THREE PHASE VERIFICATION FOR SPOKEN DIALOG CLARIFICATION

Sangkeun Jung **Cheongjae Lee** **Gary Geunbae Lee**

Department of Computer Science and Engineering
Pohang University of Science and Engineering
San 31, Hyoja-Dong, Pohang, 790-784, Korea
+82 54 279 5581
{hugman,lcj80,gblee}@postech.ac.kr

ABSTRACT

Spoken dialog tasks incur many errors including speech recognition errors, understanding errors, and even dialog management errors. These errors create a big gap between user's will and the system's understanding, and eventually result in a misinterpretation. To fill in the gap, people in human-to-human dialog try to clarify the major causes of the misunderstanding and selectively correct them. This paper presents a method for applying the human's clarification techniques to human-machine spoken dialog systems. To increase the error detection precision and error recovery efficiency for the clarification dialogs, error detection phase is organized into three systematic phases and a clarification expert is devised for recovering the errors using the three phase verification. The experiment results demonstrate that the three phase verification could effectively catch the word and utterance-level errors in order to increase the SLU (spoken language understanding) performance and the clarification experts can actually increase the dialog success rate and the dialog efficiency.

Categories and Subject Descriptors
I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *Discourse, Speech recognition and synthesis*

General Terms
Algorithm, Performance, Design, Languages, Human Factors, Verification

Keywords
Clarification Dialog, Three Phase Verification, Clarification Expert, Spoken Language Understanding, Dialog Management.

1. INTRODUCTION

Clarification Dialog (CD) is one of the dialog types used in attempting to resolve misunderstanding between human to human, or human to computer during the dialog. The following example shows a typical clarification dialog in an error-prone spoken environment in tv-guide domain.

Example)

- USER: I want to watch a ***drama***[1] ***Hae-Sin***
- SYS: ***Documentary Her-Jun*** is on ***MBC***. Do you want to watch it? (*Drama and Hae-Sin are misrecognized respectively as Documentary and Her-Jun*)
- USER: No, I want to watch a ***drama Hae-sin***.
- SYS: Please repeat the program name you want to watch. (System internally verifies that the 'drama' and "Hae-Sin" are not correctly recognized. System tries to clarify the important program name first)
- USER: Hae-Sin
- SYS: Drama ***Hae-sin*** is on ***KBS. Do you want to watch it?***
- USER: ***Yes***
- SYS: OK, showing Hae-Sin on KBS

For a successful clarification dialog like the above example, we need to solve two problems. The first problem is to select the targets to be clarified and classify the error types, and the second is to clarify and recover the targets in an intuitive and efficient way.

The first problem is conventionally called as a belief confirmation. Belief confirmation techniques have been explored by many researchers and many confidence measures have been developed. Most of the researches focused on measuring how much we can trust the speech recognition results. Many good features and the

---

[1] [***Bold-italic*** words designate the important content information to tv-guide domain dialog system]

methods of confidence measurement in the level of speech recognition decoder are summarized at [2][3].

Recently, researchers try to include more semantic level information for belief confirmation using Latent Semantic Analysis (LSA) [1] and the information from the understanding module [9]. Recent trends of confidence measuring and utterance verification techniques are well summarized in a recent survey paper [4].

The second problem has been studied using the belief confirmation techniques. Torres et. al.'s work [9] showed how to use confidence scores for calculating a transition probability in the dialog state-transition network where the confidence scores are calculated using the method of [3]. McTear et. al. [5] developed an object-oriented dialog system which can both detect and handle errors, and their system also adapted [3]'s method for detecting the errors.

Major limitation of the most previous clarification dialog researches is that the targets the system tries to correct are limited to words, and they consider only speech recognition errors even though many errors can also come from the spoken language understanding (SLU).

In the case of detecting both speech recognition and SLU errors [7], they try to use a single integrated estimator to classify the type of errors even though the characteristics of the speech recognition errors and the SLU errors are totally different, which results in low detection accuracy.

To overcome these previous researches' limitations, we devised a three-phase verification method and a clarification expert. We extend the range of errors by considering the errors coming from not only speech recognizer but also SLU module. To detect the complicated errors with high accuracy, we cascade error detection process into three phases – ***Word Error verification, Utterance Verification and Slot-Value Verification***. The multi-level rich information generated by this three phase verifier is passed to the ***Clarification Expert*** (CE), which is specially devised for handling clarification dialogs, and the CE determines adequate clarification strategy considering both error detection information and discourse status.

