社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2004－ＳＬＰ－51 （７）
2004／5／28

談話理解における対話コンテキストに基づく非線形リスコアリング

ウッティウィワッチャイ チャイ　古井 貞熙
東京工業大学大学院 情報理工学研究科 計算工学専攻
〒152-8552 東京都目黒区大岡山 2-12-1
E-mail: chai@furui.cs.titech.ac.jp, furui@furui.cs.titech.ac.jp

あらまし 本論文では、タイ語の音声対話システムへの適用を目的として、談話理解のための非線形リスコアリングアプローチについて検討した。談話理解の目的は、現在のユーザの発声と対話コンテキスト情報が与えられた状況で、最も可能性の高い発話の意味を抽出することにある。対話コンテキストスコアの線形結合に基づいて、理解仮説のリスコアを行うのが普通であるが、本論文では、これらの種々のスコアは非線形推定値を用いて組み合わせる方がよいことを示す。非線形リスコアモデルは、理解性能を上げるために適用されるだけでなく、主として音声認識誤りによって生ずる信頼性の低い発声を検出するためにも用いられる。さらに、非線形推定値に適切な信頼度を加えることにより、リスコアリングに影響を与えることなく、理解誤りを検出することができることを示す。

キーワード 談話理解、非線形リスコアリング、タイ語音声対話システム

Nonlinear Rescoring Based on a Dialogue Context in Discourse Understanding

Chai WUTIWIWATCHAI　Sadaoki FURUI
Department of Computer Science, Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku, Tokyo, 152-8552 Japan
E-mail: chai@furui.cs.titech.ac.jp, furui@furui.cs.titech.ac.jp

Abstract This paper investigates on a nonlinear rescoring approach for discourse understanding with an application to a Thai spoken dialogue system. Discourse understanding aims to find the most likely meaning of a user utterance given both the current user utterance and dialogue contextual information. While a normal approach is to rescore understanding hypotheses based on a linear combination of dialogue-context scores, this paper shows that these various scores are better combined using a nonlinear estimator. The nonlinear rescoring model is not only applied to improve understanding performance, but also to detect unreliable utterances. By adding appropriate confidence measures to the nonlinear estimator, it is capable to detect understanding-errors with no effect to the rescoring task.

Keyword discourse understanding, nonlinear rescoring, Thai spoken dialogue system

1. Introduction

Discourse understanding aims to find the most likely meaning of a user utterance given both the current user utterance and dialogue contexts (also called "system beliefs" in literatures [2]). It is known that system beliefs based on a system prompt and a dialogue history are effective in dialogue speech understanding. Incorporating the system belief into the understanding component has been performed in two ways. The first way is to augment the conceptual decoding network with additional probabilities derived from the system belief [1,2]. Adding belief scores enhances probabilities of network paths that contain potential concepts given the current dialogue state. The second way is to first produce *N*-best hypotheses of semantic representations using the existing conceptual model and then rescore the *N*-best list by belief probabilities [3,4].

Although the one-pass paradigm provided by the first approach is interesting, complexity of the augmented network is highly increased and the resulting model often requires a much larger training data. Therefore, many systems have been implemented using the two-pass paradigm, where either a concept lattice or an *N*-best list is used intermediately. In this way, each pass can be easily optimized in contrast to the single complicated decoder. Furthermore, scores from various sources that are useful for improving speech understanding, such as confidence measures, can be combined in the second pass without difficulty. A drawback is a loss of the correct path if one

defines a too small size of N-bests.

Pruning the lattice or rescoring the N-best list is normally performed by either multiplying or interpolating the original scores with probabilities conditioned on the system belief [1,3,4]. While the rescoring task is to modify several probabilities of correct patterns to a high score and probabilities of incorrect patterns to a low score, the linear interpolation technique may produce unreliable results if the probabilities cannot be linearly separated. In the first part of this paper, we propose a rescoring method based on a nonlinear estimator, which can modify the probabilities to a desired score regardless of whether they are linearly separable. Two well-known nonlinear estimators, an artificial neural network (ANN) and Support vector machines (SVM), are compared to the simple linear interpolation technique.

