MULTILINGUAL WEIGHTED CODEBOOKS

Martin Raab[1,2], *Rainer Gruhn*[1,3], *Elmar Noeth*[2]

[1]Harman Becker Automotive Systems, Speech Dialog Systems, Ulm, Germany
[2]University of Erlangen Nuremberg, Dept. of Pattern Recognition, Erlangen, Germany
[3]University of Ulm, Dept. of Information Technology, Ulm, Germany
mraab@harmanbecker.com

ABSTRACT

In this paper we present an approach for speech recognition of multiple languages with constrained resources on embedded devices. Examples of such systems are navigation systems, mobile phones and MP3 players. Speech recognizers on such systems are typically to-date semi-continuous speech recognizers, which are based on vector quantization.

Typical vector quantization algorithms can only generate vector quantization prototypes that are optimal for one language. We hypothesize and provide evidence that a certain fixed vector quantization is responsible for a significant drop of recognition performance when a recognizer is extended to recognize multiple languages at the same time.

This paper proposes an algorithm for the construction of Multilingual Weighted Codebooks (MWCs). These MWCs have the advantage that they offer significantly improved performance for the recognition of multiple languages.

Index Terms— multilingual, codebook, semi-continuous

1. INTRODUCTION

For speech recognition in car navigation systems, multilinguality is a great challenge. Place names, commands and other words in the user's main language must be recognized with maximum possible accuracy, but also words in other languages should be recognized well e.g. when driving abroad. Therefore we want to have a system that performs as well as possible for all languages under the constraint of keeping monolingual performance in the main language. To-date similar approaches [1, 2] have not addressed the constraint of conserving the main language performance when traversing from a mono- to a multi-lingual system.

For our aims we believe that the codebook generation in the training of semi-continuous speech recognizers is not optimal. To motivate our belief, it is necessary to analyze what is happening when the typical algorithms for codebook generation are applied in this multilingual scenario. We are showing this using the example of the LBG algorithm [3], a typical algorithm for codebook generation.

The aim of the LBG is to find a limited number of Gaussian prototypes in the feature space that cover the training data as well as possible. In this multilingual scenario, we have two options. Either we provide only main language training data to the LBG algorithm, or we provide data from all languages to it. In the first case the codebook is only optimized for the main language, not considering the performance on the additional languages. In the second case the codebook is optimized for all languages without prioritizing the main language.

Therefore we propose a new algorithm for the construction of a multilingual codebook. The first step is the construction of a codebook for each language. For this we use soft vector quantization based on the LBG approach. We then create a new codebook from these initial codebooks. As this new codebook is based on codebooks from many languages, we call it multilingual, and as the influence (namely the number of codebook vectors) of each original codebook can be adjusted, we call it weighted.

The remainder of this paper is organized as follows. Section 2 describes the baseline architecture that we use to train recognizers for multiple languages. Section 3 explains how MWCs are constructed from initial codebooks. Section 4 and 5 describe our experimental setup and show the results. Finally, a conclusion is drawn and suggestions for future work are made.

2. BASELINE SYSTEM

We start with a well trained monolingual semi-continuous HMM speech recognizer. While keeping the main language generated codebook constant, for each additional language we do the following.

- Add all additional language HMMs to the recognizer
- Train these additional HMMs with training data from the corresponding language, not changing the codebook

Finally, we have a system with trained HMMs for all languages. In the introduction we already stated why we believe that the main language codebook is not optimal for our scenario.

3. EXTENDED SYSTEM

This section introduces the multilingual MWC system. The only difference to the baseline system is that we replace the main language codebook with an MWC before we train the HMMs. The MWC design process is presented next.

3.1. MWC algorithm

To achieve our first priority aim of keeping the main language accuracy, the MWCs we create will always contain all Gaussians from the main language codebook. Furthermore, we will never modify any of the Gaussians that originate from the main language. To improve the performance on the additional languages, the MWCs will also contain some Gaussians which originate from codebooks from the additional languages.

Thus, our MWC is basically the main language codebook plus some additional Gaussians. Figure 1 depicts an example for the extension of a codebook to cover an additional language. From left to right one iteration of the generation of MWCs is represented in a simplified two dimensional vector space.

The picture to the left shows the initial situation. The Xs are mean vectors from the main language codebook, and the area that is roughly covered by them is indicated by the dotted line. Additionally, the numbered Os are mean vectors from the second language codebook. Supposing that both Xs and Os are optimal for the language they were created for, it is clear that the second language contains sound patterns that are not typical for the first language (Os 1,2 and 3).

The middle picture shows the distance calculation. For each of the second language codebook vectors, the nearest neighbor among the main language Gaussians is determined. These nearest neighbor connections are indicated by the dotted lines.

The right picture presents the outcome of one iteration. From each of the nearest neighbor connections, the largest one was chosen as this is obviously the mean vector which causes the largest vector quantization error. In the pictures, this is O number 2. Thus, the Gaussian O number 2 was added to the main language codebook.

The iteration described above will already lead to a reduced vector quantization error for utterances from the second language. Further iterations would further minimize this error.

