

Taking Advantage of the Situation: Non-Linguistic Context for Natural Language Interfaces to Interactive Virtual Environments

Michael Fleischman
Cognitive Machines Group
The Media Laboratory
Massachusetts Institute of Technology
mbf@mit.edu

Eduard Hovy
Natural Language Group
Information Sciences Institute
University of Southern California
hovy@isi.edu

ABSTRACT

We introduce a framework for learning situated Natural Language Interfaces (NLIs) to interactive virtual environments. The framework exploits the non-linguistic context, or situation, explicitly modeled in such interactive applications. This *situation model* is integrated with a model of word meaning in a principled manner using a noisy channel approach to language understanding. Preliminary experimentation in an independently designed interactive application, i.e. the Mission Rehearsal Exercise (MRE), shows that this situated NLI outperforms a state of the art NLI on both whole frame accuracy and F-Score metrics. Further, use of the situation model in the situated NLI is shown to increase robustness to the noise introduced by the use of automatic speech recognition.

Categories and Subject Descriptors

H.5.2 [User Interfaces]: Natural Language

General Terms

Algorithms, Performance, Design.

Keywords

Natural Language Interfaces / Understanding, Situation Models, Non-Linguistic Context, Interactive Virtual Environments, Mission Rehearsal Exercise, Plan Recognition, Situated NLI.

1. INTRODUCTION

Exploiting non-linguistic information in Natural Language Interfaces (NLIs) represents one of the most promising frontiers for language research. Interactive virtual environments, where human users and computer agents interact in shared virtual environments, provide an ideal test bed for such efforts. Not only do interactive applications pose specific challenges to traditional methodologies for learning NLIs, but also, they afford unique access to detailed models of the non-linguistic context, or situation, surrounding linguistic interactions. In the work presented here we outline a framework for learning *situated* NLIs that leverages the non-linguistic situation modeled in interactive

virtual environments. Further, using an independently designed interactive application, we show that our situated NLI is both more efficient and more accurate than a state of the art NLI.

Previous research on learning NLIs comes largely from work done in the database domain. Such systems treat the translation of natural language utterances into SQL queries as a problem of semantic parsing. Using training sets of utterances annotated with semantic representations, Machine Learning techniques have been used to learn parsers that infer semantic representations from well-formed utterances ([15], [6], [17]). Related work on text understanding has leveraged large corpora of annotated text to learn semantic taggers that identify and classify the semantic roles of phrases in a parsed sentence ([7], [3]). While semantic parsing methods have been successful in these domains, they do little to address the challenges posed by interactive virtual environments.

Unlike in the text and database domains, NLIs to interactive virtual environments must interpret language that is highly conversational and often ungrammatical. Such input often cannot be parsed using techniques that expect well-formed sentences. These issues are further exacerbated by the use of Automatic Speech Recognition (ASR) which, due to the propagation of error, further degrades the quality of input to the NLI. While these issues would be problematic for NLIs in any domain, interactive virtual environments offer unique access to situational information that can be exploited to counteract such challenges.

A number of recent efforts in both Cognitive Science and Natural Language Processing have focused on modeling the non-linguistic context, or situation, surrounding linguistic interactions ([11], [12]). While much of this research has focused on the perceptual context of language, recent efforts have focused on learning ([4], [5]) and hand-crafting [8] lexicons that exploit more abstract contextual information. Interactive virtual environments are in a unique position to benefit from such situated approaches to language processing because they model closed-domain worlds in which all aspects of the situation must be explicit and accessible. These environments provide detailed models that dictate the course of all interactions in the virtual world by defining the plans and actions that agents (both human and virtual) can perform. These models represent a rich and untapped resource for NLIs that seek to use situational information to aid in language understanding.

We propose a framework for learning situated NLIs which leverages non-linguistic information modeled in interactive virtual environments. We exploit our access to an agent's range of possible plans and actions in order to build a *situation model* that captures the context of an interaction. This situation model uses plan recognition to predict what will be said in order to prime the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IUI'06, January 29–February 1, 2006, Sydney, Australia.
Copyright 2006 ACM 1-59593-287-9/06/0001...\$5.00.