Since nonlinear estimators such as ANN and SVM have been used widely for confidence-based error detection [5,6], we expect that the nonlinear estimator applied for the rescoring task is also able to perform error detection at the same time. In the second part of this paper, several confidence measures are selected automatically and used by the nonlinear estimator in order to detect understanding errors.

The next section briefly reviews our speech understanding engine, followed by proposed methods to incorporate dialogue-contextual information. Section 4 explains how confidence features are incorporated into the proposed model. All approaches are evaluated on a Thai hotel reservation task under a project of the first Thai spoken dialogue system, namely TIRA. Experiments are described in Sect. 5 with a conclusion in Sect. 6.

2. Speech Understanding

The aim of speech understanding is to find the most likely semantic representation given an input speech signal (O). In our task, a semantic frame contains two tuples, a *goal* (G) of the input utterance and a set of *concept-values* (V) representing information items necessary for communication. Table 1 demonstrates semantic tags given to a sample utterance. Similar to many other systems, the process is separated to speech recognition and language understanding as shown in Eq. 1. Summation over all possible word strings is limited within a word graph or, in our case, an N-best list of word strings.

$$\tilde{G},\tilde{V} = \arg\max_{G,V} \sum_{W} P(G,V \mid W)P(W \mid O) \quad (1)$$

$$P(G,V \mid W) \approx \sum_{C} P(V \mid G,C)P(G \mid C)P(C \mid W) \quad (2)$$

In our understanding model, a set of *concepts* (C) contained in an input utterance is first extracted. The concepts are then used to determine a goal, and substrings of concepts that correspond to the identified goal are converted into proper values. This process is mathematically described as Eq. 2. The following subsections give more details of each sub-process. See [7,8] for more details.

2.1 Concept extraction: $P(C|W)$

Given N-best word strings, a set of concepts C is detected using a semantic n-gram tagger. Examples of concepts are shown in Table 1. Semantic labels tagged to each word are indices of defined concepts as shown in the row "*Label sequence*" of Table 1.

Table 1: *Examples of semantic tags.*

Utterance	two nights from the sixth of July	
Label sequence	(2) (2) (1) ε (1) ε (1)	
Goal	inform_prerequisite-keys	
Concept	*Concept substring*	*Concept-values*
(1) reservedate	from sixth July	2004-07-06
(2) numnight	two nights	2

2.2 Goal identification: $P(G|C)$

Concepts contained in the top hypothesis output of the concept tagger are used to construct an input pattern for an artificial neural network (ANN) in order to identify a goal [7]. Based on [9], a decision is made by

$$\tilde{G} = \arg\max_{G} P(G \mid C) \text{ with } P(G \mid C) = \frac{\exp\{y_G(\bar{x})\}}{\sum_{G}\exp\{y_G(\bar{x})\}} \quad (3)$$

where $y_G(\bar{x})$ denotes an ANN output value at the G^{th} node. The vector $\bar{x}$ is an input vector whose elements are binary values, each indicating existence of a defined concept.

2.3 Concept-value recognition: $P(V|G,C)$

Given the goal and concepts, the substrings of concepts necessary for communication are converted to concept-values using a rule set. Examples of substrings of concepts and their values are shown in Table 1. Although the top hypothesis from the concept tagger works well for concept extraction, obtaining accurate substrings used to recognize concept-values needs an extra process. In our previous work [8], a combination of statistical and structural models, called *logical n-gram modeling*, was proposed. In this model, scores of N-best label hypotheses were augmented by scores from regular grammar models of each concept. After rescoring, a hypothesis that contained the longest valid grammar was reordered to the top and used to construct its concept-value. Note that some concepts contain values such as those shown in Table 1, whereas some concepts have no value such as a concept "yesnoq" (asking by a yes-no question).

