Practical Application of Domain Dependent Confidence Measurement for Spoken Language Understanding Systems

Mahnoosh Mehrabani **David Thomson** * **Benjamin Stern**
Interactions LLC, Murray Hill, NJ, USA
{mahnoosh, bstern}@interactions.com

Abstract

Spoken Language Understanding (SLU), which extracts semantic information from speech, is not flawless, specially in practical applications. The reliability of the output of an SLU system can be evaluated using a semantic confidence measure. Confidence measures are a solution to improve the quality of spoken dialogue systems, by rejecting low-confidence SLU results. In this study we discuss real-world applications of confidence scoring in a customer service scenario. We build confidence models for three major types of dialogue states that are considered as different domains: how may I help you, number capture, and confirmation. Practical challenges to train domain-dependent confidence models, including data limitations, are discussed, and it is shown that feature engineering plays an important role to improve performance. We explore a wide variety of predictor features based on speech recognition, intent classification, and high-level domain knowledge, and find the combined feature set with the best rejection performance for each application.

1 Introduction

The purpose of an SLU system is to interpret the meaning of a speech signal (De Mori et al., 2008). SLU systems use Automatic Speech Recognition (ASR) to convert speech signal to the text of what was spoken (hypothesis), followed by semantic meaning extraction from the ASR hypothesis using Natural Language Processing (NLP). Semantic information that can be extracted from an utterance include the intent of speaker, as well as any entities such as names, products, numbers, places, etc., where depending on the application, one or more of these information are of importance.

While SLU systems have achieved considerable success during the past few decades, errors are inevitable in real applications due to a number of factors including noisy speech conditions, speaker variations such as accent, speaking style, inherent ambiguity of human language, lack of enough in-domain training data, etc. With the rise of virtual assistants and their increasing utilization from everyday voice inquiries on smart phones and voice commands in smart home scenarios to customer service applications, it is crucial to keep the accuracy of SLU systems above an acceptable threshold. Therefore, to keep the natural flow of conversation between human and automatic agent, using human agents when automatic system fails to provide an accurate response improves user satisfaction. However, the question is: "how do we know that SLU system failed?"

A confidence score is a scalar quantity that measures the reliability of an automatic system. In the literature, several studies have applied ASR-based feature vectors to train statistical models that predict word and/or utterance level confidence scores for ASR systems (Wessel et al., 2001; Jiang, 2005; Yu et al., 2011; White et al., 2007; Williams and Balakrishnan, 2009), and SLU systems (Hazen et al., 2002). Furthermore, semantic-based features have been applied in predicting confidence measures for spoken dialogue systems (San-Segundo et al., 2001; Sarikaya et al., 2005; Higashinaka et al., 2006; Jung et al., 2008), as well as other applications such as machine translation (Gandrabur et al., 2006).

The purpose of this study, is to show the importance of confidence modeling in real-world SLU applications, discuss practical challenges to train confidence models, and create a guideline to build efficient confidence models. We build domain-dependent semantic confidence models to improve the rejection of unreliable SLU results. Such rejection process is designed to maintain a high accuracy, while minimizing the number of rejected

* This work was done while at Interactions LLC.

Proceedings of NAACL-HLT 2018, pages 185–192
New Orleans, Louisiana, June 1 - 6, 2018.

utterances. Our experiments are based on improving rejection performance for three different types of dialogue states in a customer service scenario: opening (i.e., how may I help you), number capture (e.g., phone or account number), and confirmation (i.e., yes/no).

The contributions of this study are:

1. Building efficient confidence models based on domain-dependent feature engineering with limited labeled data for training, which makes confidence modeling process scalable for real applications.

2. Proposing an evaluation methodology for practical applications of rejection confidence scoring, based on which an operating point can be selected to balance cost vs. accuracy.

3. Comparing linear and nonlinear confidence models with limited training data, and proposing time-efficient nonlinear features that improve performance.

2 Problem Formulation

In this study we focus on improving confidence measure for SLU systems, where the input is a speech waveform and the output is the semantic information extracted from speech. We consider the semantic output of SLU system to be either true (i.e., all the relevant information required for the application is extracted correctly) or false. Confidence score $c \in \mathbb{R}$ in this context is a number associated with every pair of input utterance $x \in \mathbf{X}$ and estimated semantic output $\hat{y} \in \mathbf{Y}$, which computes how likely is the output of SLU system ($\hat{y}$) to be equal to the reference output (y).

