ADAPTING PHONETIC DECISION TREES BETWEEN LANGUAGES FOR CONTINUOUS SPEECH RECOGNITION

Nitendra Rajput, L. Venkata Subramaniam, Ashish Verma

IBM India Research Lab, Block I, Indian Institute of Technology, Delhi, Hauz Khas, New Delhi, India

ABSTRACT

In a continuous speech recognition system it is important to model the context dependent variations in the pronunciations of phones. In this work we have attempted to build decision trees for modeling phonetic context-dependency in Hindi. The approach followed is to modify a decision tree built to model context-dependency in American English. The reason the decision trees turn out to be different are that the English and Hindi phoneme sets are not identical. Then even for identical phonemes, the context-dependency is different for the two languages. Linguistic-Phonetic knowledge of Hindi is used to modify the English phone set. Since the Hindi phone set being used is derived from the English phone set, the adaptation of the English tree to Hindi follows naturally. Though here the adaptation is from English to Hindi, the method may be applicable for adapting between any two languages. The decision tree is built using either Hindi data or English data labeled with the correct Hindi contexts. This procedure is discussed and the limitations of both the methods are described.

1. INTRODUCTION

In continuous speech the pronunciation of a phone is heavily dependent on the context. It is important to model the context dependent variations in the pronunciation of phones. For this purpose a phonetic decision tree is used to obtain the acoustic realization of the phoneme context [1]][2][3][4][5]. In [1] the decision tree is constructed with a question set chosen to incorporate linguistic knowledge into the clustering procedure. The advantage with this is that even unseen contexts can be grouped with those, which, one would expect to be linguistically similar. In [2] a method for constructing decision trees for discrete distribution models was developed. The decision tree is constructed in a top-down fashion, with fixed splitting criteria and stopping criteria [3]. Data-based approaches that rely on clustering based question selection have been presented in [4][5]. This approach has the advantage that the phonetic structure need not be well understood to frame the questions.

The authors have not come across work on the building of decision trees for modeling phonetic context-dependency in Hindi. Hindi is a language spoken in large parts of India. In [6] a method for deriving an initial phone model for Hindi is presented. The phone set for Hindi is different from the phone set for English. However it is seen that 40 phones are common between the two languages (there the Hindi phone set has 64 phones and the English phone set 52 phones). This give the hope that a decision tree built for the English language may be adapted for Hindi.

The approach being followed by the authors is that of building a Hindi speech recognition system by bootstrapping from an existing English speech recognition system. This adaptation is especially advantageous in a language like Hindi where there are few existing recognition systems. It allows the use of an existing speech recognition system to build a system for a new language. The Hindi phone models are constructed by bootstrapping from the English phone set as described in [6]. Initial labeling of the Hindi data is done using these models. In this paper the authors construct a decision tree based on this labeled Hindi data by modifying an English decision tree. The knowledge and familiarity gained with the Hindi phone set in building the phone models in [6] is used in the revision of the English question set. In Hindi new classes of phonemes exist that are not there in English, for example, the nasal vowels AAN, AEN, EYN, IYN etc. Questions are added to the English question set for these new classes. Then certain classes are modified by adding new phonemes present in Hindi. Other classes get removed completely.

In Section 2 we describe the question set used in the decision tree. We describe how the contextual questions specific to Hindi are added to the English decision tree and used to modify it. In Section 3 we present the results of classification using the decision tree on continuous Hindi speech data. In Section 5 we briefly describe the process of generation of decision trees. In Section 6 the conclusions and future directions are given.

2. THE QUESTION SET

A binary decision tree is used to assign classes to objects. The context, i.e. the identities of the K previous phones and K following phones in the phone sequence, denoted as P_{-K}, …P_{-1}, P_1, …, P_K, defines the decision rule based on which the tree is constructed. In our experiments K=5. The tree consists of nodes that contain the decision rules and leaves (final nodes) which are labeled with the classes. At each node a binary decision criteria (contextual question) assigns the object to the left or right subtree. When the object reaches a leaf, the class label of this leaf is used as the class for the object. The classes are phonologically meaningful groups of phonemes. Each class consists of a subset of the alphabet of phones.

