SEMANTIC CONFIDENCE CALIBRATION FOR SPOKEN DIALOG APPLICATIONS

Dong Yu and Li Deng

Microsoft Research, One Microsoft Way, Redmond, WA 98034, USA
dongyu@microsoft.com, deng@microsoft.com

ABSTRACT

The success of spoken dialog applications depends strongly on the quality of the semantic confidence measure that determines the selection of the dialog strategy. However, the semantic confidence measure obtained from typical automatic speech recognition engines is not optimized for specific semantic slots and applications. We present our recent work on using a novel maximum entropy model with distribution constraints to calibrate the semantic confidence scores with the inputs of only the raw semantic confidence and the associated raw word confidence scores. We illustrate how features can be constructed from the raw confidence scores with a variable number of words and how the quality of the semantic confidence measure can be further improved by adding another calibration stage for the word confidence measure. We demonstrate the effectiveness of our approach for two types of semantic slots of practical significance. For the ZIP-code semantic slot, the new measure achieves relative 10.6% mean square error (MSE), 19.3% normalized negative log-likelihood (NNLL), and 38.5% equal error rate (EER) reduction. The counterpart of the date-time semantic slot is 37.8%, 38.7%, and 23.1%, respectively.

Index Terms— Score calibration, confidence measure, maximum entropy, distribution constraint, semantic confidence

1. INTRODUCTION

A wide range of spoken dialog applications have been deployed in recent years (e.g., [5]). In most of these applications, the goal of each dialog turn is to fill in some semantic slots. For example, in the flight booking application, the information such as date/time and departure/destination cities is obtained from the users. In the directory assistance application [5][12], the business and city names are the key semantic information to be extracted from the users' spoken responses. In all these applications, the quality of the semantic confidence measure is critical for the system to select the most appropriate strategy at each dialog turn.

Numerous techniques have been developed over the past years to improve the quality of the confidence measures [1]. However, most of these techniques focused on improving the word confidence measures, which are only indirectly related to the desired semantic confidence measure. The semantic confidence measure has several different characteristics from the word confidence measure. First, the same semantic information can be expressed in different ways. For example, the number 1234 may be expressed as "one thousand two hundred and thirty four" or "twelve thirty four". Second, the semantic information may still be correct even if some words are incorrectly recognized. For example, there is no semantic difference when November seventh is misrecognized as November seven and vice versa. Third, some words are irrelevant or redundant in conveying the semantic information. For example, *ma'am* in "yes ma'am" and *ah* in "ah yes" do not affect the semantic information and are typically filtered out using a garbage model [6] [12].

The semantic confidence measure is typically provided as the output of a classifier, which is trained using a generic data set for all types of applications. As pointed out in [14], using a generic model is not desirable for two reasons. First, the data used to train the generic confidence measure may differ vastly from the real data observed in a specific spoken dialog application involving special language models. Second, some information is application-specific and is hence not available in training the generic confidence measure [14].

In this paper, we propose to improve the quality of the semantic confidence measure by calibrating it for each specific semantic slot using both the word and semantic confidence scores provided by the generic ASR engines. Note that calibration is different from adaptation. With calibration we do not modify the model or the parameters of the generic confidence estimation module built in the ASR engine. Instead, we post-process the raw confidence scores obtained from the ASR engine and make it better. Calibration is very important for dialog application developers since they typically have no access to the internals of the ASR engines.

The key technique used in this work is the maximum entropy (MaxEnt) model with distribution constraints (MaxEnt-DC) [9], which has been successfully used to improve the quality of word confidence measures as described in the companion paper [14]. Different from the work in [14], this paper focuses on the semantic confidence score which requires different set of features, dependents on the word confidence scores, and is more important to the spoken dialog applications.

Following [14], in this paper the quality of confidence measure is evaluated using the mean square error (MSE), negative normalized log-likelihood (NNLL), equal error rate (EER), and the detection error trade-off (DET) curve [3] defined on a set of N confidence scores c_i and the associated labels y_i $\{(c_i \in [0,1], y_i \in \{0,1\}) \mid i = 1, \cdots, N\}$ with $y_i = 1$ indicating that the semantic information is correct. The exact definitions of these criteria can be found in [14].