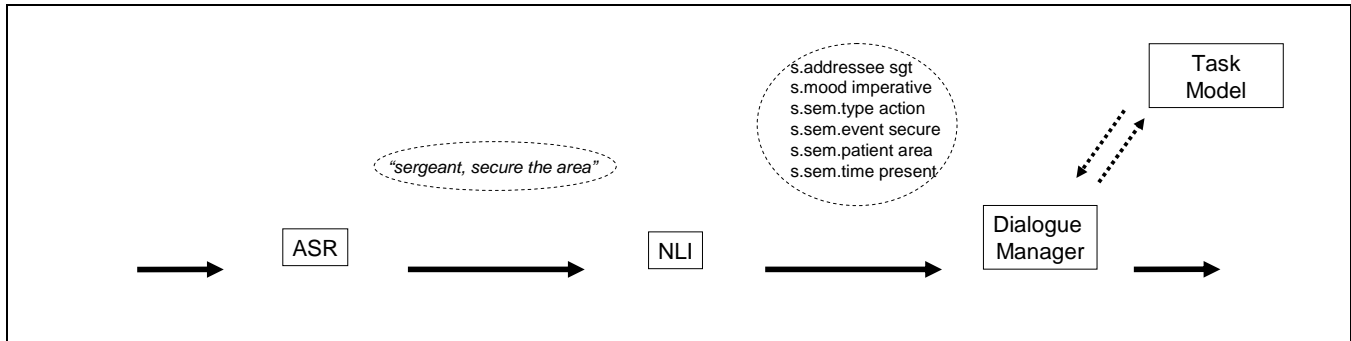


Figure 2. The Natural Language Pipeline in the MRE begins with an automatic speech recognition (ASR) which outputs text that is translated into a semantic frame by the Natural Language Interface (NLI) that is processed by a Dialogue Manager who interacts with the rest of the MRE system (including the task model) to initiate a response.

situated NLI to prefer meanings that are appropriate given the current situation.

The remainder of this paper details our framework as it is applied in an independently created, state of the art interactive virtual environment, the Mission Rehearsal Exercise (MRE) [14]. We first describe the MRE, highlighting its current NLI implementation and the details of the agent model that our situated NLI exploits. We then describe the framework for our situated NLI which uses a noisy channel approach to integrate a model of the non-linguistic situation built from information accessible in the MRE. Finally, we show preliminary evaluations comparing the situated NLI with the MRE’s state of the art NLI using both human and automatically transcribed speech from actual interactions in the MRE.

2. Mission Rehearsal Exercise (MRE)

We present our framework for learning situated NLIs within the context of the independently created interactive application, the Mission Rehearsal Exercise (MRE) [14]. The MRE is an ongoing large-scale collaborative research effort to develop a fully interactive training simulation modeled after the holodeck in Star Trek. The project brings together researchers working on graphics, 3-D audio, artificial intelligence, and Hollywood screenwriters to create a realistic virtual world in which human subjects can interact naturally with simulated agents. The agents communicate through voice and gesture, reason about plans and actions, and incorporate a complex model of emotions. Users can query and interact with one (and eventually many) agent in real-time as they proceed through scenarios developed for the particular training mission at hand.

Figure 1 shows a screen shot of the “peace-keeping mission” scenario employed in this work. The scenario is designed to train army lieutenants for eastern European peace keeping missions. It centers around the trainee, a human lieutenant, who is attempting to move his platoon to support an inspection, when one of his drivers unexpectedly collides with a civilian car. A civilian passenger, a young boy, is critically injured and the trainee must interact with his or her virtual platoon sergeant in order to arrange a medical evacuation (i.e. medevac) and stabilize the situation.

2.1. The NLI of the MRE

The current NLI of the MRE operates in a pipeline architecture (see Figure 2). The user’s speech is converted to text by a speech recognizer and sent as input to the NLI. This module converts the text to a semantic case frame representation (similar to the case



Figure 1. The Mission Rehearsal Exercise (MRE), shown here, is a state of the art training simulator where human users interact with virtual agents in a shared virtual environment (VE).

frames employed in FrameNet [2]) using a shallow semantic parser that operates in two phases (see figure 3) [1]. In the first phase, each unigram, bigram, and trigram of the input text is used to generate a semantic role based on frequency statistics collected from a training corpus of sentences hand annotated with semantic frames (see section 3 for more details). In the second phase, the large list of candidate semantic roles output from the first phase is automatically ranked using a Maximum Entropy model, and a threshold (set by cross-validation) is applied to determine which roles will be output as the final semantic frame representation of the sentence.

This shallow parsing procedure avoids some of the challenges to NLIs posed by interactive virtual environments. By using only the surface statistics described and not requiring a full parse of the input it more easily handles the complexities of conversational speech and attempts to minimize the effects of automatic speech recognition. However, as shown in section 5, by not exploiting situational information, it remains brittle to the complexities of interactive environments.

Having completed its shallow parse of the input, the NLI outputs its semantic frame to the dialogue manager who formulates the systems’ responses by interacting with various modules, including a model of the agents’ tasks. This task model is a critical resource for non-linguistic context and is detailed below.

2.2. The Task Model of the MRE

In the MRE, all actions and plans that an agent may take are represented as tasks in a task model [13]. These tasks are

represented using relatively standard plan representations, in which each task is defined by a sequence of (partially ordered) steps, or actions, that may either be primitive (i.e. a physical or sensing action in the virtual world) or abstract (i.e., a task that must be further decomposed into primitive actions). Each action can be related to each other by causal links and/or threat relations that define the pre and post conditions used in planning. Further, because abstract actions are decomposable, the task model maintains a hierarchical structure that can be seen in the graphical representation of the task [medevac], shown in Figure 4. Here the large dashed boxes represent abstract actions and the smaller solid boxes represent the primitive actions of which they are composed. (The pre and post conditions are represented by ovals, the relations between states and actions as lines). Further, as seen in the inset, each action has an internal case frame structure, similar to the intermediary semantic structure used by the current NLI.