When probabilistic models are used, posterior probability $P(y|x)$ can be applied as confidence score. However, proper normalization of posterior probabilities is important to obtain a reliable confidence score (Jiang, 2005). In this study, we define the SLU confidence measure as $P(\hat{y} = y|x, y)$. A statistical model is trained to predict the semantic correctness of SLU system. The posterior probability from this binary classifier is applied as confidence measure. While training a confidence model requires data, it outperforms unsupervised approaches. The features that are used to train the confidence model are functions of the input and output of SLU system: $f(x, y)$.

2.1 System Layout

Figure 1 illustrates the components of SLU system we used for our experiments including rejection based on confidence score. The main components of any SLU system are ASR and NL. However, we do not accept all the outputs of SLU system. A confidence model is used to decide wether or not the extracted semantic information by SLU system is accurate. The confidence model produces a score based on several predictor features including ASR scores, NL scores, and domain knowledge. If the confidence score is higher than a threshold, SLU result is accepted. The semantic information of rejected (i.e., more challenging) utterances is extracted by human labelers.

2.2 Evaluation Methodology

The performance of SLU system with an accept/reject backend, shown in Figure 1, can not simply be evaluated based on the accuracy of the output. An essential component of such system, is rejection confidence scoring, which depends on both confidence score and confidence threshold. Confidence modeling can be formulated as a binary classification problem, and be evaluated using standard measures such as Receiver Operating Characteristic (ROC) curve, or area under the curve (AUC). However, in a practical application, business objectives have to be considered in performance evaluations. In a virtual intelligent customer service scenario, it is important to maximize customer satisfaction while minimizing the cost. Customer satisfaction is directly related to the accuracy, and accuracy can be improved by using higher confidence threshold. Nevertheless, with a higher confidence threshold, more utterances that are labeled by the automated system are rejected and this will increase the cost of manual labeling. Therefore, there is a trade-off between cost (i.e., the number of rejected utterances) and precision (i.e., the accuracy of accepted utterances).

In this study, we focus on improving the confidence measurement to maintain the accuracy while reducing the rejection rate. To evaluate different confidence measures, we plot False Accept (FA) percentage on accepted utterances versus the rejection percentage. For the remaining of this study we call these plots FA-Rej. In production system, confidence threshold is set based on the required semantic accuracy for each application, and generally the higher the rejection, the lower is

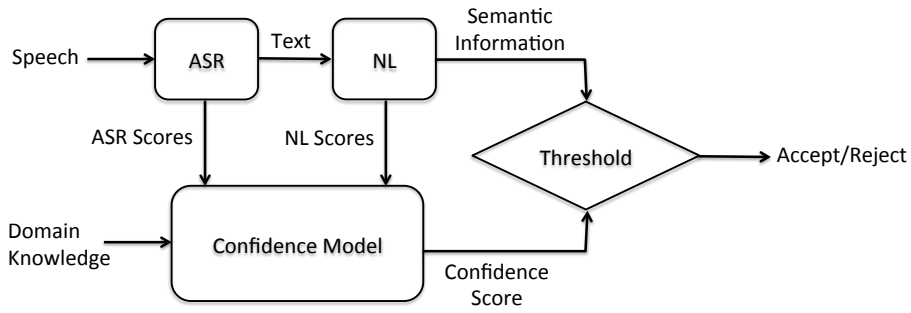

Figure 1: Flowchart of SLU system used for our experiments with an accept/reject backend based on confidence score to ensure the highest accuracy.

the error rate. If a FA-Rej plot has lower rejection rates at all FA rates compared to another plot, it shows a performance improvement.

2.3 Practical Challenges

The main challenge to build confidence models for a real-world application is data limitation. To train a confidence model, a dataset is required with true labels for each sample. In a customer service scenario, semantic information that should be captured, including intents, products, etc., vary from one client to the other. Furthermore, there are several dialog turns/states for each client with different intent sets. Our experiments show that training domain-dependent confidence models significantly improves performance. This makes the data preparation even more challenging, since creating labeled data for a large number of applications/clients is expensive. Therefore, in this study we focus on improving confidence measurement with minimum data.

