



How Predictable is ASR Confidence in Dialog Applications?

Xiang Li, Juan M. Huerta

IBM T.J. Watson Research Center
 1101 Kitchawan Road, Route 134
 Yorktown Heights, NY, 10598
 {xiangli, huerta}@us.ibm.com

Abstract

ASR confidence is a metric that reflects, to a large extent, the conditions under which a recognition task is being carried out as well as the reliability of the result. Because of this, ASR confidence constitutes a potentially useful feature in frameworks that attempt to assess the state of a dialog. In this paper we evaluate the predictability of ASR confidence based on knowledge of previously observed context-dependent confidences. We find out that the contextual confidence can be predicted with a standard prediction deviation less than 10% of the dynamic range of the confidence score, which represents a almost 40% relative reduction in standard deviation measure to a static confidence assumption baseline. Because our prediction is based on context, this predictability can be leveraged to produce an estimate of the expected average confidence until the end of a call based on the context path expected to be traversed.

Index Terms: confidence prediction, predictive analytics

1. Introduction and Background

In this paper we analyze the speech recognition confidence throughout the course of a call in a dialogue system in an attempt to determine how predictable the speech recognizer's (ASR) confidence is. If ASR confidence is to a certain extent predictable using observed evidence that becomes available during the evolution of a call (i.e., information about the speaker, the call, the context, the grammars used, etc), then, confidence prediction could represent a valuable feature in frameworks that continuously monitor and evaluate call progress, predict call completion levels, and take adequate actions. Intuitively, one can expect that confidence is predictable during the course of a call, and in this paper we will evaluate to what extent this intuition is valid.

Interest in confidence metrics is not new. Previously, many researchers have established that ASR turn confidence is a useful attribute in estimating the quality of a conversation. For example in [1], a so called *unconfident percentage* feature was used to support application performance evaluation. Because of its important role in assessing the quality of the interaction, there has also been substantial focus on finding ways to extract reliable ASR confidence annotations [2, 3, 4] as well as in semantic and interpretation engines.

Once confidence is estimated, it has found useful application in areas like pronunciation learning and modeling, as well as model training, dialog systems [5] and classification model training using confidence annotations [6].

In terms of modeling, assessing and predicting various aspects of dialog applications including performance, emotion, and user modeling, there have been multiple efforts in this direction [7, 8, 9, 10, 11]. For example, [9] is a

framework that aids in finding the factors that influence a success metric in dialog systems performance. It achieves this by applying decision theory to obtain a single objective function based on multiple disparate measurements. In [1] application assessment is based on attributes pertaining the current turn, i.e., they do not use historical dialog information to predict what will happen in the current turn. In general, their goal is not to predict success or confidence in the current or next turn, but rather, to estimate overall application state and success. In [10] a performance prediction method based on [9] is described. Given the clear advantages of performing automatic system evaluation over manual, this area has been of particular interest recently.

However, in spite of all the work in application modeling and performance prediction, so far, there has not been work in confidence prediction especially based on historical confidence applied to dynamic application adaptation. In our work, we predict confidence strictly based on previously observed confidence in previously observed contexts, and, possibly also with other users and situation combinations. Intuitively, confidence is predictable in a call because factors that impact confidence like user and environmental conditions remain constant during the course of a call, but we will assess this question from the *contextual* point of view (in this work, by context, we refer to a specific turn type -assuming that a turn taxonomy exists, e.g., an easy approach is to assess what grammar is being targeted in any given turn): *given a current turn context and a series of previously observed context-confidence pairs, can I produce an improved confidence prediction over a naïve stationary assumption and achieve a reasonable prediction accuracy?*

This paper is organized as the following: In section 2, we describe our algorithms to predict confidence based on observed contextual confidence. In section 3, we describe our experiments and results, and in section 4 we provide conclusions to this work.

2. Predicting Confidence

Our confidence prediction algorithms can be divided into two big categories: Those that use a linear prediction method under the criterion of Minimum Mean Square Error (MMSE) and those that use classification techniques to make prediction.

2.1. Linear Least Squares confidence prediction

Our linear prediction algorithms make prediction by formalizing the value of the dependent variable (i.e. the confidence score to be predicted at the current context) as a linear combination of the values of independent variables, which are the observations of past turn information including confidence scores. The prediction coefficients (including the

bias term) are determined through the minimum mean square error criterion which optimizes the coefficients to achieve the minimum empirical square error between the prediction and target value. We used two different ways to carry the MMSE-based linear prediction, one is the *direct linear prediction*, and the other is the *histogram based linear prediction*.

