LANGUAGE MODEL ESTIMATION FOR OPTIMIZING END-TO-END PERFORMANCE OF A NATURAL LANGUAGE CALL ROUTING SYSTEM

Vaibhava Goel, Hong-Kwang Jeff Kuo, Sabine Deligne, Cheng Wu

IBM T. J. Watson Research Center, Yorktown Heights, NY 10598.
{vgoel,hkuo,deligne,chengwu}@us.ibm.com

ABSTRACT

Conventional methods for training statistical models for automatic speech recognition, such as acoustic and language models, have focused on criteria such as maximum likelihood and sentence or word error rate (WER). However, unlike dictation systems, the goal for spoken dialogue systems is to understand the meaning of what a person says, not to get every word correctly transcribed. For such systems, we propose to optimize the statistical models under end-to-end system performance criteria. We illustrate this principle by focusing on the estimation of the language model (LM) component of a natural language call routing system. This estimation, carried out under a conditional maximum likelihood objective, aims at optimizing the call routing (classification) accuracy, which is often the criterion of interest in these systems. LM updates are derived using the extended Baum-Welch procedure of Gopalakrishnan et.al. In our experiments, we find that our estimation procedure leads to a small but promising gain in classification accuracy. Interestingly, the estimated language models also lead to an increase in the word error rate while improving the classification accuracy, showing that the system with the best classification accuracy is not necessarily the one with the lowest WER. Significantly, our LM estimation procedure does not require the correct transcription of the training data, and can therefore be applied to unsupervised learning from un-transcribed speech data.

1. INTRODUCTION

Spoken dialogue systems are becoming an important means of customer relations management over the telephone, having shown great value in reducing costs as well as improving the customer experience. Automatic speech recognition (ASR) systems used in spoken dialogue systems are mostly based on statistical models; in particular, the maximum *a posterior* rule is used to find the best word sequence $\widetilde{W}$ for a given acoustic speech signal A:

$$\widetilde{W} = \operatorname*{argmax}_W P(W|A) = \operatorname*{argmax}_W \frac{P(A|W)P(W)}{P(A)}, \quad (1)$$

where $P(A|W)$ is the acoustic model, and $P(W)$ is the language model.

Traditionally, acoustic and language models have been trained separately based on the maximum likelihood criterion. Because the model assumptions are often incorrect and the amount of data used to train the models insufficient, maximum likelihood training yields suboptimal solutions. Discriminative training of acoustic [1, 2] and language models [3] using the Maximum Mutual Information (MMI) [4] or Minimum Classification Error (MCE) [5] criteria often results in better speech recognition performance, in terms of reduction in sentence or word error rates. There have also been attempts at adaptation of language models for spoken dialogue systems [6, 7] but these also largely aim at reducing the word or sentence error rate.

For dictation systems, a low word error rate could be an appropriate goal to strive for. However, for spoken dialogue systems, the goal is to understand what the caller is saying (e.g. asking for information or to perform a transaction) rather than to get every single word correct. Thus metrics such as concept accuracy or classification accuracy may be more relevant criterion functions that should be optimized.

As an example, we consider the case of natural language call routing, where the goal is to automatically route a caller to the agent with the best expertise to handle the problem described by the caller in completely natural language, e.g. "hi there i was wondering um my car is kind of old and has been breaking down frequently just wondering if you've got good rates." This caller is routed to the Consumer Lending Department; he/she was not required to know or remember the correct name of the department. A common way of performing call routing is to train a statistical classifier and route to the destination (or class) $\widetilde{C}$ according to:

$$\widetilde{C} = \operatorname*{argmax}_{C_k} P(C_k|A) \approx \operatorname*{argmax}_{C_k} P(C_k|\widetilde{W}), \quad (2)$$

where $\widetilde{W}$ is the output of the speech recognizer. Traditionally, the classification model has been trained using maximum likelihood or heuristics popularly used in information retrieval, but discriminative training [8] has been shown to dramatically improve performance and reduce the requirements for domain knowledge.