The English phone set comprised of 52 phones. The existing English question set comprised of 112 questions (pruned from the 130 in [3]). The questions are of the form "does the phone at offset (say 2) from the current phone belong to class A?" Each question is applied to each element P_i of the context. For the context depth of 5 in the forward and backward direction, this leads to a total of 1120 contextual questions. The Hindi phone set we are using from [6] comprised of 64 phones. It comprises of entirely new classes of phonemes like the nasal vowels mentioned in the previous section. Then others like the stressed plosives DHH, DDN, THH etc. also don't exist in English but are close to some phonemes in English. These are added to the already existing classes. For example DHH is added to the class DH. In building the phoneme models for this particular phoneme two English phonemes (DH and HH) were combined in [6]. The grouping in this case follows naturally. Deletion of phonemes from the English question set comes about in an obvious way. Phonemes that do not exist in Hindi are removed. For example the phones IY, IH, IX belonging to a single class in English get removed. Another example is the phone AO that does not appear in Hindi. The class AE, EY, AO in English therefore gets modified to exclude AO.

As described above, to modify the English question set three things were done. One we added entirely new questions and hence new classes, two we modified some of the existing classes to include/remove phonemes and three we deleted certain classes that are not present in Hindi. This revision resulted in a question set for resolving Hindi context dependency. The resulting question set for Hindi consists of 112 questions for each context. Table 1 shows a set of new classes and, hence, new questions that appear in the Hindi question set. In Table 2 a few examples of modified classes in Hindi are given. Table 3 shows some classes that have been deleted from the English question set.

New Questions
AAN AEN AWN AXN EYN IYN OWN UHN UWN ;
UHN UWN;
DHH DXH DDN;

Table 1: Examples of new questions appearing in the Hindi question set.

Modified Questions	
Old Questions	New Questions
DH;	DH DHH;
CH;	CH CHH;
T;	T THH;
F;	F PH;
AE AH EY AO;	AE EY;
AXR ER R;	R;
DX B D G;	B D G;
IX IY IH;	IY IH;

Table 2: Examples of questions that get modified when adapting the questions to Hindi

Deleted Questions
AH;
AO OY;
AXR IX AX ;
EH;
ER;

Table 3: Examples of Questions that appear in the English question set but not in the Hindi question set

The question set has been decided based on linguistic-phonetic knowledge of the Hindi Phone set and its relation to the English phone set from which it was derived. Since the Hindi phone set used in this work is directly bootstrapped from the English phone set, meaningful phonetic classes for Hindi become obvious and the modification of the English question set straightforward. New classes in Hindi like the nasal vowels naturally lend themselves to asking new questions.

Another important concept to keep in mind when building the tree is that of garbage phones. The phones that get deleted (don't exist in Hindi but exist in English) are called *garbage phones.* In the absence of a method for labeling Hindi speech, this allows the use of English data to construct the decision trees for each of the Hindi phones. In the next section we describe two ways of building decision trees for each phone in Hindi; using labeled Hindi data and using labeled English data with appropriate mapping to the Hindi space. When using English data, phonemes like AO, AXN that do not appear in Hindi will also be present. The Hindi question set does not have questions corresponding to these. If the current phone is a garbage phone we disregard it and do not proceed with building the decision tree for it. However if it is one of the preceding or following phones, we mark it and debar it from participating in the questioning.

3. CONTEXT DEPENDENT DATA LABELLING

Since we use a K-phone context model to build the tree for each arc/phone, feature vectors are required that have been aligned to the Hindi phone set. Each of these vectors also need to have K previous and K next phone contexts on which the questions are to be asked. Context dependent labeled data can't be obtained for the language which does not have a recognition system. Isolated phone labeled data can be generated by asking the speakers to utter the isolated phonemes and then they can be labeled manually. To generate the context dependent labeling is not practically possible. We present two methods to label the data to the Hindi phone set. Although, these approaches produce approximate phone boundaries, the data so generated can be used as input to the tree generation algorithm in the first iteration.