3. Incorporating Dialogue-Contextual Information

Based on the speech understanding model, several strategies to improve system performance by incorporating belief or dialogue contextual information can be conducted as follows.

(I.1) A dialogue-state dependent semantic model (DD-SM). In this case, the general n-gram tagger for concept extraction is replaced by a dialogue-state dependent n-gram model. The $P(C|W)$ described in Sect. 2.1 is rewritten as

$$P(C|W) \approx \sum_{B} P(W|C,B)P(C|B)P(B) \quad (4)$$

where B refers to a system belief. The term $P(W|C,B)$ represents DD-SM and a weight $P(C|B)$ represents possibility of the concept appearing in the given belief state. $P(B)$ is a priori probability of B. The DD-SM can be constructed based on either maximum a posteriori (MAP) or interpolation as

$$P(W|C,B) = \max_{d}\{P(W|C,B_d)\}$$
$$\text{or} = \sum_{d} \alpha_d P(W|C,B_d) \quad (5)$$

where B_d denotes the d^{th} dialogue state. These dialogue-state dependent models are often combined to the general dialogue-independent model in order to preserve system robustness.

(I.2) Rescoring N-best ANN outputs by the system belief. This is the main focus of this paper, details of which will be given in the next subsection.

The rescoring approach is attractive, since we have observed a high accuracy in an oracle test on N-best hypotheses provided by the ANN goal identifier.

3.2 Rescoring of N-best goal hypotheses

The idea of using belief information to rescore N-best hypotheses of the understanding component is not new. However, a new aspect is that the N-best list is produced by an ANN-based goal identifier. We can convert the ANN outputs to probabilistic values as shown in Eq. 3 and treat the values as that produced by other stochastic conceptual models. Denoting a probability $P(G|C)$ by $P_{ANN}(G)$ and a belief-based conditional probability $P(G|B)$ by $P_B(G)$, one who assumes these two sources independent to each other can simply combine both scores by multiplication [2,4] as

$$P_{Combine}(G) = P(G|C,B)$$
$$\approx P(G|C)P(G|B) = P_{ANN}(G)P_B(G) \quad (6)$$

Note that one can apply scaling factors to the two probabilistic terms in order to give different weights. Another technique is to linearly interpolate between the two probabilities with interpolation weights estimated normally by an EM algorithm [1,3].

$$P_{Combine}(G) = \lambda P_{ANN}(G) + (1-\lambda)P_B(G) \quad (7)$$

Taking a logarithm to Eq. 6 results in an additive operation of the two terms, which is similar to Eq. 7. Therefore, we observe both techniques based on the same criterion of a linear combination.

3.2 Belief probability estimation

A belief often reflects the latest system prompt and the dialogue history. In this paper, the belief probability $P_G(B)$ described in Eqs. 6 and 7 is estimated from two sources, $P(G_t|S_t)$ and $P(G_t|S_t,G_{t-1})$, where G_t denotes the current user goal, S_t is the latest system prompt, and G_{t-1} is the goal of previous user turn. These two probabilities, referred to as $P_{B1}(G)$ and $P_{B2}(G)$ hereafter, can also be combined using linear interpolation.

$$P_B(G) = \beta P_{B1}(G) + (1-\beta)P_{B2}(G)$$
$$= \beta P(G_t|S_t) + (1-\beta)P(G_t|S_t,G_{t-1}) \quad (8)$$

The $P_{B1}(G)$ is an explicit model of a goal given a system prompt. It can be computed by counting on a training set with a simple additive smoothing technique,

$$P_{B1}(G) = P(G_t|S_t) \approx \frac{n(G_t,S_t)+\delta}{\sum_{G}(n(G_t,S_t)+\delta)} \quad (9)$$

where $n(G_t,S_t)$ is a co-occurrence count of G_t and S_t, and δ is an appropriate constant added for smoothing. The $P_{B2}(G)$ is calculated by a back-off smoothed trigram.

