WORD CONFIDENCE CALIBRATION USING A MAXIMUM ENTROPY MODEL WITH CONSTRAINTS ON CONFIDENCE AND WORD DISTRIBUTIONS

Dong Yu [a], *Shizhen Wang* *[b], *Jinyu Li* [a], *Li Deng* [a]

[a] Microsoft Corporation, One Microsoft Way, Redmond, WA 98034, USA
[b] University of California, Los Angeles, CA 90095, USA
dongyu@microsoft.com, szwang@ee.ucla.edu, jinyli@microsoft.com, deng@microsoft.com

ABSTRACT

It is widely known that the quality of confidence measure is critical for speech applications. In this paper, we present our recent work on improving word confidence scores by calibrating them using a small set of calibration data when only the recognized word sequence and associated raw confidence scores are made available. The core of our technique is the maximum entropy model with distribution constraints which naturally and effectively make use of the word distribution, the raw confidence-score distribution, and the context information. We demonstrate the effectiveness of our approach by showing that it can achieve relative 38% mean square error (MSE), 39% negative normalized likelihood (NNLL), and 23% equal error rate (EER) reduction on a voice mail transcription data set and relative 35% MSE, 45% NNLL, and 35% EER reduction on a command and control data set.

Index Terms— confidence calibration, confidence measure, maximum entropy, distribution constraint, word distribution

1. INTRODUCTION

Despite the significant progress made in improving automatic speech recognition (ASR) accuracy over the last three decades, the recognition results of spontaneous ASR systems still contain a large amount of errors, esp. under the noisy conditions. For speech applications (e.g., interactive dialog systems) to make wise decisions, it is important for the ASR engines to provide speech applications with the word confidence score representing an estimate of the likelihood that each word is correctly recognized.

Numerous techniques have been developed over the past years to improve the quality of the confidence measures [1]. These techniques can be classified into three categories. Techniques in the first category build a two-class (true or false) classifier based on the information (e.g., acoustic and language model scores) obtained from the ASR engine. The confidence measure on a specific word is then considered as the likelihood that the classifier's output is true. Techniques in the second category consider the posterior probability of a word given the acoustic signal, which is typically estimated from the ASR lattices, as the confidence measure. Techniques in the third category consider the confidence estimation problem as an utterance verification problem and use the likelihood ratio between the null hypothesis (the word is correct) and the alternative hypothesis (the word is incorrect) as the confidence measure.

No matter which technique is used, the confidence measure is typically provided by the ASR engines which use one fixed set of model parameters, trained on a generic data set, for all applications. This approach has two drawbacks. First, the data used to train the confidence measure may differ vastly from the real data observed in a specific speech application due to different language models used and different environments in which the applications are deployed. Second, some information such as distribution of the words (see Section 2.2 for detailed discussions) cannot be used in the generic confidence model since such information is application specific and cannot be reliably estimated from the generic data set. As a result, the confidence measure provided by the ASR engines can be far from optimal for a specific application.

In this paper we propose to improve the quality of confidence measure by calibrating (post-processing) it for each specific application. We assume that we have access to a small amount of transcribed calibration data collected under the real usage scenario for the specific application. We further assume that the only information we can obtain from the ASR engines is the recognized word sequence and the associated "raw" confidence scores. Our calibration technique described in this paper is especially useful for dialog application developers who cannot modify the confidence estimation module and/or access to the information inside the ASR engine.

Given a set of N confidence scores and the associated labels $\{(c_i \in [0,1], y_i \in \{0,1\}) \mid i = 1, \cdots, N\}$, where $y_i = 1$ if the word is correct and $y_i = 0$ otherwise, the quality of confidence measure can be evaluated using four popular criteria. The first criterion is mean square error (MSE):

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(c_i - y_i)^2. \quad (1)$$

The second criterion is negative normalized log-likelihood (NNLL):

$$NNLL = -\frac{1}{N}\sum_{i=1}^{N} log\big(c_i\delta(y_i = 1) + (1 - c_i)\delta(y_i = 0)\big), \quad (2)$$

where $\delta(x) = 1$ if x is true and 0 otherwise. The third criterion is equal error rate (EER). And the fourth criterion is the detection error trade-off (DET) curve [2], the crossing of which with the $(0,0) - (1,1)$ diagonal line gives the EER.

