International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:6, No:2, 2012

Comparison among Various Question Generations for Decision Tree Based State Tying in Persian Language

Nasibeh Nasiri, Dawood Talebi Khanmiri

***Abstract*—**Performance of any continuous speech recognition system is highly dependent on performance of the acoustic models. Generally, development of the robust spoken language technology relies on the availability of large amounts of data. Common way to cope with little data for training each state of Markov models is tree-based state tying. This tying method applies contextual questions to tie states. Manual procedure for question generation suffers from human errors and is time consuming. Various automatically generated questions are used to construct decision tree. There are three approaches to generate questions to construct HMMs based on decision tree. One approach is based on misrecognized phonemes, another approach basically uses feature table and the other is based on state distributions corresponding to context-independent sub-word units. In this paper, all these methods of automatic question generation are applied to the decision tree on FARSDAT corpus in Persian language and their results are compared with those of manually generated questions. The results show that automatically generated questions yield much better results and can replace manually generated questions in Persian language.

***Keywords*—**Decision Tree, Markov Models, Speech Recognition, State Tying.

I. Introduction

MOST automatic speech recognition systems are based on HMM clustering or state tying of sub-words. This tying improves recognition accuracy when systems are trained with limited data, and is performed by classifying the sub-phonetic units using a series of binary tests based on speech production, called "linguistic questions". Generating questions automatically has a main advantage on human generated questions which is without deep knowledge in phonetic studies for each language, we can produce these questions. The number of complete set of triphones in a standard language is very high so estimation process runs into data insufficiency problem. To counter these, it becomes necessary to group triphones into a statistically estimable number of clusters.

Nasibeh Nasiri is with Information Technology and Computer Engineering Department, Azarbaijan University of Tarbiat Moallem (email:nasiri@azaruniv.edu).

Dawood Talebi Khanmiri is with Electrical and Computer Engineering Department, Islamic Azad University-Bonab Branch, Bonab, Iran (email:dtalebi@gmail.com).

Automatic question generation has several advantages over manual procedures such as: the generation speed, consistent judgment and lower cost.

In this paper first we discuss about important rule-based clustering and then we explain three different approaches for generating linguistic questions. All these methods of automatic question generation are applied to the decision tree on FARSDAT corpus in Persian language and their results are compared with those of manual question generation. The results show that automatically generated questions can be replaced by manually generated questions in Persian language.

II. Phonetic Decision Tree Construction

To achieve good performance in continuous density HMM system, it is necessary to use mixture Gaussian output probability distributions together with context dependent phone models. In practice, this creates a data insufficiency problem due to the resulting large number of model parameters.

Question sets in splitting evaluation function M(q,S) and decision tree for stopping criteria are very important. Assume, S is states of hidden Markov model, L(S) is logarithm Maximum likelihood from training Frames (F) (with this assumption that all states have been tied in S) [1,2].

$$L(S) = \sum_{f \in F} \sum_{s \in S} \log(N(x_f; \mu(s), \sum(S)) \gamma_s(x_f)) \quad (1)$$

For each node which is split into two subsets $s_y(q)$, $s_x(q)$, evaluation function M(q,S) was defined as follows:

$$M(q,S) = L(S_y(q)) + L(S_n(q)) - L(S) \quad (2)$$

Constructing decision tree is as follows [1,2]:

1. Initialization: Start from the elements of the root.
2. Induction: Continue until there is not any untested node. Evaluate M(q,s) for each question in this node , if stopping criteria is met, declare this node as a final node and then go to the step '2', else construct two substituted node that q(n) = max $M(q_t,s)$
   For all elements:
   If the answer is positive: go to the right side,
   If the answer is negative: go to the left side.

3. End: all leaf nodes are final nodes and all clusters have a tree based rule.

III. CORPUS

Small and Big FARSDAT corpus are used to train our models. Small FARSDAT includes 6080 Persian sentences from 304 speakers who each of them uttered twenty sentences and Big FARSDAT includes 70 hours speech from 100 speakers with different Persian dialects. In FARSDAT corpus five microphones were used which one of them was fixed. The other four microphones were altered to increase its data [3].

IV. GENERATING QUESTIONS USING A FEATURE-TABLE

Feature tables are commonly used in phonological descriptions of most languages and are thus readily available. The feature table we are using is a list of sub-word units tagged with their features. Features can be placed on many different tiers according to their classification, e.g., manner, place, voicing, vowel type, vowel height, etc. In Persian language, for example, "ah" is tagged with "Voiced" voice, "Glottal" place, "Stop" manner, "nil" round and "nil" height where "nil" means there is no classification on that tier. The feature table that we used in this research is in Persian language and the features of some phonemes are shown in Table I.

