A Probabilistic Approach to the Interpretation of Spoken Utterances

Ingrid Zukerman, Enes Makalic, Michael Niemann, and Sarah George

Faculty of Information Technology, Monash University
Clayton, VICTORIA 3800, AUSTRALIA
{enes,ingrid,niemann,sarahg}@csse.monash.edu.au

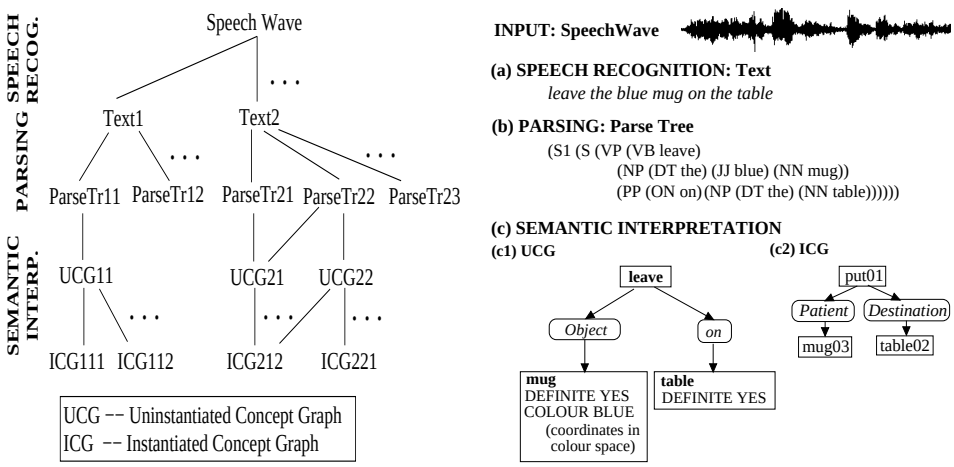
Abstract. In this paper we describe *Scusi?*, the speech interpretation component of a spoken dialogue module designed for an autonomous robotic agent. *Scusi?* postulates and maintains multiple interpretations of the spoken discourse, and employs a probabilistic formalism to assess and rank hypotheses regarding the meaning of spoken utterances. These constituents in combination enable *Scusi?* to cope gracefully with ambiguity and speech recognition errors. The results of our evaluation are encouraging, yielding good interpretation performance for utterances of different types and lengths.

1 Introduction

The *DORIS* project aims to develop a spoken dialogue module for an autonomous robotic agent, which supports the generation of responses that require physical as well as dialogue actions. In this paper, we describe *Scusi?*, *DORIS*'s language interpretation component, focusing on the techniques used to postulate and assess hypotheses regarding the meaning of a spoken utterance.

Minimally, a language interpretation component must be able to postulate promising interpretations, and decide whether there is a clear winner or several likely candidates to be passed to the dialogue system. These capabilities provide the basis for additional desiderata, viz recovering from erroneous interpretations, and adjusting interpretations dynamically as new information becomes available. The dialogue system in turn must determine an appropriate action. For example, consider the request "get me the blue mug". If there is an aqua mug, an indigo mug and a light blue mug in view, the robot could do one of the following: (1) pick the 'bluest' mug among these candidates, (2) select one of these mugs at random, (3) ask a clarification question, or (4) look for a mug that better fits the request. The chosen action depends on the certainty associated with the options returned by the language interpretation module, and the decision procedures applied by the dialogue system.

In order to support the above capabilities, a discourse interpretation system should (1) maintain multiple interpretations, and (2) apply a ranking process to assess the relative merit of each interpretation. *Scusi?* does this, employing a probabilistic mechanism for the ranking component. Its interpretation process comprises three stages: speech recognition, parsing and semantic interpretation. Each stage produces multiple candidate options, which are ranked according to their probability of matching the speaker's



(a) Stages of the interpretation process (b) Sample structures for the interpretation stages

Fig. 1. *Scusi?*'s spoken language interpretation process

intention (Section 3). This probabilistic framework, together with the maintenance of multiple interpretations at each stage of the process, enable *Scusi?* to cope with ambiguity and speech recognition errors (Section 5). In addition, these constituents support the re-ranking of interpretations as new information becomes available, and hence the recovery from erroneous interpretations; and they enable *Scusi?* to abstract features of the interpretations which support the generation of appropriate dialogue or physical actions. Examples of these features are: number of highly ranked interpretations, the difference in their probability, and the similarity between them.