Thus in natural language call routing, the objective is to minimize the routing (classification) error rate (CER), rather than the word error rate (WER). It is thus suboptimal to estimate the acoustic and language models in isolation; instead it may be beneficial to jointly estimate the acoustic, language, and classification models to optimize the CER. In this paper, we consider how to improve the call classification performance through tighter coupling of the models. In Section 2, we describe in more detail how to achieve tighter model coupling using lattices or N-best lists. In Section 3, we describe a novel method of re-estimating the language model using a training method that directly optimizes the classification accuracy objective. In Section 4, we describe our experimental setup and in Section 5, we present some preliminary results. In Section 6, we conclude with some discussions.

2. MODEL COUPLING

Using the single best word sequence hypothesis from the recognizer as input to the call classifier (as in Equation 2) is clearly suboptimal because of recognition errors. Better accuracy can be achieved by using a word lattice, sausage, or N-best list from the

speech recognizer. [9, 10] Using a list of N-best hypotheses from the recognizer, a better classification rule is

$$\widetilde{C} = \underset{C_k}{\operatorname{argmax}}\, P(C_k|A) \approx \underset{C_k}{\operatorname{argmax}} \sum_n P(C_k|W_n)P(W_n|A), \quad (3)$$

where W_n is the n^{th} best word sequence hypothesis. The posterior probability of class C_k given the acoustic signal can be further expanded as

$$P(C_k|A) \approx \sum_n P(C_k|W_n)\frac{P(A|W_n)P(W_n)}{P(A)} \quad (4)$$

$$= \frac{\sum_n P(C_k|W_n)P(A|W_n)P(W_n)}{\sum_j P(A|W_j)P(W_j)}. \quad (5)$$

An appropriate objective that is matched to the classification rule of Equation 3 is the following conditional probability

$$R(\Lambda, \theta, \Phi) = \prod_{i=1}^{M} P(C_i^*|A_i, \Lambda, \theta, \Phi), \quad (6)$$

where Λ, θ, and Φ are the acoustic, language, and classification models respectively, M is the total number of training sentences, and C_i^* is the correct class label for sentence i. We expect that $R(\Lambda, \theta, \Phi)$ is correlated with the classification accuracy and that its maximization with respect to the component models would lead to an increase in accuracy. In this paper we focus on estimating only the language model, while keeping other models constant.

3. LANGUAGE MODEL PARAMETER ESTIMATION

Our baseline language model is a modified Kneser-Ney interpolated bigram [11]; it defines the bigram probability of word w_2 conditioned on word w_1 as

$$P(w_2|w_1) = f(w_2|w_1) + b(w_1)u(w_2), \quad (7)$$

where $f(w_2|w_1)$ is a discounted bigram probability, $b(w_1)$ is a history dependent interpolation weight, and $u(w_2)$ is a unigram probability. These are estimated from training data as [11]

$$f(w_2|w_1) = \frac{c(w_1w_2) - \gamma(c(w_1w_2))}{\sum_{w_1} c(w_1w_2)} \quad (8)$$

$$b(w_1) = 1 - \sum_{w_2} f(w_2|w_1). \quad (9)$$

$c(w_1w_2)$ is the count of word pair w_1w_2 in the training data, and $\gamma(c(w_1w_2))$ is a count dependent discounting weight. The unigram $u(w_2)$ in (7) is chosen so that the unigram marginals of the resulting bigram match the data marginals. $u(w_2)$ is sometimes further interpolated with a uniform distribution to give a non-zero probability to unseen words. An excellent in-depth discussion of modified Kneser-Ney parameter estimation is given by Chen and Goodman [11].

The parameter estimation under the objective of Equation 6 is carried out using the extended Baum-Welch procedure of Gopalakrishnan et. al. [4]; this procedure is chosen because Equation 6 is a rational function of polynomials. To keep the estimated language model smooth, we update only the $f(w_2|w_1)$ portion of the bigram model while keeping $b(w_1)$ and $u(w_2)$ fixed at their original values. This choice of keeping the interpolation weights fixed was made somewhat arbitrarily, to keep the smoothing mass of each history unchanged from the ML estimated model of (7).