We show the importance of feature engineering to select the best predictive feature set from a combination of ASR and NLP features, as well as using domain knowledge to improve performance of confidence models for each domain. In this context, domain is defined as a group of dialog states with similar intent types. We show that with low amounts of training data, Maximum Entropy (MaxEnt) model with a linear feature function is the most efficient classifier. We also apply several other classifiers including neural networks and random forest and compare performance for different feature sets. However due to train data limitations, nonlinear classifiers do not significantly outperform MaxEnt. Another advantage of MaxEnt is lower runtime, which is very important in practical applications.

2.4 Data Specification

We present results on three different types of dialogue states, which are widely used for customer service applications in automatic spoken dialogue systems. Our data is selected from real conversations between enterprise customers and the automatic agent. The first dialogue state is the response to an open-ended question, asking "how may I help you". For the remaining of this paper we call this dataset: "opening". In this dialogue state, the user is prompted to explain why they are calling customer service using natural language. The second dataset is based on a "number capture" dialogue state, where the system prompts users to provide an identification number, such as phone or account number. The third dataset is a "confirmation" dialogue state, where the user is prompted to confirm some information.

"Opening", "number capture", and "confirmation" datasets include approximately 15k, 11k, and 10k utterances, respectively. We use 10-fold cross validation for evaluation with a baseline MaxEnt model. These datasets were labeled manually to create the reference intents for each utterance. The "Opening" dataset consists of a large number of intents due to speakers being allowed to use an open language. Furthermore, an "opening" utterance might have more than one intent. For instance, if the speaker says: "I would like to talk to a live agent about my bill", the intent will be "live-agent/billing". In addition to intents, other semantic information such as products are also extracted from "opening" utterances.

For "number capture" dataset, if the speaker provides a number, SLU system is considered accurate if the hypothesized phone or account number exactly matches the reference number. We considered a few more intents for when speak-

ers do not provide a number, such as "don't-have" (i.e., speaker does not have an account number) or "live-agent" (i.e., speaker would like to talk to a live agent). The main intents for "confirmation" dataset are: "true" and "false". A few other intents such as "live-agent" were also considered for this dialogue state. We used a statistical language model and intent classifier for "opening" and "confirmation" datasets, while a Speech Recognition Grammar Specification (SRGS), which is a rule-based language model that also provides the intent was used for "number capture". Note that our objective in this study is to improve rejection based on confidence modeling without any modifications in the SLU (i.e., ASR and NL) system.

3 Combining ASR and NL Features

During speech recognition, several scores are created that can be aggregated at word or utterance level and be applied to estimate ASR confidence. Since speech understanding process is a combination of speech recognition and natural language understanding of ASR hypothesis, additional semantic information and intent classification scores can also be used to predict the semantic confidence measure associated with a spoken utterance.

3.1 ASR Features

Previous studies have used a variety of speech recognition predictor features, such as posterior probabilities, acoustic and language model scores, n-best and lattice related scores, etc., to estimate the ASR confidence for different applications (Jiang, 2005; Yu et al., 2011; White et al., 2007; Williams and Balakrishnan, 2009; Hazen et al., 2002; San-Segundo et al., 2001). We examined several feature sets to achieve the best performance on rejecting the utterances with inaccurate semantic interpretation for "opening", "number capture", and "confirmation" domains. Particularly, two groups of ASR predictor features were applied: scores extracted from Word Confusion Network (WCN) (i.e., a compact representation of lattice (Mangu et al., 2000)), and delta scores that are based on comparing the best path score to an alternative path. Williams et al. (Williams and Balakrishnan, 2009) showed the effectiveness of these two feature types to estimate the probability of correctness for each item in an ASR n-best list.

