ROBUST DIALOGUE-STATE DEPENDENT LANGUAGE MODELING USING LEAVING-ONE-OUT

Frank Wessel and Andrea Baader

Lehrstuhl für Informatik VI, RWTH Aachen – University of Technology
52056 Aachen, Germany
wessel@informatik.rwth-aachen.de

ABSTRACT

The use of dialogue-state dependent language models in automatic inquiry systems can improve speech recognition and understanding if a reasonable prediction of the dialogue state is feasible. In this paper, the dialogue state is defined as the set of parameters which are contained in the system prompt. For each dialogue state a separate language model is constructed. In order to obtain robust language models despite the small amount of training data we propose to interpolate all of the dialogue-state dependent language models linearly for each dialogue state and to train the large number of resulting interpolation weights with the EM-Algorithm in combination with Leaving-One-Out.

We present experimental results on a small Dutch corpus which has been recorded in the Netherlands with a train timetable information system and show that the perplexity and the word error rate can be reduced significantly.

1. INTRODUCTION

In automatic inquiry systems, e.g. train timetable information systems or switchboards, speech recognition and understanding can be improved using contextual knowledge as an additional constraint during the recognition process. If the prediction of the state a dialogue system is currently in is possible, this knowledge can be used to improve the language model of the recognizer. Previous work has focussed on the statistical prediction of dialogue states in a speech-to-speech translation system [7]. In automatic inquiry systems, the prediction of the dialogue states is easier. In [1] and [6] the dialogue state is defined by the question the user is replying to. Using this simple definition, the language model training corpus is split according to the dialogue states and a separate language model for each dialogue state is then trained. Since in a train timetable information system the system question for the station of arrival will most probably be answered by providing a station name, this approach seems very reasonable and, in fact, yields good results.

One of the main drawbacks of this approach is that, with the very limited amount of training material in the domain of automatic inquiry systems, the number of words in the language model training corpus for each dialogue state is very small and several dialogue states might even remain unobserved in the training material. Possible ways to overcome this problem are to generalize dialogue states until a sufficient amount of training material for each state is obtained [1] or to decide between the dialogue-state dependent language model and a global, context-independent language model [6], if the first is not robust enough.

Although this 'hard' decision between a state dependent and an independent model performs very well, there might be other dialogue states which condition similar user utterances. Thus, it might be desirable to use a combination of several dialogue-state dependent and a dialogue-state independent language model. We therefore propose to train a language model for each dialogue state and use a linear interpolation of all dialogue-state dependent and a global language model for each dialogue state instead of deciding between just the dialogue-state dependent and the independent language model.

The rather large number of resulting interpolation weights can be trained efficiently on the language model training corpus with the EM-Algorithm in combination with Leaving-One-Out. In doing so, we do not need to hold out a part of the small training corpus for the estimation of the interpolation weights which would have further reduced the amount of training material.

2. DESCRIPTION OF SYSTEM AND CORPUS

The corpus which we have used for our experiments has been recorded in the Netherlands with the prototype of a train timetable information system [3]. The language model training material is identical to the transcriptions of the user utterances. We have split the corpus randomly into two parts, reserving a large part for testing purposes, so that each dialogue state is observed often enough in the testing corpus. Table 1 specifies the Dutch corpus. The vocabulary which has been used throughout all of the following experiments consists of 985 words, the phoneme inventory of 36 phonemes. Since we did not have access to the online version of the information system we run all experiments off-line and restrain ourselves to the evaluation of the impact of the dialogue-state dependent language models on the word error rate. For our experiments, we have generated a word lattice with our own large vocabulary continuous speech recognition system [4]. The generation

Table 1: Specification of the Dutch corpus

	training	testing	total
dialogues	2364	453	2817
sentences	23234	4330	27564
words	97838	18491	116329
hours	16.5	3.1	19.6

This work was partly funded by the European Commission in the framework of the ARISE project under grant LE3-4229. The responsibility for the contents of this study lies with the authors.

of the lattice is based on the word pair approximation and makes use of a bigram language model during the recognition process. The only difference to the system described in [4] is the modified perceptual linear predictive analysis (MF-PLP) which has been applied to the signal in the acoustic front-end.