Applying the extended Baum-Welch procedure, we can derive the following parameter update equations

$$\hat{f}(w_2|w_1) = (1 - b(w_1))\frac{c^{num}(w_1w_2) - c^{den}(w_1w_2) + Df(w_2|w_1)}{\sum_{w_2} c^{num}(w_1w_2) - c^{den}(w_1w_2) + Df(w_2|w_1)}, \quad (10)$$

where $c^{num}(w_1w_2)$ and $c^{den}(w_1w_2)$ denote numerator and denominator counts, respectively, obtained from a list of N most likely state sequences corresponding to each training sentence

$$c^{num}(w_1w_2) = \sum_i \sum_{W_n \in \mathcal{N}_i} P_{\theta^0}(W_n|C_i^*, A_i) \sum_{w_1w_2 \in W_n} \frac{f(w_2|w_1)}{f(w_2|w_1) + b(w_1)u(w_2)}, \quad (11)$$

$$c^{den}(w_1w_2) = \sum_i \sum_{W_n \in \mathcal{N}_i} P_{\theta^0}(W_n|A_i) \sum_{w_1w_2 \in W_n} \frac{f(w_2|w_1)}{f(w_2|w_1) + b(w_1)u(w_2)}. \quad (12)$$

$\mathcal{N}_i$ denotes the N-best list corresponding to utterance i. θ^0 is used to denote the language model parameter values at the current iteration.

The posterior probabilities of W_n used in Equations 11 and 12 are obtained from the N-best lists and current language model as

$$P_{\theta^0}(W_n|C_i^*, A_i) = \frac{P(C_i^*|W_n)P_{\theta^0}(W_n, A_i)^\alpha}{\sum_{W \in \mathcal{N}_i} P(C_i^*|W)P_{\theta^0}(W, A_i)^\alpha} \quad (13)$$

$$P_{\theta^0}(W_n|A_i) = \frac{P_{\theta^0}(W_n, A_i)^\alpha}{\sum_{W \in \mathcal{N}_i} P_{\theta^0}(W, A_i)^\alpha}. \quad (14)$$

The parameter α is a *log-likelihood scale* that is used to get "reasonable" posterior probabilities; it is tuned on a held out set as discussed below in Section 5. Note that the LM update according to Equation 10 does not involve the knowledge of the reference word script.

The extended Baum-Welch procedure [4] suggests a D value to be used in (10). However, instead of using this value, we follow a strategy that is analogous to choosing D for conditional maximum likelihood (CML) estimation of acoustic models [1] and was found to be useful in CML estimation of language models [12]. We select D as

$$D = \lambda D^* \quad (15)$$

$$D^* = \max_{w_1, w_2} \frac{c^{num}(w_1w_2) - c^{den}(w_1w_2)}{f(w_2|w_1)}, \quad (16)$$

where $\lambda \geq 1.0$ is an empirically selected parameter. We note that this choice of D simply ensures positivity of the numerator on RHS in (10), and consequently ensures validity of $\hat{f}(w_2|w_1)$. We experimented with history w_1 dependent $D(w_1)$ values, something that is also found to be of value in CML estimation of acoustic models [1]. Details of these experiments are reported in Section 5.

4. EXPERIMENTAL SETUP

The experiments reported in this paper were conducted on an IBM internal corpus collected for a natural language call routing system. The system routes incoming calls to one of 35 destinations.

The training data consisted of 27685 sentences containing 180K words. These sentences were divided into a *trainset* (24503 sentences) that was used in model training and a *devset* (3182 sentences) set that was used for selecting heuristic parameters. A separate data set containing 5589 sentences and 38K words was used

iteration	λ	trainset objective function	trainset classification accuracy	devset objective function	devset classification accuracy	testset objective function	testset classification accuracy
0		-7146	93.01	-1787	85.95	-3134	86.19
2	2.0	-6804	93.43	-1777	86.05	-3112	86.37
4	2.0	-6711	93.58	-1775	86.02	-3108	86.35
6	1.1	-6576	93.78	-1770	85.92	-3098	86.38
8	2.0	-6492	93.91	-1768	86.05	-3092	86.42
10	2.0	-6425	93.99	-1768	86.08	-3087	86.37

Table 1. N-best based classification accuracy and objective function values with iterations of language model update. Iteration 0 corresponds to the baseline language model.

as the *testset*. All of the trainset, devset, and testset sentences were hand labeled with correct destination classes.

A maximum entropy model [13] was trained as the statistical classifier $P(C_k|W)$ used for call classification. For simplicity, only single word features (unigrams) were used, and the model was trained on the transcribed text of the trainset.

