ACOUSTIC LANGUAGE MODEL CLASSES FOR A LARGE VOCABULARY CONTINUOUS SPEECH RECOGNIZER

V. Fischer, S.J. Kunzmann

IBM Voice Systems, European Speech Research,
Vangerowstr. 18, D-69115 Heidelberg, F.R. of Germany
{vfischer,kunzmann}@de.ibm.com

ABSTRACT

In a maximum a posteriori probability approach to speech recognition stochastic n-gram language models are used for the estimation of a word sequence's a priori probability. In any practical implementation of a large vocabulary speech recognition system the language model acts as a hypotheses filter that has to differ between candidate words with similar acoustic evidence. For that purpose, the combination of word based and class based language models is attractive, because it allows to fall back to the more reliable estimates of the class based model in case of sparse training data. However, class language models can differ between words from the same class only in terms of a priori probability. To improve the discriminative power for words with similar acoustic score, it is therefore useful to put similar sounding words into different classes.

Based on the above considerations, the paper presents an automatic procedure for the optimal classification of a large vocabulary into classes with acoustic dissimilar words. In combination with a standard word based trigram model the so created *acoustic class language model* provides a relative reduction in word error rate of up to 16 percent and performs slightly better than a perplexity minimizing automatically created class language model.

1. INTRODUCTION

Today's large vocabulary continuous speech recognition systems compute a word sequence

$$\begin{aligned}\widehat{\mathbf{W}} &= \text{argmax}_{\mathbf{W}}\{P(\mathbf{W}|\mathbf{A}\} \\ &= \text{argmax}_{\mathbf{W}} P(\mathbf{A}|\mathbf{W}) \cdot P(\mathbf{W}) \qquad (1)\end{aligned}$$

with maximum a posteriori probability given an acoustic evidence $\mathbf{A}$ derived from the speech signal. While the computation of $P(\mathbf{A}|\mathbf{W})$ is referred to as acoustic modeling and is usually done in a Hidden Markov Model framework, the estimation of the a priori probability

$$P(\mathbf{W}) = \prod_{i=1}^{L} P(w_i|w_1,\ldots,w_{i-1}) \qquad (2)$$

of a word sequence of length L is referred to as (stochastic) language modeling; see [10] for a detailed treatment of statistical methods in both acoustic and language modeling.

Since the number of conditioning events $w_1,\ldots,w_{i-1}$ is far too large for both the storage and reliable estimation of probabilities $P(w_i|w_1,\ldots,w_{i-1})$, word histories are usually classified into a manageable number of equivalence classes. The state of the art is provided by trigram language models which consider two histories equal if they end in the same two words,thus

$$P_{3g}(\mathbf{W}) = \prod_{i=1}^{L} P(w_i|w_{i-2},w_{i-1}). \qquad (3)$$

Besides smoothing or back-off techniques, class language models [4] have been introduced to deal with the problem of sparse training data. For class language models a mapping $C : \mathcal{V} \mapsto \mathcal{C}$ of the training vocabulary $\mathcal{V}$ into a system $\mathcal{C}$ of — usually pairwise disjoint — classes is defined and the computation of $P(\mathbf{W})$ according to Eqn. 3 modifies to

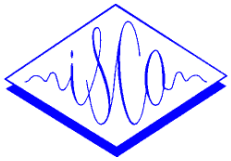
$$\begin{aligned}P_{3c}(\mathbf{W}) = \prod_{i=1}^{L} & P(w_i|C(w_i)) \cdot \qquad (4) \\ & P(C(w_i)|C(w_{i-2}),C(w_{i-1})).\end{aligned}$$

Methods for the classification of a large vocabulary $\mathcal{V}$ into a class system $\mathcal{C}$ rely on the maximization of the triclass probability $P_{3c}(\mathbf{W})$ on a given training corpus, or — equivalently — minimize the text perplexity [11]. Combinatorial optimization procedures, like e.g. simulated annealing, may start from a grammatically motivated classification of words (e.g. into classes like *noun*, *verb*, *adjective*,...) and are well suited to find optimal solutions that do not depend on the initial partitioning of the vocabulary [9].