This paper is organized as follows: An expert-based dialog management architecture as our base-line architecture for the clarification is described in section 2. Based on this architecture, a three-phase verification method along with new ***Information Potential*** measure will be introduced in section 3. The detail of the clarification expert will be described in section 4. Extensive experiments and analyses are shown in section 5, and finally a conclusion will be drawn in section 6.

2. SITUATION BASED SPOKEN DIALOG MANAGEMENT SYSTEM

O'Neill et. al. proposed an object-oriented dialog system in [6]. Inspired by O'Neill et. al.'s work, and motivated to overcome the conventional dialog systems' weaknesses, we developed a situation-based spoken dialog management system using the following two dialog modeling principles:

- Dialog management should be state-transition free and based on the current situation for general response generation (Situation-based dialog management)
- Domain-dependent dialog management should be based on a specific expert for more efficient management.

Most state-transition-based dialog management systems rely on the fixed state transition to determine the dialog status using a finite state transition model. This state transition-based dialog modeling guarantees fast system build-up and easy dialog modeling. But, it is not flexible to handle various natural language dialog phenomena, because the next state of the dialog is fully determined by the fixed transition state. Also, the state transition dialog modeling makes it difficult to transfer a current domain dialog model to another domain, because we would need to re-design the whole transition network again. To avoid this rigidity in management, we developed a state-transition-free dialog management model. Our system does not use a state-transition network, but uses a situation-based dialog management strategy. The definition of the ***situation*** in our system is as follows:

- A situation is determined by various information of the current dialog status including:
  - A. User's utterance and intention (dialog acts)
  - B. Set of semantic slots and values
  - C. Confidence status of each slot
  - D. History of dialog in a current session
  - E. System's previous intention
  - F. Database query results of the current user query

To determine the system's intention and proper responses, we consider all the above situation-related information and use the three kinds of situation-based rules as follows:

- Situation-**action rules**: rules for describing the system's actions under the current situation.
- **Constraint-relax rules**: rules for relaxing the constraints on database queries.
- **Frame-reset rules**: rules for restarting a new dialog frame for the case of domain switch and dialog closing

Like O'Neill et. al.'s domain experts [6], we pursue an expert-based dialog management strategy to conduct a specific domain-oriented dialog. Each expert is designed as a specialist for handling specific dialog patterns. For example, the tv-guide expert handles tv-guide related utterances, and the movie-guide expert handles movie-guide related utterances. Experts of each domain are implemented by designing the 'situation-based rules' for the corresponding domain

The biggest advantages of the expert based dialog management are that the system not only conducts a specific domain-oriented dialog efficiently but also provides an architectural beauty of implementing the separate clarification dialog experts. In other words, if we view the clarification dialog phenomena as another special form of dialog patterns like a tv-guide and a movie-guide, clarification dialog model also can be designed as a special expert, e.g., ***Clarification Expert***.

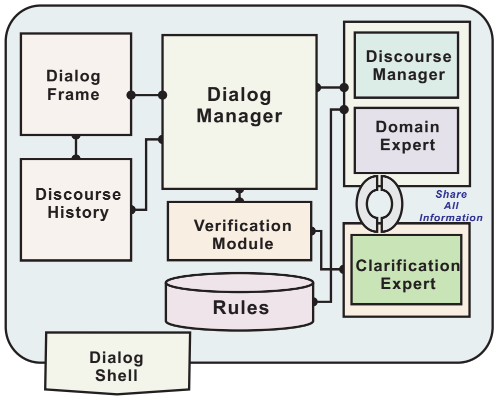

Fig 1 Situation Based Dialog Management Architecture with the Clarification Expert.

Fig 1 illustrates a situation-based dialog system architecture with the connection to the clarification expert. Each component respectively has the following role:

- Dialog Manager: A hub module that communicates with ASR (Automatic Speech Recognizer), SLU (Spoken Language Understander) and an error verification module. It also manages other dialog components in this architecture.
- Dialog History: Consists of the following two parts:
  - Dialog Frame: Provides a current semantic frame for a dialog
  - Discourse History: Stores history information extracted from user utterances and dialog frame status
- Discourse Manager: Can handle the dialog by inheriting a general expertise to one of the proper Domain Expert. The discourse manager handles very basic dialog patterns such as updating dialog status, saving and restoring a dialog history.
- Domain Expert: Takes the responsibility in handling domain specific dialogs. Expert itself is a domain specific, but it takes a generic dialog strategy inherited from the Discourse Manager. Each domain expert has its own situation-action rules, dialog frames and the history.