This paper is organized as follows. Section 2 outlines the interpretation process. The estimation of the probability of an interpretation is presented in Section 3, and the semantic interpretation procedure is described in Section 4. Section 5 details our evaluation. Related research and concluding remarks are given in Sections 6 and 7 respectively.

2 Multi-stage Processing

Scusi? processes spoken input in three stages: speech recognition, parsing and semantic interpretation (Figure 1(a)). Our approach generally resembles that described in [2], but there are significant differences. Miller *et al.* considered textual (rather than spoken) input, and used semantic grammars tailored to a slot-filling application. In contrast, our grammars are syntactic, and we incorporate domain-related information only in the final stage of the interpretation process, which yields Conceptual Graphs – a more general structure than frames.

In the first stage of our interpretation process, *Scusi?* runs Automatic Speech Recognition (ASR) software (Microsoft Speech SDK 5.1) to generate candidate texts from a speech signal. Each text is assigned a score that reflects the probability of the words

given the speech wave. The second stage applies Charniak's probabilistic parser (`ftp://ftp.cs.brown.edu/pub/nlparser/`) to generate parse trees from the texts. The parser generates up to N ($= 50$) parse trees for each text, associating each parse tree with a probability. During semantic interpretation, parse trees are successively mapped into two representations based on Conceptual Graphs [3]: first *Uninstantiated Concept Graphs (UCGs)*, and then *Instantiated Concept Graphs (ICGs)*. UCGs are obtained from parse trees deterministically – one parse tree generates one UCG (but a UCG can have more than one parent parse tree).

A UCG represents syntactic information, where the concepts correspond to the words in the parent parse tree, and the relations between the concepts are directly derived from syntactic information in the parse tree and prepositions. Each UCG can generate many ICGs. This is done by nominating different instantiated concepts (relations) from *DORIS*'s knowledge base as potential realizations for each concept (relation) in a UCG (Section 4). Instantiated concepts are objects or actions in the domain, and instantiated relations are similar to semantic role labels [4]. Figure 1(b) illustrates the generation of one ICG for the request "leave the blue mug on the table". The noun "mug" in the parse tree is mapped to the concept `mug` in the UCG, which in turn is mapped to the instantiated concept `mug03` in the ICG. The preposition 'on' in the parse tree is mapped to the relation *on* in the UCG, and then to the relation *Destination* in the ICG. Noun modifiers, such as colour and size, are treated as features to be matched to those of instantiated objects in the knowledge base. For instance, the colour `BLUE` is represented as a set of colour coordinates, which are then matched against the colour coordinates of stored objects [5].

The consideration of all possible options at each stage of the interpretation process is computationally intractable. *Scusi?* uses two computational devices to generate interpretations in real time: (1) an *anytime* algorithm [6], and (2) a processing threshold.

The **anytime algorithm** ensures that the system can return a list of ranked interpretations at any point after completing an expansion. In each stage of the interpretation process, the algorithm applies a selection-expansion cycle to add an element to a search graph (Figure 1(a)) as follows. First, it selects an option for consideration (speech wave, textual ASR output, parse tree or UCG), and expands this option to the next level of interpretation. When an option is expanded, a single candidate is returned for this next level, but additional options reside in a buffer, which is created the first time the option is expanded. For example, when we expand a particular text, the parser returns the next most probable parse tree (but the first time this text is expanded, a buffer with at most N parse trees is created). Similarly, when we expand a UCG, the ICG-generation module returns the next most probable ICG, but as indicated in Section 4, the first time the UCG is expanded, a buffer of at most k_{max} ICGs is created. Buffers are used, rather than piecemeal generation of alternatives, due to two reasons: (1) the ASR and parser return all the options at once; and (2) owing to the complex interactions between the components of ICGs, ICGs are not generated in descending order of probability (i.e., the best ICGs are often generated later on). By maintaining an ICG buffer for each UCG, higher-probability ICGs that are generated later can be slotted into the buffer (and considered by the selection-expansion process) in the order that reflects their probability. The selection-expansion process is repeated until one of the following happens: all op-

tions are fully expanded, a time limit is reached, or the system runs out of memory. At any point after completing an expansion, *Scusi?* can return a list of ranked interpretations with their parent sub-interpretations (text, parse tree(s) and UCG(s)).