The WCN feature set that we used includes utterance-level best path score, as well as statistics of word-level scores such as mean, min, max (adding standard deviation did not improve the results), total number of different paths in WCN, and number of segments in WCN. Delta feature set includes two categories: gdelta and udelta. Gdelta score is the log likelihood difference between the best path and the best path through garbage model (i.e., a filler model that is trained with non-speech and extraneous speech), while udelta is the log likelihood difference between the best path and best possible path without any language model constraint (if hopping from phone to phone was allowed). We used average utterance-level gdelta and udelta, as well as min, max, and mean of the word-level gdelta and udelta scores. Our best ASR feature set is a combination of WCN and delta feature sets with the addition of a few more features including number of speech frames, total number of frames, and number of n-best. Table 1 summarizes the ASR features that were used for confidence modeling in all three domains.

Feature Number	Feature Description
F1	WCN utterance-level best path score
F2 – F4	Mean, min, max of WCN word-level scores
F5	Total number of paths in WCN
F6	Number of WCN segments
F7	Average utterance-level gdelta score
F8 – F10	Mean, min, max of gdelta word-level scores
F11	Average utterance-level udelta score
F12 – F14	Mean, min, max of udelta word-level scores
F15	Number of n-best
F16	Number of Speech frames
F17	Total number of frames

Table 1: List of ASR features

3.2 Semantic Features and NL Scores

As speech recognition errors contribute to semantic inaccuracy, ASR confidence predictor features, which mainly predict the probability of correctness of speech recognition hypothesis, can be applied in predicting the semantic confidence. Nevertheless, there are other factors that affect the semantic accuracy, even with an accurate ASR hypothesis. Such factors are related to the meaning interpreted from the text. Therefore, using semantic and high-level features that include domain knowledge can improve the rejection perfor-

mance for an SLU system, especially with limited training data. A number of studies have applied semantic features for confidence prediction (San-Segundo et al., 2001; Sarikaya et al., 2005; Higashinaka et al., 2006; Jung et al., 2008). In this study, we identify domain-dependent features and show that semantic features based on domain knowledge for "opening" and "number capture" domains, as well as using statistical intent classifier scores for "opening" and "confirmation" dialogue states considerably improve performance.

Opening Dialogue State: Confidence predictor features based on word distribution and word identity have been previously studied (Yu et al., 2011; Huang et al., 2013). In this study, we created word distributions using a separate training dataset. Next, we tested various methods of creating predictor features based on the most common words in each application. For "opening" dataset this type of predictor features improved performance, and the best results were achieved by using the occurrence of top 450 words via a bag of words feature vector. Larger and smaller number of words were also tested, which deteriorated the performance. Furthermore, we tested using the word scores from WCN instead of binary occurrence vector, which did not improve the performance. Features based on significant or top words did not improve performance for "number capture" and "confirmation" datasets, which can be due to more limited vocabulary in those domains compared to "opening".

We also applied the top three intent scores from classifier as additional confidence predictor features, which significantly improved the results. For "opening" application, an SVM model was used to classify intents. Intent scores in this context are the raw scores computed based on classifier's decision function. Figure 2 shows the FA-Rej results of using NL features in addition to ASR features. As shown, compared to the best performance with ASR features, using significant words feature vector improves the performance. The best performance is achieved by combining ASR features with intent classifier scores. Our experiments show that when intent classification scores are used, adding the significant word feature vector deteriorates rejection performance. Figure 2 also shows the result of using the top intent score as final confidence measure for rejection, which has better performance than ASR features. However, if intent classification scores are not available, the combination of ASR features and top word features obtains the best results.