The relationship of the discourse manager and the domain expert is complementary each other. The discourse manager decides only generic dialog strategy, and it is totally domain independent. Therefore it can be used by any domain expert. Each domain expert inherits generic discourse manager knowledge, and can handle generic dialog along with domain specific dialog patterns.

For implementing ASR, we developed a speech recognizer based on the Hidden Markov Model Toolkit (HTK). We modified the HTK for making and providing decoder level information to the following three-phase verification process.

3. THREE PHASE VERIFICATION

To select targets to be clarified, we develop a three-phase error verification method. The first phase is a word error verification which is conventional belief confirmation on words that are recognized by the speech recognizer. The second phase is utterance verification which examines the whole utterance's properness to progress the dialog management further. To do this, we devise the concept of information potential which can measure the properness of whole utterance in the sense of ASR and SLU confidence. The third phase is slot-word verification which examines the slot and the value which are extracted by SLU from user utterances.

The target of the verification in each step along with the proper example is shown as follows:

Example 1: Examples of the three-phase verification

User Utterance & Speech Recognized Utterance	
I want to watch Drama Hae-Sin on KBS (User utter.) I want to watch Drama Bae-Sin on SBS (ASR Result)	
1st Phase	Word Error Verification
I(confidence scores : 89.45% / correct) want (78% / correct) to(45.32% / error) watch(78.67% / correct) Drama(87.32% / correct) Bae-sin(65.2% / correct) on(93% / correct) SBS(29.24% / correct)	
2nd Phase	Utterance Verification
I want to watch Drama Bae-Sin on SBS Information potential : 67.42%	
3rd Phase	Slot-Value Verification
Channel – SBS: 65% Genre – Drama: 97% Program_Name – Bae-Sin: 45%	

3.1 Word Error Verification

Verification in the first phase is similar to the conventional belief confirmation approaches [2][3][4]. It examines every word that is recognized by the speech recognizer. However, we don't use the Word Error Verification (WEV) results directly; the goal of this step is to provide rich information to the utterance verifier. This is one of the big difference from the previous clarification dialog researches [5][6][9].

We adopted some of the good confidence measures from [3][4]. We used a Maximum Entropy (MaxEnt) classifier [8] for combining good confidence features and calculating the confidence scores for classifying the word recognition errors. The followings are the description of the classes and the features for our MaxEnt classifier.

- ***Classes***: Correct/Error
- ***Normalized acoustic scores***: Frame normalized acoustic scores of a node in the lattice.
- ***Language model scores***: Word Trigram scores $P(W_i|W_{i-2},W_{i-1})$

- ***N-best purity***: The fraction of the N-best hypotheses in which the hypothesized word appears in the same position in the utterance
- ***NFrames***: The number of time frames of the word
- ***Word Length:*** The length of the word.
- ***Word Lexical:*** The lexical form of the recognized words.

3.2 Utterance Verification

Goal of the clarification dialog is to fill a gap between user's intention and the intention that the system finally understands. It is closely related to the measure of the amount of distortion in the channel between users and the dialog system. In other words, calculating the distortions in the series of the noisy channels – user's intention is formed as a spoken language form, the spoken language is recognized by the ASR, the result of the ASR is understood by the SLU, and finally the SLU results are passed to the dialog manager – is the natural approach to start a clarification dialog. To do this, we devised a concept of information potential by the following:

Information Potential = Ratio of correctly carrying user's information (intention) to the dialog system in the noisy channel between user and the system.

= How much we can trust the information that the dialog system understands.