The combination of trigram and triclass language models (e.g. [12]), can be achieved by the use of a combined a priori score

$$P(\mathbf{W}) = \alpha P_{3g}(\mathbf{W}) + (1-\alpha) P_{3c}(\mathbf{W}), \qquad (5)$$

$0 \leq \alpha \leq 1$,and is attractive, because it allows to fall back to the more reliable parameters of the triclass language model in cases of a poor trigram estimate.



6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

The work described here also utilizes this approach, but presents a new method for the computation of language model classes. Instead of focussing on the minimization of perplexity, we employ a class language model that directly aims on a better distinction between acoustic similar candidate words. After reviewing the role of the language model in our large vocabulary continuous speech recognizer in Section 2, we describe a method for the computation of the acoustic similarity between words and a clustering procedure based on this distance measure. Section 4 reports some experimental results and finally Section 5 gives a conclusion.

2. THE STACK DECODER

The general framework for the work presented here is provided by IBM's large vocabulary continuous speech recognition system, an asynchronous stack decoder that is described in some detail in [1].

The acoustic front end extracts 13 Mel-Frequency-Cepstrum coefficients (including $C0$) and their first and second order derivatives every 10 msec. The labeled speech frames are passed to the acoustic fast match [2] which uses continuous density Hidden Markov Models to quickly obtain a short list of candidate words from a very large vocabulary. Most of the words on the short list will be acoustically similar to the correct word, but may be unlikely in the linguistic context, which is provided by the last words on the current search path. Therefore, the final fast match score

$$F(w) \quad = \quad F_A(w)^{\lambda} \cdot p(w)^{1-\lambda} \tag{6}$$

for a word w is obtained by a combination of the coarse acoustic score $F_A(w)$ and the language model score $p(w)$, where λ allows to adjust the relative importance of the two scores. The best scoring word candidates are re-scored by a computationally more expensive detailed match that also provides a more exact end point distribution for the candidate words. Finally, the best scoring hypotheses are used to form new partial paths, and the best path is extended in the next decoder iteration.

The language model score $p(w)$ that is used to differ between words with similar acoustic score may be either drawn from a word based trigram language model, or may be a combination of a trigram and triclass score according to Eqn. 5. Note however, that for the latter members of the same class can be distinguished only in terms of their a priori probability (cf. Eqn. 4). Thus, for an efficient disambiguation a class language model should use different equivalence classes for similar sounding words, like e.g. the nouns *pain* and *gain* or homonyms like *(the) sea* and *(to) see*. An automatic method for the computation of such classes is described in the following section.

3. ACOUSTIC LM CLASSES

Because of some experiments in cross domain acoustic modeling we are primarily interested in a similarity measure that is independent of a particular acoustic model, but easy to modify, once reliable domain dependent acoustic information is available. Moreover, useful information from the acoustic model, like e.g. phone confusion probabilities, leaf rank histograms [3], or HMM statistics, may be unavailable to developers aiming on the customization of the recognizer vocabulary.

Acoustic similarity. Following the above considerations, the computation of the similarity $s(w_i, w_j)$ of any two words w_i and w_j is based solely on the distance $D(.)$ between the two words' standard pronunciations ϕ_i and ϕ_j:

$$s(w_i, w_j) \quad = \quad -D(\phi_i, \phi_j). \tag{7}$$

Since pronunciations are usually available as phonetic baseforms $\phi_i = \varphi_{i,1} \ldots \varphi_{i,N}$ and $\phi_j = \varphi_{j,1} \ldots \varphi_{j,M}$, a distance measure is required that allows for the treatment of arbitrary length (phone) symbol strings. We use the *Levenshtein distance*

$$\begin{aligned} D_L(\phi_i, \phi_j) &= d(\varphi_{i,N}, \varphi_{j,M}) + \overline{D}_L \\ d(\varphi_{i,n}, \varphi_{j,m}) &= \begin{cases} 0 & \text{iff} \quad \varphi_{i,n} = \varphi_{j,m} \\ 1 & \text{else} \end{cases} \\ \overline{D}_L &= \min\{D_L(\phi_i^{N-1}, \phi_j^{M-1}), \\ &\qquad D_L(\phi_i^{N}, \phi_j^{M-1}), \\ &\qquad D_L(\phi_i^{N-1}, \phi_j^{M})\}, \end{aligned} \tag{8}$$