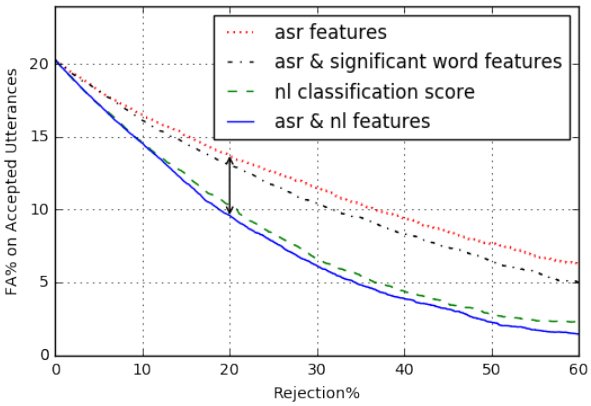

Figure 2: FA-Rej plots on"opening" dataset

Number Capture Dialogue State: The impact of using semantic and high-level features in addition to ASR features to predict semantic confidence for "number capture" application is shown in Figure 3. Since a rule-based grammar is used to perform speech recognition for this dataset, which also generates the intent (i.e., a sequence of digits or another intent), there are no intent classification scores to be applied to predicting the confidence. The additional feature set that we used as NL features include: encoded intent category, digit sequence length as a bag of words vector, binary feature showing the occurrence of the word 'oh', and binary feature comparing the first and second best intents. Our experiments show that using the length of digit sequence as a predictor feature vector improves confidence prediction. We used a 20-dimensional vector for length feature (the length of digit sequences in our dataset varied from zero to nineteen). Encoded intent identity (i.e., number, live agent, etc.) as another feature improved the performance for "number capture" domain. The occurence of the word 'oh' was used as another feature, since it is ambiguous and can mean 'zero' in a digit sequence or be used to show exclamation. Finally, the first and second intents based on the first best and second best ASR hypotheses were compared to generate another semantic feature that shows the certainty of SLU response. If both intents were numbers, but the digits did not exactly match, we set this feature to zero. As shown in Figure 3, using semantic features based on domain knowledge significantly improves the

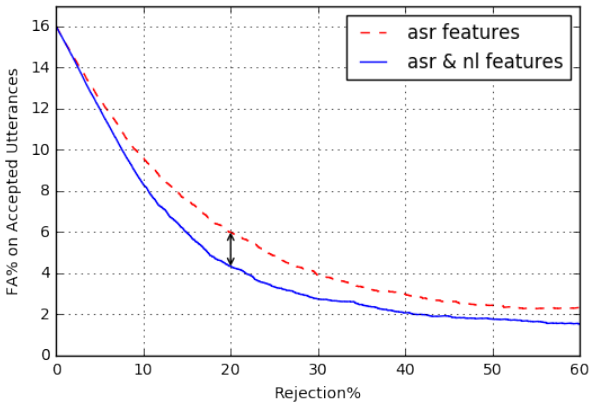

Figure 3: FA-Rej plots on "number capture" dataset

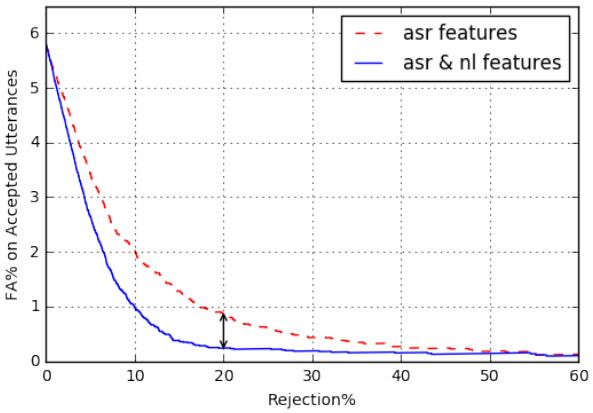

Figure 4: FA-Rej plots on "confirmation" dataset

rejection performance, and performance improvement (i.e., the difference between the number of utterances that have to be rejected to obtain a specific FA on accepted utterances) is higher at lower FA rates. This is especially of importance when the system is expected to have a high accuracy.

Confirmation Dialogue State: The result of integrating NL classification scores with ASR scores for "confirmation" dataset is compared to using ASR scores in Figure 4. As shown, considerable improvement is achieved by using intent classification scores. Due to the high accuracy of "confirmation" domain compared to the other domains, using other semantic features did not improve the performance.

Table 2 summarizes effective semantic and NL features for each domain. Relative performance improvement using the best semantic feature set in addition to ASR features at 20% rejection rate (i.e. when 80% of utterances are accepted based on confidence score) is shown in Table 3. As shown, while "confirmation" dialogue state achieves the highest accuracy compared to other applications, it has the highest relative improvement by using NL scores in addition to ASR scores. The difference in FA rates at 20% rejection when using ASR features versus using both ASR and NL features is illustrated by arrows in Figures 2-4.