TABLE I
FEATURE TABLE OF SOME PERSIAN PHONES

Phoneme	Consonant	manner	Place	round
Aa	Non_con	Vocalic	Back	Rounded
P	Consonant	Plosive	Bilabial	Nil
Ah	Consonant	Stop	Glottal	Nil
Ch	Consonant	Affricate	Alveopalata	Nil
Je	Consonant	Affricate	Alveopalata	Nil
Sh	Consonant	Fricative	Alveopalata	Nil
Zh	Consonant	Fricative	Alveopalata	Nil

The algorithm for generating linguistic questions is as follows [6]:

1. Read sub-word units and their features.
2. List unique features from each tier.
3. Add a "blank" feature in each tier in order to bypass some tiers in some combinations. In other words, assume that every unit has a "blank" feature in every tier.
4. Group sub-word units according to each feature.
5. Combine each feature from different tiers with "and" and sub-word unit groups according to the features with set intersection.
6. Reject empty, similar-member and "nil" groups.
7. Generate questions according to the groups from cross-tier feature combination.

We use these classes which are 33 numbers as linguistic questions in constructing decision tree.

V. GENERATING QUESTIONS USING MISRECOGNITION

Confusable phone classes are generated using phone substitution errors hypothesized by context-independent acoustic models (phone deletion and insertion are ignored).

Although the idea of using confusable phone classes as phonetic questions appears reasonable, the weak point of this approach is the same as other automatic phonetic question generation systems, namely that the quality of phonetic questions greatly depends on the quality of speech used to generate the questions. In order to fully use confusable phone classes as phonetic questions, a number of techniques are applied to reduce this error. Firstly, all confusable classes are used directly. This is the simplest use of misrecognition. However, this may be risky because of out-of-class misrecognition errors. The second technique is called count-limited misrecognition. This technique comes from the assumption that the number of phones that are misrecognized out of class should be small and if the number of misrecognized phones is less than the threshold, that phone should not be counted in the class. Finally, a "cross constraint" technique is tested. For this technique, only two-way misrecognitions are accepted. For example, if "p" is recognized as "b" and "b" is recognized as "p", "p" and "b" are in the same class. However, if "a" is recognized as "l" while "l" is not recognized as "a", "a" and "l" are not in the same class [7].

After the misrecognition classes are obtained, the class intersections can be generated by combining all of these classes. The concept of generating class intersections is simple. The algorithm starts from a class and cluster with other classes until there is only one member in the class or no more classes left for intersection. To clearly explain this, let us assume that a phone recognizer misrecognized phones as shown in Table1. The number in each cell indicates percentage of recognition a row phone as a column phone. For example, "I" can be recognized as /I/ 94.2%, /E/ 4.95, /U/ 0.563, /O/ 0.0445, /AA/ 0.0148 and /P/ zero percent. Classes from this table are (from each column) {"I", "E", "AA"}, {"I", "E"}, {"I", "AA"} and {"P"}. {"I", "E", "AA"} means "I", "E" and "AA" can be recognized as /I/. An example of misrecognition table for some phonemes is shown in Table 2 which shows the percentage of recognized phonemes for each phoneme.

TABLE II
PERCENTAGE OF MISRECOGNITION IN PERSIAN LANGUAGE

	I	E	AA	P
I	94.2	4.95	0.0148	0
E	5.5	85.6	0	0
AA	0.0218	0	89.1	0
P	0	0	0	87.9

The algorithm to generate the class intersections is as follows.

1. Read each column from the misrecognition table.

2. List all classes with the phones where the number of recognition is higher than a threshold, e.g. from table 1 /I/ {“I”, “E”, “AA”} (threshold = 0), etc.
3. Add a special class called the /A/ class to the class list. This class contains all phones in the table ({“I”, “E”, “AA”, “P”} in this case).
4. Use /A/ class as the root node of the tree.
5. For each column
   5.1. List all classes in the column. This includes /A/ class.
   5.2. For each activated leaf node in the tree.
      5.2.1. Split the node according to class list constructed in step 5.1.
      5.2.2. For each split node
      5.2.2.1. Find the intersection of classes between the node and its parent node.
      5.2.2.2. If the node contains the empty set, deactivate the node.
      5.2.2.3. If the node contains the same phones as any node in step 5.2.1, deactivate the node.
6. All leaf nodes are confusable phone classes.

The results of speech recognition are not completely correct, so we can categorize misrecognized sets. For each phoneme we find all other phonemes that misrecognized in recognition process. Then we find all possible intersections among these sets and finally after eliminating repetitive resulted sets, we use these sets as linguistic questions in constructing decision tree. These classes are very useful ones in clustering technique. The number of questions obtained from this way is 2467. Trees are then pruned down in order to eliminate nodes for which splitting results in the largest increase in likelihood, since this indicates that the child nodes are in reality very dissimilar. Each node in each resulting pruned tree is then used as a phone grouping which forms a linguistic question.