The acoustic model training data consists of about 1000 hours of audio data. The acoustic feature vectors were obtained by first computing 13 Mel-cepstral coefficients (including energy) for each time slice under a 25.0 msec. window with a 10 msec. shift. Nine such vectors were concatenated and projected to a 60 dimensional space using LDA. An acoustic model was built on these features with a phone set containing 50 phones. Each phone was modeled with a three state left to right HMM. This, in addition to six states of two silence phones, resulted in 156 context independent states which were decision-tree clustered into 2198 context dependent states and modeled using state dependent Gaussian Mixture Models. There are 222620 Gaussians all together in the acoustic model.

This acoustic model was MAP adapted [14] using the in-domain training data (trainset) with a weight that was found to yield minimum word error rate on the devset. The resulting acoustic model was the one used in all the experiments reported in this paper.

The language model that formed the baseline for our experiments was a bigram model described in Equation 7. This model was estimated on the trainset using an IBM internal language modeling toolkit developed by Stanley Chen.

Using the acoustic and language models described above, 500-best lists were generated for trainset, devset, and testset. The top (most likely) hypothesis in the testset had word/ sentence error rates of 23.30/ 46.02% . On the devset, the word/ sentence error rates of the top hypothesis were 23.17/ 47.05% and on the train set they were 14.81/ 35.89%.

5. EXPERIMENTAL RESULTS

5.1. One-Best Vs N-Best Based Classification

As our first experiment, we compared the classification accuracy of using the most likely hypothesis according to Equation 2 with that of using the N-best lists according to Equation 3.

The testset classification accuracy using the most likely hypothesis was 85.42%, for the devset it was 84.70%, and for the trainset it was 92.23%. To see the effect of ASR on classification accuracy, we computed the accuracy using the reference text. Resulting numbers were 89.32% on testset and 89.53% on devset and 96.41% on the trainset. Thus there is an absolute degradation of about 4-5% in classification accuracy when using the top hypothesis from the ASR compared to the correct transcription.

To carry out N-best based classification according to Equation 3, the probabilities $P(W_n|A)$ were computed as described in Equation 14. A line search was carried out for parameter α and the value that gave optimal classification accuracy on the devset was chosen. This resulted in $\alpha = 2.0$ and devset classification accuracy of 85.95%. Using $\alpha = 2.0$ on the testset resulted in classification accuracy of 86.19%, a relative reduction of about 5% in classification error rate.

5.2. Updating Language Model

In the next set of experiments, we estimated the language model from N-best lists, as discussed in Section 3.

Several iterations of LM update were carried out where each iteration consisted of gathering c^{num} and c^{den} according to Equations 11 and 12, finding history dependent $D^*(w_1)$ values (Equation 16 with maximization carried out only over w_2), conducting a line search for λ in Equation 16 so as to maximize the objective function value (Equation 6) on the devset, and updating the language model (Equation 10 with D replaced by $D(w_1)$). The resulting LM was used as the starting LM for the next iteration. We note that the objective function maximization on devset was carried out in an attempt to avoid over-training the language model.

Table 1 presents the objective function values and N-best based classification (Equation 3) results on the trainset, devset, and testset. From this table, we note that the LM updates result in a significant increase in the objective function value and classification accuracy on the trainset. The improvements on the testset are small, but there is a very encouraging positive movement. Note that we stopped at iteration 10 since the objective function stopped improving on the devset. However, the trainset and testset objective function and the error rate are still improving, suggesting that other stopping criteria may be more useful.

We also looked at the effects of our language model updates on the word error rate and classification accuracy of the one-best hypothesis. Use of one-best corresponds to classification according to Equation 2 and may be useful in cases when N-best lists are only available at training time but testing is done on one-best hypothesis (possibly due to limited resources during run-time). From results presented in Table 2, we note that, interestingly (and somewhat expectedly), there is a consistent degradation in the word error rates while the test set classification accuracy increases with increasing iterations of LM updates. This reinforces our original motivation that direct optimization of the classification accuracy may be a better goal to strive for than word error rate even when there is only the most likely hypothesis available at test time.

	trainset		devset		testset	
	word error rate	classification accuracy	word error rate	classification accuracy	word error rate	classification accuracy
0	14.81	92.23	23.17	84.70	23.30	85.42
2	14.82	93.09	23.19	84.79	23.43	85.76
4	14.98	93.36	23.41	84.88	23.67	85.81
6	15.31	93.58	23.72	84.60	24.04	85.95
8	15.81	93.70	24.09	84.85	24.40	85.94
10	16.32	93.82	24.63	84.70	24.82	85.92

Table 2. One-best word error rate and classification accuracy with iterations of LM updates. Iteration 0 corresponds to the baseline language model.