Feature(s)	Application(s)
NL classification scores	Opening, Confirmation
Occurrence of top words	Opening
Intent category	Number Capture
Length of digit sequence	Number Capture
Occurrence of "oh"	Number Capture
Comparing 1st and 2nd intents	Number Capture

Table 2: List of domain-dependent semantic features

	Opening	Number Capture	Confirmation
Performance Improvement	29.98 %	27.92 %	72.46 %

Table 3: Relative performance improvement on accepted utterances at 20% rejection

4 Confidence Models

So far we have explored a variety of features using MaxEnt classifier with a linear feature function. In this section, we apply nonlinear feature functions with MaxEnt, as well as nonlinear models. Previous studies have shown the success of MaxEnt models for confidence prediction (Yu et al., 2011; White et al., 2007). The principle of maximum entropy states that given a set of training samples (x_i, y_i), the best estimation of the distribution $p(y|x)$ subject to a set of constraints is the one with the largest entropy (Jaynes, 1957). A typical constraint is that the empirical average from the training samples for each feature function $f_j(x, y)$ should match the expected value. The MaxEnt distribution with this constraint can be characterized with a log-linear form (White et al., 2007):

$$p(y|x) = \frac{exp(\sum_j \lambda_j f_j(x,y))}{\sum_y exp(\sum_j \lambda_j f_j(x,y))} \quad (1)$$

In this study, x is in fact a confidence predictor feature vector $\vec{x}$, and y is a binary random variable. The predictor feature vector includes binary, categorical, and continuous random variables.

As our baseline classifier, we used MaxEnt with a linear predictor feature function f. Philips et al. (Phillips et al., 2006) applied a number of methods to use nonlinear relations in data to improve performance of a MaxEnt classifier, from which we evaluated quadratic function and product of features. Furthermore, we tested binning, where bins were defined based on the Cumulative Distribution Function (CDF) of each continuous feature, which did not improve performance. In addition to the nonlinear feature functions proposed in previous studies, we used a logarithmic function of predictor features: $f(x) = \ln(|x| + \epsilon)$, where ϵ is a very small number used to prevent the log of zero. We also applied nonlinear models such as Neural Networks (NN) and Random Forest. The best NN performance was achieved using a feedforward fully-connected network with one hidden layer, and Adam (Kingma and Ba, 2014) optimizer. Due to limited training data, DNN with larger number of hidden layers did not show any improvements.

Our experiments showed that performance improvement using nonlinear methods is limited due to data limitation, and depends on the domain and the feature set. As shown in Figure 5 using logarithmic function of features that we proposed in this study, in addition to linear features improves the rejection performance for "number capture" when ASR features are used. The advantage of logarithmic features is time efficiency in both training and runtime compared to previously used nonlinear features. Figure 6 illustrates the performance improvement in low FA when applying nonlinear classifiers on "opening" dataset with the largest feature dimension (ASR features combined with top word features described in 3.2). However, with the best predictor feature set for each domain, nonlinear methods did not improve performance.

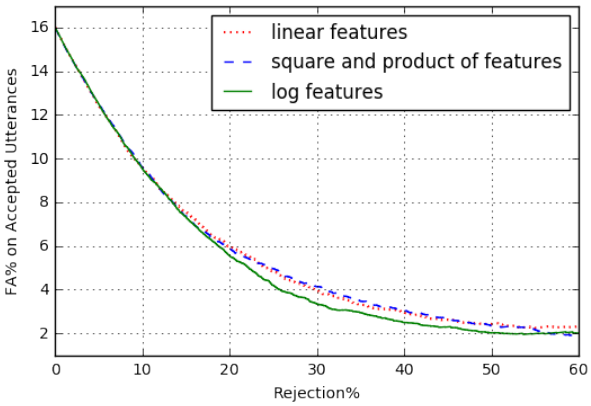

Figure 5: FA-Rej plots on "number capture" dataset with MaxEnt linear and nonlinear features

