Codebook Design for Speech Guided Car Infotainment Systems

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

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

Abstract. In car infotainment systems commands and other words in the user's main language must be recognized with maximum accuracy, but it should be possible to use foreign names as they frequently occur in music titles or city names. Previous approaches did not address the constraint of conserving the main language performance when they extended their systems to cover multilingual input.

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

We provide evidence that common vector quantization algorithms are not optimal for such systems when they have to cope with input from multiple languages. Our new method combines information from multiple languages and creates a new codebook that can be used for efficient vector quantization in multilingual scenarios. Experiments show significant improved speech recognition results.

1 Introduction

Imagine a German tourist looking forward to a wonderful holiday in Marseille, France. Imagine he is driving there with his new car with the latest, speech driven car infotainment system. Of course, he will most of the time interact with the system in German. On his way, however, he will drive through Italy and France. His holiday is long enough, so he wants to explore some cities on his way.

So, apart from Milano and Cannes, he wants to tell his navigation system to drive to Rozzano, Italy and to Roquebrune-Sur-Argens, France. As these city names do not belong to the language of the user interface, current systems are not able to recognize such names when they are spoken. Therefore a multilingual system is needed, that can recognize the German commands the user will utter as well as the foreign city names.

But there are further reasons why the user would want to have a multilingual speech recognizer on his journey. For example, he prefers to listen to his own

E. André et al. (Eds.): PIT 2008, LNCS 5078, pp. 44–51, 2008.

music collection rather than to radio stations in foreign languages. Even the most convenient haptic music selection systems however will distract the tourist on his long way. It would be much better if it would be possible just to say the name of the artist, instead. But as for the city names above, many artists names are not German. Again, without multilingual speech recognition such a system is not possible.

By now, we have motivated why multilingual speech recognition is a essential feature of future car infotainment systems. But, it should not be forgotten, that the user lives in most cases in one country and speaks one language. Thus, he will most of the time travel within his country, and the commands he uses will always belong to one language. Therefore, while we want multilingual recognition, we can not allow multilingual recognition to degrade performance on the user main language, as this is the language he will most of the time use to interact with the system.

To recapitulate, we want to built a Man-Machine Interface (MMI) for car infotainment that

* recognizes commands and other words in the user's main language with maximum accuracy
* but it should be possible to recognize foreign names as well as possible, because they frequently occur in music titles or city names

Previous approaches did not address the constraint of conserving the main language performance when they extended their systems to cover multilingual input [Koehler, 2001, Gruhn et al., 2004, Schultz and Waibel, 2001]. In one case improved performance on non-native speech was achieved without loosing performance on the main language, but only for the limited task of digit recognition [Fischer et al., 2003].

In theory it is hardly a problem to use recognizers for each language and all problems are solved. However, in practice this is currently a rather unrealistic approach. Car systems are still very restricted regarding computing power. Therefore, we build one recognizer that has trained HMMs for each language, but otherwise uses the same small set of Gaussians for all languages. This reduces both computing demands and memory consumption drastically. Using a small set of Gaussians is not new, and generally referred to as semi continuous speech recognition. There are also established methods (LBG algorithm, [Linde et al., 1980]) for the creation of such collections of Gaussians. Such collections are usually called codebook.

For our aims, we believe traditional vector quantization algorithms to be suboptimal. Either only main language training data is provided to the LBG algorithm, or data from all languages is provided. 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. For the car infotainment scenario, neither of these options is optimal.

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

with soft vector quantization based on the LBG approach. From these initial codebooks a new codebook is created. This new codebook should still achieve monolingual performance on the main language, but at the same time improve performance on the additional languages. As the new codebook is based on codebooks from many languages, we call it multilingual, and as the influence of each original codebook can be adjusted, we call it Multilingual Weighted Codebook (MWC).

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

3 Extended System

To improve the performance on the additional languages, we replace the monolingual codebook with an MWC. The 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.

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. We use the Mahalanobis distance as distance measure.

