STATIC DICTIONARY FOR PRONUNCIATION MODELING

K.Subramanyam[1], D. Arun Kumar[2]
[1] *Assistant Professor, IT Department, R.V.R&J.C College of Engineering, Guntur, A.P, India*
[2]*Assistant Professor, ECE Department, R.V.R&J.C College of Engineering, Guntur, A.P, India*
subramanyamkunisetti@gmail.com, arundhupam@gmail.com

Abstract

Speech Recognition is the process by which a computer can recognize spoken words. Basically it means talking to your computer and having it correctly recognize what you are saying. This work is to improve the speech recognition accuracy for Telugu language using pronunciation model. Speech Recognizers based upon Hidden Markov Model have achieved a high level of performance in controlled environments. One naive approach for robust speech recognition in order to handle the mismatch between training and testing environments, many techniques have been proposed. some methods work on acoustic model and some methods work on Language model. Pronunciation dictionary is the most important component of the Speech Recognition system. New pronunciations for the words will be considered for speech recognition accuracy.

----------***----------

STATEMENT OF THE PROBLEM

Most current leading edge speech recognition systems are based on an approach called Hidden Markov Modeling (HMM). By adding new pronunciations to the Static Dictionary the accuracy can be improved. There are 2 methods to add pronunciations to the static dictionary. First, calculating Levenshtein distance between the strings in the confusion pairs. Second, add the pronunciation by frequency of occurrence.

Automatic Speech Recognition (ASR) or Speech-to-text conversion is a sequential pattern recognition problem. It comprises of three major components- acoustic models, language model and the pronunciation dictionary, which aims to correctly hypothesize a spoken utterance into a string of words. During the training, the system is provided with speech data, the corresponding transcription and a pronunciation dictionary. At the decoding time, the acoustic models and language models trained on the task are used along with one of the standard dictionary (CMUdict) as lexicon. After Decoding is completed, the confusion pairs are used as arguments for Levenshtein Distance algorithm, which gives the maximum number of operations required to convert one string into another. The pair which has minimum distance will be considered for adding to the static dictionary. Later the decoding process will be repeated to find improved recognition accuracy. In the other method the word which has occurred more number of times will be considered for new entry in the static dictionary.

Automatic base form learning:

The simple method of learning pronunciation variants is to learn each word's various pronunciations on a word-by-word basis. Typically a phone recognizer is used to determine possible alternatives for each word by finding a best-fit alignment between the phone recognition string and canonical pronunciations provided by a Static Dictionary.

DICTIONARY REFINEMENT:

Sometimes Dictionary pruning is used to improve the Speech Recognition accuracy. Dictionary pruning is done based on the speech training database. we may arrive 2 types of problems

1. Words that are not included in the data do not have information to be treated with and
2. Some words tend to keep pronunciations that were rarely observed.

To solve the Unobserved words problem, we can use central or summary pronunciations in the pruned Dictionary [3] . The aim of this sort of pronunciation is to capture the phonetic contents included in the set of pronunciation variants of each word and to consolidate them in a reduced pronunciation set.

Enhanced Tree Clustering:

This approach is contrast to decision tree based approach, which allows parameter sharing across phonemes [4]. In this approach a single decision tree is grown for all sub-states. The clustering procedure starts with all polyphones at the root. Questions are asked regarding the identity of the center phone and its neighboring phones. At any node, the question that yields the highest information is chosen and the tree is split. This process is repeated until either the tree reaches a certain size or a minimum count threshold is crossed. Compared to the traditional multiple-tree approach, a single tree allows more flexible sharing of parameters any nodes can potentially be shared by multiple phones.

Present work

We tried with various approaches to improve the speech recognition accuracy for Telugu sentences. The approaches are Calculating distance using Levenshtein distance algorithm and minimum distance variants are added to the Static Dictionary.

1. Addition of the frequently occurring errors.
2. Addition of variant in Language model.
3. Changing the probability.
4. Transcription Modification.

Levenshtein Distance Algorithm:

Levenshtein distance algorithm is to calculate the distance between the variants. The variants which are having minimum distance will be added to the Static Dictionary. Then we can observe the improved accuracy.

Addition of the Frequently Occurring Errors:

In the result file we get confusion pairs with number of times the error was repeated. I took frequently occurring errors and added to the Static Dictionary. Then i got the improved Accuracy.

Addition of Variant in Language Model:

I also tried to include the variant in the Language model also. But i got reduced accuracy. So i did not tried this procedure later.

Changing the Probability:

I tried to change probabilities of the states, because I want to know whether the accuracy will be increased or decreased. But i could unable to open the following files in the folders which are in model_parameters.

(i) means.
(ii) mixture_weights.
(iii) transition_matrices .
(iv) variances.

Transcription Modification:

when I get some type of errors , I modified the Transcription for those words, i observed improved accuracy. The following are the Examples of errors.