= P (Trust | Information that the dialog system understands)

Namely, measuring the information potential can be formulated by calculating the confidence of recognizing and understanding module in the channel between user and the dialog system. In other words, it can be put into words as follows:

Information Potential ~ < Confidence of ASR >

~ < Confidence of SLU >

~ <Other information that the system has>

('~' means that there exists a relationship)

Based on the above definition, we calculated the information potential by combining various information from ASR and SLU modules using logistic regression. Used features are as follows:

- **Mean of normalized acoustic scores**: Mean of the frame normalized acoustic scores of each word in the sentence.
- **Mean of language model scores**: Mean of the language model scores of each word in the sentence
- **Mean of the N-best purity**: Mean of the N-best purity scores of each word in the sentence.
- **Mean of the understanding scores**: Mean of the log-likelihood scores of the SLU.
- **Mean of the word verification scores**: Mean of the word error confidence scores generated by the word-error verifier in the first phase.
- **Number of words**: The number of words in the utterance
- **Predicted word error rate**: The ratio of the word errors predicted by the word error verifier.

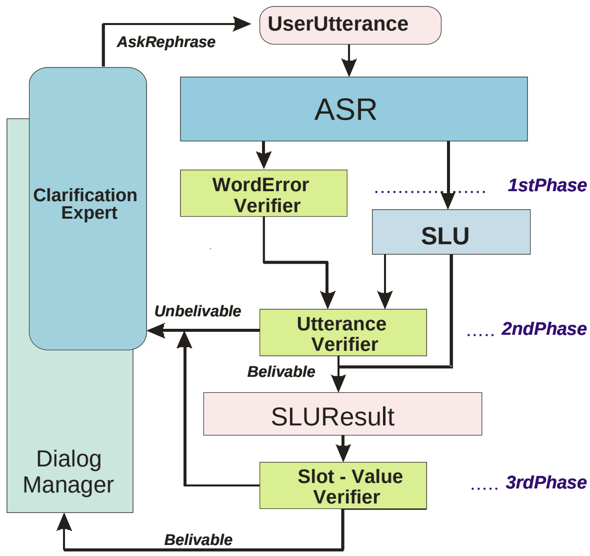

Fig 2 Flow of three phase verification and clarification

Shown as in Fig 2, after calculating the information potential, the dialog system decides the utterance level clarification strategy. If the information potential is lower than the threshold, system tags the utterance as "Unbelievable" and passes the information to the clarification expert to decide proper system's responses. In this case, system will ask the user to rephrase the whole utterance again because the utterance itself is estimated as "Unbelievable" in the sense of both ASR and SLU.

However, if the information potential is higher than the threshold, system tags the utterance as "Believable" and continues to the third phase verification – **Slot-Value Verification**. Even though the utterance itself can be estimated as "Believable," there may be some errors in the level of slot and value recognition. The goal of the third phase verification is to find the slot-value level errors in detail.

3.3 Slot-Value Verification

The slot-value verification is executed when the utterance which is recognized by the ASR and understood by the SLU, is estimated as "Believable." In other words, it verifies every slot and value pair which is extracted by the SLU. The difference between this verification and the conventional belief confirmation is that we calculate the confidence of the slot-value pair by considering not only the ASR information but also the SLU information. So, we can focus on the recognition performance of the more important content words which are more critical to the successful dialog completion.

We can turn the slot-value confidence measuring problem to classification problem as we did for the word error verification. We used the same features and the MaxEnt classification method that we have used in the word-error verification in section 3.1. The only difference is the following new feature:

- **Understanding Scores**: the likelihood of slot and value of the spotted words generated by the SLU

If all of the slot and value of the utterance are classified as "Correct," the utterance and the slot-values are directly passed to the dialog manager. However the "Error" tagged slot-values and the confidence are passed to the clarification expert to make a decision of proper clarification strategy.

4. CLARIFICATION DIALOG STRATEGY

From the results of the utterance and slot-value verifier, we can get the targets which should be clarified. The targets can be a full sentence or a set of words. For clarifying these targets efficiently and systemically, we introduce a clarification expert in our situation-based dialog management architecture.

As previously mentioned, our dialog system is strongly based on expert system architecture. Each expert takes the responsibility of handling a certain domain dialog, and it is designed to manage domain specific dialog patterns. Therefore if we reformulate the 'clarification' as specific dialog patterns, we could model the clarification as one of the expert system. However, there should be some differences between a clarification expert and the other domain experts. The differences are as follows:

- The clarification expert is a secondary expert different from the primary working domain experts.
- The clarification expert should be domain independent.
- The clarification expert should be able to share all the information of the current primary working domain expert.