The **thresholding** approach is based on the observation that the probabilities of the texts returned by the ASR drop quite dramatically after the first few texts, as do the probabilities of the parse trees. We take advantage of this observation to prevent the consideration of unpromising alternatives as follows. When the probability of the next child of a parent node n drops below a threshold *Thr* relative to the probability of the most probable child of n, no additional children of n are considered. For example, for *Thr* $= 50\%$, if the probability of the next parse tree for text T_i is less than half of the probability of the first (best) parse tree generated for T_i, no more parse trees are considered for T_i.

3 Probability of an Interpretation

Scusi? ranks candidate ICGs according to their probability of being the intended meaning of a spoken utterance. The principles of this calculation were set out in [6]. Here we refine this process, focusing on the calculation of the probability of ICGs.

Given a speech signal W and a context $\mathcal{C}$, the probability of an ICG I is represented as follows.

$$\Pr(I|W,\mathcal{C}) \propto \sum_{\Lambda} \Pr(I|U,\mathcal{C}) \cdot \Pr(U|P) \cdot \Pr(P|T) \cdot \Pr(T|W) \tag{1}$$

where the UCG, the parse tree and the textual interpretations are denoted by U, P and T respectively. The summation is taken over all possible paths $\Lambda = \{P, U\}$ from the parse tree to the ICG, because a UCG and an ICG can have more than one parent. The ASR and the parser return an estimate of $\Pr(T|W)$ and $\Pr(P|T)$ respectively. In addition, $\Pr(U|P) = 1$, since the process of generating a UCG from a parse tree is deterministic. Hence, we still have to estimate $\Pr(I|U,\mathcal{C})$.

Consider an ICG I containing concepts $c^{ICG} \in \Omega_c$ and relations $r^{ICG} \in \Omega_r$ (Ω_c and Ω_r are the concepts and relations in the domain knowledge respectively). The parent UCG (denoted by U) comprises concepts $c^{UCG} \in \Gamma_c$ and relations $r^{UCG} \in \Gamma_r$ (Γ_c and Γ_r are the concepts and relations from which UCGs are built). The probability of I given U and context $\mathcal{C}$ can be stated as follows.

$$\begin{aligned}\Pr(I|U,\mathcal{C}) &= \prod_{\substack{c^{ICG} \in \Omega_c \\ r^{ICG} \in \Omega_r}} \Pr\left(c^{ICG}, r^{ICG} | c^{UCG}, r^{UCG}, \Omega_c^-, \Omega_r^-, \mathcal{C}\right) \\ &= \prod_{\substack{c^{ICG} \in \Omega_c \\ r^{ICG} \in \Omega_r}} \left\{ \Pr\left(r^{ICG} | c^{ICG}, c^{UCG}, r^{UCG}, \Omega_c^-, \Omega_r^-, \mathcal{C}\right) \times \Pr\left(c^{ICG} | c^{UCG}, r^{UCG}, \Omega_c^-, \Omega_r^-, \mathcal{C}\right) \right\}\end{aligned} \tag{2}$$

where c^{UCG} and r^{UCG} denote the UCG concept and relation corresponding to the ICG concept c^{ICG} and relation r^{ICG} respectively; and Ω_c^- and Ω_r^- denote the sets Ω_c and Ω_r without the concept c^{ICG} and relation r^{ICG} respectively.

It is difficult to estimate Equation 2, as each concept and relation in an ICG depends on the other ICG concepts and relations. We therefore make the following simplifying assumptions.

- The probability of an ICG relation r^{ICG} depends only on the corresponding UCG relation, the parent ICG concept of r^{ICG}, and the context.
- The probability of an ICG concept c^{ICG} depends only on the corresponding UCG concept, the parent ICG relation and grandparent concept of c^{ICG}, and the context (e.g., the parent relation of `mug03` in Figure 1 is *Patient*, and its grandparent concept is `put01`).

These assumptions are justified by the information in the knowledge base, which stores the location and ownership of many objects, and by the available linguistic information regarding concepts and relations (e.g., the action `fetch01` has a mandatory *Patient* relation, but an optional *Beneficiary*, and any mug is a suitable *Patient* for most actions). Now, say we have the request "get the mug from the table", and one of the candidate ICGs has the fragment [`mug03`→*Location*→`table01`] (*Location* is the parent of `table01`, and `mug03` is its grandparent). If `mug03` is indeed on `table01`, the probability of this ICG increases, otherwise it decreases. In the absence of this information, we back off to bigram probabilities (e.g., whether `table01` a possible *Location*). These assumptions yield