3. DEFINITION OF DIALOGUE STATES

As in [1] and [6] we have decided to define the dialogue states in a natural way. In order to generate a database query, the system has to fill several slots and has to prompt questions to the user. Typically, the user will answer these questions in the desired way and provide the necessary information. E.g., the answer to a question for the station of departure and arrival will in most cases contain two station names. In the automatic inquiry system under consideration, the slots which have to be filled before a database query can be started are *station of departure* **(DE)**, *station of arrival* **(AR)**, *date* **(DA)** and *time* **(TI)**. One of our main aims was to avoid a hand-driven analysis of the user utterances, which would have been necessary to find out similarities between different dialogue states and to construct robust language models for the different dialogue states. Instead we have regarded all possible combinations of these slots and have decided to leave the might-be combination of different dialogue states to later and automatic steps. With the four different slots defined above, $2^4 - 1 = 15$ potential dialogue states have to be considered. In addition, the system is capable of asking whether the user wants a repetition of the connection which has been retrieved from the database **(REPEAT)**, whether he wants a later connection **(LATER)** or whether he would like to obtain another, completely different one **(OTHER)**. In combination, the system prompt can contain 18 different sets of parameters which can either be part of a question for this set **(Q)** or a verification of it **(V)**. An additional garbage state **(GARBAGE)** has been defined to be able to classify dialogue states which have obviously resulted from errors within the system.

With this definition of a dialogue state we have implemented a very simple parser which is able to classify each system prompt non-ambiguously. We have split the corpus according to the dialogue state of each utterance and have thus obtaining a separate training corpus for each dialogue state. In summary, we have observed 22 of the 37 possible dialogue states in the language model training and 18 of these 22 in the testing corpus. For the rest of this paper we will use the following notation: let S denote the number of different dialogue states, s the current dialogue state, C_s the language model training corpus for dialogue state s and N_s the number of words in this corpus.

4. MATHEMATICAL MODELS

In the following we define different language models which we have use in our evaluation experiments. Let $N_s(h, w)$ denote the frequency of event (h, w) in training corpus C_s, $n_{s_0}(h)$ the number of different words which have not been observed after history h and W the size of the vocabulary.

4.1. Dialogue-State Dependent Language Model

For each dialogue state s we have constructed a trigram language model with the dialogue-state dependent training corpus C_s. The models for each dialogue state are based on absolute discounting.

Table 2: Number of words in the corpus for each dialogue state

	dialogue state	testing	training
0	GLOBAL	18491	97838
1	GARBAGE	80	845
2	Q DE	318	2239
3	Q DE AR	3906	19671
4	Q AR	486	1952
5	Q DA	880	5806
6	Q TI	5322	27434
7	Q REPEAT	2495	13334
8	Q OTHER	1839	9117
9	Q LATER	438	3236
10	V DE	25	286
11	V DE AR	13	84
12	V DE AR DA	3	16
13	V DE AR DA TI	68	167
14	V DE DA	0	2
15	V DE DA TI	0	85
16	V DE TI	0	2
17	V AR	25	422
18	V AR TI	0	12
19	V DA	615	2829
20	V DA TI	543	2909
21	V TI	1179	6359
22	V REPEAT	150	1031

For smoothing, the relative frequencies are discounted with a discounting weight b_s and are interpolated with a generalized singleton backing-off probability distribution $\beta_s(w|\overline{h})$. Details are described in [5].

$$p_s(w|h) = \max\left\{0; \frac{N_s(h,w) - b_s}{N_s(h)}\right\} + b_s \cdot \frac{W - n_{s_0}(h)}{N_s(h)} \cdot \beta_s(w|\overline{h}) \quad . \tag{1}$$