We demonstrate the effectiveness of our approach for two types of semantic slots of practical significance. For the ZIP-code semantic slot, the new measure achieves relative 10.6% mean square error (MSE), 19.3% normalized negative log-likelihood (NNLL), and 38.5% equal error rate (EER) reduction. The counterpart of the date-time semantic slot is 37.8%, 38.7%, and 23.1%, respectively.

The rest of the paper is organized as follows. In Section 2, we review the MaxEnt-DC model and explain the specific treatment needed for the continuous-valued features and multi-valued nominal features. In Section 3, we describe how the features can be constructed for the semantic confidence calibration purpose in the MaxEnt-DC framework and illustrate three approaches to exploiting the word confidence score distribution information. We report empirical results on two representative semantic slots in Section 4, and conclude the paper in Section 5.

2. MAXIMUM ENTROPY MODEL WITH DISTRIBUTION CONSTRAINTS

We have explained the MaxEnt-DC model and the special treatment needed for the continuous and multi-valued nominal variables in detail in [14]. For the sake of self-completeness, we briefly review it here.

The MaxEnt-DC model is an extension to the MaxEnt model with moment constraints (MaxEnt-MC), which is a popular discriminative model widely used for classifier design, e.g., for confidence measures [2][6]. MaxEnt-DC was proposed in [9] and successfully applied to several tasks [9][11]. It consistently outperforms MaxEnt-MC when sufficient training data is available.

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 input x and output y, we first classify features 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 hence no change is needed. For the continuous features, each feature $f_i(x, y)$ is expanded to K features

$$f_{ik}(x, y) = a_k\big(f_i(x, y)\big) f_i(x, y), \tag{1}$$

where $a_k(.)$ is a weight function whose definition and calculation method can be found in [8][9][10] and the number K needs to be determined based on the amount of training data available. For the multi-valued nominal features (e.g., words), the feature values are sorted first in a descending order of their occurrence frequencies. 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

$$f_{ij}(x, y) = \delta(f_i(x, y) = j). \tag{2}$$

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

$$p(y|x) = \frac{1}{Z_{\boldsymbol{\lambda}}(x)} \exp\Big(\sum_{i \in \{binary\}} \lambda_i f_i(x, y) + \sum_{i \in \{continuous\},k} \lambda_{ik} f_{ik}(x, y) + \sum_{i \in \{nominal\},j} \lambda_{ij} f_{ij}(x, y)\Big) \tag{3}$$

where $Z_{\boldsymbol{\lambda}}(x)$ is a normalization constant to fulfill the probability constraint $\sum_y p(y|x) = 1$. λ_i, λ_{ik}, and λ_{ij} are learned to maximize the log-conditional-likelihood

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

over the entire training set and can be optimized in the same way as that used in MaxEnt-MC. In our experiments, we used the RPROP [4] algorithm for the optimization.

3. FEATURE CONSTRUCTION

In our setting, we assume that we have access to the raw semantic confidence score c_n^s of the n-th trial (utterance) as well as the recognized words $w_{n,t}$ and the corresponding raw word confidence scores $c_{n,t}$ for each semantic slot from the ASR engine. That is, we have the observation vector of

$$x_n = \langle c_n^s, \begin{bmatrix} w_{n,1} \\ c_{n,1} \end{bmatrix}, \begin{bmatrix} w_{n,2} \\ c_{n,2} \end{bmatrix}, \cdots, \begin{bmatrix} w_{n,T} \\ c_{n,T} \end{bmatrix} \rangle. \tag{5}$$

Our goal is to derive a better semantic confidence score $c_n^{s'} = p(y_n|x_n; \boldsymbol{\lambda})$ for each trial by post-processing x_n. We also assume that we have a training (calibration) set that tells us whether the derived semantic information and each recognized word is correct (true) or not (false), from which we train the parameters of the MaxEnt-DC model.