$$\Pr(I|U,\mathcal{C}) \approx \prod_{\substack{c^{ICG} \in \Omega_c \\ r^{ICG} \in \Omega_r}} \left\{ \Pr(r^{ICG}|r^{UCG}, c_p^{ICG}, \mathcal{C}) \times \Pr(c^{ICG}|c^{UCG}, r_p^{ICG}, c_{gp}^{ICG}, \mathcal{C}) \right\} \tag{3}$$

where the parent concept of relation $r^{ICG} \in \Omega_r$ is $c_p^{ICG} \in \Omega_c$, and the grandparent concept and parent relation of concept $c^{ICG} \in \Omega_c$ are $c_{gp}^{ICG} \in \Omega_c$ and $r_p^{ICG} \in \Omega_r$ respectively.

After applying Bayes rule, and making additional simplifying assumptions about conditional dependencies, we obtain

$$\Pr(I|U,\mathcal{C}) \approx \prod_{\substack{c^{ICG} \in \Omega_c \\ r^{ICG} \in \Omega_r}} \left\{ \begin{array}{l} \underbrace{\Pr(r^{UCG}|r^{ICG})\ \Pr(c^{UCG}|c^{ICG})}_{\text{segment 1}}\ \underbrace{\Pr(r^{ICG}|c_p^{ICG})\ \Pr(c^{ICG}|r_p^{ICG}, c_{gp}^{ICG})}_{\text{segment 2}}\ \underbrace{\Pr(c_p^{ICG}|\mathcal{C}) \times}_{\text{segment 3}} \end{array} \right\} \tag{4}$$

The first segment in Equation 4 represents the probability that a user who intended r^{ICG} and c^{ICG} said r^{UCG} and c^{UCG} respectively; the second segment represents the probabilities of relations and concepts in the ICG in light of their parent and grandparent elements; and the third segment represents the prior probabilities of the concepts in the ICG (judicious conditionalization obviates the need to calculate the prior probabilities of relations in the ICG). Ideally, all these probabilities should be estimated from data, but this would require the development of a large database of UCGs and ICGs corresponding to different speech signals. Such a database is currently not available. Hence, *Scusi?* employs a heuristic approach to estimate the necessary probabilities, as follows.

- The probabilities in the first segment of Equation 4 are estimated on the basis of the goodness of the match between candidate instantiated concepts (relations) in the

ICG and concepts (relations) mentioned in the UCG. For relations, this probability depends on the lexical match between a stated relation and an instantiated relation. For concepts we also take into account the left modifiers of the head noun — at present we consider colour and size. For example, if the user said "blue mug", then a light blue cup will yield a lower probability than a royal blue mug [5].

– The probabilities in the second segment are estimated based on how well children nodes match the expectations of their parent (and grandparent) nodes in the ICG. For example, the probability of the ICG bigram [`go02` → *Destination*] depends on whether *Destination* is a compulsory complement of `go02` (high probability) or optional (lower probability); the probability of the trigram [`cup05` → *Owned-by* → `Susan01`] is 1 if `Susan01` owns `cup05`, and 0.5 if ownership is unknown. At present, grandparent concepts are considered only for location and ownership of objects, which may be determined from the system's knowledge base. In other cases or if the information is unknown, we back-off to the parent relation of a concept, e.g., the probability of `kitchen` being a *Location*.
– The prior probability of an instantiated concept depends on the context, which at present includes only domain knowledge, i.e., all instantiated concepts have the same prior. In the future, we propose to estimate these prior probabilities by combining salience scores obtained from dialogue history [7] with visual salience [8].

4 Generating ICGs

The process of generating ICGs from a UCG and estimating their probability is carried out by Algorithm 1, which refines the procedure presented in [6]. This algorithm generates a buffer containing up to k_{max} ($= 400$) ICGs ranked in descending order of probability the first time a UCG is expanded (the size of the buffer was empirically determined). Every time a new ICG is requested for that UCG, the next ranked ICG is returned. The algorithm has two main stages: *concept and relation postulation* (Steps 2–10), and *ICG construction* (Steps 11–16).

4.1 Postulating concepts and relations

In this stage, the algorithm proposes instantiated concepts (relations) from the knowledge base for each UCG concept (relation), and sorts each candidate list of instantiated concepts (relations) in descending order of probability.