The confidence calibration approach proposed in this paper is based on our recently developed maximum entropy (MaxEnt)

* Shizhen Wang contributed to this work when he was an intern at Microsoft Research.

model with distribution constraints [6] and uses both the raw confidence score distribution and the word distribution information. We demonstrate the effectiveness of our approach in this paper by showing that it can achieve relative 38% MSE, 39% NNLL, and 23% EER reduction on a voice mail transcription (VM) data set and relative 35% MSE, 45% NNLL, and 35% EER reduction on a command and control (C&C) data set.

The rest of the paper is organized as follows. In Section 2 we review the MaxEnt model with distribution constraints (MaxEnt-DC) and the specific treatment needed for continuous features and multi-valued nominal features. In Section 3 we first argue that the word distribution information differs vastly for different applications and hence should be effectively exploited to calibrate the confidence scores. We then describe three different approaches to exploiting the word distribution information. We evaluate our approach empirically on a VM data set and a C&C data set in Section 4, and conclude the paper in Section 5.

2. MAXIMUM ENTROPY MODEL WITH DISTRIBUTION CONSTRAINTS

The MaxEnt model with moment constraints (MaxEnt-MC) is a popular discriminative model that is widely used for classifier design. Given an N-sample training set $\{(x_n, y_n) \mid n = 1, \cdots, N\}$ and a set of M features $f_i(x, y), i = 1, \cdots, M$ defined on the input x and output y, the posterior probability

$$p(y|x; \boldsymbol{\lambda}) = \frac{1}{Z_{\boldsymbol{\lambda}}(x)} \exp\left(\sum_i \lambda_i f_i(x, y)\right) \tag{3}$$

is in a log-linear form, where $Z_{\boldsymbol{\lambda}}(x) = \sum_y \exp(\sum_i \lambda_i f_i(x, y))$ is a normalization constant to fulfill the probability constraint $\sum_y p(y|x) = 1$, and λ_i is optimized to maximize the log-conditional-likelihood

$$O(\boldsymbol{\lambda}) = \sum_{n=1}^{N} \log p(y_n|x_n) \tag{4}$$

over the whole training set.

While the MaxEnt-MC model can achieve impressive classification accuracy when binary features are used, it was not as successful when continuous features are used. We have recently developed the MaxEnt model with distribution constraints (MaxEnt-DC) [6] and proposed that the information carried in the feature distributions be used to improve classification performance. Our model is a natural extension to the MaxEnt-MC model by observing that the moment constraints are the same as the distribution constraints for binary features.

To use the MaxEnt-DC model, features are first classified into three categories: binary, continuous, and multi-valued nominal features. For the binary features, the distribution constraint is the same as the moment constraint and so no change is needed. For the continuous features, each feature $f_i(x, y)$ is expanded to K features

$$f_{ik}(x, y) = a_k\big(f_i(x, y)\big) f_i(x, y), \tag{5}$$

where $a_k(.)$ is a weight function whose definition and calculation method can be found in [5][6][7] and the number K needs to be determined based on the amount of training data available. For the multi-valued nominal features, the feature values are sorted first in the descending order of their number of occurrences. The top $J-1$ nominal values are then mapped into token IDs in $[1, J-1]$, and all remaining nominal values are mapped into the same token ID J, where J is chosen to guarantee the distribution of the nominal features can be reliably estimated. Each feature $f_i(x, y)$ is subsequently expanded to J features

