AUTOMATIC MODELING OF USER SPECIFIC WORDS FOR A SPEAKER INDEPENDENT RECOGNITION SYSTEM

Jacques Duchateau and Dirk Van Compernolle

K. U. Leuven - E.S.A.T., Kardinaal Mercierlaan 94, B-3001 Heverlee, Belgium
Email: Jacques.Duchateau@esat.kuleuven.ac.be

ABSTRACT
The problem addressed in this paper, is the incorporation of user specific words in a speaker independent speech recognition system. No transcription is used to model the new words, modeling is based on a very small number of training utterances only.

We investigated two different modeling methods. The first is intended for small vocabulary recognisers. The HMM models for the new words are enhanced by averaging their states with the nearest speaker independent state. This way, the recognition error was reduced by a factor two, and even the noise robustness of the speaker independent models seems to be transferred to the new models. The second method can be used in large vocabulary recognisers. Using a CSR algorithm, a transcription for the new words is found in terms of the subword models in the recogniser. The resulting models perform equally well as the models based on phonetic transcriptions.

1. INTRODUCTION
The vocabulary of a speaker independent speech recognition system, whether it is small or large, is limited: it is always possible that an application user wants to add his own vocabulary, unknown to the recogniser (proper names, technical terminology). This paper describes how these new words can be modeled without phonetic or textual transcription, but based on three or less example utterances per word and using the given speaker independent models.

The solution for this problem depends on the type of speaker independent recogniser the user specific words are added to. We distinguish two different recogniser types.

- Small vocabulary recognisers
- Large vocabulary recognisers

In this case, the situation is just the opposite. We need a concatenation of existing subword models (e.g. phoneme models) for each new word, but now the models are (contextually) rich enough to do the job. A system based on this idea is described in section 3.

2. ENHANCED TRAINING IN SMALL VOCABULARY RECOGNISERS

2.1. The modeling method

It is well known that default HMM training with too few training data results in poor models: the parameters in the HMM can’t be estimated well, and the state boundaries don’t shift sufficiently during training, they get stuck in their initial positions. The main idea behind this method is to improve these poor models using the well trained speaker independent models, and transferring their good properties (well estimated parameters, robustness against different SNR values, ...).

The method is summarised as follows:
1. Train an HMM word model for each user specific word.
2. Find for each of these trained states the nearest speaker independent state.
3. Replace each trained state by the average of itself and that nearest speaker independent state.
4. Add the modified model for the user specific word to the recogniser.
For the second step in this modeling method, a measure of distance between states is needed. We defined the nearest speaker independent state as the state, that has the highest probability for emitting the labels aligned to the trained state. The according measure of distance for states of discrete HMMs is determined as follows.

Suppose there are C codebooks with \( L_i \) labels \( l_{ij} \) (\( i = 1 \) to \( C \), \( j = 1 \) to \( L_i \)). Denote the output probabilities in the speaker independent state as \( p_{ij} \) and those in the trained state as \( P_{ij} \). Then, with \( N \) the number of frames aligned to the trained state, the number of occurrences of label \( l_{ij} \) in those frames, is \( N \times P_{ij} \). So, the total probability for the speaker independent state to emit all the labels in the frames aligned to the trained state, is given by

\[
\prod_{i=1}^{C} \prod_{j=1}^{L_i} p_{ij}^{N \times P_{ij}}.
\]

Taking the logarithm, this gives

\[
N \times \sum_{i=1}^{C} \sum_{j=1}^{L_i} (P_{ij} \times \log(p_{ij})),
\]

where \( N \) can be dropped if one looks for the emission probability per frame. The resulting distance thus is the inner product of the probabilities in the trained state, and the logarithm of the probabilities in the speaker independent state. As the distance problem involves two different types of states, it is not strange that we get an asymmetric distance measure. Note that this distance measure only applies for discrete density HMMs, which is the state-of-the-art approach for HMM-based small vocabulary recognition systems.

2.2. Experimental results

2.2.1. Baseline speaker independent system

Characteristics of the baseline speaker independent isolated word recognition system, fully described in [1]:

- LPC-cepstrum based signal processing
- Multiple VQ with 4 codebooks
- Discrete density HMMs
- Word models with 10 states

Database for the baseline system:

- 8 kHz, recorded over a telephone line
- 500 speakers for training, 80 for tests
- Vocabulary of 23 Dutch words, including the ten digits

This baseline recognition system gives a 3.4 % recognition error.