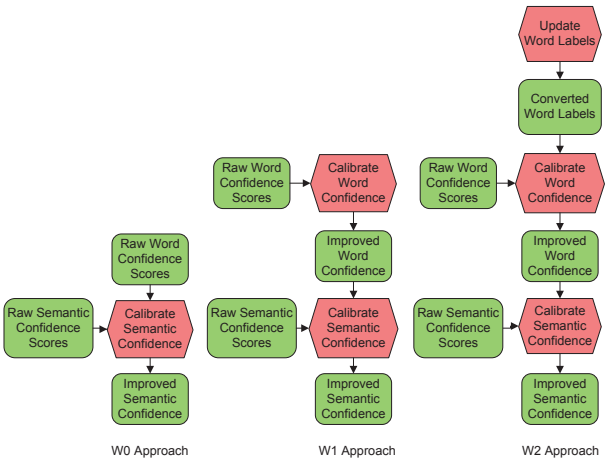

Fig. 1. Three approaches (W0, W1, and W2) of exploiting raw word confidence scores in calibrating the semantic confidence

The word and word confidence score sequences contain a variable number of elements, while the MaxEnt-DC model requires a fixed number of features. Hence, a promise should be made to fulfill the requirement. Since whether the semantic information retrieved is correct or not is determined primarily by the least confident words, we sort the word confidence scores in the ascending order and keep only the top M word confidence scores and the associated words. The discarded information is thus kept at a minimum. We denote the top M sorted words and confidence scores as

$$\begin{bmatrix} \bar{w}_{n,1} \\ \bar{c}_{n,1} \end{bmatrix}, \begin{bmatrix} \bar{w}_{n,2} \\ \bar{c}_{n,2} \end{bmatrix}, \cdots, \begin{bmatrix} \bar{w}_{n,M} \\ \bar{c}_{n,M} \end{bmatrix}. \tag{6}$$

We then construct two features

$$f_1(x_n, y_n) = \begin{cases} c_n^s & if\ y_n = true \\ 0 & otherwise \end{cases} \tag{7}$$

$$f_2(x_n, y_n) = \begin{cases} c_n^s & if\ y_n = false \\ 0 & otherwise \end{cases} \tag{8}$$

one for the class true and one for the class false, based on the raw semantic confidence scores. In addition, we construct four (two pairs of) features

$$f_{4t-1}(x_n, y_n) = \begin{cases} \bar{c}_{n,t} & if\ y_{n,t} = true \\ 0 & otherwise \end{cases} \quad (9)$$

$$f_{4t}(x_n, y_n) = \begin{cases} \bar{c}_{n,t} & if\ y_{n,t} = false \\ 0 & otherwise \end{cases} \quad (10)$$

$$f_{4t+1}(x_n, y_n) = \begin{cases} \bar{w}_{n,t} & if\ y_{n,t} = true \\ 0 & otherwise \end{cases} \quad (11)$$

$$f_{4t+2}(x_n, y_n) = \begin{cases} \bar{w}_{n,t} & if\ y_{n,t} = false \\ 0 & otherwise \end{cases} \quad (12)$$

for each pair of word $\bar{w}_{n,t}$ and confidence $\bar{c}_{n,t}$ in the top M list with the total number of features equals to $4M + 2$. In this above formulation, the raw word confidence scores are directly used when constructing features. This is denoted as W0 approach in Fig. 1, which illustrates three ways of exploiting the word confidence scores. The remaining two ways are described below.

As shown in [14], the quality of word confidence scores can be greatly improved using the calibration algorithm described therein. Better word confidence scores often translate to better features and subsequently better calibrated semantic confidence scores. This is denoted as W1 approach in Fig. 1.

The quality of the calibrated semantic confidence scores may be further improved since some word recognition errors do not affect the desired semantic information. As such, we should disregard these errors when calibrating the word confidence scores. Specifically, if the semantic information is correct in the calibration set, we convert the labels of all related words to true and calibrate the word confidence scores using the updated labels. This is denoted as W2 approach in Fig. 1.

4. EMPIRICAL EVALUATION

To evaluate the effectiveness of the semantic confidence calibration technique we just described, we have conducted a series of experiments on a range of semantic slots using the data collected under realistic usage scenarios. In this paper, we report the evaluation results on two representative semantic slots: a ZIP-code slot and a date-time slot, to unveil the pros and cons of each approach described in Section 3.