MEEREKKADA- MEE
EKKADIKI - EKKADA
HYDERAABAD- MEERAEMI
BHAARATHADESAM-BHAARATHEEYULANDARU

The modified Transcription words are

MEERREKKADA
EKKADAKKI
HHYDERAABAD

BHAARATTHADESAM

In all the approaches, in which i succeeded one main observation is that when I add new variant to the dictionary, there is reduced error rate. Which is contradiction to other papers [3-5]. For all approaches initially I used Praat to eliminate the noise present in the wave files. if we have noise in the wave files I got insertion errors, because of these errors the error rate is increasing. After deleting noise using Praat the error rate is decreased.

Database

The speech database consists of 24 speaker's voice and each speaker spoken 40 sentences.

The database is verified with different times with increased training data. We observed improved accuracy.

TRAINING (number of speakers)	TESTING (number of speakers)	ACCURACY (%)	ERRORS (%)	AFTER DICTIONARY ADDITION	
				ACCURACY(%)	ERRORS(%)
12	12	51.159	85.797	69.420	88.478
16	8	58.370	71.196	76.413	65.000
20	4	59.826	73.696	79.130	65.870

The following are the Results when the wave files are noisy

EXPERIMENT NAME	WORD ACCURACY(%)	ERRORS(%)
50	61.364	59.091
51	88.696	21.739
52	84.348	31.304
53	80.870	65.217
54	93.913	16.522
55	87.826	24.348
56	73.913	68.696
57	96.522	16.522
58	89.565	18.261
59	77.391	61.739
60	88.696	38.261
61	62.609	95.652
62	90.435	21.739

63	93.043	16.522
64	93.043	11.304
65	92.174	26.087
66	81.739	20.870
67	91.304	20.000
68	93.913	12.174
69	66.957	72.714
70	83.478	24.348
71	83.478	36.522
72	76.522	31.304
73	55.000	50.833

After eliminating the noise using Praat the results are as follows

EXPERIMENT NAME	WORD ACCURACY(%)	ERRORS(%)
50	80.870	40.000
51	88.696	19.130
52	86.957	21.739
53	81.739	45.217
54	91.304	18.261
55	92.174	14.783
56	87.826	21.739
57	94.783	12.174
58	93.913	10.435
59	84.348	37.391
60	96.522	10.435
61	73.913	53.043
62	92.174	13.043
63	93.043	12.174
64	92.174	13.043
65	94.696	22.957
66	81.739	20.870
67	93.043	14.783
68	93.913	12.174
69	78.261	51.304
70	80.870	29.565
71	85.217	21.739
72	86.957	19.130
73	93.913	9.565

After addition to the Dictionary the results are as follows

EXPERIMENT NAME	WORD ACCURACY(%)	ERRORS(%)
50	99.130	22.609
51	95.652	12.174
52	98.261	10.435
53	96.522	28.696
54	99.130	10.435
55	97.391	8.696
56	97.391	16.522
57	98.261	8.696
58	99.130	5.217
59	96.522	23.478
60	99.130	7.826
61	94.696	30.739
62	98.261	6.957
63	98.261	6.957
64	96.522	8.696
65	100.000	15.652
66	94.783	7.826
67	99.130	6.957
68	97.391	6.957
69	93.913	36.522
70	93.913	16.522
71	93.913	13.913
72	96.522	9.565
73	99.130	4.348

CONCLUSIONS

The approaches discussed in this dissertation working well. Initially the wave files are noisy. I removed the noise with sound recorder .But sound recorder can remove noise present at both the ends , not in the middle. so I used Praat to remove the noise present in the middle part of wave files . I given these wave files to the Sphinx tool ,i got word accuracy, errors and confusion pairs .These confusion pairs are added to the Dictionary using the approaches discussed in this thesis. finally I observed improved accuracy and decreased errors.

FUTURE WORK

I worked with the database of 24 speakers, each 40 sentences. it's better to work with large database. we can try to achieve 100% accuracy for individual speakers. and also we can try for improving accuracy for large database. Extending the work for Dynamic Dictionary. Try to record wave files in noise less environment.

Levenshtein Distance algorithm

Steps	Description
1	Set n to be the length of s. Set m to be the length of t. If n = 0, return m and exit. If m = 0, return n and exit. Construct a matrix containing 0..m rows and 0..n columns.
2.	Initialize the first row to 0..n. Initialize the first column to 0..m.
3	Examine each character of s (i from 1 to n).
4.	Examine each character of t (j from 1 to m).
5.	If s[i] equals t[j], the cost is 0. If s[i] doesn't equal t[j], the cost is 1.
6.	Set cell d[i,j] of the matrix equal to the minimum of: a. The cell immediately above plus 1: d[i-1,j] + 1. b. The cell immediately to the left plus 1: d[i,j-1] + 1. c. The cell diagonally above and to the left plus the cost: d[i-1,j-1] + cost.
7.	After the iteration steps (3, 4, 5, 6) are complete, the distance is found in cell d[n,m].

REFERENCES:

[1] Gopala krishna Anumanchipalli, "Modeling Pronunciation Variation for Speech Recognition", M.S Thesis , IIIT Hyderabad, February 2008.