Note that, although there exists a large difference in recording conditions between the train and test databases (telephone line versus office or car environment, see below), we used in our experiments a signal processing without normalisation step for noise or for channel effects. Including such step can improve the results considerably.

2.2.2. The 10 word task

The database with user specific words was recorded in an office by one speaker. Each of ten Dutch words is recorded 40 times: 20 utterances with a SNR of about 25 dB (\( \text{clean} \)) and 20 utterances with a SNR of about 10 dB (\( \text{noisy} \)). For this database, the test corpus always consists of the utterances that are not used for training. The test corpus thus depends on the experiment.

Experiments with different weighting factor for trained and speaker independent models showed that values between 25 % and 75 % are equally good. For all reported results, the weighting factor was set to 50 %. For both trained and averaged models, an optimal value for the minimal probability \( \epsilon \) for all labels in the models was used.

In table 1, the recognition results (error rate) for the 10 user specific words in the 33 (23 + 10) word recogniser are given, comparing trained models with averaged models. Results are given for three different training sets with a different number of utterances per word. One can see that by averaging, the error reduction for the user specific words is larger when only \( \text{clean} \) words are used as training data: 70 % reduction in both cases, compared with 30 % if also a \( \text{noisy} \) utterance is given.

<table>
<thead>
<tr>
<th>Training utterance(s):</th>
<th>trained models</th>
<th>averaged models</th>
</tr>
</thead>
<tbody>
<tr>
<td>One ( \text{clean} ) utterance</td>
<td>14.9 %</td>
<td>4.4 %</td>
</tr>
<tr>
<td>Three ( \text{clean} ) utterances</td>
<td>4.6 %</td>
<td>1.6 %</td>
</tr>
<tr>
<td>One ( \text{clean}, \text{one noisy} )</td>
<td>4.5 %</td>
<td>3.2 %</td>
</tr>
</tbody>
</table>

Table 1: Results for the 33 word recogniser

Also the results for the speaker independent words in the combined 33 word recogniser are important. The newly trained models didn’t influence these results, but using the averaged models (which are nearer to the speaker independent models), the error rate for the speaker independent words increased from 3.4 % for the baseline system to 3.5 %.

2.2.3. Method validation task

We checked the results found with our 10 word task on an other, larger database. This database consists of
20 different words, 15 names and 5 Dutch nouns. All recordings are made in a car, with the engine on, but at a different speed. The SNR varies from 2 to 29 dB (with the highest SNR in a stationary car, these utterances will be called clean, the others noisy). Each word is recorded by 10 speakers in at most 11 situations, resulting in a database of about 1700 utterances. As for the first database with user specific words, the test corpus consists of the utterances that are not used for training, it thus depends on the experiment.

In table 2, the recognition results (error rate) for the 20 user specific words in the 43 (23 + 20) word recogniser are given, averaged over all 10 speakers. Again, the results are given for different training sets and compared between trained models and averaged models.

<table>
<thead>
<tr>
<th>Training utterance(s):</th>
<th>trained models</th>
<th>averaged models</th>
</tr>
</thead>
<tbody>
<tr>
<td>One clean utterance</td>
<td>42.0 %</td>
<td>23.1 %</td>
</tr>
<tr>
<td>Three clean utterances</td>
<td>29.6 %</td>
<td>18.9 %</td>
</tr>
<tr>
<td>Three noisy utterances</td>
<td>27.3 %</td>
<td>14.5 %</td>
</tr>
<tr>
<td>One clean, one noisy</td>
<td>17.8 %</td>
<td>13.7 %</td>
</tr>
</tbody>
</table>

Table 2: Results for the 43 word recogniser

One can see that similar improvements are found as for our 10 word task. The error reduction by averaging the models is again larger when the training data consists of only one noise type (or clean, or noisy): for the first three cases, the improvement is about 40 %, for the last only 20 %. Note that although the experiment with one clean and one noisy utterance gives the best result, this option should be avoided in practice because it requires a recording in a stationary car and an other recording in a moving car. So adding a user specific word becomes difficult for an application user, and our first reason for restricting the number of utterances per word is to make a user-friendly system.

We can conclude from the experiments with both recognition tasks in this section that the trained word models for the user specific words can be improved considerably by averaging their states with the nearest well trained speaker independent word model state. Since the error reduction is about two times larger for systems using a train set that only consists of utterances with one SNR level than for systems using a train set with both clean and noisy utterances, we can say that by averaging, the noise robustness of the speaker independent models in the given small vocabulary recognizer has been transferred to the new user specific models.