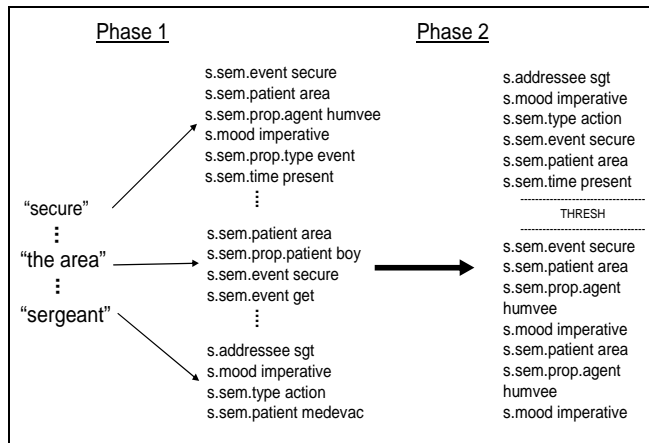


Figure 3 The shallow semantic parser used in the current NLI for the MRE operates in two phase.

The primitive and abstract actions represented in the MRE’s task model delimit what the virtual agents in the MRE can do, and thus, encode the mission (or pedagogical purpose) of the training scenario itself. Importantly, the task model also delimits what human users can do in the MRE; for anything not explicitly encoded in the task model, by definition, cannot be interpreted by the system. This makes the task model an incredibly rich resource for a NLI as it not only explicitly describes what can and cannot be interpreted by the system, but also, explicitly describes all possible ways in which agents and users can behave. In the next section we describe a situated NLI that exploits these aspects of the task model.

3. Situated Natural Language Interface

Learning a situated NLI for the MRE demands a framework for exploiting the rich situational information represented in the task model. One possible approach to achieving this is to extend the current NLI in a manner that takes advantage of the information available in the task model. However, it is unclear how having access to such information would influence the current NLI’s parsing of an utterance. The difficulty arises because the actions in the task model and the meanings in the current NLI are not represented in the same manner. In order to have one influence the other, however, a common framework is required that links the representations of meaning and action.

The remainder of this section describes this framework as well as our general framework for a situated NLI. First, we detail the

common representation that links how meaning and action are formalized in the MRE. Then, we describe how this common representation facilitates a principled method for integrating non-linguistic context in language understanding, i.e. the noisy channel framework. Finally, we show how this non-linguistic context is encoded in a situation model that exploits information represented in the task model.

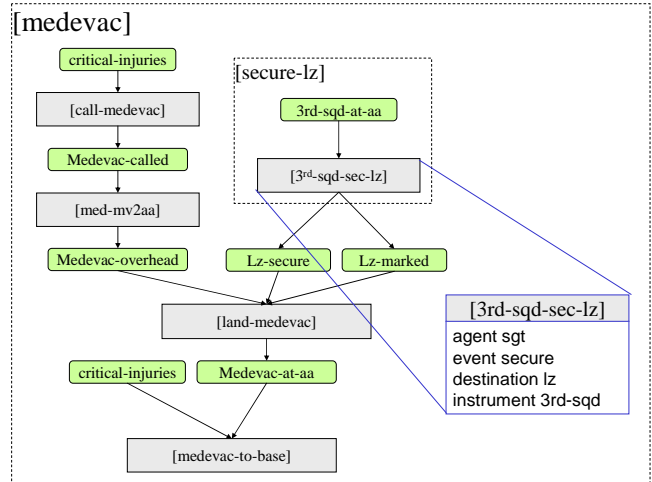


Figure 4. The task model maintains representations of every task that can be performed by agents (human or virtual) in the MRE. In this graphical representation of the task [medevac], which describes a medical evacuation procedure), ovals represent states, lines represent steps in the plan, solid boxes represent primitive actions, and dashed boxes represent abstract actions. Importantly, the actions can be hierarchical (if abstract) and have internal structure (see inset).

3.1. Representing Meaning and Action

As described above, in order to exploit the non-linguistic information encoded by the actions in the task model, a link must be made between how those actions, and the meanings of utterances, are represented in the MRE. Although a number of ways to link these representations can be imagined (e.g. hand written rules that map each semantic frame to a frame in the task model), we choose to adopt a simpler approach. Instead of attempting to generate such linkings, we simply drop the use of the semantic frames entirely, in favor of representing the meaning of an utterance directly with the frame representation used in the task model.

Recall that these task frame representations (see inset Figure 4) are similar to the semantic frames used by the current NLI, but encode less information and are more impoverished than those intermediary representations. Our choice to rely only on these impoverished frames to represent the meaning of even complex utterances rests on the fact that, in interactive applications such as the MRE, there is no need for representations of meaning that are more expressive than what is in the task model. This is the case because, as described above, any information that is not explicitly encoded in the task model cannot be interpreted by the system anyway.

