaldi, http://kaldi.sourceforge.net/

[4]. Shigeki Matsuda, Xinhui Hu, Yoshinori Shiga, Hideki Kashioka, Chiori Hori, Keiji Yasuda, Hideo Okuma, Masao Uchiyama, Eiichiro Sumita, Hisashi Kawai, and Satoshi Nakamura, "Multilingual speech-to-speech translation system: VoiceTra," 14th International Conference on Mobile Data Management (MDM), Italy, 2013, pp.229-233.

[5]. Doan Thien Thuat, Ngu am tieng Viet (Vietnamese Acoustic), Vietnam National University Press, Ha Noi, 2003.

[6]. B.S.Atal, Automatic Speaker Recognition Based on Pitch Contours, Ph.D.Thesis, Polytechnic Institute of Brooklyn, Michigan, 1986.

[7]. Kevin Kilgour, Christian Mohr, Michael Heck, Quoc Bao Nguyen, Van Huy Nguyen, Evgeniy Shin, Igor Tseyzer, Jonas Gehring, Markus Muller, Matthias Sperber, Sebastian Stuker and Alex Waibel, "The 2013 KIT IWSLT Speech-to-Text Systems for German and English," International Workshop on Spoken Language Translation (IWSLT), Germany, Dec 2013.

[8]. Jonas Gehring, Yajie Miao, Florian Metze, Alex Waibel, "Extracting deep bottleneck features using stacked auto-encoders," Acoustics - Speech and Signal Processing (ICASSP), Vancouver – Canada, ISSN 1520-6149, May 2013, pp.3377-3381.

[9]. Geoffrey Hinton, A Practical Guide to Training Restricted Boltzmann Machines, Department of Computer Science, University of Toronto, 2010.

[10]. Shakti P. Rath, Daniel Povey, Karel Vesely, Jan H., "Improved feature processing for Deep Neural Networks," Proceedings of Interspeech 2013, Lyon, 2013, pp.109-113.

[11]. Léon Bottou, "Stochastic Gradient Learning in Neural Networks," Proceedings of Neuro-Nimes 91, EC2, Nimes, France, 1991.

[12]. Phonetisaurus G2P kit, http://code.google.com/p/phonetisaurus/

[13]. The SRI Language Modeling Toolkit, http://www.speech.sri.com/projects/srilm/

[14]. Van Huy Nguyen, Chi Mai Luong, Tat Thang Vu, "Adapting bottle neck feature to multi space distribution for Vietnamese speech recognition," Proceedings of Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA), Phuket-Thailan, Oct 2014.

[15]. Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, Van Huy Nguyen, Florian Metze, Zaid A. W. Sheikh, Alex Waibel , "Models of tone for tonal and non-tonal languages," Automatic Speech Recognition and Understanding (ASRU), Czech Republic, Dec-2013.