VI. Generating Questions Using Automatic Clustering

The clustering algorithm is a hybrid of the top down and bottom-up clustering techniques. Bottom-up clustering is performed until the number of partitions of the resulting clusters can be exhaustively evaluated, resulting in two maximally-separated clusters. On each of these clusters, the bottom-up clustering is performed as described above, followed by exhaustive partitioning (Fig. 1). Each recursion of this procedure constitutes one step of a top-down evolution of the clustering. The resultant pattern of top-down clusters forms a tree (Fig. 2).

Using the clustering technique described above, the first n/2 states of the CI-phones modeled by n-state HMMs are used to generate right-context questions. Clustering is done separately for each of these n/2 states. This results in n/2 trees. The trees are then pruned down in order to eliminate nodes for which splitting results in the largest increase in likelihood, since this indicates that the child nodes are in reality very dissimilar. Each node in each resulting pruned tree is then used as a phone grouping which forms a linguistic question. Similarly, the last n/2 states are used to generate left-context linguistic questions. This procedure ensures that all phones that are similar in their right most portions are considered as possible groupings for left context questions. Grouping of phones that are similar in the leftmost portions are considered as questions for the right contexts of triphones [4].

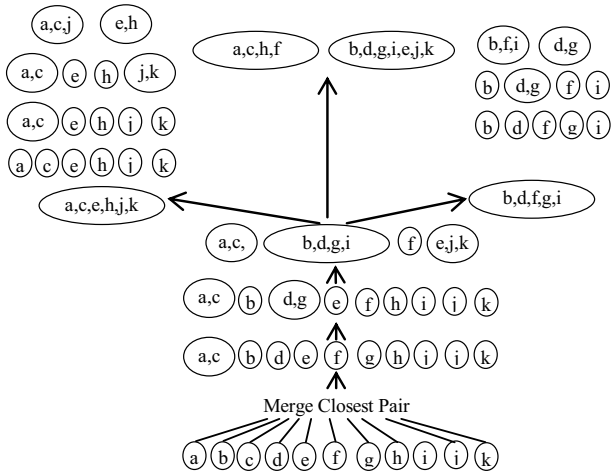

Fig. 1 Hybrid clustering of sample sub-word units

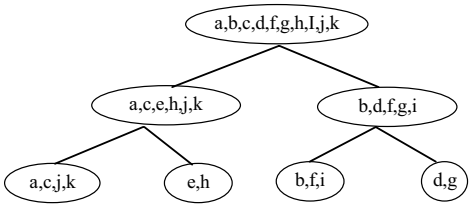

Fig. 2 Resultant top-down evolution of clusters

For example generated questions for the first state of HMM in our experiment are as stated in Table III.

TABLE III
The Question Sets Generated For the First State Using Automatic Clustering

A, AA, E, O, U	G, JE, K
A, AA	G, JE
E, O, U	AH, H, I, L, M, N, Q,
AH, B, CH, D, F, G, H, I,	R, V, Y, Z, ZH
JE, K, L, M, N, P, Q, R,	I, Y, ZH
S, SH, T, V, X, Y, Z, ZH	Y, ZH
B, CH, D, F, G, JE, K, P,	AH, H, L, M, N, Q, R, V, Z
S, SH, T, X	M, V
F, S, SH, X	AH, H, L, N, Q, R, Z
S, SH	AH, H, Q
F, X	AH, Q
B, CH, D, G, JE, K, P, T	L, N, R, Z
B, CH, P, T	L, R

VII. Experimental Recognition Results

All experiments are trained and tested by using the SPHINX system. Three ways to automatically generate phonetic questions have been tested. Furthermore manually generated questions in Persian language by a linguist have been tested. The first technique employs feature table to generate questions of decision tree and the second technique employs misrecognitions to generate classes where each class is assumed to have similar properties and the last technique is a hybrid of the top down and bottom-up clustering techniques. The quality of these questions is proved by the recognition results. The recognition results are presented in figures 3 through 6. Figure 3 shows the percentage of correctly recognized phonemes for various numbers of senones and mixtures using generated questions from feature table.