3.2. Distance Measures

A key element for the algorithm described above is the distance measure. In the literature two measures are frequently used to determine distances between multi-dimensional Gaussians. These are the well-known Mahalanobis distance and Kullback-Leibler divergence.

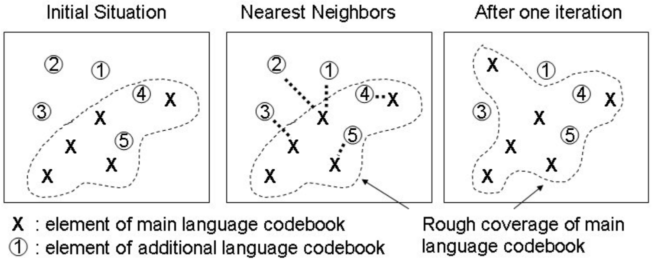
Initial Situation
Nearest Neighbors
After one iteration
X : element of main language codebook
① : element of additional language codebook
Rough coverage of main language codebook

Fig. 1. Basic Idea of Multilingual Weighted Codebooks

Table 1. Testset description (Utterances/Grammar)

	German	English	Italian	French	Spanish
#U.	2005	852	2000	3308	5143
#G.	2498	500	2000	2000	3672

In addition to these distances, we also experimented with a distance measure that tries to minimize the gain in variance when two Gaussians are merged. We called this distance measure Log Variance Minimization (LVM).

In the LVM distance, as well as in some experiments, we refer to merging of Gaussians. When merging two Gaussians, we replace these two by one Gaussian that would have been estimated from all the training samples that led to the estimation of the two original Gaussians. This can be done without the need to actually know all the training samples, a formula can be found in [4].

4. EXPERIMENTAL SETUP

The Harman Becker semi-continuous speech recognizer uses 11 MFCCs with their first and second derivatives per frame and LDA for feature space transformation. The codebook contains Gaussians with full covariance. In order to concentrate on the acoustic effect of the changes to the codebook, no statistical language model was used. All recognizers are trained on 200 hours of Speecon data [5].

Information about test sets for all languages is given in Table 1. The first row contains the number of test utterances, the second row the number of different entries in the grammar used. All utterances are city names. For each test language we created a codebook with 1024 Gaussians as input for our MWC algorithm. The German 1024 Gaussian codebook is also the baseline codebook. For non-native experiments, two further test sets are used. 500 sentences from the German accented English part of Verbmobil [6] and 1100 sentences from the ISLE corpus [7].

5. EXPERIMENTS

We performed four sets of experiments. First we determine which of the distance measures we propose leads to good re-

Table 3. Effect of Codebook Size (Word Accuracies)

1024 + US. Gaussians	**0**	**26**	**76**	**126**	**176**	**276**	**1024**
US Cities	67.3	67.8	69.3	72.0	72.5	73.4	74.5

Table 4. Effects on Non-Native Speech (Word Accuracies)

1024+US. Gaussians	**0**	**26**	**76**	**126**	**176**	**276**	**1024**
Verbmobil	79.1	78.2	77.2	78.0	77.3	79.4	79.2
ISLE	80.9	81.8	81.7	82.7	81.1	81.0	82.8

sults. Second, we analyze how the codebook size increase relates to the performance gain. Third, we evaluate on non-native speech. These first three sets of experiments are performed on a bilingual setting with German as main language and English as additional language. Our final set of experiments will then evaluate the effect on a truly multilingual system with five languages.

5.1. Distance Measure Evaluation

We made some initial experiments to determine which of the distances we proposed performs best. Each of the three distances, LVM, Mahalanobis distance and the Kullback-Leibler divergence was evaluated on the English test set when 176 and 276 English Gaussians were added to the German codebook with 1024 Gaussians. The results are given in Table 2.

Table 2. Evaluation of Distance Measures (Word Accuracies)

Codebook	US Cities	%Gain
- / 0	67.3	0.0 %
LVM / 176	70.3	4.6 %
KLD / 176	72.4	7.7 %
MAH / 176	72.5	7.9 %
LVM / 276	73.0	8.6 %
KLD / 276	72.8	8.2 %
MAH / 276	73.4	9.1 %

All three distance measures are a reasonable choice for Gaussians, but for our application the Mahalanobis distance achieved the best results. Therefore we will use the Mahalanobis distance for the following experiments.

5.2. Codebook Size Determination

The recognition performances with different sizes of MWCs are shown in Table 3. The baseline, when only the German Codebook is used is the leftmost score. The more English Gaussians are added (as indicated in the top row), the better the recognition score gets.

The nice aspect of these results is that large parts of the maximum possible improvement can already be achieved when the codebook size increase is rather small. This is made visible in Figure 2.

Obviously, the larger the final codebook can be, the better the score. The results from this section are a guidance for setting codebook sizes for our final results on five languages.