Table I summarizes the number of trials (utterances) and words in the training (calibration), development, and test sets for each semantic slot. The semantic error rates (SERs) are 5.2% and 9.8% on the ZIP-code and date-time slots, respectively. There are three major differences between these two semantic slots. First, only 11 key words are used in the ZIP-code slot. As shown in Table II in which the top 10 words and their frequencies are illustrated, the frequencies of these 11 words are close to uniform. In contrast, more than 100 key words are used in the date-time slot and the word distribution in the date time slot is rather non-uniform. Second, there are many variations in expressing the same date/time concept while the way to express the ZIP code is relatively consistent. Third, while a word recognition error in the ZIP-code slot usually indicates a semantic error, this is not necessarily true for the date-time slot. For example, the semantic information is unchanged when eleventh is misrecognized as eleven.

Table III compares different approaches described in Section 3 using the MSE, NNLL, and EER criteria. The confidence measure before calibration was obtained directly from a speaker-independent ASR engine, which uses a Gaussian mixture model classifier discriminatively trained with generic training sets. The W0 setting uses the word confidence measure retrieved directly from that ASR engine. The W1 and W2 settings use word confidence measures calibrated using the approach described in [14], with the word labels unadjusted and adjusted, respectively, as described in Section 3. All the results reported in the table are obtained using the lowest three (i.e. M=3) word confidence scores, which slightly outperforms the setting using the lowest two word confidence scores. The vocabulary size J is automatically determined using the training set by assigning distinct word token IDs to words occur more than 20 times in the training set. This yields 12 word token IDs (i.e., $J = 12$) for the ZIP-code slot and 65 word token IDs for the date-time slot. When the amount of calibration data increases, J will also increase slightly and automatically.

TABLE I
SUMMARY OF DATA SETS

	ZIP code		Date time	
	# trials	# words	# trials	# words
train	4010	20062	7300	16393
dev	4010	20055	7299	16464
test	4009	20054	7299	16368

TABLE II
TOP 10 WORDS AND THEIR FREQUENCIES IN THE DATA SETS ASSOCIATED WITH THE ZIP CODE AND DATE TIME SEMANTIC SLOTS

ZIP code			Date Time		
word	**count**	**percentage**	**word**	**count**	**percentage**
one	2284	11.38%	twenty	1819	11.10%
three	2277	11.35%	April	998	6.09%
two	2159	10.76%	March	651	3.97%
four	2022	10.08%	May	609	3.72%
zero	1986	9.90%	February	582	3.55%
seven	1823	9.09%	July	517	3.15%
five	1673	8.34%	June	514	3.14%
eight	1629	8.12%	eight	497	3.03%
six	1611	8.03%	August	470	2.87%
nine	1464	7.30%	first	459	2.80%

TABLE III
CONFIDENCE QUALITY COMPARISON USING FEATURES CONSTRUCTED FROM DIFFERENT WORD CONFIDENCE MEASURES

	ZIP code			Date time		
	MSE	NNLL	EER	MSE	NNLL	EER
No Calibration	0.047	0.202	**39.5**	0.074	0.271	**18.2**
W0	0.042	0.163	**24.3**	0.053	0.191	15.6
W1	0.043	0.169	24.3	0.048	0.174	14.6
W2	0.044	0.169	24.8	0.046	0.166	**14.0**

Several observations are made by examining Table III. First, in the simplest W0 setting, we already reduce MSE, NNLL and EER by relatively 10.6%, 19.3% and 38.5% on the ZIP-code slot, and 28.4%, 29.5% and 14.3% on the date-time slot, respectively. That is, even without any change in the word confidence measure, our calibration approach can already significantly improve the semantic confidence measure. Second, if we further calibrate the word confidence scores before using them to calibrate the semantic confidence measure (W1 setting), even better results are obtained (for the date-time slot) -- outperforming the no-calibration setting with relative MSE, NNLL, and EER reductions of 35.1%, 35.8% and 19.8%. No further improvement was observed on the ZIP-code slot. Third, by using the word confidence measure calibrated using the adjusted word labels we can furthermore improve the MSE,

NNLL and EER on the date time slot, with the relative reduction over the no-calibration setting by 37.8%, 38.7% and 23.1%, respectively. This is because many word recognition errors in the date-time slot do not affect the semantic information.

Figures 2 and 3 illustrate the DET curves on the ZIP-code and date-time slots under four different settings including no-calibration setting as the baseline. Consistent improvements over the baseline are observed on the date-time slot across different false-alarm and miss probability ranges when either the original word confidence measure or the calibrated word confidence measures are used. For the ZIP-code slot, however, we can observe gains using W2 settings over the W1 and W0 settings when the false alarm probability is low but observe no significant difference when the false alarm probability is high.