In **Step 5**, for each concept c^{UCG} in the UCG, the algorithm estimates the probability that each instantiated concept in the knowledge base matches c^{UCG}. The same is done for relations. The probability of this match, which corresponds to the first segment in Equation 4, is estimated by means of comparison functions [5]. The probability of a match between an instantiated relation and a UCG relation depends only on the goodness of the lexical match between these relations. In contrast, the probability of a concept match also depends on the match between intrinsic features mentioned in the UCG, such as colour and size, and the actual values of these features for a candidate instantiated concept. For instance, given the UCG concept *cup*, the instantiated concepts `mug01, ..., mug05` and `cup01, ..., cup04` have a good lexical match with the UCG concept. If

Algorithm 1 Generate candidate ICGs for a UCG

Require: UCG U comprising concepts c^{UCG} and relations r^{UCG}, context $\mathcal{C}$

1: Initialize buffer $\mathcal{I}_U$ of size $k_{\max}$ (=400)

{ **Postulate concepts and relations for UCG** }

2: **for all** concepts c^{UCG} (relations r^{UCG}) in U **do**

3: Initialize a list of candidate concepts $L_{c^u} \leftarrow \emptyset$ (list of relations $L_{r^u} \leftarrow \emptyset$)

4: **for all** instantiated concepts $\mathbf{c}^I$ (instantiated relations $\mathbf{r}^I$) **do**

5: Compare c^{UCG} with $\mathbf{c}^I$ (r^{UCG} with $\mathbf{r}^I$), yielding a probability for the match (segment 1 in Equation 4)

6: Calculate the prior probability of $\mathbf{c}^I$ according to the context $\mathcal{C}$ (segment 3 in Equation 4)

7: Multiply the probabilities obtained in Steps 5 and 6

8: Insert $\mathbf{c}^I$ in the list L_{c^u} ($\mathbf{r}^I$ into L_{r^u}) in descending order of probability

9: **end for**

10: **end for**

{ **Construct ICGs** }

11: **for** $j = 1$ to $k_{\max}$ **do**

12: Generate the "next best" ICG I_j by going down each list L_{c_u} and L_{r_u} in turn

13: Perform internal consistency checks to calculate the probabilities of the concepts and relations in ICG I_j (segment 2 in Equation 4)

14: Estimate $\Pr(I_j|U, \mathcal{C})$ by multiplying the probabilities obtained in Step 7 with the probabilities obtained in Step 13

15: Insert I_j into buffer $\mathcal{I}_U$ in descending order of probability

16: **end for**

the UCG concept had been *blue cup*, then the colour coordinates of the mugs and cups in the knowledge base would be matched against the coordinates for the term 'blue'.

Upon completion of Step 5, we prune ICG candidates that do not have a good match with the concept (relation) in the UCG. For example, the UCG concept *chair* could refer to an armchair, a stool, a pouf, etc. Hence, all the armchairs, stools, poufs, etc in the knowledge base are retained for further processing, while lamps, tables, cups, etc are discarded. Similarly, red cups are discarded if a blue mug is requested, and there are blue mugs in the knowledge base.

The prior probability of the retained candidate concepts (third segment in Equation 4) is estimated in **Step 6**. In **Step 7**, this probability is multiplied by the probability calculated in Step 5.

This stage of the algorithm yields a list of candidate instantiated concepts L_{c^u} for each UCG concept c^{UCG}, and a list of candidate instantiated relations L_{r^u} for each UCG relation r^{UCG}. These lists are sorted in descending order of probability. For example, Table 1 shows the sorted lists of concepts and relations postulated for the request in Figure 1 "leave the blue mug on the table": there are four objects that are a good match for the concept *blue mug*, three candidate tables, three candidate actions for *leave* (leave the room (`leave01`), put in a specific place (`put01`), and put down (`put02`)), two relations for *on*, and one for *object*.

Table 1. Concepts and relations used to build ICGs for the utterance "leave the blue mug on the table"

leave	*blue mug*	*table*	*on*	*object*
`leave01`	`mug02`	`table01`	*Destination*	*Patient*
`put01`	`cup01`	`table02`	*Location*	
`put02`	`mug03`	`table03`		
	`cup02`			

4.2 Constructing ICGs

In this stage, the algorithm uses the list of instantiated concepts (relations) built for each concept (relation) in a UCG to construct candidate ICGs for this UCG, and sorts these ICGs in descending order of probability. First, **Step 12** applies an enumerative process to generate different combinations of concepts and relations from the list L_{c_u} (L_{r_u}) maintained for each UCG concept (relation). This is done by iteratively selecting one candidate concept (relation) from each list. For instance, the concepts and relations in Table 1 are combined as follows to build candidate ICGs. First, the top line $\{$`leave01,mug02,`$\ldots\}$, which has the highest probability, is used. The next four combinations are generated by replacing one element from this line at a time, i.e., `leave01` is replaced with `put01`, yielding $\{$`put01,mug02,`$\ldots\}$; then `mug02` is replaced with `cup01`, yielding $\{$`leave01,cup01,`$\ldots\}$; and so on.