which is the minimum number of substitutions, insertions, and deletions of phones that are needed to transform one baseform into the other. Here, $\varphi_{i,n}$ denotes the n-th phone of baseform ϕ_i, and $\phi_i^l = \varphi_{i,1} \ldots \varphi_{i,l}$ is the baseform's prefix phone string of length l. The Levenshtein distance can be efficiently computed by a dynamic programming procedure and, moreover, it allows for an easy modification via either a different local distance measure $d(.,.)$ or the consideration of different predecessors in the minimum term of Eqn. (8).

Classification algorithm. The classification of a large vocabulary $\mathcal{V}$ (typically $|\mathcal{V}| > 60.000$) into a class system $\mathcal{C} = \bigcup_v C_v$ ($|\mathcal{C}| \approx 5.000$) is a computational hard problem that can be efficiently solved by combinatorial optimization [8].

The algorithm employed here starts from a uniform but random assignment of words into a fixed number of classes. In each iteration k the average intra-class distance

$$S_v^{(k)} \quad = \quad \frac{1}{|C_v|^2} \sum_{w_i \in C_v} \sum_{w_j \in C_v} -s(w_i, w_j) \tag{9}$$

is used to select two classes C_u and C_v that are allowed to tentatively exchange a single word. While words are chosen randomly from each class, we employ a roulette wheel selection mechanism (see e.g. [7]) that prefers those classes with a small intra-class distance.

For the evaluation of a tentative word swap the overall goodness of the classification result is computed as

$$G^{(k)} = \frac{1}{|\mathcal{C}|}\sum_{v=1}^{|\mathcal{C}|} S_v^{(k)}. \quad (10)$$

A tentative swap of words can result in either an increased or decreased intra-class similarity for both classes C_u and C_v, or it can lower S for one class while increasing it for the other. While we reject (accept) all moves that increase (decrease) the similarity within both classes, we found the best speed of convergence for the optimization procedure if a *deterministic* acceptance criterion

$$\begin{aligned} \text{yes}(\mathcal{C}^{(k)}) &= \begin{cases} 1 & \text{iff} \quad G^{(k)} > \theta \cdot G^{(best)} \\ 0 & \text{else} \end{cases} \\ G^{(best)} &= \max_{i=1\ldots k}\{G^{(i)}\} \end{aligned} \quad (11)$$

(see e.g. [5]) is used that accepts all moves that are not significantly worse than the best solution obtained so far ($\theta \approx 0.95$).

4. EXPERIMENTS

In the experiments described here we compare speaker independent word error rates for the use of a trigram language model (cf. Eqn. 3) and the use of a combined trigram and triclass language model (cf. Eqn. 5). For the latter we compare two different triclass models that were automatically created by use of a combinatorial optimization procedure:

- Simulated annealing was employed to create a class language model with 4096 classes that minimizes the perplexity of the training corpus (referred to as *pclass* model below).
- The algorithm described in Section 3 was used for the classification of the 65.000 words from the fast match tree into 4096 language model classes (denoted *aclass* model below).

Both triclass language models have approximately the same size and were evaluated in combination with the same word based trigram model. We used a closed vocabulary, i.e. there are no OOV words in the test script, and two different sets of test speakers that were drawn from an in-house data base:

- The first set comprises 20 German speakers (A1 – V1, 10 female, 10 male) that mostly use standard pronunciations as provided by the phonetic baseforms employed for the creation of the acoustic class language model in Section 3.
- The second set comprises 20 Austrian dialect speakers (A2 – V2, again 10 female, 10 male) whose pronunciation in some cases significantly differs for many words in the vocabulary.