Thus, whereas in the current NLI each training utterances is hand annotated with a semantic frame, in our situated NLI, each training utterance is linked to a task frame encoded in the task model. Because some frames are missing from the task model that would be useful to the NLI (e.g. [3rd-sqd-sec-lz] exists, but

[1st-sqd-sec-lz] does not) we extract a mini-ontology of actions and objects from the task model, and use it to generate out a more complete set of candidate representations for the training utterances. This mini-ontology contains approximately 15 roles and 50 fillers (compared with the nearly 80 roles and 280 fillers used in the hand annotated semantic frames) that combine to form about 110 unique frames (compared to over 400 frames used by the current NLI) that represent about 500 utterances.

Representing meaning using the task model has a number of advantages from a design perspective. First, as seen in Table 1, task frames are generally smaller than the intermediary semantic frames alleviating problems of sparsity associated with learning from small datasets. Second, using task frames yields comparable results in testing (see section 5) but is more efficient than using semantic frames. This is because simply matching utterances to elements of the task model requires fewer human-hours of labor than the full human annotation required by the current NLI. Finally, unifying meaning and action under one representation facilitates the use of a noisy channel approach to language understanding and provides a principled framework for exploiting non-linguistic information.

Table 1. Comparison of meaning representations for NLI input. Semantic frames are hand-annotations used by current (baseline) NLI; Task frames, already used by MRE for planning and reasoning, are adopted for use in situated NLI.

<i>roger we have an injured boy that's been struck by one of our vehicles we need to get him medevaced asap over</i>	
Semantic Frame	Task Frame
s.mood declarative	speech_act request
s.sem.prop.type event	agent It
s.sem.prop.agent humvee	event call
s.sem.prop.patient boy	patient medevac
s.sem.prop.event struck	
s.mood imperative	
s.sem.type action	
s.sem.event get	
s.sem.patient medevac	
s.sem.time present	

3.2. Noisy Channel Framework

The framework for exploiting non-linguistic context, or situation, that we present is based on a cognitive model of early word learning [4]. Like the current NLI to the MRE, this model represents linguistic meaning (i.e., the meaning of words) using conditional probability distributions of words given elements of a case frame representation [i.e. $p(\text{word} | \text{element})$]. Following work in Machine Translation (MT), the Expectation Maximization (EM) algorithm is used to find the maximum likelihood estimates of such conditional distributions over a training set of paired utterances and case frames. As in Brown et al. [16], we make the IBM Model 1 assumption that each word in an utterance is generated from exactly one element of its corresponding case frame representation (i.e. a role plus its role filler). This learned conditional distribution (i.e. word meaning model) is then used along with a model of the non-linguistic context (i.e. the situation model described below) in a noisy channel model of language understanding.

This noisy channel framework, also used widely in MT research, allows for a principled integration of linguistic and non-linguistic information. The framework relies on a Bayesian decomposition of understanding in which the (posterior) probability of some meaning given an utterance is proportional to

the (channel) probability of the utterance given that meaning multiplied by the prior (source) probability of the meaning itself.

$$p(\text{meaning} | \text{utterance}) \approx \tag{1}$$

$$p(\text{utterance} | \text{meaning})^\alpha \bullet p(\text{meaning})^{(1-\alpha)}$$

Here, the channel probability (i.e. word meaning model) represents the contribution of linguistic information, while the source probability (i.e. situation model) represents the contribution of non-linguistic information to understanding. Further, the weighting coefficient α provides a way to modulate the relative importance of one over the other.

Just as in MT, finding the most likely meaning given an utterance (i.e. the argmax of the posterior probability) involves searching for the meaning that generates the highest combined source and channel probabilities. Given our choice of representation, it is clear that the space of meanings that is searched during understanding is just that same set of task frames maintained in the task model. The intuition behind using such a search space follows from the fact that, since we know that only a set number of things can be interpreted by the system, we do not need to waste time considering any interpretations outside of that set.¹ Because this space of task frames is limited by the sophistication of the task model, the search is both finite and tractable, and thus, can be performed exhaustively with little concern for computational cost.²

It should be noted here that treating understanding as a search problem is not equivalent to treating it as a multi-class classification problem (in which each task frame is a unique class). Just as in MT, in the proposed framework, a novel utterance can yield a frame that had never before been seen in the training data. This is possible because the word meaning model learns structure within a task frame; thus, as long as an unseen frame is composed of elements that *had* been seen in separate training instances, it can be hypothesized by the system. In fact, this application of the noisy channel framework can no more be considered classification than can MT; the only difference being in the size of the space of possible meanings (sentences) that must be searched.

This noisy channel approach allows us a principled way to incorporate non-linguistic information as the source probability in Equation 1. In the next section we detail how this distribution can be built using information represented in the MRE's task model.