The probabilities of the concepts and relations in each ICG (second segment in Equation 4) are then estimated in **Step 13**. These probabilities reflect the extent to which the relationships between neighbouring nodes in an ICG match the known reality. As mentioned in Section 3, for relations this calculation is done based on the type of relations admitted by each concept (e.g., compulsory, optional or absent), and for concepts the calculation reflects the current state of the world. For example, if we request "the mug on the table" and according to the knowledge base, `mug03` is on `table02`, the probability of an ICG that contains [`mug03`$\rightarrow$*Location*$\rightarrow$`table02`] is increased, whereas if `cup01` is not on a table, the probability of an ICG containing `cup01` is decreased. In **Step 14**, this 'structural' probability is combined with the probability calculated in Step 7 (candidate postulation stage) to obtain the final probability of an ICG produced for a given UCG. This ICG is inserted in the buffer for that UCG in descending order of the ICG's probability.

5 Evaluation

Our evaluation test set comprised 100 utterances: 43 declarative (e.g., "the book is on the desk", "in the kitchen", "the red mug") and 57 imperative (e.g., "open the door").[1] These utterances were based on interactions between users and a "robot" (enacted by one of the authors) in a virtual home scenario; they were recorded by one of the authors,

[1] We acknowledge the modest size of this test set compared to that of some publicly available corpora, e.g., ATIS and GeoQuery. However, we must generate our own test set since our task differs significantly from the slot-filling tasks where these large corpora are used. This is due to the domain itself and the open-ended nature of the utterances.

Table 2. *Scusi?*'s interpretation performance

	# Gold ICGs with prob in top 1	# Gold ICGs with prob in top 3	Average adj rank (rank)	Not found	Avg # of ICGs to Gold ICG (avg # of iters)
BASELINE	53	53	0 (0)	47	0 (4)
No Thrsh	69	82	3.85 (1.15)	7	9 (38)
10%	67	81	2.63 (0.91)	8	8 (37)
20%	70	83	2.47 (0.87)	7	8 (39)
50%	70	84	2.37 (0.81)	7	8 (37)
80%	70	85	2.31 (0.80)	7	8 (37)
90%	70	85	2.31 (0.78)	7	8 (37)
Total	100	100			

as the ASR software is speaker dependent, and at present we do not handle features of spontaneous speech. The utterances (which were *not* used during system development) were chosen to test *Scusi?*'s ability to identify target objects (the intended book, mug, table, etc), and its ability to handle phenomena such as synonyms (e.g., "wash" and "clean") and homonyms (e.g., "*leave* the mug on the table" versus "*leave* the room"). The average utterance length was 8.5 words, with a maximum length of 12 words.

Scusi? was set to generate at most 300 sub-interpretations in total (including texts, parse trees, UCGs and ICGs) for each utterance in the test set. An interpretation was deemed successful if it correctly represented the speaker's intention within the limitations of *Scusi?*'s knowledge base, which comprises 135 items (24 relations and 111 concepts). This intention was represented by one or more *Gold ICG*s that were manually constructed by one of the authors. Multiple Gold ICGs were allowed if there were several objects in the knowledge base that matched a requested object, e.g., "get a mug".

Ideally, we would like to evaluate separately the impact of our probabilistic framework and that of maintaining multiple interpretations. However, the design of an alternative, baseline hypothesis-ranking framework is outside the scope of this project. We therefore designed our experiments to measure (1) *Scusi?*'s overall interpretation performance, (2) the impact of maintaining multiple interpretations on performance, and (3) the impact of different thresholds (Section 2). Thus, tests were conducted under the following settings.

- BASELINE – a beam search was executed, where only the best ASR result was parsed, and only the best parse tree yielded a UCG. We then selected the top ICG among those in the buffer for the UCG (Section 4). Note that the selection of the top-ranked item for each of these stages is still done on the basis of its probability. Hence, our baseline enables us to isolate only the impact of maintaining multiple interpretations.
- Threshold – No Threshold, and thresholds of 10%, 20%, 50%, 80% and 90%, e.g., a 10% threshold discards texts T such that $\Pr(T) < 0.1 \times \Pr(\text{highest-probability text})$.