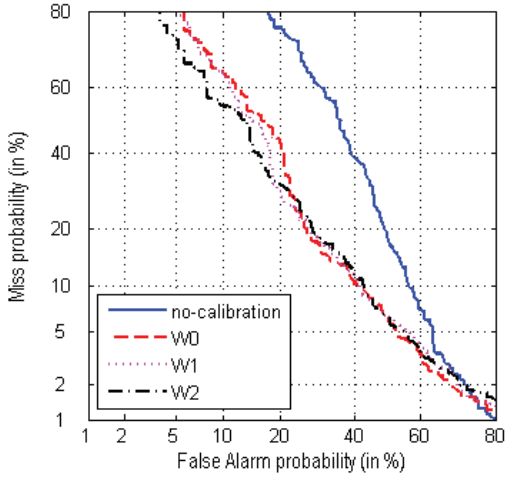

Fig. 2. The DET curve for the ZIP code semantic slot.

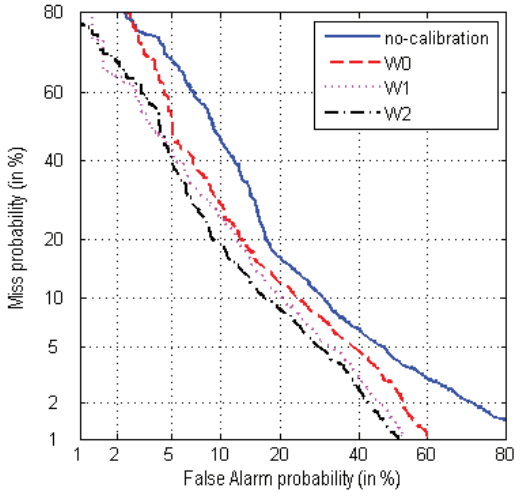

Fig. 3. The DET curve for the date time semantic slot.

5. CONCLUSIONS

In this paper, we propose to use the MaxEnt-DC model [9] to calibrate the semantic confidence scores by utilizing the full distributional information over the raw semantic confidence score and over the word confidence scores. Our approach is shown to have significantly boosted the quality of the semantic confidence scores. This happens even without calibrating the word confidence scores in the first place. We also report that additional improvements can be obtained on some semantic slots using calibrated word confidence scores. We believe the technique and features described in this paper is very useful for spoken dialog application developers to improve the semantic confidence scores without modifying the ASR engines.

6. ACKNOWLEDGEMENTS

We would like to thank Drs. Yifan Gong, Jian Wu, Jinyu Li, Nikko Strom 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.

7. REFERENCES

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

[2] D. van Leeuwen and N. Brümmer. "On calibration of language recognition scores," in Proc. *IEEE Odyssey: The Speaker and Language Recognition Workshop*, 2006.

[3] 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.

[4] 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.

[5] Y.-Y. Wang, D. Yu, Y.-C. Ju, and A. Acero. "An Introduction to Voice Search," *IEEE Signal Processing Magazine*, vol. 25, no. 3, pp. 28-38, May 2008.

[6] J. Wilpon, L. Rabiner, and C.-H., Lee, "Automatic recognition of keywords in unconstrained speech using hidden Markov models", IEEE Trans. ASSP 38, pp. 1870-1990.

[7] 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.

[8] 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.

[9] 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.

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

[11] D. Yu, L. Deng, and A. Acero. "Hidden conditional random field with distribution constraints for phonetic classification," in Proc. *Interspeech,* pp. 676-679, 2009.

[12] D. Yu, Y.-C. Ju, Y.-Y. Wang, A. Acero. "N-gram based filler model for robust grammar authoring", in Proc. *ICASSP*, vol. I, pp. 565-568, 2006.

[13] D. Yu, Y.-C. Ju, Y.-Y. Wang, G. Zweig, A. Acero. "Automated directory assistance system - From theory to practice," in Proc. *Interspeech*, pp. 2709-2712, 2007.

[14] D. Yu, S. Wang, J. Li, and L. Deng. "Word confidence calibration using a maximum entropy model with constraints on confidence and word distributions," in Proc. *ICASSP* 2010.