$$f_{ij}(x, y) = \delta(f_i(x, y) = j). \tag{6}$$

After the feature expansion for the continuous and the multi-valued nominal features, the posterior probability in the MaxEnt-DC model can be evaluated as

$$\begin{aligned} p(y|x) = \frac{1}{Z_{\boldsymbol{\lambda}}(x)} \exp\Big(&\sum_{i \in \{binary\}} \lambda_i f_i(x, y) \\ &+ \sum_{i \in \{continuous\},k} \lambda_{ik} f_{ik}(x, y) \\ &+ \sum_{i \in \{nominal\},j} \lambda_{ij} f_{ij}(x, y)\Big) \end{aligned} \tag{7}$$

and parameter estimation can be carried out in the same way as that used in the MaxEnt-MC model. In our experiments we have used the RPROP [3] training algorithm.

We have applied the MaxEnt-DC model to several tasks [6] [8] and consistently observed improvement over the MaxEnt-MC model when sufficient training data is available. In the work described in this paper, we use the MaxEnt-DC model to calibrate the confidence scores. As the related earlier work, White et al. [4] used the MaxEnt-MC model for confidence measurement in speech recognition. Our approach performs significantly better than the earlier approaches in that we used the MaxEnt-DC model with the constraints on both continuous raw confidence scores and multi-valued word tokens.

3. INFORMATION SOURCES AND FEATURES

In our confidence calibration setting, it is assumed that we only have access to the word and "raw" confidence score sequences of

$$\left\{x_{n,t} = \begin{bmatrix} w_{n,t} \\ c_{n,t} \end{bmatrix} \middle| t = 1, \cdots, T\right\} \tag{8}$$

from the ASR engine, where $w_{n,t}$ is the t-th word in the n-th utterance and $c_{n,t}$ is the associated confidence score. The goal of confidence calibration is to derive a better confidence score $c'_{n,t} = p\big(y_{n,t}\big|x_{n,t}; \boldsymbol{\lambda}\big)$ for each word $w_{n,t}$ from the raw scores. We also assume that we have a training (calibration) set that tells us whether each recognized word is correct (true) or not (false), from which we train the parameters of the calibration model.

At first glance, there seems to be little information we can exploit to calibrate the raw confidence score from the ASR engine. The information at hand is the current word's confidence score $c_{n,t}$ and the previous and next words' confidence scores $c_{n,t-1}$ and $c_{n,t+1}$, since an error in one place can affect the adjacent words. After a careful examination, however, we noticed that the recognized word itself also contains information as shown in Table I, where the top 10 words and their frequencies in VM and C&C data sets are displayed. From the table we observe that the distributions are significantly different across words and tasks. Hence, constraints on the distribution of the words would supply useful information to the MaxEnt model. In addition, the distribution of the confidence scores across words is also vastly different, and hence, constraints on the joint distribution of words and confidence scores can also help.

The above analysis suggests three ways of using the word and confidence distribution information in the MaxEnt-DC model. In the first approach, we construct four features

$$f_1(x_{n,t}, y_{n,t}) = \begin{cases} c_{n,t} & if\ y_{n,t} = true \\ 0 & otherwise \end{cases} \quad (9)$$

$$f_2(x_{n,t}, y_{n,t}) = \begin{cases} c_{n,t} & if\ y_{n,t} = false \\ 0 & otherwise \end{cases} \quad (10)$$

$$f_3(x_{n,t}, y_{n,t}) = \begin{cases} w_{n,t} & if\ y_{n,t} = true \\ 0 & otherwise \end{cases} \quad (11)$$

$$f_4(x_{n,t}, y_{n,t}) = \begin{cases} w_{n,t} & if\ y_{n,t} = false \\ 0 & otherwise \end{cases} \quad (12)$$

for each frame (n,t). If context information is used, features constructed for the previous and next frames can be exploited. In this first approach the weight on the raw confidence score is shared across all the words. However, different bias weights are used for different words since

$$p(y|x) = \frac{1}{Z_{\lambda}(x)} \exp\left(\sum_{i\in\{1,2\},k} \lambda_{ik} f_{ik}(x,y) + \sum_{i\in\{3,4\},j} \lambda_{ij}\, \delta(f_i(x,y) = j) \right) \quad (13)$$