3.3. The Situation Model

Situation models for NLIs attempt to exploit the domain knowledge about what agents (human and virtual) are likely to do in a given context. In the MRE domain, we do this by taking advantage of the plan representations maintained in the task model. For example, by examining the plan represented in Figure 4, we know that if a medevac is to come to the user's position (to evacuate the wounded boy) two actions must occur: 1) the medevac must be called (i.e. $[call_medevac]$) and 2) the landing zone must be secured (i.e. $[secure_lz]$). Further, we know that these actions can occur in either order. It follows, thus, that if we knew that a user was going to perform the $[medevac]$ action, we could predict that they would do this by first performing either $[call_medevac]$ or $[secure_lz]$.

¹ As in all NLIs, handling out of domain utterances is a challenge. Anecdotal examination suggests that this problem may be solvable by filtering utterances based on out of vocabulary words.

² See discussion for questions of scalability of this approach.

Table 2. Probabilistic Context Free Grammar (PCFG) rules converted from hierarchical plan representation in task model. Probabilities are estimated over held out data.

Num	Rule	Prob
1	complete_mission -> collision help_boy support_inspection	.5
2	complete_mission -> collision support_inspection help_boy	.083
3	complete_mission -> collision help_boy support_inspection reinforce_lz	.25
4	complete_mission -> collision support_inspection help_boy reinforce_lz	.083
5	complete_mission -> collision help_boy reinforce_lz support_inspection	.083
6	help_boy -> secure_area evaluate_boy medevac	.5
7	help_boy -> secure_area evaluate_boy ambulance	.083
8	help_boy -> evaluate_boy secure_area medevac	.25
9	help_boy -> evaluate_boy secure_area ambulance	.083
10	help_boy -> evaluate_boy treat_on_scene	.083
11	medevac -> secure_lz call_medevac	.777
12	medevac -> call_medevac secure_lz	.222
13	ambulance -> call_ambulance	1
14	support_inspection -> squad_to_celic	1

To take advantage of such knowledge, we use plan recognition to estimate the probability of what a user is likely to say, given the sequence of actions that have already been performed [5]. We follow Pynadath [10] in treating plan recognition as a probabilistic parsing problem. Thus, in the same way that a syntactic parser infers a syntactic structure over words in a sentence, our plan recognition parser, infers a plan structure over observed actions in an environment.

In building such a parser, we must first convert each abstract action in the task model into a rule in a probabilistic context free grammar (PCFG). This is done by first removing the states and threat relations from the task representation, leaving the set of actions (both primitive and abstract) and the causal links between them. By following these causal links we get a set of possible sequences of actions, each one of which is a valid right hand side for a PCFG rule (where the left hand side is just the abstract action itself). We can simplify these rules further by removing all actions that the human user cannot directly affect (e.g., in figure 4, the action *[med_move_to_area]* can only be affected by performing *[call-medevac]*). PCFG Rules 11 and 12 in Table 2 are the results of converting the task in Figure 3 (the probabilities are estimated on held out data, see section 4).

Given a grammar of rules converted from the task model, the contextual history of an utterance can now be exploited during language understanding. Figure 5 shows a typical situation in which, having completed a series of actions (i.e. *collision*, *secure_area*, *evacuate_boy*) the user produces an utterances that is poorly recognized by the ASR module. Even though the linguistic evidence here is weak, by setting the prior probabilities (in Equation 1) of each frame appropriately, the utterance can still be understood. Setting these prior probabilities is done using the PCFG parser and the grammar rules described above.

As each action is performed in the VE, the PCFG parser infers a hierarchical plan structure, which acts as a memory of what

actions have already been performed in the system (see Figure 5). Whenever a new action is performed, the system uses this memory to build a probability distribution over which action will be performed next. This is done by simply cycling through each action in the task model, parsing it with the situation model as though it had actually occurred, and storing the probability of each resulting tree. These probabilities are then used as the priors (see Equation 1) used during understanding.

In order to generalize the effect of this procedure, we collect task frames that operate on the same objects into *priming sets* (see Table 3) which act as equivalence classes when assigning these prior probabilities. So, for example, the prior probability assigned to *[call_medevac]* in Figure 5, will be assigned to all five task frames in *[call_medevac]*'s priming set. This allows the situation model to recover from the fact that, although human users must follow plans in general, they do not always take the exact steps written into the task model.

Unlike previous work using plan recognition in dialogue (e.g. [20]), this work focuses on using the predictive power of plan recognition to determine from non-linguistic context what is likely to have been said, even given weak linguistic evidence. Using PCFGs to model the situation surrounding human utterances allows us to capture this predictive power while still maintaining a relatively simple and tractable representation. Although a number of other plan recognition algorithms have been proposed, this approach represents a compromise between simpler models (such as HMMs [18]) that are easier to learn and parameterize, but are not in general hierarchical, and fully abductive planners [19] which can handle even more complex representations (such as interleaved plans) but suffer from greater complexity. Both learning and interleaving plans will be examined in future work.