In the first method, we generate the alignments for continuous speech in English language using the English recognition system. This produces the exact alignment of the feature vectors with the English phone set as the English recognition system is well trained for English speech. After the vectors are aligned, each vector has (1) a phone-id to which it has been aligned and, (2) the phone contexts of this vector. These phone contexts and the phone-ids are in accordance with the English phones. So we use the mapping described in [6], to convert the phone-ids and the phone contexts from the English phone set to the Hindi phone set. The vectors that are assigned to phones in English which do not occur in Hindi are all mapped to a garbage phone. No tree is built for this garbage phone. Same mapping is applied to the ids existing in the phone context of each feature vector. Although the vector is of English speech and it represents the acoustics of the English language, the mapping creates the closest Hindi phonetic contexts. This way we can have each vector being represented by a Hindi phone-id and having a Hindi phonetic context.

Another way to generate the labeled data in Hindi is to use the novel language data labeling technique as described in [6]. This needs Hindi continuous speech data and uses the English recognition system to align the Hindi vectors. Using the lexeme contexts in the two languages we generate the Hindi data labeled to the Hindi phone set.

These methods produce alignments that may not represent, both, the contexts and the corresponding feature vectors exactly. Building a tree from the first method may not result in the best models as none of the feature vectors in this method are from the Hindi language and so they may not be able to model any characteristically new sound of the language in the vector space. However the alignments give the exact phone boundaries, as it is a trained system for the English data. On the other hand the second method uses Hindi data and so the feature vectors do cover the acoustic space of the Hindi language. But aligning Hindi speech with the English recognition system does not give the exact phone boundaries. So the alignment generated by this method is not very accurate. We use both the methods to generate the labeled data and build trees on the two different sets of data.

Hindi data was collected for 20 speakers having a total of 2000 utterances totaling around 6 hours of continuous Hindi speech. The Hindi phonetic vocabulary contains 900 Hindi words. The English data consisted of 3000 utterances of 30 different speakers. The English training sentences are chosen from a vocabulary consisting of 96,000 words. This constituted about 7 hours of continuous speech.

4. BUILDING THE DECISION TREES

Once we have the set of questions to ask and the labeled data, we ask these questions to all the vectors of a particular phone to build the tree. We choose the best question at each step by asking all the questions to all the contexts of all vectors in the phone. The question that gives the best split is taken as the best question for that stage [3]. Iteratively we prune down the tree and we stop when we get the terminal leaves. We create Gaussian mixture models for each leaf in the tree to model the vectors in each leaf. These models represent the context dependent models for the new language using the English language recognition system.

5. CONCLUSIONS

We have described an effective means of building decision trees for Hindi. The procedure followed is that of modifying the English contextual questions based on Linguistic-Phonetic knowledge. Once the question set is fixed, we present ways of generating Hindi or English data labeled with the correct Hindi contexts for the construction of the decision trees. Questions are then asked on this set of data for building the trees for each of the phones. We then generate context dependent models for the Hindi phone set. These can be used iteratively to further generate the labeled data and the tree can be refined. Viewed in conjunction with the companion paper [6] the first steps in building a Hindi recognition system by bootstrapping an existing English recognition system is presented.

The main focus in the near future will be

- Refine the tree by further iterations,
- Build a language model for Hindi speech,
- To build a complete Hindi speech recognition system by iteratively refining the steps discussed in this paper .

6. REFERENCES

1. J. J. Odell, *The use of context in large vocabulary speech recognition*, PhD Thesis, Cambridge University, 1995.
2. L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo and M. A. Picheny, "Robust methods for using context-dependent features and models in a continuous speech recognizer," *Proceedings ICASSP*, Adelaide, Australia, pp. 533-536, 1994.
3. L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo and M. A. Picheny, "Decision trees for phonological rules in continuous speech, " *Proceedings ICASSP*, Toronto, Canada, pp. 185-188, 1991.
4. R. Singh, B. Raj and R. M. Stern, "Automatic clustering and generation of contextual questions for tied states in hidden Markov models, " *Proceedings ICSLP*, 1999.
5. K. Beulen and H. Ney, "Automatic question generation for decision tree based state tying, " *Proceedings of ICASSP*, Vol 2, pp. 805-809, May 1998.
6. N. Mukherjee, N. Rajput, L. V. Subramaniam, A. Verma, “On deriving a Phoneme model for a new language,” *Proceedings ICSLP*, Beijing, China, 2000.