TABLE I
TOP 10 WORDS AND THEIR FREQUENCIES IN THE VOICE MAIL TRANSCRIPTION AND COMMAND AND CONTROL DATA SETS

VM			C&C		
word	**count**	**percentage**	**word**	**count**	**percentage**
i	463	3.03%	three	716	4.81%
you	451	2.95%	two	714	4.80%
to	446	2.92%	five	713	4.79%
the	376	2.46%	one	691	4.64%
and	369	2.42%	seven	651	4.38%
uh	356	2.33%	eight	638	4.29%
a	302	1.98%	six	627	4.21%
um	287	1.88%	four	625	4.20%
that	215	1.41%	nine	616	4.14%
is	213	1.39%	zero	485	3.26%

In the second approach the distribution of the words and confidence scores are jointly modeled and two features

$$f_{2j-1}(x_{n,t}, y_{n,t}) = \begin{cases} c_{n,t} & if\ w_{n,t} = j\ \&\ y_{n,t} = true \\ 0 & otherwise \end{cases} \quad (14)$$

$$f_{2j}(x_{n,t}, y_{n,t}) = \begin{cases} c_{n,t} & if\ w_{n,t} = j\ \&\ y_{n,t} = false \\ 0 & otherwise \end{cases} \quad (15)$$

are constructed for each word token j at each frame. This approach essentially uses distinct weights on the raw confidence score but share the same bias weight for different words.

In the third approach we add two more features as in (11) and (12) for each frame, in addition to the features used in the second approach. This approach uses different weights on the confidence scores and different bias weights for different words.

4. EMPIRICAL EVALUATION

To evaluate the effectiveness of the confidence calibration technique we just described, we have conducted a series of experiments on two data sets: VM and C&C. Table II summarizes the number of utterances and words in the training (calibration), development, and test sets in each data set. The word error rate (WER) obtained from a speaker-independent ASR engine on the test sets are 28% and 8%, respectively, for the VM and C&C data sets. The confidence measure before calibration was obtained directly from the same ASR engine which used a Gaussian mixture model classifier discriminatively trained using generic training sets unrelated to these two data sets.

Table III compares different approaches using the MSE, NNLL, and EER criteria. A setting with continuous features expanded to k features, with w-th approach used to incorporate the word distribution information, and with the adjacent words' information used ($c = 1$) or not used ($c = 0$) is denoted as KkWwCc. $w = 0$ indicates that the word distribution information is not used. In all these settings, we assign a unique token ID for words that occur more than 20 times in the training (calibration) set and assign a same token ID to all other words. This yields 133 and 109 word tokens (i.e., J=133 and 109) in the VM and C&C calibration models respectively. Note that with the same threshold (20 in this case) J will be automatically reduced when smaller calibration set is available. Further improvements can be obtained by tuning the threshold but won't affect the main message.

TABLE II
SUMMARY OF DATA SETS

	VM		C&C	
	# utterances	# words	# utterances	# words
train	352	15274	4381	14877
dev	368	15265	4391	14642
test	371	15300	4371	15164