Table 2 summarizes our results, which were obtained with an ASR that had a 20% error rate (the correct text was top ranked by the ASR in 80% of the cases). Column 1 displays the test condition (baseline and threshold value). Columns 2-3 show how many utterances had Gold ICGs whose probability was among the top 1 or top 3, e.g., the 20%

threshold yielded 70 Gold ICGs with the highest probability (top 1), and 83 within the top 3 probabilities. The average *adjusted rank* and *rank* of the Gold ICG appear in Column 4. The rank of an ICG I is its position in a list sorted in descending order of probability (starting from position 0), such that all equiprobable ICGs are deemed to have the same position (recall that the baseline returns a single ICG, whose rank is therefore 0). The adjusted rank of an ICG I is the mean of the positions of all ICGs that have the same probability as I. For example, if we have 3 top-ranked equiprobable ICGs, each has a rank of 0, but an adjusted rank of $\frac{0+2}{3}$. Column 5 shows how many utterances didn't yield a Gold ICG, and Column 6 indicates the average number of ICGs created and iterations done until the Gold ICG was found (from a total of 300 iterations).

As seen in Table 2, the baseline yielded significantly fewer top-ranked Gold ICGs than our anytime algorithm ($p < 0.05$).[2] The 20% ASR error was substantially exacerbated by the baseline approach, which failed to return Gold ICG(s) in 27 of the 80 cases where it was presented with the correct text. In contrast, *Scusi?* performed significantly better, failing to produce top-ranked Gold ICG(s) in only 10 of those cases. Further, for the top-3 ranking, *Scusi?* overcame the ASR error (i.e., its error rate is less than 20%).[3] These results confirm the need to maintain multiple interpretations in combination with a probabilistic hypothesis assessment.

Interestingly, the threshold did not affect the number of top-ranked Gold ICGs and not found ICGs, and the number of iterations to Gold. However, the average rank of the Gold ICGs decreases (improves) as the threshold increases, which is consistent with the slight improvement in the number of top-3 ICGs. We also performed additional experiments that examined the effect of using a different threshold for each level of interpretation. However, the new scheme did not yield any improvement over a single system-wide threshold.

6 Related Research

This research builds on the work described in [6]. Here we refine the process for considering multiple alternatives during the generation of interpretations, and its combination with the probabilistic hypothesis assessment formalism. In particular, we focus on the calculation of the probabilities of ICGs.

Many researchers have investigated numerical approaches to the interpretation of spoken utterances in dialogue systems, e.g., [9–12]. Pfleger *et al.* [9] and Hüwel and Wrede [10] employ modality fusion to combine hypotheses from different analyzers (linguistic, visual and gesture), and apply a scoring mechanism to rank the resultant hypotheses. In contrast, He and Young [11] and Gorniak and Roy [12] apply a probabilistic approach to spoken language interpretation, using Hidden Markov Models for the ASR stage. Additionally, as mentioned above, *Scusi?*'s probabilistic formalism resembles in style that employed by Miller *et al.* for discourse interpretation in a text-

[2] Sample paired t-tests were used for all statistical tests.

[3] Clearly, it is not fair to compare *Scusi?*'s top-3 rank with the baseline's top-1 rank. However, the top-3 rank supports the generation of clarification questions for ICGs with similar probabilities — an option that is not available if only the top-ranked interpretation is returned.

based system [2]. However, all these systems employ semantic grammars, while *Scusi?* employs a three-stage interpretation process, which uses generic, syntactic tools, and incorporates semantic- and domain-related information only in the final stage of the interpretation process. Knight *et al.* [13] compare the performance of a dialogue system based on a semantic grammar to that of a system based on a statistical language model and a robust phrase-spotting grammar. The latter performs better for relatively unconstrained utterances by users unfamiliar with the system. The probabilistic approach and intended users of our system are in line with this finding.