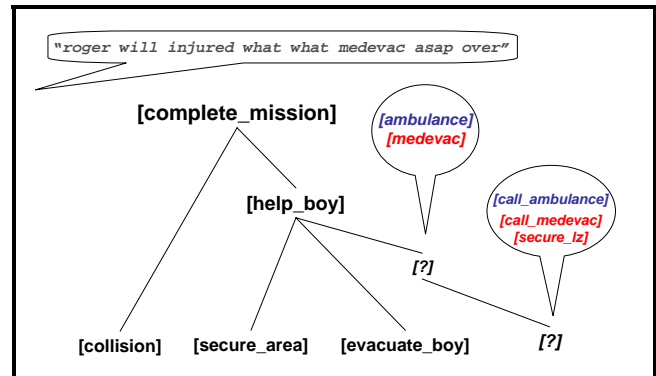


Figure 5. Plan recognition in the situation model operates as PCFG parsing (see rules in table 2). As new actions occur in the system, a hierarchical plan representation is updated and possible next actions [?] are predicted. The model is shown as a new utterance (output by ASR) is given to the system.

Table 3. Task frames primed by system action *[call_medevac]* in the situation model.

<i>[call_medevac]</i>		
attribute location	agent lt	agent medic
object medevac	event call	event call
	object medevac	object medevac
attribute status	agent sgt	
object medevac	event call	
	object medevac	

Table 4. Sample transcript of interaction between human and virtual agents (only human utterances shown). Both human transcriptions and output of automatic speech recognition (ASR) are shown along with actions taken by the system (e.g., [secure_area]).

Human Transcribed	ASR Transcribed
[collision]	
sergeant secure the assembly area	i sergeant secure the assembly area
[secure_area]	
[evaluate_boy]	
eagle base this is eagle two six over	out eagle base this is eagle two six over
roger we have an injured boy that's been struck by one of our vehicles we need to get him medevaced asap over	roger will injured what what medevac asap over
[call_medevac]	
sergeant have third squad secure the lz	that sergeant have third squad secure the lz
sergeant have third squad secure the lz	that's sergeant have third squad secure the lz
do it	Do it
[secure_lz]	
sergeant send fourth squad to celic	i sergeant send one squad tucci what
roger eagle one six this is eagle two six we are at a medevac site	roger eagle one six is eagle two six we have a medevac site
roger eagle eagle one six i have one squad inbound over	i route to eagle one six that was squad about
sergeant send fourth squad to celic	sergeant send one squad the child's
[squad_to_celic]	
sergeant let's move the boy to the lz	sergeant what happened to the lz
sergeant move the boy to the lz	sergeant move the boy to the lz
sergeant control the crowd	l sergeant route
sergeant have first	sergeant have third
sergeant have first squad reinforce the lz	that sergeant have third squad reinforce the lz
[reinforce_lz]	

4. EVALUATIONS

We evaluate the situated NLI on transcripts of eight test runs of the MRE system (totaling 150 individual test utterances). Test runs represent actual interactions between military personnel and the virtual agents in the MRE system. Due to error propagation amongst the modules (e.g. ASR, Dialogue Manager, etc.), these test runs are far from the type of clean interactions one sees with Wizard of Oz data. However, evaluating on actual interactions more accurately represents the system's true performance, as such noisy interactions are the norm in interactive applications. Table 4 shows a portion one of these test interactions.

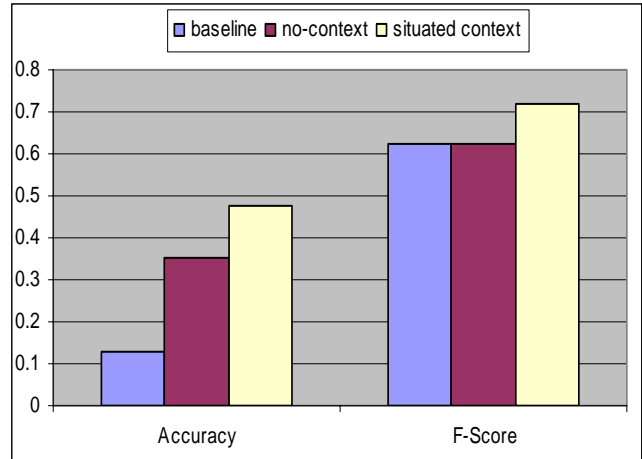


Figure 6. Performance of the situated NLI both with and without the use of the situation model as context versus performance of the current (baseline) system to the MRE.

Approximately 360 manually transcribed utterances, paired with frame representations from the task model, were used to train the word meaning model for the situated NLI. The number of iterations, beam search, and other initialization parameters for the EM algorithm (see [9]) were optimized using cross-validation. For each test run, maximum likelihood estimates for the PCFG rules used in the situation model were generated using all *other* runs in the test set.