3gram only		3gram + pclass	
German	Austrian	German	Austrian
A1: 5.94	A2: 9.09	A1: 5.24	A2: 3.85
B1: 6.99	B2: 4.90	B1: 7.69	B2: 9.09
C1: 5.24	C2:11.19	C1: 2.45	C2: 9.44
D1: 8.39	D2: 7.69	D1: 5.24	D2: 8.04
E1: 9.44	E2: 8.04	E1: 8.04	E2: 5.94
F1: 7.34	F2: 6.29	F1:16.78	F2: 5.94
G1:11.19	G2: 4.55	G1: 6.64	G2: 5.24
H1:12.24	H2: 5.59	H1: 9.09	H2: 4.55
J1:10.49	J2:10.14	J1: 6.29	J2: 6.44
K1:18.53	K2: 9.79	K1:17.83	K2: 7.34
L1: 6.29	L2:13.64	L1: 4.90	L2:15.03
M1: 9.09	M2: 8.39	M1:10.14	M2: 5.94
N1: 8.39	N2: 9.09	N1: 6.64	N2: 6.99
P1: 9.44	P2: 6.29	P1: 9.44	P2: 4.55
Q1: 5.94	Q2:12.24	Q1: 4.55	Q2:12.24
R1: 5.24	R2: 7.69	R1: 7.69	R2: 6.29
S1:10.49	S2:25.17	S1: 6.64	S2:22.03
T1: 6.64	T2:14.34	T1: 7.34	T2:13.64
U1:11.19	U2:20.28	U1: 3.85	U2:23.78
V1:21.33	V2:11.89	V1:18.53	V2: 8.74
ave.: 9.49	ave.:10.31	ave.: 8.25	ave.: 9.27

Table 1: Word error rates for the word based trigram LM (left) and for the combination with a perplexity minimizing class LM (right).

Table 1 compares the baseline results, i.e. speaker independent word error rates (WER) for the use with the trigramm language model (left column), to the results obtained in combination with the perplexity minimizing triclass model (right). We can observe a relative decrease in WER of 13.1 percent for the German speakers and of 10.1 for the Austrian speakers; for both test sets, 14 out of 20 speakers show an improved recognition accuracy when using the combined LM score according to Eqn. 5.

Table 2 gives the results for the combination of the trigram LM and the acoustic class language model. The use of the acoustic class LM results in a 16.0 percent relative decrease in WER for the standard speakers, but yields a smaller gain for the dialect speakers (left column). While throughout these experiments the relative importance λ of the language model score (cf. Eqn. 6) remains unchanged, in combination with the acoustic class LM we observed better results for an increased contribution α of the word based LM (cf. Eqn. 5) to the overall LM score. Finally, from the right column it can be seen that the combination of all three language models yields the best results for both sets of speakers. However, it has to be noted that in this case the LM access is more time consuming.

5. CONCLUSION

The main idea of the work described here was to achieve a better discrimination of candidate words by the construction of a class language model that considers the acoustic similarity of words. For that purpose, we have presented an algorithm for the automatic classification of words from a large vocabulary that utilizes the Levenshtein distance of the words' phonetic baseforms and a combinatorial op-

3gram + aclass		3gram + pclass + aclass	
German	Austrian	German	Austrian
A1: 3.50	A2: 7.34	A1: 3.85	A2: 4.20
B1: 6.29	B2: 3.85	B1: 5.24	B2: 4.20
C1: 5.24	C2: 10.84	C1: 4.55	C2: 11.19
D1: 8.04	D2: 6.64	D1: 6.29	D2: 7.69
E1: 6.64	E2: 6.99	E1: 6.99	E2: 8.39
F1: 6.99	F2: 6.29	F1: 7.69	F2: 6.29
G1: 9.44	G2: 2.45	G1: 6.29	G2: 4.90
H1: 10.14	H2: 4.20	H1: 8.74	H2: 5.24
J1: 6.99	J2: 9.09	J1: 6.99	J2: 8.39
K1: 16.43	K2: 9.09	K1: 19.58	K2: 10.49
L1: 4.55	L2: 13.29	L1: 4.90	L2: 13.99
M1: 6.99	M2: 7.34	M1: 10.84	M2: 6.99
N1: 7.34	N2: 9.09	N1: 4.90	N2: 6.64
P1: 8.04	P2: 4.90	P1: 6.64	P2: 4.55
Q1: 4.55	Q2: 12.24	Q1: 6.29	Q2: 9.09
R1: 4.55	R2: 6.99	R1: 6.29	R2: 5.94
S1: 9.09	S2: 23.08	S1: 10.14	S2: 20.98
T1: 4.90	T2: 13.29	T1: 6.99	T2: 10.14
U1: 9.44	U2: 21.68	U1: 6.99	U2: 21.68
V1: 20.28	V2: 10.14	V1: 18.18	V2: 10.84
ave.: 7.97	ave.: 9.44	ave.: 7.92	ave.: 9.09