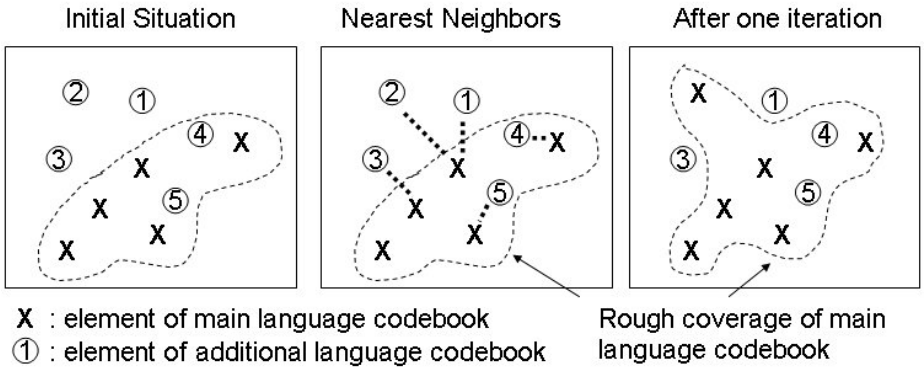

Fig. 1. Basic Idea of Multilingual Weighted Codebooks

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

4 Experimental Setup

Our experiments base on 11 MFCCs with their first and second derivatives per frame and LDA for feature space transformation. All recognizers are trained on 200 hours of Speecon data for each language [Iskra et al., 2002]. The HMMs are context dependent.

We used five languages for the training of our recognizer (US English, French, German, Spanish and Italian). For each training language a codebook with 1024 Gaussians was created by the LBG algorithm. Information about test sets for native speech of 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.

The non-native tests in Section 5.2 are conducted on the HIWIRE data [Segura et al., 2007]. The HIWIRE database contains human input in a command and control aeronautics application by 81 speakers (31 French, 20 Greek, 20 Italian, 10 Spanish). The spoken language is always English. Each speaker produced 100 utterances which were recorded as clean speech with a close talking microphone. The database provides the clean speech signal, as well as three further signals that are convolved with cockpit noise. We test on the clean speech test which is provided with the data (50% of the HIWIRE data). The non-native adaptation data provided with the database is not used. For a comparison of the HIWIRE corpus to other non-native corpora we refer to [Raab et al., 2007].

To reduce the number of experiments, German is always the main language when we build MWCs.

Table 1. Testset description (Utterances/Grammar)

	German	English	Italian	French	Spanish
#Utterances	2005	852	2000	3308	5143
#Grammar Size	2498	500	2000	2000	3672

5 Results

In [Raab et al., 2008] we describe preparatory several experiments to get setups for creating MWCs. For example, experiments with different distance measures and varying codebook sizes. The first part of this section presents results on five languages (German, English, Italian, French, Spanish).

The second part of this section deals with a problem that we have not introduced yet, but which is commonly known: non-native speech recognition. While it is one aspect to produce a better codebook for the recognition of multiple languages, this does not need to actually improve the performance in our target scenario. The reason is that speech by non-native speakers differs significantly from native speakers. Due to the unpredictable deviations, non-native speech is well known to degrade speech recognition performance severely [Witt, 1999, Tomokiyo, 2001]. Therefore the second parts presents results on several non-native accents of English.

5.1 Multilingual Evaluation

When 5 languages at once are considered 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. At first all additional language codebooks are thrown together, resulting in a codebook with 4096 Gaussians. Then very similar Gaussians in this 4096 codebook are merged until the codebook has 2048 Gaussians. By merging we mean replacing 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. It is not necessary to know all the training samples to perform the merging, a formula can be found in [Steidl, 2002]. This 2048 Codebook (the additional languages codebook) is the input to the MWC algorithm, together with the unmodified German codebook.

The top row of Table 2 is the baseline experiment which uses only the German codebook. The other recognizers contain a 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 due to the fact that the LBG already produces an optimal codebook for German. Thus, the extensions to the codebook can not improve performance on German, but they do not hurt.

Table 2. Word Accuracies on five languages

Total Gaussians	Added Gaussians	German	English	Italian	French	Spanish
1024	0	84.1	65.6	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

However, for the additional languages the results show significant improvements. In general, the performance increase is correlated to the amount of Gaussians that were added to the codebook. The MWCs contain relatively few additional Gaussians and can cover all five languages significantly better.

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.

5.2 Results on Non-native Accents

In the previous experiments native speech of several languages was evaluated. In this section accented English is evaluated, i.e. uttered by Spanish, French, Italian and Greek speakers. Basically, the same systems as before are tested. As the MWCs base on the German codebook, we can actually regard two systems as baseline. Our first baseline is a system that uses a standard English codebook. This system will be optimal for native English speech, but the performance on non-native speech is not clear. The system with the German codebook is our second baseline. These two baselines are given in the top rows of Table 3. As a reference the first column gives again the performance of each system on the native English cities test.