From the view point of application domain, robot-mounted dialogue systems were also studied in [14, 15, 10]. Matsui *et al.* [14], like Gorniak and Roy [12], use contextual information to constrain the alternatives returned by the ASR early in the interpretation process. This allows their system to process expected utterances efficiently, but makes it difficult to interpret unexpected utterances. In contrast, *Scusi?* incorporates contextual information in the final stage of the interpretation process. Unlike *Scusi?*'s probabilistic reasoning formalism, Bos *et al.* [15] use a logic-based language interpretation framework to understand instructions and descriptions, and employ formal proofs for conflict resolution. Consequently, alternatives are not considered when an utterance is ambiguous or the preferred option proves undesirable. This is also the case for Hüwel and Wrede's [10] system, which considers only a single alternative as a result of each stage of the interpretation process.

7 Conclusion and Future Work

We have described *Scusi?*, a spoken language interpretation system that maintains multiple options at each stage of the interpretation process, and ranks interpretations based on estimates of their posterior probability. In particular, we presented the algorithm used by *Scusi?* to postulate hypotheses regarding the meaning of a spoken utterance, and detailed the estimation of the probabilities of these hypotheses.

Our empirical evaluation shows that *Scusi?* performs well for declarative and imperative utterances of varying length, with the Gold ICG(s) receiving one of the top three probabilities for most test utterances. Our results also show that using a threshold has a small impact on *Scusi?*'s performance (by slightly improving the number of ICGs ranked top-3, and hence the average rank of the Gold ICGs). In the near future, we will further investigate the impact of thresholds on performance speed and accuracy. An additional avenue of investigation pertains to the consideration of different weightings for combining the scores obtained from the three interpretation stages.

References

1. Shankaranarayanan, S., Cyre, W.: Identification of coreferences with conceptual graphs. In: ICCS'94 – Proceedings of the Second International Conference on Conceptual Structures, College Park, Maryland (1994)
2. Miller, S., Stallard, D., Bobrow, R., Schwartz, R.: A fully statistical approach to natural language interfaces. In: ACL96 – Proceedings of the 34th Conference of the Association for Computational Linguistics, Santa Cruz, California (1996) 55–61

3. Sowa, J.: Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, Reading, MA (1984)
4. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics **28**(3) (2002) 245–288
5. Zukerman, I., Makalic, E., Niemann, M., Schmidt, D.: A probabilistic model for understanding composite spoken descriptions. In: Submitted. (2008)
6. Niemann, M., Zukerman, I., Makalic, E., George, S.: Hypothesis generation and maintenance in the interpretation of spoken utterances. In: AI'07 – Proceedings of the 20th Australian Joint Conference on Artificial Intelligence, Gold Coast, Australia (2007)
7. Zukerman, I., George, S.: A probabilistic approach for argument interpretation. User Modeling and User-Adapted Interaction, Special Issue on Language-Based Interaction **15**(1-2) (2005) 5–53
8. Wyatt, J.: Planning clarification questions to resolve ambiguous references to objects. In: Proceedings of the 4th IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems, Edinburgh, Scotland (2005) 16–23
9. Pfleger, N., Engel, R., Alexandersson, J.: Robust multimodal discourse processing. In: Proceedings of the Seventh Workshop on the Semantics and Pragmatics of Dialogue, Saarbrücken, Germany (2003) 107–114
10. Hüwel, S., Wrede, B.: Spontaneous speech understanding for robust multi-modal human-robot communication. In: Proceedings of the COLING/ACL Main conference poster sessions, Sydney, Australia (2006) 391–398
11. He, Y., Young, S.: A data-driven spoken language understanding system. In: ASRU'03 – Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding, St. Thomas, US Virgin Islands (2003) 583–588
12. Gorniak, P., Roy, D.: Probabilistic grounding of situated speech using plan recognition and reference resolution. In: ICMI'05 – Proceedings of the Seventh International Conference on Multimodal Interfaces, Trento, Italy (2005) 138–143
13. Knight, S., Gorrell, G., Rayner, M., Milward, D., Koeling, R., Lewin, I.: Comparing grammar-based and robust approaches to speech understanding: A case study. In: EUROSPEECH 2001 – Proceedings of the Seventh European Conference on Speech Communication and Technology, Aalborg, Denmark (2001) 1779–1782
14. Matsui, T., Asoh, H., Fry, J., Motomura, Y., Asano, F., Kurita, T., Hara, I., Otsu, N.: Integrated natural spoken dialogue system of Jijo-2 mobile robot for office services. In: AAAI99 – Proceedings of the Sixteenth National Conference on Artificial Intelligence, Orlando, Florida (1999) 621–627
15. Bos, J., Klein, E., Oka, T.: Meaningful conversation with a mobile robot. In: EACL10 – Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics, Budapest, Hungary (2003) 71–74