4.2. Interpolation with a Global Language Model

As Table 2 shows, several of the dialogue states have hardly been observed in the language model training corpus. Therefore, we have combined the dialogue-state dependent and the global language model linearly to achieve a smoother probability distribution. p_0 denotes the probability distribution provided by the global dialogue-state independent language model which has been trained on the whole training corpus and $\lambda_s(i)$ the interpolation weight for dialogue-state dependent language model i in dialogue state s:

$$\widetilde{p}_s(w|h) = \lambda_s(s) \cdot p_s(w|h) + \lambda_s(0) \cdot p_0(w|h) \quad , \tag{2}$$

$$\text{where} \quad \lambda_s(s) + \lambda_s(0) = 1 \quad \forall s \quad .$$

4.3. Interpolation of all Language Models

The final language model combines all of the dialogue-state dependent language models and the global model linearly for each dialogue state. As described above, the motivation for this model was to investigate if other dialogue states might be comparable to the current one and might thus contribute to the prediction of what the user is going to say. This approach is similar to the models presented in [2], the main difference being that the interpolation

weights are not estimated dynamically in order to adapt to a change of topic, but statically and beforehand:

$$\widetilde{p}_s(w|h) = \sum_{i=0}^{S} \lambda_s(i) \cdot p_i(w|h) \quad , \tag{3}$$

$$\text{where} \quad \sum_{i=0}^{S} \lambda_s(i) = 1 \quad \forall s \quad .$$

The main problem with this last model is the large number of $(S+1)^2$ interpolation weights. In order to obtain fair results we would have had to split the training corpus into two parts using one of them for the training of the language models and the other as a cross-validation set for the estimation of the interpolation weights. With only 97838 words, this would have further deteriorated the language models and probably no improvement would have been possible. Instead, we have decided to use the training corpus itself for the estimation of the $\lambda_s(i)$ with the Expectation-Maximization algorithm.

The iteration formula for the estimation of the interpolation weights is usually given as:

$$\overline{\lambda}_s(i) = \frac{1}{N_s} \sum_{n=1}^{N_s} \frac{\lambda_s(i) \cdot p_i(w_n|h_n)}{\sum_{j=0}^{S} \lambda_s(j) \cdot p_j(w_n|h_n)} \quad . \tag{4}$$

Using this formula on the training corpus would have lead to setting $\lambda_s(s) = 1$ and $\lambda_s(i) = 0 \quad \forall s \neq i$. Therefore we have computed Leaving-One-Out probabilities on the training corpus and have used them in Equation 4. These probabilities are given by:

$$\begin{aligned} p_s(w|h) &= \max\left\{0; \frac{N_s(h,w) - 1 - b_s}{N_s(h) - 1}\right\} \\ &+ b_s \cdot \frac{W - n_{s_0}(h)}{N_s(h)} \cdot \beta_s(w|\overline{h}) \quad , \end{aligned} \tag{5}$$

where $\beta_s(w|\overline{h})$ and $n_{s_0}(h)$ are also modified accordingly. The modification of these quantities is very convenient in our language model software, since we store the counts of trigrams, bigrams and unigrams and compute the language model probabilities when needed. For details, the reader is referred to [8]. Using these modified probabilities, a reliable estimation of the interpolation weights is possible, for both, the model defined in Equation 3 and the interpolation between the dialogue-state dependent and the global model, defined in Equation 2.

5. EXPERIMENTAL RESULTS

In order to evaluate the different language models we have measured the perplexities and the word error rates on the word lattice. Throughout this paragraph, let **GLOB** denote the dialogue-state independent language model, **DEP** the dialogue-state dependent one, **BOTH** the interpolation between both and **ALL** the interpolation of all language models for each dialogue state. Table 3 summarizes the perplexities on the testing corpus for the different trigram language models. The third column clearly indicates that the use of dialogue-state dependent language models without any further smoothing only has a small effect on the perplexities. The perplexities for several of the dialogue states even increase. On the other hand, this effect is not surprising bearing in mind the small number of words in the corresponding training corpora, summarized in Table 2.