Table 3. Word Accuracies with MWCs on HIWIRE

Codebook	US City	Hiwire SP	Hiwire FR	Hiwire IT	Hiwire GR
English 1024	**75.5**	82.5	83.9	81.6	83.1
German 1024	65.6	85.4	**86.9**	82.8	85.5
5ling_1224	68.6	86.2	86.6	84.6	85.3
5ling_1424	68.4	86.4	86.7	**85.7**	85.8
5ling_1824	70.9	**86.9**	86.2	84.2	**86.3**

At first the focus is on the performance relative to the system with the English codebook. Compared to this system, the MWCs steadily improve the performance, the larger the codebook is, the better the performance.

Now the focus is on the performance relative to the system with German codebook. Oddly, even the system using only the German codebook outperforms the system with English codebook in the recognition of non-native English. This shows how strong non-native speech differs from native English. For most accents, however, MWCs perform better than systems with only the German codebook.

While future work is needed to explain why a German codebook can outperform a native English codebook, one conclusion can already be made from these results. MWCs consistently provide good performance for all tested accents and are always significantly better than the native English codebook. One assumption why the German codebook is better as the English codebook could be that the German language is phonetically richer, thus better suited to cover the sound deviations non-native speakers produce.

6 Conclusion

With our algorithm for the creation of multilingual codebooks, we have successfully introduced a vector quantization that can satisfy the aims of our car infotainment scenario. We can fix the performance of a system on a given main language, but at the same time improve performance on additional languages.

The results show clearly significant improvements on native speech from five different languages. While open questions remain for the recognition of non-native speech, we can show that MWCs can produce significant better performance on non-native accents of English. A task left to future experiments is to analyze more closely why a German codebook can outperform a native English codebook for non-native English tests.

Regarding non-native speech, our results show the strong acoustic differences, but there are also differences in the choice of words. Thus, language modeling approaches like in [Fuegen, 2003] could be used with special treatment for non-native speech. A more recent review of language model techniques an be found in [Raab, 2007].

Acknowledgments

We thank the HIWIRE (Human Input That Works In Real Environments) project from the EC 6th Framework IST Programme for the non-native test data [Segura et al., 2007].

References

[Fischer et al., 2003] Fischer, V., Janke, E., Kunzmann, S.: Recent progress in the decoding of non-native speech with multilingual acoustic models. In: Proc. Eurospeech, pp. 3105–3108 (2003)

[Fuegen, 2003] Fuegen, C.: Efficient handling of multilingual language models. In: Proc. ASRU, pp. 441–446 (2003)

[Gruhn et al., 2004] Gruhn, R., Markov, K., Nakamura, S.: A statistical lexicon for non-native speech recognition. In: Proc. Interspeech, Jeju Island, Korea, pp. 1497–1500 (2004)

[Iskra et al., 2002] Iskra, D., Grosskopf, B., Marasek, K., van den Huevel, H., Diehl, F., Kiessling, A.: Speecon - speech databases for consumer devices: Database specification and validation. In: Proc. LREC (2002)

[Koehler, 2001] Koehler, J.: Multilingual phone models for vocabulary-independent speech recognition tasks. Speech Communication Journal 35(1-2), 21–30 (2001)

[Linde et al., 1980] Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantization design. IEEE Transactions on Communications 28(1), 84–95 (1980)

[Raab, 2007] Raab, M.: Language Modeling for Machine Translation. Vdm Verlag, Saarbruecken (2007)

[Raab et al., 2007] Raab, M., Gruhn, R., Noeth, E.: Non-native speech databases. In: Proc. ASRU, Kyoto, Japan, pp. 413–418 (2007)

[Raab et al., 2008] Raab, M., Gruhn, R., Noeth, E.: Multilingual weighted codebooks. In: Proc. ICASSP, Las Vegas, USA (2008)

[Schultz and Waibel, 2001] Schultz, T., Waibel, A.: Language-independent and language-adaptive acoustic modeling for speech recognition. Speech Communication 35, 31–51 (2001)

[Segura et al., 2007] Segura, J., et al.: The HIWIRE database, a noisy and non-native English speech corpus for cockpit communication (2007), `http://www.hiwire.org/`

[Steidl, 2002] Steidl, S.: Interpolation von Hidden Markov Modellen. Master's thesis, University Erlangen-Nuremberg (2002)

[Tomokiyo, 2001] Tomokiyo, L.: Recognizing Non-native Speech: Characterizing and Adapting to Non-native Usage in Speech Recognition. PhD thesis, Carnegie Mellon University, Pennsylvania (2001)

[Witt, 1999] Witt, S.: Use of Speech Recognition in Computer-Assisted Language Learning. PhD thesis, Cambridge University Engineering Department, UK (1999)