Actually, useful information derived from the dialogue history includes the number of user turns, the number of repetitions, the sub-dialogue state, and completed pre-requisite keys [10]. In the case of TIRA, a dialogue manager decides to prompt to the user by considering internal variables, which include information items input by the user. Therefore, the prompt itself implies what the user has stated. Since we defined unique prompts to each sub-dialogue, the prompt also reflects the sub-dialogue state. Tracking back to the previous user turn by $P_{B2}(G)$ helps capturing repetitions.

3.3 Nonlinear rescoring

Although linear interpolation techniques have been successfully used in various rescoring tasks [1,3], reliable interpolation weights cannot be estimated when combined scores are not linearly separable. When there is no such guarantee, nonlinear estimators are expected to be more effective.

Various kinds of nonlinear estimators such as ANN, probability density estimation (PDE), and Support vector machines (SVM) can be adopted for this task. In this paper, ANN, which is one of the classical algorithms for probability estimation, and SVM, which has been extensively employed, are compared to a typical linear interpolation model.

To make an ANN output a probabilistic value, we apply a normalization function as shown in Eq. 3. For the SVM, an algorithm for transforming an SVM prediction value to a probability has been described in [11], which uses a sigmoid function with parameters trained by an ML algorithm. See [11] for details.

4. Understanding-Error Detection

A common practice to construct an error detection mechanism is to extract potential confidence measures and use them to train an accept/reject classifier. Several algorithms such as linear discriminant analysis (LDA), ANN, and SVM have been successfully applied for the

classifier [5,6,12]. Since in the previous section, we have proposed to use a nonlinear estimator for the rescoring task, we expect that the estimator is also able to perform understanding-error detection. To achieve this task, several confidence measures are incorporated into the nonlinear estimator in addition to the $P_{ANN}(G)$, $P_{B1}(G)$, and $P_{B2}(G)$. After combining all scores including confidence measures, a simple criterion for utterance rejection is adopted.

$$\tilde{G} = \arg\max_{G_n \in \Phi_N} \{P_{Combine}(G_n)\} \text{ if } P_{Combine}(G_n) \geq \delta \qquad (10)$$

where Φ_N is an N-best list of semantic hypotheses and δ is a rejection threshold.

Sixteen features including confidence measures as shown in Table 2 are derived from the speech understanding component. Note that an error-detection mechanism was developed once in our previous speech understanding engine, where concept extraction was not a statistical model, and ASR confidence features were not taken into account [13]. In this paper, features derived from our new speech understanding model including ASR-based confidence scores are incorporated. See details of our proposed *concept similarity* (the 4th feature) in [13].

Table 2: *Features used for rescoring and error-detection.*

ID	*Feature*
1	$P_{ANN}(G)$
2	$P_{B1}(G)$
3	$P_{B2}(G)$
4	Concept similarity
5	Distance from $P_{ANN}(G)$ to the best ANN output
6	Best ANN output value
7	Second ANN output value
8	Third ANN output value
9	Difference between features 6 and 7
10	Difference between features 6 and 8
11	Difference between features 7 and 8
12	No. of ANN outputs with values greater than 0.5
13	No. of ANN outputs with values greater than 0.1
14	Concept tagger likelihood score
15	No. of extracted concepts
16	ASR likelihood score

In [13], we successfully applied an automatic feature selection process for error detection. The process successively selected a feature that maximized the accept/reject classification rate when combining with the previous selected features. In this paper, this process is also conducted for selecting the best subset of features.

5. Experiments

Experiments were performed on Thai hotel reservation corpora, which were collected under a project of the first Thai spoken dialogue system, TIRA. We collected data in two ways. First, we obtained a large utterance text via our specific web site simulating expected dialogues. Thai natives were requested to answer to system prompts by typing in the web page. So far, 5,869 utterances from 150 natives have been semantically annotated. They were used for a training set (TR) of the understanding model. Second, real speech signals were collected during evaluations of the TIRA system. Out of 1,101 automatically recognized utterances of these speech files, 500 were reserved for a development test set (DT) and the rest were for an evaluation test set (ET). Table 3 presents characteristics of data sets.