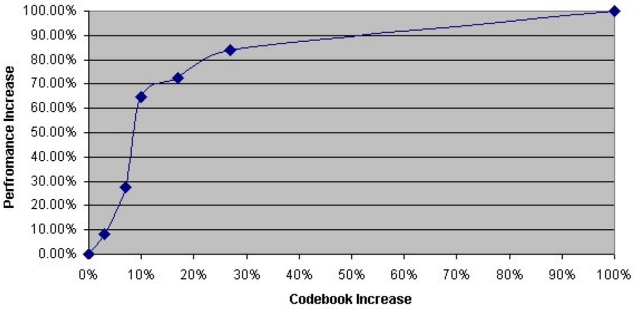

Fig. 2. Relative Improvements on English Data with MWC

5.3. Examination of Non-Native Speech

Our previous experiments have already shown, that MWCs help for the recognition of multiple languages. However, non-native speech is known to be significantly different from native speech. Therefore, in this section we evaluate English speech spoken by Germans. The results are given in Table 4, which has the same format as Table 3.

The results are not what we wished. We actually hoped that MWCs might even be better than the full English codebook, as they might implicitly model the fact that Germans tend to use German sound artifacts in their English. However, this was not the case for either of the test sets. Furthermore, there is no clear tendency in the results.

Nevertheless, these are interesting results, but we are not yet able to give a coherent explanation for them. In the future we will perform further experiments which will hopefully provide a better source for analysis.

Table 5. Results on five languages (Word Accuracies)

Total Gaussians	Added Gaussians	German	English	Italian	French	Spanish
1024	0	84.1	67.3	85.2	68.7	88.3
1224	200	83.8	68.4	88.3	69.0	90.2
1424	400	84.0	70.9	87.9	71.3	91.5
1824	800	84.3	72.0	89.7	72.9	91.0

5.4. Multilingual Evaluation

In this section we evaluate how well our previous results transfer to the case when we actually consider multiple languages at once. Thus, we evaluate on German, English, Italian, French, and Spanish, while we keep German as the main language. As for each language a separate test set is needed, the results in Table 5 show also how consistent MWCs perform for different test sets.

When we consider 5 languages at once in the MWC generation, it is likely that the four codebooks of the additional languages will contain some similar entries, as there will be sound patterns that occur in all languages. To remove these multiple representations of one sound pattern, the following is done. We first throw all additional language codebooks together, resulting in a codebook with 4096 Gaussians. Then we merge (see 3.2) very similar Gaussians in this 4096 codebook until we have gained a codebook with 2048 Gaussians. This 2048 Codebook (the additional languages codebook) is the input to the MWC algorithm, together with the unmodified German codebook.

Table 5 shows the following results. The top row is the baseline experiment which uses only the German codebook. The other recognizers contain the full German codebook with 1024 Gaussians and 200, 400 and 800 Gaussians from the additional languages. The total codebook sizes are 1224, 1424 and 1824, respectively.

The first column with word accuracies shows that the performance on the German test set varies insignificantly. This is what we expected, as the LBG produces already an optimal codebook for German. Thus, the extensions we make to the codebook can not improve performance on German, but they also do not hurt.

However, for the additional languages we see significant improvements. In general, the performance increase is correlated to the amount of Gaussians we add to the codebook, meaning the more we add, the better the performance.

It is interesting to see that performance on Italian and Spanish improves significantly as soon as some additional Gaussians are added to the German codebook. Improvements in English and French are less pronounced at first. When further Gaussians are added, however, improvements in English and French are stronger than in Spanish or Italian. This might be due to similarities and differences between languages.

6. CONCLUSION

To summarize, the MWCs we propose offer significantly improved performance on additional languages, while the increase of the codebook size is still rather moderate. Furthermore, we did also show that the performance on the main language is not affected.

We also performed some experiments on non-native speech. No consistent performance gains were observed on these test sets. In further experiments we will evaluate these phenomena of non-native speech more thoroughly.

7. REFERENCES

[1] J. Koehler, “Multilingual phone models for vocabulary-independent speech recognition tasks,” *Speech Communication Journal*, vol. 35, no. 1-2, pp. 21–30, 2001.

[2] T. Schultz and A. Waibel, “Language-independent and language-adaptive acoustic modeling for speech recognition,” *Speech Communication*, vol. 35, pp. 31–51, 2001.

[3] Y. Linde, A. Buzo, and R. Gray, “An algorithm for vector quantization design,” *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, 1980.

[4] S. Steidl, “Interpolation von Hidden Markov Modellen,” M.S. thesis, University Erlangen-Nuremberg, 2002.

[5] D. Iskra, B. Grosskopf, K. Marasek, H. van den Huevel, F. Diehl, and A. Kiessling, “Speecon - speech databases for consumer devices: Database specification and validation,” in *Proc. LREC*, 2002.

[6] University Munich, “The Verbmobil project,” `http://www.phonetik.uni-muenchen.de/Forschung/Verbmobil/VerbOverview.html`.

[7] W. Menzel, E. Atwell, P. Bonaventura, D. Herron, P. Howarth, R. Morton, and C. Souter, “The ISLE corpus of non-native spoken English,” in *LREC*, Athens, Greece, 2000, pp. 957–963.