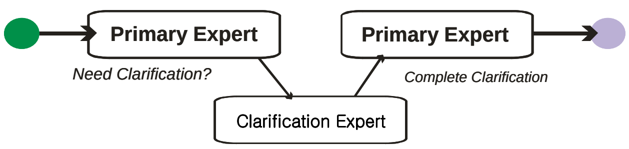

Fig 3 Switch to the clarification expert

As shown in **Fig 3**, if the three-phase verification alarms the dialog manager that the utterance or some words should be recovered, the dialog manager pauses the working primary domain expert and gives the control to the clarification expert. In this step, dialog manager makes the clarification expert access to all the primary expert's information including the dialog frame and the discourse history.

Clarification expert decides clarification dialog strategy by considering the confidence scores provided by the three-phase verification module and other domain-related information.

To decide more efficient and systematic clarification dialog strategy, clarification expert is considering the following properties:

- Property 1: Dependency between each error information and dynamic change of the error information relationship
- Property 2: Relative importance of the error information

Most of the error information has a strong dependency with other information in the same dialog domain. For example, `Larry King Live' is always broadcasted on `CNN' and the Korean popular drama `Hae-Sin' is always on `KBS'. There is a certain dependency between the program title and the channel. Without considering these dependencies, we end up taking unnecessary clarification steps. The following clarification target example demonstrates the importance of considering the dependency in deciding proper clarification strategy.

Example 2. Target example of selected slot-values for clarifying

User Utterance	I want to watch drama Hae-Sin on KBS
ASR Result	I want to watch drama Bae-Sin on SBS
Targets needed to be clarified	Program_Name - Bae-Sin Channel - SBS

In this example, if we don't know there is a strong dependency between 'Hae-Sin' and 'KBS', the clarification expert asks users to rephrase both channel and program name. However if we know the dependency, clarification expert doesn't need to clarify channel in the moment that 'Bae-sin' is clarified to 'Hae-sin' because system already knows 'Hae-Sin' is always on 'KBS'. Like this, we can implement more efficient clarification dialog strategy by considering property 1.

Property 2 can be used for choosing the clarification order among multiple targets. .As in the example 2, when there are more than two slot-values to be clarified, the clarification expert considers the relative importance property to set the priority of clarification. In most of the cases, the priority from relative importance is closely related to the range of the slot types. For example, **Fig 4** illustrates the range of information types on tv-guide domain. As we can see, in most of the cases, if the 'playing actor/actress' is determined surely, 'program_name', 'channel' and 'genre' are determined automatically. Therefore, in example 2, the clarification expert tries to clarify 'program name' first

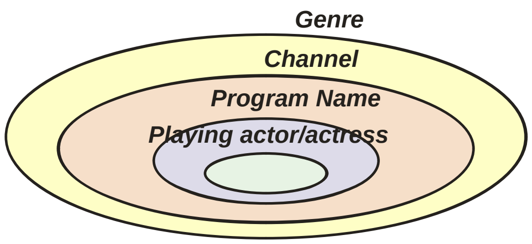

Fig 4 Range of the information types on tv-guide domain

In addition to the priority of each slot, clarification expert also considers the slot-value confidence scores that are generated by the Slot-Value Verifier. The clarification ordering algorithm can be described as follows:

> For each slot-and-value, calculate
>
> **Clarification Order Score(k)**
>
> $= \alpha$*Priority_Score(k) + β*Confidence(k)
>
> where k is the slot-and-value pair
>
> Choose the slot-and-value k which has the largest **Clarification Order Score**

5. EXPERIMENT AND ANALYISIS

We collected monthly TV schedule for developing and testing our dialog system. The ASR's language model and the SLU are trained to be able to handle all kinds of possible utterances which might happen in the selected tv schedule. Our ASR is based on HTK and uses pre-trained dialog acoustic model[2] and adopted tv-guide domain specific language model. The performance is word error rate (WER) 15.3%.

To verify our system's performance, we did experiments on both three phase verifier and end-to-end dialogs.

5.1 Experiments on the three-phase verification

Our three-phase verifier is designed to use other verifier's information in a cascade manner. As we mentioned in section 3.2 and 3.3, utterance verifier and slot-value verifier heavily depends on the confidence scores from the word-error verifier. Because of this dependency, the performance of the word-error verification is very important.