Table 2: Word error rates for the combination of a word based trigram and acoustic class LM (left), and for the combination of word based, acoustic, and perplexity minimizing class LM.

timization procedure to separate similar sounding words into different classes.

In combination with a standard word based trigram LM the created language model classes have been shown to provide a good disambiguation of hypotheses from the acoustic fast match list. By considering multiple baseforms and more sophisticated distance measures we hope to further improve the results for non-standard or dialect speakers. Initialization of the algorithm with a perplexity minimizing class system may be helpful to obtain a single class language model that shows the advantages of both methods. Further work should also consider investigations on the optimal number of LM classes. Finally, we plan to use other evaluation criteria, like e.g. *speech decoder entropy* [6], that also aim on a better interaction between the decoder's acoustic and linguistic components for the computation of powerful class language models.

Acknowledgement We would like to thank our colleagues in the IBM European Speech Research Group (located in Cairo, Heidelberg, Hursley, Paris, Rome, and Seville) and in the IBM Human Language Technology Research Group (Thomas J. Watson Research Center, Yorktown Heights) for many valuable suggestions and the continuous exchange of ideas. In particular, we wish to thank Giulio Maltese for some valuable discussions and Martin Herzog for help in the creation of the acoustic class language model.

6. REFERENCES

1. L. Bahl, S. Balakrishnan-Aiyer, J. Bellegarda, M. Franz, P. Gopalakrishnan, D. Nahamoo, M. Novak, M. Padmanabhan, M. Picheny, and S. Roukos. Performance of the IBM Large Vocabulary Continuous Speech Recognition System on the ARPA Wall Street Journal Task. In *Proc. of the IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, pages 41–44, Detroit, 1995.
2. L. Bahl, S. De Gennaro, P. Gopalakrishnan, and R. Mercer. A fast approximate acoustic match for large vocabulary speech recognition. *IEEE Trans. on Speech and Audio Processing*, 1(1):59–67, January 1993.
3. L. Bahl, P. de Souza, P. Gopalakrishnan, D. Nahamoo, and M. Picheny. Robust methods for using context-dependent features and models in a continuous speech recognizer. In *Proc. of the IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, Adelaide, 1994.
4. P. Brown, V. Della Pierra, P. de Souza, J. Lai, and R. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–4798, 1992.
5. G. Dueck. New optimization heuristics: The great deluge algorithm and the record-to-record-travel. *Journal of Computational Physics*, 104(1):86–92, 1993.
6. M. Ferretti, G. Maltese, and S. Scarci. Measuring information provided by language model and acoustic model in probabilistic speech recognition: therory and experimental results. *Speech Communication*, 9(5/6):531–539, December 1990.
7. D. Goldberg. *Genetic Algorithms: Search, Optimization and Machine Learning*. Addison–Wesley Publ. Co., Reading, Mass., 1989.
8. M. Jardino. Multilingual Stochastic n-Gramm Class Language Models. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Atlanta, N.Y., 1996.
9. M. Jardino and G. Adda. Automatic word classification using simulated annealing. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Minneapolis, 1993.
10. F. Jelinek. *Statistical Methods for Speech Recognition*. The MIT Press, Cambridge, Ma., 1997.
11. S. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 35(3):400 – 401, March 1987.
12. T. Niesler and P. Woodland. Combination of word-based and category-based language models. In *Proc. of the 6th Int. Conf. on Spoken Language Processing*, Philadelphia, 1996.