TABLE III
CONFIDENCE QUALITY COMPARISON USING DIFFERENT FEATURES AND APPROACHES

	VM			C&C		
	MSE	NNLL	EER	MSE	NNLL	EER
No Calibration	**0.235**	**0.749**	**33.8**	**0.085**	**0.362**	**32.7**
K1W0C0	0.177	0.532	33.7	0.059	0.226	32.7
K4W0C0	0.177	0.531	33.8	0.059	0.223	32.7
K1W0C1	0.171	0.515	31.7	0.058	0.219	32.3
K4W0C1	0.171	0.514	31.9	0.057	0.217	30.2
K1W1C0	0.149	0.458	27.4	0.055	0.202	23.4
K4W1C0	0.149	0.458	27.5	0.055	0.202	23.1
K1W1C1	0.146	0.449	26.3	0.054	0.200	22.4
K4W1C1	0.145	0.447	26.6	0.054	0.200	21.7
K1W2C1	**0.146**	**0.455**	**26.1**	**0.055**	**0.198**	**21.1**
K4W2C1	0.155	0.480	27.6	0.057	0.209	21.5
K1W3C1	0.145	0.451	26.6	0.055	0.203	21.7
K4W3C1	0.153	0.474	27.7	0.056	0.204	23.2

From Table III, we make several observations. First, if neither the word distribution nor the adjacent words' information is used (settings K1W0C0 and K4W0C0), no EER reduction can be obtained. However, we still can reduce MSE and NNLL by relatively 25% and 29% on the VM test set and 31% and 38% on the C&C test set, respectively. This indicates that even without using additional information, our calibration approach can still make the confidence scores more closely related to the probability that the word is correct. Second, if the adjacent words' confidence scores are used but the word distribution information is not used (settings K1W0C1 and K4W0C1), MSE, NNLL, and EER can all be improved. However, only 6% and 8% relative EER can be achieved and additional MSE and NNLL reduction over the settings K1W0C0 and K4W0C0 is very small. This means although the adjacent words' confidence scores carry information, the improvement they bring is relatively small. This conclusion is

corroborated by comparing the results obtained under settings K1W1C1 and K4W1C1 with that achieved under settings K1W1C0 and K4W1C0. Third, if the word distribution information is used (settings K*W1C*, K*W2C*, and K*W3C*), significant MSE, NNLL, and EER reduction is achieved no matter how the word distribution information is used. For example, the K4W1C1 setting outperforms the no-calibration setting with relative MSE, NNLL, and EER reductions by 38%, 40%, 21%, respectively, on the VM test set, and 36%, 45%, 34% on the C&C test set. Similarly, the K1W2C1 setting reduces the MSE, NNLL, and EER by 38%, 39%, and 23% on the VM test set and 35%, 45%, and 35% on the C&C test set over the no-calibration setting. Fourth, expanding the continuous features to four features helps when the word distribution information is not used or the first approach is used. But it does not help when the second and third approaches are used to exploit the word distribution information. This is due to the fact that in this case each word has its own set of confidence weight and bias, and the training (calibration) set size is not large enough especially for the words that occur only about 20 times in the calibration set. The above observations can also be made from Figs. 1 and 2 where the DET curve for the VM and C&C test sets are illustrated for the settings of no-calibration, K4W0C1, K4W1C1, and K1W2C1.

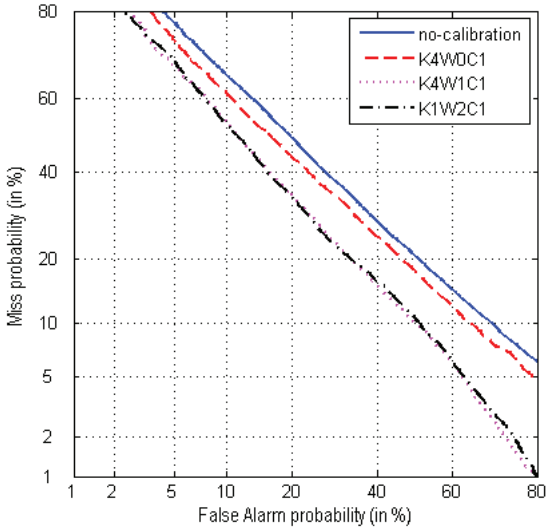

Fig. 1. The DET curves for the VM test set.