As we can see in **Table 1**, our word-error verifier shows 9.07% for false rejection rate and 7.21% for false acceptance rate. Both rates are lower than 10 %, which means that our belief confirmation method is quite accurate. Furthermore low false acceptance rate (7.21%) is quite noticeable, because it guarantees accurate filtering of the erroneous words.

Table 2 shows the performance of the information potential based utterance verifier. The UV-Positive utterances are the utterances that the utterance verifier tags 'Believable'. We tried to perform the SLU on the set of UV-Positive tagged, UV-Negative tagged and normal utterances. As we can see in table 2, the UV-Positive utterances can be more accurately SLU decoded than normal utterances and UV-Negative utterances. It means that the utterance verifier can guarantee for the dialog system not to try to decode improper utterances that have high chance of mis-interpretation.

Fig 5 depicts the relationship between the SLU performance and the information potential threshold. The SLU performance increases according to the increase of the information potential threshold, and converges finally to 100%. This means that our information potential measure works perfectly as a verifier that distinguishes proper utterances from improper utterances.

Table 1 Confusion Matrix on Word Error Verification

Predict / Reference	Correct	Error
Correct	129535	10931
Error	10076	109469
False Reject Rate	9.07%	
False Acceptance Rate	7.21%	

Table 2 SLU performance based on the utterance verification (Threshold = 0.85)

	SLU performance
UV Positive utter.	95.65%
UV Negative utter.	79.05%
All utter.	81.38%

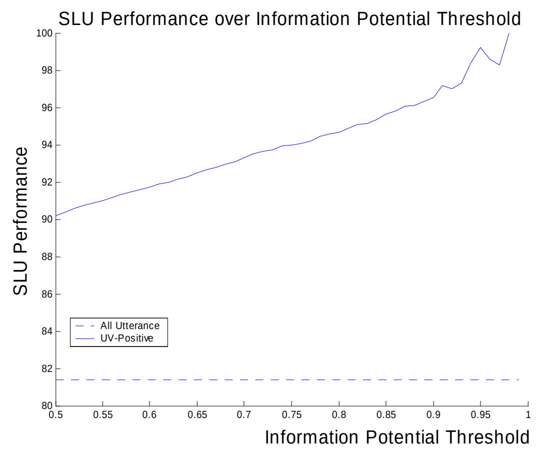

Fig 5 SLU performance vs. Information Potential Threshold

---

[2] Model was trained on Travel Plan DB, which contains 225 dialog set, 18 hour recoding. It was courteously provided from Seoul National University.

5.2 Experiments on the clarification dialog

To demonstrate the effect of the clarification expert based on the three-phase verification method, we did end-to-end dialog tests on various input modes.

First, we asked the test volunteers to set the 10 dialog goals (tasks) that they want to achieve using the dialog system. Each volunteer is asked to make conversations with the dialog system in three different modes – text input mode, speech input mode and speech input with clarification mode.

The volunteers are asked to evaluate every system's response and clarification's success. This dialog history and evaluating records are saved and analyzed. The results are summarized in **Table 3**. It summarized 6 volunteer's 10 different dialogs on each mode.

As we can see in **Table 3**, the dialog success rate of the text input surpasses both the speech input without and with the clarification dialog. However, we can verify that the speech input with clarification dialog has higher success rate than the one without clarification dialog.

Another noticeable point of this table is that the average system action turns per dialog decrease from 13.00 in speech input to 11.58 in our clarification dialog mode. This decrease comes from the preventive effects of the system's mis-interpreation by pre-verifying the properness of the utterances and the slot-values. Namely, the three-phase verifier discourages the dialog system from doing actual actions on improper utterances and this discouragement contributes to the decrease of average total turns per dialog. The user, the system action and the total dialog turns are defined as follows:

- **User turn**: the number of user's utterances
  : + 1 cost for one utterance
- **System Action turn** : Sum of the two action turns
  - System response utterances : + 1 cost for one system response utterance
  - Physical Action: including following actions
    : turn on/off TV, moving channel, DB Accessing
    : + 1 cost for each physical action
- **Total turn**: Sum of the total cost of user's turn and the system action's turn.

As shown in the experiment results, our clarification approach is successful at decreasing the system action turn cost. It makes the conversation shorter and therefore increases the user's satisfaction.