For purposes of comparison, the current NLI to the MRE was treated as a baseline system. It was trained using the same 360 utterances paired with human annotated semantic frames. All parameters were set using cross-validation. Comparative evaluation between the baseline and the situated MRE proposed here is not trivial. Due to logistical issues, a full task based evaluation of the system was not practical. Further, because the situated NLI necessarily employs a different representation of meaning than the current NLI (i.e. task frames instead of semantic frames), the systems' outputs cannot be directly compared. Instead, we compare the NLIs using two system-relative metrics: a strict measure of the proportion of complete frames that the system generates which are correct (i.e. accuracy), and a relaxed measure of the F-Score over the elements of those frames³. Because of these constraints, experimental results must be considered preliminary.

In examining the performance of the situated NLI we perform three experiments. In Experiment 1, we examine the performance of the situated NLI both with and without (alpha=0.5 and alpha=1 from Equation 1, respectively) the use of the situation model against performance of the baseline system (note that the baseline system cannot be tested with context for reasons described in section 3). We measure both total frame accuracy and F-Score on automatically recognized speech. In Experiment 2, we look more closely at the effect of context on F-Score performance in the situated NLI by varying the setting for the weighting coefficient alpha (see Equation 1). In these experiments, alpha=1 corresponds to using only the word meaning model for

³F-Score is equivalent to $(2 * \text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$, where $\text{Precision} = (\text{Correct Gussed Elements} / \text{Total Gussed Elements})$ and $\text{Recall} = (\text{Correct Gussed Elements} / \text{Total Correct Elements})$

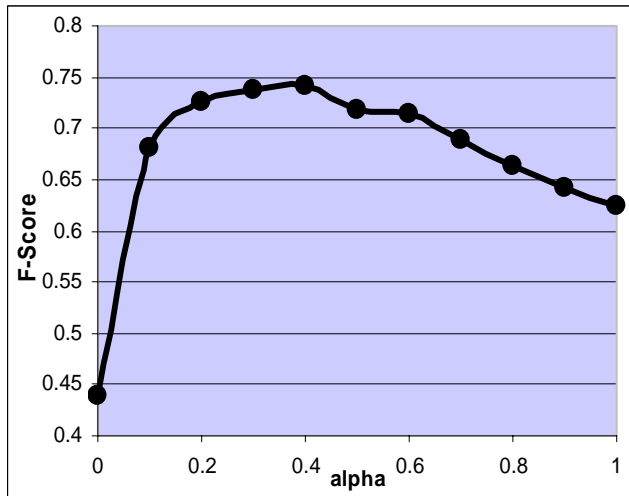


Figure 7. Effect of situation model on the situated NLI.

understanding, while $\alpha=0$ corresponds to using only the situation model (i.e. no linguistic evidence at all). Finally, in Experiment 3, we directly examine the robustness of the situated NLI to automatically recognized speech by comparing performance of the system, both with and without the situation model, on hand transcribed text versus text output by the ASR module.

5. RESULTS

Results from Experiment 1 are shown in Figure 6. No significant difference was found between the F-Scores of the baseline versus the situated NLI without context.⁴ However, accuracy for the situated system was significantly improved over baseline ($p<0.01$). Further, the situated NLI with context performed significantly better on both accuracy ($p<0.01$) and F-Score ($p<0.05$) than both the baseline and the system without context.

Figure 7 shows results from Experiment 2. Again, incorporating the situation model in understanding increases performance on F-Score versus using no context at all. Interestingly, using a model with only context ($\alpha=0$), yields performance well above the simplistic baseline of always guessing the most common frame (F-Score = 0.11).

Experiment 3's results are shown in Figure 8. Not surprisingly, the situated NLI both with and without situated context performs better on transcribed speech than on ASR output. However, the relative drop in performance is nearly 50% greater for the non-context condition when moving from transcribed to ASR text.

6. DISCUSSION

Results from the three experiments above provide preliminary evidence that the situated NLI proposed here performs better than the shallow semantic parser currently used in the MRE. As shown in Experiment 1, the situated NLI significantly outperforms the baseline system on F-Score, and is over four times more accurate in predicting an utterance's complete frame representation. This dramatic increase in accuracy arises directly

⁴ Although the baseline uses more complex models of word meaning than the situated NLI (i.e. bigrams and trigrams), the benefits of these complexities are often lost due to noise from the ASR system.

from viewing understanding as a search problem. Unlike the baseline system which makes no guarantee that the output will be a well formed case frame, the situated NLI's outputs can only be complete frames already represented in the task model. Thus, given only partial linguistic evidence (as is often the case with ASR output), the baseline system will generate only a partial semantic frame. However, given the same limited evidence, the situated NLI is forced to find the complete task frame that is most likely associated with the input.

Even more critical to the success of the situated NLI than the understanding as search paradigm is its incorporation of a situation model to capture the context of an input utterance. Such context allows the system to perform reasonably even when given no linguistic information at all, as seen in Experiment 2. Further, as Experiment 3 shows, using the situation model makes the system more robust to the noise introduced by ASR than the same system without such situated context. Thus, such situational context can aid in recovering from the problem of error propagation that can so often cripple complex NLIs to interactive applications.