[2] Eric John Fosler-Lussier, "Dynamic Pronunciation Models for Automatic Speech Recognition", Ph.d Thesis, University of California, Berkeley, Berkeley, CA, 1999.

[3] Gustavo Hernandez-Abrego, Lex Olorenshaw, "Dictionary Refinement based on Phonetic Consensus and Non-uniform Pronunciation Reduction", INTERSPEECH 2004 -ICSLP 8th International Conference on Spoken Language Processing, October 4-8, 2004.

[4] Hua Yu and Tanja Schultz, "Enhanced Tree Clustering with Single Pronunciation Dictionary for Conversational Speech Recognition", in Proceedings of Eurospeech, pp. 1869-1872, 2003.

[5] Dan Jurafsky, Wayne Ward, Zhang Jianping, Keith Herold, Yu Xiuyang, and Zhang sen, "What Kind of Pronunciation Variation is Hard for Triphones to Model", in proceedings of ICASSP, pp. 577-580, 2001.

[6] Adda-Decker M. and Lamel L, "Pronunciation Variants Across System Configuration", Speech Communication, 1999.

[7] J. M. Kessens, M. Wester, and H. Strik , "Improving the Performance of a Dutch csr by Modeling within-word and cross-word Pronunciation Variation", Speech Commun., vol. 29, no. 2-4, pp. 193–207, 1999.

[8] R. Rosenfeld, "Optimizing Lexical and N-gram Coverage via Judicious use of Linguistic data", in Proc. Eurospeech, (Madrid), 1995.

[9] Sk . Akbar, " Comparing Speech Recognition Accuracy using HMM and Multi-layer Perceptrons ", M.Tech dissertation, June 2008.

[10] kenneth A. Kozar, "Representing Systems with Data Flow Diagrams", Spring, 1997.
http://spot.colorado.edu/~kozar/DFD.html

[11] Finke Michael and Waibel Alex, "Speaking Mode Dependent Pronunciation Modeling in Large Vocabulary Conversational Speech Recognition", in Procedures of Eurospeech, 1995.

[12] M. Ravishankar and M. Eskenazi, "Automatic Generation of Context-Dependent Pronunciations", in Proc. Eurospeech '97, (Rhodes, Greece), pp. 2467–2470, 1997.

[13] T. Sloboda and A. Waibel, "Dictionary Learning for Spontaneous Speech Recognition", in Proc. ICSLP '96, (Philadelphia), pp. 2328–2331, 1996.

[14] A. Xavier and D. Christian, "Improved Acoustic-Phonetic Modeling in Philip's Dictation System by Handling Liaisons and Multiple Pronunciations" , in Proc. Eurospeech '95, (Madrid), pp. 767– 770, 1995.

[15] P. S. Cohen and R. L. Mercer, "The Phonological Component of an Automatic Speech Recognition System ", Reddy , D.R. (Ed) Speech Recognition. Invited Papers Presented at the 1974 IEEE Symposium., pp. 275–319, 1975.

[16] N. Cremelie and J.-P. Martens, "In Search of Better Pronunciation Models for Speech Recognition", Speech Communication, vol. 29, no. 2-4, pp. 115–136, 1999.

[17] B. C. S. and Y. S. J., "Pseudo-Articulatory Speech Synthesis for Recognition using Automatic Feature Extraction from X-ray Data", in Proc. ICSLP '96, (Philadelphia), pp. 969–972, 1996.

[18] I.Amdal, F. Korkmazskiy, and A. C. Surendran, "Joint Pronunciation Modeling of Non-native Speakers using Data-Driven Methods", in Proc. ICSLP '00, (Beijing, China), pp. 622–625, 2000.

[19] M. Bacchiani and M. Ostendorf, "Joint Lexicon, Acoustic Unit Inventory and Model Design", Speech Commun., vol. 29, no. 2-4, pp. 99–114, 1999.

[20] T. Fukada, T. Yoshimura, and Y. Sagisaka, "Automatic Generation of Multiple Pronunciations based on Neural Networks", Speech Commun., vol. 27, no. 1, pp. 63–73, 1999.

[21] S. Greenberg, "Speaking in Shorthand – A Syllable-centric Perspective for Understanding Pronunciation Variation", in Proc. of the ESCA Workshop on Modeling

Pronunciation Variation for Automatic Speech Recognition, (Kekrade, Netherlands, May 1998. ESCA.), 1998.
[22] T. Holter and T. Svendsen, "Maximum Likelihood Modeling of Pronunciation Variation", Speech Commun., vol. 29, no. 2-4, pp. 177–191, 1999.
[23] H. Strik and C. Cucchiarini, "Modeling Pronunciation Variation for ASR: a survey of the literature", Speech Commun., vol. 29, no. 2-4, pp. 225–246, 1999.
[24] M. Riley, W. Byrne, M. Finke, S. Khudanpur, A. Ljolje, J. McDonough, H. Nock, M. Saraclar, C. Wooters, and G. Zavaliagkos, "Stochastic Pronunciation Modeling from Hand-Labeled Phonetic Corpora", Speech Commun., vol. 29, no. 2-4, pp. 209–224, 1999.