Table 3 also shows the occurrence and success rates of the utterance verification (UV) level and the slot-value verification (SV) level clarification. While the occurrence rate of the UV is quite high, the occurrence rate of the SV is very low. It is because that the utterance verification is working as a first-phase remedy against the errors. This shows that the utterance verification and clarification is more suitable strategy for the dialog clarification.

The reason of lower correction rate of the slot-value clarification can be due to the low speech recognition performance, especially for the short word recognition used in the slot-values.

This problem could be solved by using a multi-level speech recognizer according to each different situation. For example, if the system asks users to repeat a partial word, the isolated speech recognizer has much more advantage than the continuous speech recognizer that we have used in this research.

Table 3 Quantitative performance measures for the clarification dialog effects

	Text input	Speech input without clarificat ion	Speech input with 3-phase verification based clarification
Dialog Success Rate	0.92	0.76	0.85
Average total turn per dialog	16.60	19.50	17.95
Average user turn per dialog	5.53	6.50	6.37
Average system action turn per dialog	11.07	13.00	11.58
Good system response rate per system utterance	0.88	0.65	0.67
UV occurrence rate per user utterance	-	-	0.095
UV occurrence rate per dialog	-	-	0.604
Utterance clarification success rate	-	-	0.690
SV occurrence rate per user utterance	-	-	0.010
SV occurrence rate per dialog	-	-	0.063
Slot-Value clarification success rate	-	-	0.333

From the experiment results, we can conclude that our three-phase verification based clarification approach is viable and feasible: First, it increases the dialog success rate for better task completion. Second, it reduces the total turn for dialog from 19.5 in speech input to 17.9 in three-phase verification based approach. Third, it prevents possible mis-interpretation by verifying the utterances and slot values carefully before actual system action.

6. Conclusion

In this paper, we tried to solve the two different but inter-related problems of error detection and dialog clarification. To achieve accurate error detection, we systemically separate error detecting process into three-phase error verification. In this process, we consider that errors come from both ASR and SLU. To measure the comprehensive confidence of the user utterance, we devised the information potential measure. For modeling efficient clarification dialog strategy, we modeled a clarification expert to specially handle the clarification dialogs based on our situation-based dialog management model. Clarification expert considers both three-phase verification's information and two essential properties to decide efficient and systematic clarification strategies. Through various experiments of both three-phase verifier and clarification dialogs, we can confirm that our three-phase error verification and clarification expert approaches are successful for implementing robust clarification dialogs in the spoken human-to-machine interfaces.

7. ACKNOWLEDGMENTS

This research was supported by the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea.

8. REFERENCES

[1] S.J. Cox and S. Dasmahapatra, "A semantically-based confidence measure for speech recognition," in Proceedings of the ICSLP 2000, Beijing, 2000.

[2] T.J. Hazen, J. Polifroni, and S. Seneff, "Recognition confidence scoring and its use in speech language understanding systems," Computer Speech and Language, vol. 16, no. 1, pp. 49–67, January 2002.

[3] T. J. Hazen, T. Burianek, J.Polifroni, and Stephanie Seneff, "Recognition confidence scoring for use in speech understanding systems," in Proceedings of the ISCA ASR2000 Tutorial and Research Workshop, Paris, September 2000.

[4] H. Jiang, "Confidence measures for speech recognition," Speech Communication, vol. 45, no. 4, pp. 455–470, April 2005.

[5] M. McTear, I. O'Neill, P. Hanna, and X. Liu, "Handling errors and determining confirmation strategies -an object-based approach," Speech Communication, vol. 45, no. 3, pp. 249–269, March 2005.

[6] I. O'Neill, P. Hanna, X. Liu, D. Greer and M. McTear, "Implementing advanced spoken dialogue management in Java," Speech Communication, vol. 54, no. 1, pp. 99–124, January 2005.

[7] T. Paek & E. Horvitz, "Conversation as action under uncertainty," Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence, 2000, pp. 455-464.

[8] A. Ratnaparkhi, "Maximum Entropy Models for Natural Language Ambiguity Resolution," Ph.D. Dissertation. University of Pennsylvania, 1998

[9] F. Torres, L.F. Hurtado, F. Garcia, E. Sanchis, and E. Segarra, "Error handling in a stochastic dialog system through confidence measures," Speech Communication, vol. 45, no. 3, pp. 211–229, March 2005.