Table 3: Perplexities for the different language models

	dialogue state	GLOB	DEP	BOTH	ALL
0	GLOBAL	12.11	10.62	9.58	9.48
1	GARBAGE	10.36	14.40	9.33	8.93
2	Q DE	41.66	44.20	28.99	27.51
3	Q DE AR	18.64	14.74	14.13	14.01
4	Q AR	21.55	23.62	15.90	15.11
5	Q DA	26.45	27.68	21.26	20.46
6	Q TI	16.96	15.05	14.16	14.09
7	Q REPEAT	4.80	3.72	3.66	3.63
8	Q OTHER	6.79	5.87	5.50	5.47
9	Q LATER	5.78	5.42	5.00	4.78
10	V DE	7.94	9.49	7.38	9.48
11	V DE AR	22.78	70.35	22.78	25.25
12	V DE AR DA	6.95	418.78	6.95	6.95
13	V DE AR DA TI	14.69	54.85	13.74	18.70
14	V AR	19.04	55.01	19.13	19.01
15	V DA	8.29	8.54	6.67	6.63
16	V DA TI	8.91	8.42	7.19	6.91
17	V TI	6.15	4.69	4.42	4.37
18	V REPEAT	13.23	14.99	10.20	9.73

The interpolation between the dialogue-state dependent and the independent language model performs better than the dialogue-state dependent models alone. The perplexities are lower for all dialogue states, except for dialogue state 14.

The combined model which interpolates all dialogue-state dependent models for each dialogue state further reduces the perplexity for most of the dialogue states. Unfortunately, the perplexity increases for some of the verification states. The perplexity for dialogue state 11 rises from 22.8 with the global model to 25.3 with the model defined in Equation 3. On the other hand, the testing corpus for this dialogue state consists of only 13 words and the increase in perplexity can be regarded as statistically insignificant.

A comparison between Figure 1 and Figure 2 confirms our assumption that several dialogue states contribute to the current one. The x-axis in both figures represents the dialogue state, the y-axis the language model index and the z-axis the interpolation weight. Whereas in Figure 1 the interpolation weights for the global model in several dialogue states are assigned a high value close to unity (e.g. dialogue state 13), because of the insufficient amount of training material for the dialogue-state dependent model, the interpolation weight for the global model in Figure 2 is remarkably smaller for dialogue state 13 and several other dialogue-state dependent models are included in the combined language model.

Table 4 presents the word error rates (WER) on the word lattice with the different language models. The graph error rate of the word lattice we have used is 7.2%. Although the reduction in WER between the **BOTH** and **ALL** trigram language model is very small, it indicates, that the large number of interpolation weights can be estimated reliably with our method. Our experiments show, that the linear combination of several dialogue-state dependent models is able to detect similarities between different dialogue states and can therefore be use to exploit supplementary information contained in the different language models.

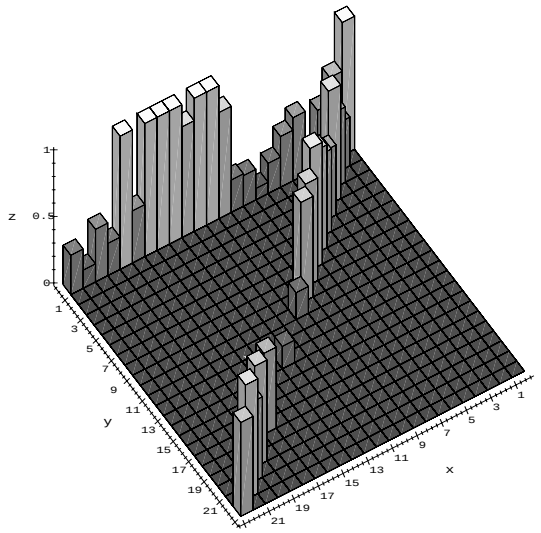

Figure 1: Interpolation weights using a linear interpolation between the dialogue-state dependent language models and the global language model.