5.1 Dialogue-state dependent semantic modeling

This section explains an experiment on the use of a dialogue-state dependent semantic n-gram model (DD-SM), the (I.2) method described in Sect. 3. An important point is a criterion for dialogue-state clustering. In our case, user utterances can be clustered based on either system prompts or user goals. Utterances responding to each system prompt can be various kinds of goals especially in mixed-initiative dialogues. In the case of clustering by goals, 42 kinds of goals were grouped into a smaller number of clusters. The grouping criterion was based on an n-gram similarity [14].

Semantic n-gram models were constructed for each dialogue-state and merged with the general n-gram model using either the MAP or interpolation criterion as shown in Eq. 5 with interpolation weights estimated by an EM algorithm. The experiment showed that the DD-SM clustered by system prompts and merged in the linear interpolation technique achieved the best result. This model is incorporated with N-best goal rescoring in the next experiment.

5.2 *N*-best goal rescoring

An oracle test showed that over 10% improvement of goal accuracy could be obtained given a few N-best hypotheses produced by the ANN goal identifier. Based on the results of the oracle test, N of 5 was used through out our experiments.

In practice, we have two sources of collected data as previously explained. Since the TR set contained only pairs of Q&A, not whole dialogues, it was used to estimate the $P_{B1}(G)$, whereas the $P_{B2}(G)$ could be estimated only by the DT set.

Two nonlinear estimators, ANN and SVM, were compared with the simple linear interpolation algorithm (LI) in rescoring N-best goal hypotheses. The nonlinear estimators utilized training samples derived from the DT set. The training set of the ANN contained 2500 samples (500 utterances with $N = 5$) of $(\vec{x}, t)$ pairs, where $\vec{x}$ was a 3-dimensional vector of $x_i \in \{P_{ANN}(G), P_{B1}(G), P_{B2}(G)\}$ and $t \in \{1, 0\}$ was a target value ($t = 1$ for a correct-goal sample, and $t = 0$ otherwise). The ANN estimator was constructed using the SNNS tool [15].

Training samples for the SVM were similar to that of the ANN, except that the target values of negative samples were set to -1. The SVM^{light} toolkit [16] was utilized in experiments. LI weights were also estimated by the DT set. Several constraints in each algorithm

including the number of ANN hidden nodes and the SVM kernel functions as well as their parameters were optimized separately for each case. Three kinds of kernel functions including linear, polynomial, and radial basis functions (RBF) were evaluated.

Two measures were used in evaluations, *goal accuracy* (GAcc) and *concept-value accuracy* (VAcc). The latter was the number of concepts, whose values were correctly matched to their references, divided by the total number of concepts that contained values. Concepts in consideration were only those necessary for communication given an identified goal.

Table 3: *Characteristics of data sets.*

Characteristic	*TR*	*DT*	*ET*
# Goal types	42	40	40
# Concept-value types	20	18	18
# Concept-values	6,365	366	439
% Out-of-goal		5.2	5.3
% Word error rate		22.8	21.0

Table 4: *Goal accuracies of the DT set after rescoring by various algorithms.*

Algorithm	*GAcc*
LI	66.4
ANN	**78.4**
SVM (linear)	73.4
SVM (RBF)	75.0
SVM (poly)	73.6
No rescoring	75.0

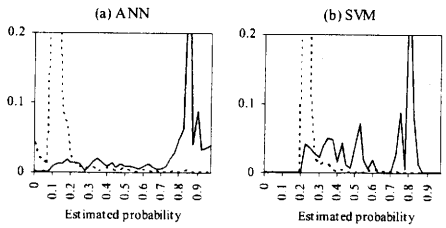

Figure 1: *Histograms of estimated probabilities, solid lines: correct-goal samples, dotted lines: incorrect-goal samples.*