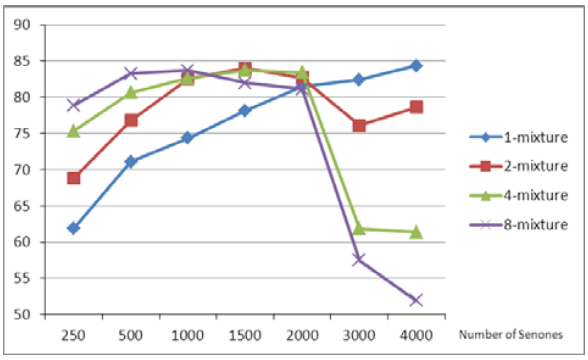

Fig. 3 Recognition results from generated questions using feature table

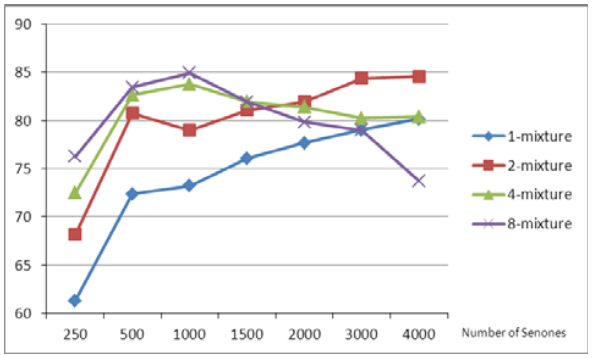

Fig. 4 Recognition results from generated questions using misrecognition

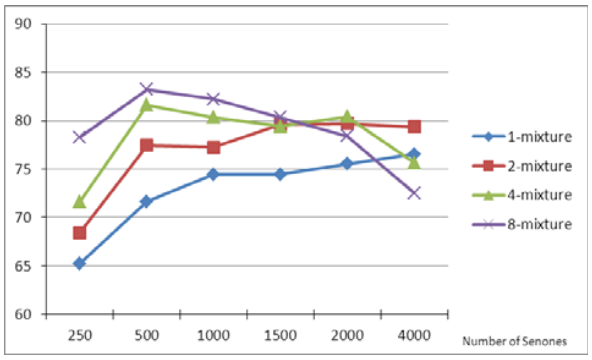

Fig. 5 Recognition results from generated questions using automatic clustering

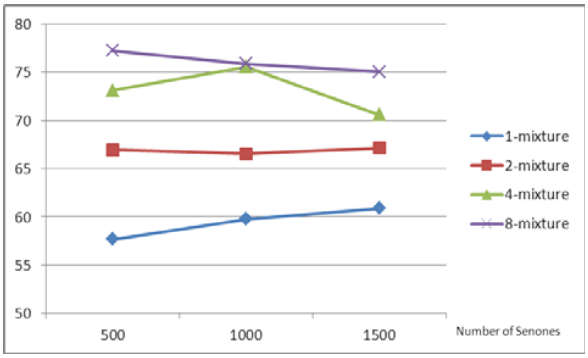

Fig. 6 Recognition results from manually generated questions

Figures 4 and 5 show the same for generated questions using misrecognition table and automatic clustering respectively. The recognition results for manually generated questions are shown in figure 6 for various numbers of senones and mixtures.

We can see from the figures that the accuracy of the models trained from automatically generated questions is better than manually generated questions. These questions are more consistent than handmade questions which rely on judgment of human experts and also more easily implemented in any new language without specific language linguistic knowledge.

VIII. Conclusion

Manually generated questions suffer from human errors and are time consuming and need deep knowledge in phonetic studies for each language. Various automatically generated questions are used to construct decision tree that all these methods of automatic question generation are applied to the decision tree on FARSDAT corpus in Persian language and their results show that automatically generated questions yield much better results and can replace manually generated questions in Persian language.

References

[1] K. Beulen, H. Ney, "Automatic question generation for decision tree based state tying," Proc. Of ICASSP '98, pp. 805-808, 12-15 May, Seattle, WA, USA, 1998.

[2] J. J. Odell, "The Use of Context in Large Vocabulary Speech Recognition", Ph.D. Thesis, Cambridge University, 1995.

[3] M. Bijankhan et al., "FARSDAT – The Speech Database of Farsi Spoken Language", Proc. 5th Australian Int. Conf. On Speech Science and Tech., Vol. 2, perth, 1994.

[4] Singh, R., Raj, B., Stern, R. M.: Automatic Clustering and Generation of Contextual Questions for Tied States in Hidden Markov Models. In Proc. ICSLP, Vol. 1, pp.117-1202, 1999

[5] Kanokphara, S., Geumann, A.,Carson-Berndsen, J.: Accessing Language Specific Linguistic Information for Triphone Model Generation: Feature Tables in a Speech Recognition System., 2nd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, 2005.

[6] Kanokphara, S. and Carson-Berndsen, J.: Automatic Question Generation for HMM State Tying using a Feature Table. Proc. Australian Int. Conf. on Speech Science & Technology (ASST) 2004.

[7] Kanokphara. S. , Carson-Berndsen, J. "Phonetic Question Generation Using Misrecognition , ", In Proc. The Ninth International Conference on TEXT , SPEECH and DIALOGE(TSD), Brno, Czech Republic, September, pp. 407-414, 2006.