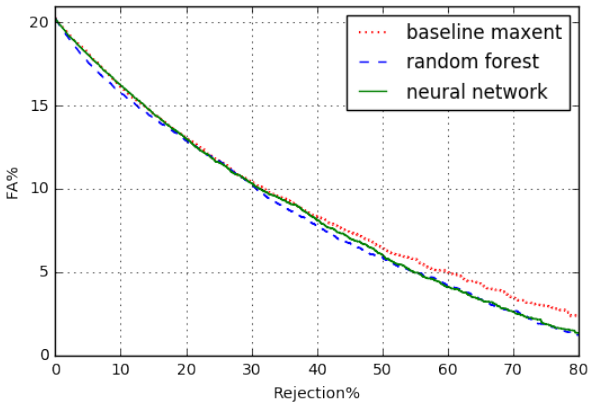

Figure 6: FA-Rej plots on "opening" dataset with baseline MaxEnt, random forest, and neural networks

5 Discussion and Conclusions

The focus of this study was on the practical application of confidence measurement in rejecting unreliable SLU outputs with an important impact on the quality of spoken dialogue systems by re-prompting or using human annotations for challenging (e.g., noisy or vague) utterances. We performed a comprehensive feature engineering to identify the best set of features to train statistical semantic confidence models for three common types of dialogue states in a customer service scenario. It was shown that applying a combination of ASR confidence scores, NL-based features and domain-dependent predictors significantly improves the confidence measure performance. Our experiments showed that with a predictive set of features, MaxEnt is a proper classifier for confidence estimation in terms of performance and computational efficiency. Due to practical challenges, such as the limitation of application-specific supervised data to train confidence models and the importance of real-time rejection (and therefore confidence prediction), the application of more complex models requires a significant performance improvement.

References

Renato De Mori, Frédéric Bechet, Dilek Hakkani-Tur, Michael McTear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding. *IEEE Signal Processing Magazine*, 25(3).

Simona Gandrabur, George Foster, and Guy Lapalme. 2006. Confidence estimation for nlp applications. *ACM Transactions on Speech and Language Processing (TSLP)*, 3(3):1–29.

Timothy J Hazen, Stephanie Seneff, and Joseph Polifroni. 2002. Recognition confidence scoring and its use in speech understanding systems. *Computer Speech & Language*, 16(1):49–67.

Ryuichiro Higashinaka, Katsuhito Sudoh, and Mikio Nakano. 2006. Incorporating discourse features into confidence scoring of intention recognition results in spoken dialogue systems. *Speech Communication*, 48(3):417–436.

Po-Sen Huang, Kshitiz Kumar, Chaojun Liu, Yifan Gong, and Li Deng. 2013. Predicting speech recognition confidence using deep learning with word identity and score features. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7413–7417. IEEE.

Edwin T Jaynes. 1957. Information theory and statistical mechanics. *Physical review*, 106(4):620.

Hui Jiang. 2005. Confidence measures for speech recognition: A survey. *Speech communication*, 45(4):455–470.

Sangkeun Jung, Cheongjae Lee, and Gary Geunbae Lee. 2008. Using utterance and semantic level confidence for interactive spoken dialog clarification. *JCSE*, 2(1):1–25.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lidia Mangu, Eric Brill, and Andreas Stolcke. 2000. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer Speech & Language*, 14(4):373–400.

Steven J Phillips, Robert P Anderson, and Robert E Schapire. 2006. Maximum entropy modeling of species geographic distributions. *Ecological modelling*, 190(3):231–259.

Rubén San-Segundo, Bryan Pellom, Kadri Hacioglu, Wayne Ward, and José M Pardo. 2001. Confidence measures for spoken dialogue systems. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 1, pages 393–396. IEEE.

Ruhi Sarikaya, Yuqing Gao, Michael Picheny, and Hakan Erdogan. 2005. Semantic confidence measurement for spoken dialog systems. *IEEE Transactions on Speech and Audio Processing*, 13(4):534–545.

Frank Wessel, Ralf Schluter, Klaus Macherey, and Hermann Ney. 2001. Confidence measures for large vocabulary continuous speech recognition. *IEEE Transactions on speech and audio processing*, 9(3):288–298.

Christopher White, Jasha Droppo, Alex Acero, and Julian Odell. 2007. Maximum entropy confidence estimation for speech recognition. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–809. IEEE.

Jason D Williams and Suhrid Balakrishnan. 2009. Estimating probability of correctness for asr n-best lists. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 132–135. Association for Computational Linguistics.

Dong Yu, Jinyu Li, and Li Deng. 2011. Calibration of confidence measures in speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(8):2461–2473.