2.1.1. Direct linear prediction

The direct linear prediction method, as described in Equation 1, directly formalizes the predicted confidence score C_t at context t as a linear combination of the past observed discrete events X_i (e.g., no-input, rejection, etc), confidence scores Y_j and a bias term, and it estimates one set of prediction coefficients $A_t = [\alpha_{t,i,x}, \alpha_{t,j,y}, \beta]^T$ for every target context t :

$$\begin{aligned} C_t &= \sum_i \alpha_{t,i,x} X_i + \sum_j \alpha_{t,j,y} Y_j + \beta_t \\ &= [\overline{X} \quad \overline{Y} \quad 1][\overline{\alpha_{t,x}} \quad \overline{\alpha_{t,y}} \quad \beta]^T \end{aligned} \quad (1)$$

In formalizing the independent variables of the prediction, we removed the sequence information of the past context events and confidences, and only used a “bag-of-word” representation of the history. X_i is an integer variable that indicates how many times certain event i has happened in the history of the current call. For example, X_{65} equal 1 could represent that we observed one past event of “no-match” in the context number 3 in the current call history. Y_j is a real value variable that indicates the observed confidence score of the context j in the history. If context j has been observed multiple times, then the value used is the average confidence observed for such context. If it has not been observed in the past, then its Y_j is 0. This produced a “per turn” characterization of the past turn event and confidence score, and has been proven to be effective in our experiments. One advantage of this “bag-of-word” representation is that the number of independent variables will be fixed no matter how long the history was. Since the history of a call will get longer and longer as the call evolves, this invariance to the history length of our feature makes it possible for us to use one set of model parameters to carry continuous real time prediction as the call evolves.

Having formalized the prediction as in Equation 1, the prediction coefficients $A_t = [\alpha_{t,i,x}, \alpha_{t,j,y}, \beta]^T$ are estimated according to the MMSE criterion. It turns out that the vector of those coefficients is the product of the pseudo inverse of the feature matrix and the target confidence vector as in Equation 2:

$$A_t = (Z^T Z)^{-1} Z^T C \quad (2)$$

Where Z is the feature matrix and C is the target vector as in Equation 3, and N is the total number of observations of context t ,

$$Z = \begin{bmatrix} \overline{X_{i,1}} & \overline{Y_{j,1}} & 1 \\ \dots & \dots & \dots \\ \overline{X_{i,N}} & \overline{Y_{j,N}} & 1 \end{bmatrix}, C = [C_{t,1}, \dots, C_{t,i}, \dots, C_{t,N}]^T \quad (3)$$

2.1.2. Histogram based linear prediction

Instead of using real confidence score of the past event as independent variable, our histogram based linear prediction method quantizes the confidence score of each context according to its corresponding histogram distribution and carries linear prediction using the index of the quantized confidence scores. The specific process of histogram based quantized confidence linear prediction can be described as the following:

- We first compute the confidence histogram using a fixed number of bars for each of the contexts.
- We then index the histogram of each context by marking the highest count (i.e. the most likely confidence region) as 0, all the regions on its right side as positive integers, and all the regions on the left side as negative numbers. The specific value of each region’s index depends on the distance of that region to the region with the highest count (e.g. 1 for the bar that is immediate right to the highest bar, -2 for the bar sits second left to the highest bar, etc.).
- The real confidence score of each context is then quantized based on its corresponding histogram distribution index as generated above. The quantized confidence scores are used as independent variable to estimate the prediction coefficients in a similar way as in Equations 1 and 2.
- To make prediction, we first generate the quantized confidence score from linear prediction in a similar way as in Equation 1, then we translate the quantized confidence score into its real confidence value based on the corresponding histogram distribution, as in Equation 4:

$$\begin{aligned} C_R &= F(C_p) \\ &= (C_p + MaxH) \cdot Scale - Scale / 2 \end{aligned} \quad (4)$$

where C_R and C_p are the real value and quantized confidence score prediction respectively. $MaxH$ is the index of the highest count histogram bar, and $Scale$ is the range of the bar. The bias term $-Scale/2$ makes C_R the central point of that region.

The motivation of the above confidence score quantization is the following: although we believe that the rationale behind the confidence score prediction is that the user’s behavior (or performance) remains stationary during the call, we observed that different contexts have different confidence score distributions. Two confidence scores with the same real value but observed in two different contexts may represent two different levels-of-performance of the user if those two scores fall into different quantization regions in those two different contexts. Consequently, using only the real confidence score in prediction without the appropriate confidence score distribution normalization may make us miss this valuable level-of-performance information of the user and affect the prediction result. Quantizing the confidence score into regions makes it possible for us to normalize the effect of different confidence score distributions in the prediction process and provide a better description of the user’s level-of-performance, and as a result, give us higher prediction accuracy (as described in section 3, experimental results did confirm this point).