5. CONCLUSIONS

We have argued that calibrating the confidence scores for different speech applications is important and proposed to use the MaxEnt-DC model [6] to calibrate the word confidence scores by utilizing the raw confidence and word distribution information. We have shown on two data sets that our approach significantly boosted the quality of the confidence scores even though only the recognized word sequence and the associated raw confidence scores are available, which is the typical situation for most dialog application developers. We have also observed that the performance gain is mostly from using the word distribution information and the three approaches proposed to exploit this information perform equally well.

The quality of the calibrated confidence scores can be further improved if the additional information such as the N-best results and the acoustic and language model scores are available to the MaxEnt-DC model. In the companion paper [9], we discussed how the same technique can be used to significantly improve the semantic confidence measure.

6. ACKNOWLEDGEMENTS

We would like to thank Drs. Yifan Gong, Jian Wu, and Alex Acero at Microsoft Corporation, Prof. Chin-Hui Lee at Georgia Institute of Technology, and Dr. Bin Ma at Institute for Infocomm Research (I^2R), Singapore for valuable discussions. Thanks also go to Wei Zhang and Pavan Karnam at Microsoft Corporation for their help in preparing experimental data.

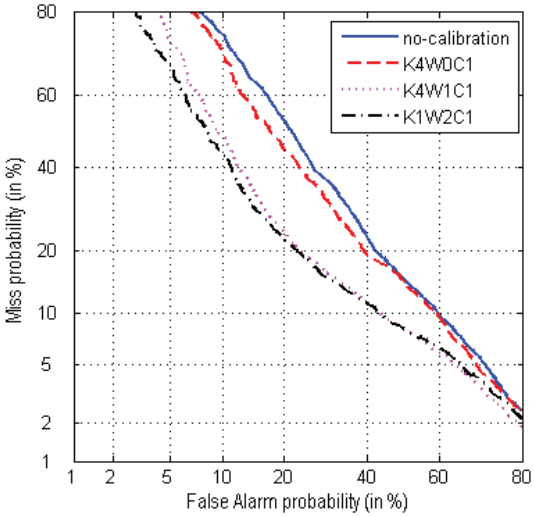

Fig. 2. The DET curves for the C&C test set.

7. REFERENCES

[1] H. Jiang, "Confidence measures for speech recognition: a survey," *Speech Communication*, vol. 45, no. 4, pp. 455-470, Apr. 2005.

[2] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve assessment of detection task performance," in Proc. *EuroSpeech*, vol. 4, pp. 1895-1898, 1997.

[3] M. Riedmiller, and H. Braun, "A direct adaptive method for faster back-propagation learning: The RPROP algorithm," in Proc. *IEEE ICNN*, vol. 1, pp. 586-591. 1993.

[4] C. White, J. Droppo, A. Acero, and J. Odell, "Maximum entropy confidence estimation for speech recognition," in Proc. *ICASSP*, vol. IV, pp. 809-812, 2007.

[5] D. Yu, L. Deng, Y. Gong, and A. Acero, "A novel framework and training algorithm for variable-parameter hidden Markov models," *IEEE trans. on Audio, Speech, and Language Processing*, vol 17, no. 7, pp. 1348-1360, September 2009.

[6] D. Yu, L. Deng, and A. Acero, "Using continuous features in the maximum entropy model", *Pattern Recognition Letters*. vol. 30, no. 8, pp.1295-1300, June, 2009.

[7] D. Yu, and L. Deng, "Solving nonlinear estimation problems using Splines," *IEEE Signal Processing Magazine*, vol. 26, no. 4, pp.86-90, July, 2009.

[8] D. Yu, L. Deng, and A. Acero, "Hidden Conditional Random Field with Distribution Constraints for Phonetic Classification," in Proc. *Interspeec* , pp. 676-679, 2009.

[9] D. Yu, L. Deng, "Semantic Confidence Calibration for Spoken Dialog Applications", in Proc. *ICASSP* 2010.