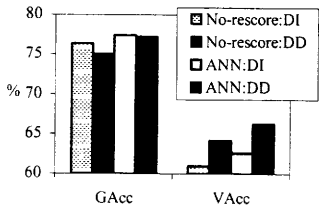

Figure 2: *Goal accuracy results of the ET set.*

Table 4 presents goal accuracy results of the DT set after rescoring by each algorithm. It is shown that the ANN can improve the goal accuracy, whereas no improvement can be obtained by SVM and LI approaches. We then analyzed the probabilities produced by the ANN and SVM estimators. Figure 1 plots histograms over estimated probabilities where solid lines and dotted lines denote distributions of correct and incorrect goal samples. The graphs produced by the ANN indicate clearer separation of the right and wrong samples compared to the SVM. This is probably due to the fact that the SVM, which has been proven to be efficient for classification tasks, is inappropriate for probability estimation, at least in our task.

Figure 2 shows evaluation results on the ET set using the optimized ANN estimator with a comparison between the use of DD-SM and a dialogue-state independent semantic model (DI-SM). Compared to the DI-SM without rescoring, the ANN improved the GAcc relatively by about 1.3% regardless of whether the DD-SM was use. The DD-SM highly contributed to improving concept-value recognition, as it increased the VAcc by relatively 8.7%. We concluded that the use of an ANN estimator for belief-based N-best rescoring with the DD-SM was the most effective for our task.

5.3 Incorporating error detection

As described in Sect. 4, we expect that the nonlinear estimator used in the rescoring task can also be able to detect understanding-errors. In this section, the best subset of features was automatically selected from the sixteen features shown in Table 2 using the MCE-based feature selection method. Note that the first three features were those conducted for the rescoring task, whereas the rest were added for a purpose of error detection. The selected features were used to train the nonlinear estimator for an accept/reject classification task. An additional measure, *Accept/reject accuracy* (ARAcc), was considered in evaluations. It represented overall error-detection performance and could be calculated by

$$\text{ARAcc} = (n(CA) + n(CR))/n_T \qquad (11)$$

where $n(CA)$ and $n(CR)$ were the number of utterances correctly accepted and correctly rejected, and n_T was the total number of test utterances.

Table 5 shows results on the DT set when using ANN and SVM estimators for rescoring and error detection. The DD-SM was utilized in every case. Results clearly showed that the ANN again achieved the best performance.

We then varied and chose a δ (described in Eq. 10) that maximized ARAcc and used it as an operating point in evaluations on the ET set. Table 6 presents evaluation results, where "Rescoring-1" denotes the system with a rescoring process based only on the first three features (the system set up in Sect. 5.2), and "Rescoring-2" denotes the system with rescoring and error-detection processes using selected features in Table 5. According to Table 6, goal accuracies after rescoring by both the "Rescoring-1" and "Rescoring-2" systems are almost equal. Therefore, we concluded that the nonlinear estimator could not only be used for a rescoring task, but

also for an error-detection task with no effect to the rescoring goal accuracy.

Table 5: *Rescoring and error-detection results of the DT set at the MCE operating point (δ = 0.5).*

System	*Selected features*	*GAcc*	*ARAcc*
No rescoring		75.0	
ANN	1-4,8,11,13,16	**84.8**	**97.2**
SVM (linear)	1-4,13,16	77.6	88.2
SVM (RBF)	1-4,13,15	78.8	90.6

Table 6: *Evaluation results of the ET set using the ANN with error detection.*

System	*GAcc*	*ARAcc*	*CA*	*CR*
No-rescoring	75.0			
Rescoring-1	77.2			
Rescoring-2	77.5	84.4	94.5	53.1

6. Conclusions

Two issues were addressed in this paper. First, we showed that in an *N*-best rescoring task, which incorporated system-belief scores, a nonlinear estimator was superior to a normal linear interpolation method. The use of a nonlinear estimator together with a dialogue-state dependent semantic model was optimal for our task.