In addition to the benefit of providing a better description of a user’s level-of-performance, another advantage of histogram quantization is that it can transform the prediction problem into a classification problem, and enables us to use the conventional powerful pattern classification techniques to

solve the prediction problem. The following sub-section specifically talks about how we use classification techniques to solve prediction problems in the context of dialog system confidence score prediction.

2.2. Classification-based confidence prediction

As mentioned above, one benefit of quantization is that we can categorize the target confidence score into individual regions and carry out classification processing to make prediction. We can treat the quantized and discrete confidence score, which we call it pseudo confidence score, as class label and build a classifier to predict the index of the pseudo confidence-score, then convert it back into the real confidence score. We have tried two different methods along this direction: maximum entropy model (MaxEnt) and Multi-layer perception (MLP) neural network.

2.2.1. MaxEnt based confidence score prediction

Our MaxEnt approach works in the following way:

- We first quantize the target confidence score of each context using the histogram quantization method described above, and we use the resulting quantization index as the class label.
- Then we construct the classification feature in two different ways: one was to use the quantized past confidence score together with the past context event as the feature, in the same way as we did in histogram quantization based linear prediction, which we call it *quantization-based feature*; the other was to still use the real value of past confidence score and past context event as the feature, as we did in direct linear prediction, and we call it *Plain feature*.
- We then build a classification model using the class label and features defined above, carry classification for the unknown case and generate the posterior probability distribution for each class.
- Having the posterior probability of each pseudo confidence score, we also used two different ways to generate the real confidence score as in Equation 5:

$$\begin{aligned} C_{R,MAP} &= F(C_p(\operatorname{argmax}_h P(h))) \\ C_{R,mean} &= \sum_h P(h)F(C_p(h)) \end{aligned} \quad (5)$$

Where h is the histogram index and $F()$ is the conversion function as defined in Equation 4.

As indicated by Equation 5, the first way was to pick the index with the highest posterior probability and use the corresponding central point of that region as the confidence score, which we call *MAP conversion*. The other was to compute the mean value of the central points of all quantization regions based on their corresponding posterior probabilities from the model, and we call it *Mean conversion*.

2.2.2. MLP based confidence score prediction

Multi-layer perceptron (MLP) neural network was the other approach we tried. The features used in MLP were the same as in MaxEnt (*quantization-based feature and plain feature*). But different from the MaxEnt model, MLP is capable of predicting both the posterior probabilities of discrete labels and continuous confidence score value, depending on how the output units and target were constructed. When we use multiple outputs in MLP and construct the target as a binary

vector, we make the MLP as a classifier and we can apply the same technique as we used in the MaxEnt model to predict the confidence index then convert it into real confidence scores. On the other hand, if we only use one output unit in MLP and make the target as the normalized confidence score (e.g. normalize the target as a real number between 0 and 1 if we use the sigmoid function), then the MLP can be applied to predict the real confidence score directly. We label the above two different ways of using MLP as *MLP classification* and *MLP prediction*, and their performances are reported in section 3 of experimental results.

3. Experiments and Discussions

In order to evaluate the predictability of the ASR confidence score in dialogs using the methods described in section 2, we conducted various experiments. The experimental dataset was collected from the caller-system interaction log of a real deployment of an automatic customer service application.

The total number of caller-system interaction logs was around 21000, with average number of events per call as 5.1, and was divided into training and testing sets with 85 and 15 percent amount of the data. The number of possible contexts was 61, each context had 4 different possible events: *no-input*, *no-match*, *stopped* and *accepted*. The range of the confidence score was from 0 to 99.

Since we are predicting continuous real value contextual turn confidence score, we chose the square root of the mean square error (RMSE) of the predicted confidence to its truth level as the performance measure. The confidence score truth level was produced by the ASR engine in the immediate next user-machine interaction turn.

The baseline performance system was established using the empirical mean value of the confidence score using the matching context from the training data as prediction. The corresponding RMSE baseline performance on the training data thus matched the empirical standard deviation (i.e., the average deviation magnitude from the baseline predicted value is the standard deviation when the predicted value is the mean).

The baseline performance corresponds to a 15.38 RMSE. In other words, if we use the mean of the context (average confidence per grammar) as predictor, the root of the mean square error rate we get is on 15% off.

We now compare it with various algorithms described in section 2. In the histogram normalization based linear prediction system, the whole confidence score distribution was covered by 10 equal width bars in the histogram. In the MLP prediction, we used 10 and 20 hidden units for *MLP prediction* and *MLP classification* respectively. The experimental results of various methods are listed in Table 1.