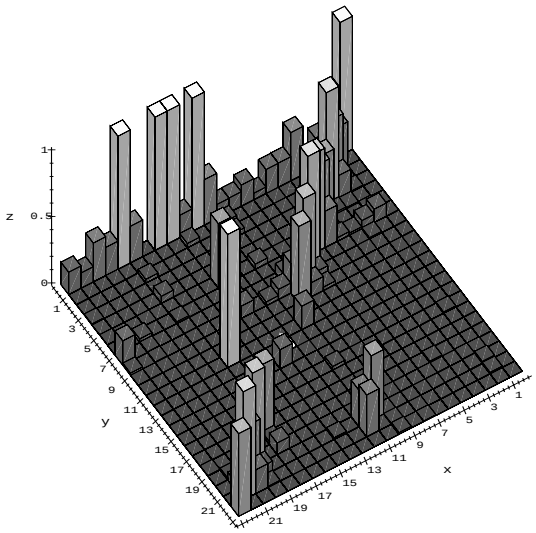

Figure 2: Interpolation weights using all dialogue-state dependent and the global language model for each dialogue state.

Table 4: Word error rates for the different language models

trigram LM	Perplexity	errors [%] del / ins / WER
GLOB	12.1	2.1 / 2.7 / 14.3
DEP	10.6	2.1 / 2.7 / 14.2
BOTH	9.6	1.9 / 2.5 / 13.6
ALL	9.5	1.9 / 2.5 / 13.5

6. CONCLUSION

We have presented experiments with dialogue-state dependent language models on a very small Dutch database which has been acquired with an automatic train timetable information system in the Netherlands. We have defined and investigated two models which are based on the linear interpolation between several of the dialogue-state dependent and a global, dialogue-state independent model and we have trained the interpolation weights on the training corpus using Leaving-One-Out probabilities. Our experiments indicate that the parameters can be estimated reliably, despite the very small number of words in the language model training corpus. The perplexity on the test corpus has been reduced by 27% and the word error rate by 6% relative, from 14.3% with a dialogue-state independent language model to 13.5% with our best dialogue-state dependent model.

We will obtain a larger database consisting of 12000 dialogues in the close future. With this additional training material we expect a more distinct effect of the combined model on the word error rate.

7. REFERENCES

[1] W. Eckert, F. Gallwitz, H. Niemann: 'Combining Stochastic and Linguistic Language Models for Recognition of Spontaneous Speech', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1996, Atlanta, USA, pp. 423-426, May 1996.

[2] S. Martin, J. Liermann, H. Ney: 'Adaptive Topic-Dependent Language Modeling Using Word-Based Varigrams', in Proc. Fifth European Conference on Speech Communication and Technology, Rhodes, Greece, pp. 1447-1450, September 1997.

[3] J. Mariani, L. Lamel, 'An Overview of EU Programs Related to Conversational / Interactive Systems', in Proc. of the 1998 Broadcast News Transcription and Understanding Workshop, Lansdowne, USA, pp. 247-253, February 1998.

[4] H. Ney, L. Welling, S. Ortmanns, K. Beulen, F. Wessel: 'The RWTH Large Vocabulary Continuous Speech Recognition System', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1998, Seattle, USA, pp. 853-856, May 1998.

[5] H. Ney, S. Martin, F. Wessel: 'Statistical Language Modeling Using Leaving-One-Out', in 'Corpus Based Methods in Language and Speech Processing', S. Young, G. Bloothoft (eds.), pp. 174-207, Kluwer Academic Publishers, The Netherlands, 1997.

[6] C. Popovici, P. Baggia: 'Specialized Language Models Using Dialogue Predictions', in Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing 1997, Munich, Germany, pp. 815-8181, April 1997.

[7] N. Reithinger, E. Maier: 'Using Statistical Dialogue Act Processing in Verbmobil', in Proc. 33rd Annual Meeting of the Association for Computational Linguistics 1995, Cambridge, USA, pp. 116-121, June 1995.

[8] F. Wessel, S. Ortmanns, H. Ney: 'Implementation of Word Based Statistical Language Models', in Proc. SQEL (*Spoken Queries in European Languages*) Workshop on Multi-Lingual Information Retrieval Dialogues, Pilsen, Czech Republic, pp. 55-59, April 1997.