Second, we showed that by incorporating confidence measures into the nonlinear estimator, it could not only perform the *N*-best rescoring task, but also able to detect understanding errors. Adding confidence measures had no negative effect to the rescoring performance. An ANN was proven to be efficient for both the rescoring and error-detection tasks.

Although we could gain an improvement by *N*-best rescoring, the improvement was rather small compared to its upperbound observed in the oracle test. According to an error analysis, we found many errors caused by semantically similar goals such as "request_cost" and "request_listcost". The former goal aims to specify a desired cost, whereas the latter asks for a list of cost. These errors can be managed by a dialogue manager. With an appropriate dialogue strategy, the system can respond to these similar goals in the same manner with no need to recover the error.

References

[1] Bousquet-Vernhettes, C., and Vigouroux, N., "Context use to improve the speech understanding processing", *Proc. SPECOM 2001*, pp. 89-92.

[2] Raymond, C., Estève, Y., Béchet, F., De Mori, R., and Damnati, G., "Belief confirmation in spoken dialog systems using confidence measures", *Proc. ASRU 2003*, pp. 150-155.

[3] Higashinaka, R., Nakano, M., and Aikawa, K., "Corpus-based discourse understanding in spoken dialogue systems", *Proc. ACL 2003*, pp. 240-247.

[4] Seide, F., Rueber, B., and Kellner, A., "Improving speech understanding by incorporating database constraints and dialogue history", *Proc. ICSLP 1996*, pp. 1017-1020.

[5] Weintraub, M., Beaufays, F., Rivlin, Z., Konig, Y., and Stolcke, A., "Neural-network based measures of confidence for word recognition", In *Proc. ICASSP 1997*, pp.887-890.

[6] Ma, C., Randolph, M., and Drish, J., "A support vector machines-based rejection technique for speech recognition", In *Proc. ICASSP 2001*, pp.381-384.

[7] Wutiwiwatchai, C., and Furui, S., "Combination of finite state automata and neural network for spoken language understanding", *Proc. Eurospeech 2003*, pp.2761-2764.

[8] Wutiwiwatchai, C., and Furui, S., "Hybrid statistical and structural semantic modeling for Thai multi-stage spoken language understanding", *To appear in Workshop of HLT/NAACL 2004*.

[9] Bridle, J. S., "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition", *Neurocomputing: Algorithms, Architectures and Applications*, Fogleman Soulie, F. and Herault, J. (eds.), Springer-Verlag, 1990.

[10] Walker, M., Wright, J., and Langkilde, I., "Using natural language processing and discourse features to identify understanding errors in a spoken dialogue system", *Proc. ICML 2000*, pp. 1111-1118.

[11] Platt, J., "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods", *Large Margin Classifiers*, Smola, A., Bartlett, P., Scholkopf, B., Schuurmans, D. (eds.), MIT Press, 1999.

[12] Hazen, T., Seneff, S., and Polifroni, J., "Recogniton confidence scoring and its use in speech understanding systems", *Computer Speech and Language*, 16, 49-67, 2002.

[13] Wutiwiwatchai, C., and Furui, S., "Confidence scoring for ANN-based spoken language understanding", *Proc. ASRU 2003*.

[14] Damashek, M., "Gauging Similarity with ngrams: Language-Independent Categorization of Text", *Science*, Vol. 267, pp. 843-848, 1995.

[15] Zell, A., Mamier, G., Vogt, M., Mach, N., Huebner, R., Herrmann, K. U., Doering, S., and Posselt, D., "SNNS Stuttgart neural network simulator, user manual", University of Stuttgart.

[16] Joachims, T., "Making large-scale SVM learning practical. Advances in kernel methods - support vector learning", Schölkopf, B., Burges, C., and Smola, A. (eds.), MIT-Press, 1999.