Method	RMSE
Baseline	15.38
Direct linear prediction	10.7
Histogram quantization linear prediction	10.09
MaxEnt quantization feature, MAP conversion	11.75
MaxEnt quantization feat, Mean conversion	9.56
MaxEnt plain feat, MAP conversion	12.65
MaxEnt plain feat, Mean conversion	10.07
MLP prediction	9.92
MLP classification with mean conversion	9.91

Table 1 Context-based confidence score prediction performance

Our first impression from Table 1 is that the confidence score can be predicted with reasonable accuracy. Compared with the baseline performance of using the prior mean information, all the proposed methods have generated significant performance improvements. The best system so far, which is the MaxEnt model with quantization based feature and mean conversion, reduced the RMSE of confidence prediction by almost 40% relative to the baseline.

The difference between *Mean conversion* and *MAP conversion* is also worth mentioning. On average, *Mean conversion* has 20% lower RMSE than *MAP conversion*. The reason may be the discrepancy between classification processing and prediction processing. Different classes are mutual exclusive in a classification task, and it's more reasonable to pick the class with highest probability as label. On the other hand, those different classes in prediction scenario indeed have strong correlations with each other, and it's more appropriate to treat the posterior probability as the inverse distance measure of the target prediction to each region's central point. Since *mean conversion* provides a better fit to the prediction scenario, it's without surprise that it outperforms *MAP conversion*.

Another interesting observation is the comparison between the two different linear prediction methods using different features. By normalizing the confidence score variation between different contexts, the *quantization based feature* approach provided a better characterization of the level-of-performance of the user, and consequently resulted in better prediction accuracy. As explained in section 2, we argue that the quantization approach provides a better description than the un-quantized method.

4. Conclusions

Using the confidence associated to previous turns in a dialog, and using knowledge of the current context and previous contexts we can improve our prediction of the confidence by reducing the RMSE by almost 40% over the baseline. In other words we can do significantly better than simply assuming that confidence remains stationary during the course of a call. Our approach can be used for real time prediction of what's going to happen at the next turn simply based on observed contextual confidence information. This could be extended to a prediction of the expected behavior of the remainder a call. This, in turn, could allow us to automatically and in real time monitor system performance (between user and automatic system), and trigger an alarm if the predicted output falls below acceptable thresholds, for example. Future work should follow this direction.

In terms of existing and upcoming assessment frameworks, due to its contextual predictability, ASR confidence should be part of the set of features/attributes used in existing and upcoming dialog application analysis frameworks [9, 12] especially if high-level dialog structure is lacking in the application.

5. References

- [1] P. Carpenter, C. Jin, D. Wilson, R. Zhang, D. Bohus, A. Rudnicky; *Is this conversation on track?*, Eurospeech, 2001
- [2] D. Bansal, M. Ravishankar; *New Features for Confidence Annotation*, ICSLP, 1998
- [3] L. Gillick, Y. Ito, J. Young; *A Probabilistic Approach to Confidence Estimation and Evaluation*, ICASSP, 1997
- [4] T. Kemp, T. Schaaf, *Estimating Confidence Using Word Lattice*, Eurospeech, 1997
- [5] TJ Hazen, S Seneff, J Polifroni; *Recognition confidence scoring and its use in speech understanding systems* Computer Speech and Language, 2002
- [6] R. E. Schapire, Y. Singer; *Improved Boosting Algorithms Using Confidence-rated Predictions*, Machine Learning, Vol 37 Number 3, 1999
- [7] K. Forbes-Riley, D. J. Litman; *Modelling User Satisfaction and Student Learning in a Spoken Dialogue Tutoring System with Generic, Tutoring, and User Affect Parameters*. HLT-NAACL, 2006
- [8] K. Forbes-Riley, D. J. Litman; *Predicting Emotion in Spoken Dialogue from Multiple Knowledge Sources*. HLT-NAACL, 2004
- [9] M. A. Walker, D. J. Litman, C. A. Kamm, A. Abella; *PARADISE: A Framework for Evaluating Spoken Dialogue Agents*. ACL, 1997
- [10] H. Bonneau-Maynard, L. Devillers, S. Rosset; *Predictive performance of dialog systems* LIMSI-CNRS, BP 133 91403
- [11] I. Zukerman, W. Albrecht; *Predictive Statistical Models for User Modeling*, User Modeling and User-Adapted Interaction, Volume 11, Numbers 1-2 March, 2001
- [12] T. Paek, *Toward Evaluation that Leads to Best Practices: Reconciling Dialog Evaluation in Research and Industry*, HLT-NAACL, 2007