We note that in the experiments reported here, the N-best lists were kept fixed. This may have provided us some protection from degrading the classification performance but may also have limited the gains that we see from this technique. One of our immediate future experiments is to re-generate N-bests at each iteration under the updated language model.

6. CONCLUSIONS

In this paper we have presented a novel language model estimation procedure that aims to directly optimize the call classification accuracy of a natural language call routing system. Rather than estimating acoustic, language, and classification models in isolation, we believe better coupling among the models and joint optimization should provide better end-to-end performance. In particular, the maximum likelihood criterion commonly used to train language models or even the word error rate (WER) metric used to benchmark speech recognition systems may not be the correct criteria to use, since they are only indirectly related to the call classification accuracy.

In contrast, we have proposed an objective function that is more closely related to the ultimate goal of classification performance. Preliminary experiments show modest improvements. The results also show that improvements in classification accuracy may be uncorrelated with the word error rate, providing evidence for our hypothesis that end-to-end optimization of the classification accuracy is more important than optimizing the WER. Importantly, our objective function does not require knowledge of the correct transcription. Therefore, our proposed algorithm can be used in unsupervised training of language models using un-transcribed speech, and can potentially provide substantial gains.

7. ACKNOWLEDGMENTS

The authors thank Stanley Chen for providing his marvelous codebase infrastructure that enabled the experiments described in this paper. We also thank EE Jan for help with the baseline language model and other members of the Human Languages Technology department who provided help with this project.

8. REFERENCES

[1] P. Woodland and D. Povey, "Large scale discriminative training of hidden Markov models for speech recognition," *Computer Speech and Language*, vol. 16, pp. 25–47, 2002.

[2] B.-H. Juang, W. Chou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 3, pp. 257–265, May 1997.

[3] H.-K. J. Kuo, E. Fosler-Lussier, H. Jiang, and C.-H. Lee, "Discriminative training of language models for speech recognition," in *Proc. ICASSP 2002*, Orlando, Florida, May 2002.

[4] P. Gopalakrishnan, D. Kanevsky, A. Nadas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Transactions on Information Theory*, vol. 37, pp. 107–113, 1991.

[5] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2345–2373, Nov. 1998.

[6] G. Riccardi, A. Potamianos, and S. Narayanan, "Language model adaptation for spoken language systems," in *Proc. ICSLP 1998*, Sydney, Australia, Dec. 1998.

[7] B. Souvignier and A. Kellner, "Online adaptation for language models in spoken dialogue systems," in *Proc. ICSLP 1998*, Sydney, Australia, Dec. 1998.

[8] H.-K. J. Kuo and C.-H. Lee, "Discriminative training of natural language call routers," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 1, pp. 24–45, Jan. 2003.

[9] G. Tür, D. Hakkani-Tür, and G. Riccardi, "Extending boosting for call classification using word confusion networks," in *Proc. ICASSP 2004*, Montreal, Canada, May 2004.

[10] G. Tür, J. Wright, A. Gorin, G. Riccardi, and D. Hakkani-Tür, "Improving spoken language understanding using confusion networks," in *Proc. ICSLP 2002*, Denver, Colorado, Sept. 2002.

[11] S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Tech. Rep. TR-10-98*, Center for Research in Computing Technology, Harvard University, MA, 1998.

[12] V. Goel, "Conditional maximum likelihood estimation for improving annotation performance of N-gram models incorporating stochastic finite state grammars," in *Proc. ICSLP 2004*, Jeju Island, Korea, Oct. 2004.

[13] A. L. Berger, S. A. D. Pietra, and V. J. D. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[14] J. L. Gauvain and C.-H. Lee, "Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.