3. TRANSCRIBING NEW WORDS IN LARGE VOCABULARY RECOGNISERS

3.1. The modeling method

Different algorithms can be used to find a good transcription for the new words in terms of the speaker independent subword models. In [2], a tree search algorithm based system finds one optimal concatenation for all given utterances at a time. The use of a Continuous Speech Recognition (CSR) algorithm is a more simple possibility, but this generally results for each given utterance in a different concatenation. In [3], only the best of these concatenations is used in the recogniser. As we only use three or less utterances per word, we decided to put one model in parallel for each utterance. A user specific word is then recognised if one of its models has the best score, no matter which score its other models have.

Our CSR modeling method is summarised in the next algorithm steps:

1. Select a set of speaker independent subword units.
2. Use CSR to find for each utterance of each user specific word a sequence of subword units.
3. Add each of these new transcriptions as model for the new word to the recogniser.

Note that the CSR, implemented following the Dynamic Programming Algorithm described in [4], comes with a variable sequence length. A Model Transition Cost (MTC) for starting a new model was added to the algorithm to be able to influence (but not set) the number of units in the sequences.

3.2. Experimental results

3.2.1. Baseline speaker independent system

Characteristics of the baseline speaker independent isolated word recognition system:

- Mel-cepstrum based signal processing with channel adaptation
- Semi-continuous density HMMs (described in [5])
- 40 context-independent phoneme models with 5 states

Database for the baseline system:

- 16 kHz, recorded in different locations (classrooms, offices, pubs, ...), the SNR ranges from 5 to 35 dB.
- 440 speakers for training, 105 for tests
- Vocabulary of 77 Dutch words, including the ten digits, a command set and a set of phonetically balanced words

This baseline recognition system gives a 2.0 % recognition error.
3.2.2. The 10 word task

A new database was recorded with the same 10 Dutch user specific words as for our experiments with the small vocabulary system. Due to the more silent office and a better microphone, the SNR of these recordings is about 35 dB. Each of the words is recorded 43 times: 3 utterances used for modeling only, and 40 utterances used for testing.

The 40 context-independent phoneme models were chosen as subword unit set. When we use the correct phonetic transcriptions for the new words, the error rate for the user specific words in the combined 87 (77 + 10) word recogniser is only 0.25% (1 error on all 400 testing utterances). The error rate for the speaker independent words increases from 2.0% in the baseline experiment to 2.2% in this combined recogniser. This result is worse than for the user specific words due to the difference in SNR in both databases.

To find automatic transcriptions for the new words with the CSR algorithm, we first had to fix an optimal MTC. We therefore did a phoneme recognition experiment with different values for the MTC on our baseline 77 word database and chose that value that gave the lowest phoneme recognition error (sum of insertions, deletions and substitutions). With this MTC in the CSR algorithm, we found all automatic transcriptions for the new words.

First we did some experiments adding only one transcription for a new word to the baseline recogniser. Since we have three utterances per word for training, we can do this experiment three times with the three different utterance sets. We found an error rate for the user specific words of 1.75%, 0.75% and 4.50% respectively, an average of 2.3%. But using the first and the last set, all errors were due to only one of the 10 new words. Using only one automatic transcription per word thus is risky, an automatic transcription can go wrong for a word.

Therefore we tried with two transcriptions per new word, the global recogniser thus had 97 (77 + 20) models in parallel for 87 words. This way, the recognition error reduces to 0.2%, as for the phonetic transcriptions.

The error for the speaker independent words increases from 2.0% for the baseline system to 2.1% when 10 models (one per new word) are added, and to 2.2% with 20 additional models (two per new word). This is also the same result as for the system with new models based on phonetic transcriptions.

We can conclude from these experiments that, using the current context-independent phoneme models, it is advisable to use two utterance transcriptions instead of one to model a new word, this in order to decrease the risk for badly modeled words. With two models per word, we get the same results as with the phonetic transcriptions. But it is probable that in future, with enhanced speaker independent models (e.g. context-dependent phoneme models), one transcription per word will suffice.

4. CONCLUSION

In this paper, two methods for modeling user specific words in a speaker independent recognition system are proposed. The first proved to be a good method to enhance trained models for new words in small vocabulary recognisers, transferring the noise robustness from the speaker independent models to the new models. The second gives promising results for adding new words to a large vocabulary recognition system. In future, this system will be improved using context-dependent phonemes and it will also be tested in continuous word recognition circumstances.

5. REFERENCES