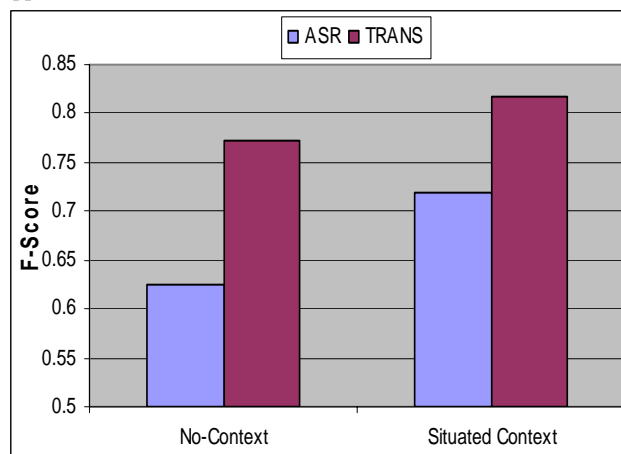


Figure 8. Performance of the system with and without situation model context on hand transcribed (TRANS) versus automatically recognized (ASR) text.

7. CONCLUSION

We have introduced a framework for learning situated Natural Language Interfaces (NLIs) to interactive virtual environments. The framework exploits the non-linguistic information explicitly modeled in such interactive applications and provides a principled way to incorporate such information by means of a noisy channel approach to language understanding. Preliminary experimentation in an independently designed interactive application, i.e. the Mission Rehearsal Exercise (MRE), shows that the situated NLI outperforms the MRE's existing NLI on both whole frame accuracy and F-Score metrics. Further, use of the situation model in the situated NLI increases robustness to the noise introduced by the use of automatic speech recognition.

The benefits of this framework lie not only in its improved performance, but also in its increased efficiency. By using pre-existing representations from the task model to represent the meaning of utterances, the situated NLI avoids the need for the costly human annotation of utterances with intermediary semantic frames. Because using task frames requires only matching training utterances to actions in the task model, annotation can be accomplished with markedly fewer human hours of work.

Further, the framework we have presented opens up a number of avenues for future work. One area that has already shown promise [4] is the use of situation modeling in cognitive models of language acquisition. Such research uses situated NLP techniques to show how social and pragmatic information can affect how children learn language. Another area of future work is the application of situated NLI to other domains, such as databases. These domains present new challenges in that they do not afford access to the rich non-linguistic information seen in interactive virtual environments such as the MRE, and therefore, require different methodologies for the construction of a situation model. Perhaps the most interesting area of future work, though, is the development of a new field of situation modeling which, like language modeling, would be useful to a number of other language technologies (e.g. MT, ASR, etc.). Such models could incorporate other non-linguistic aspects of the situation, such as gestures and eye gaze, which could then be incorporated into NLI in a principled manner.

Finally, although the results presented here are preliminary, it is clear that situated NLI have the potential to be more efficient and accurate than the NLI currently used in interactive applications. Importantly, these situated NLI are not less scalable than their non-situated counterparts. Many attempts to create situation models have been attacked because of the difficulties involved in scaling representations of contextual knowledge, and the situation model here is no exception. However, it is important to distinguish issues of scalability that are associated with interactive virtual environments in general, and those issues associated with learning NLI in particular. For, it is true that scaling task models is not easy (and is often considered the “art” of a good interactive application), but it is not true that, given an already designed task model of arbitrary size, the algorithms used in this framework do not scale. Rather, the underlying strategy of this work is to exploit the fact that interactive virtual environments operate over very limited domains as a wedge into the very difficult problem of language understanding. Taking advantage of this situation does not imply a lack of scalability for situated NLI, but rather, it suggests that by exploiting the lack of scalability already present in interactive applications we can begin to make learning situated NLI feasible.

8. ACKNOWLEDGMENTS

The authors would like to thank David Traum and Jon Gratch for useful discussions and their expertise on Virtual Interactive Environments. We would also like to thank Rahul Bhagat and Susan Robinson for their continued assistance, without which this work would not have been possible. Finally, we thank Deb Roy, who was essential to the development of the ideas presented here.

9. REFERENCES

- [1] Bhagat, R., Leuski, A., and Hovy, E. Statistical Shallow Semantic Parsing despite Little Training Data. In prep.
- [2] Fillmore, C. Frame Semantics and the Nature of Language. *Annals of the New York Academy of Science Conference on the Origin and Development of Language and Speech*, Volume 280 (pp. 20-32), 1976.
- [3] Fleischman, M., Kwon, N., and Hovy, E.. 2003. Maximum Entropy Models for FrameNet Classification. *EMNLP*, Sapporo, Japan, 2003.
- [4] Fleischman, M. B. and Roy, D. Why Verbs are Harder to Learn than Nouns: Initial Insights from a Computational Model of Intention Recognition in Situated Word Learning. 27th Annual Meeting of the Cognitive Science Society, Stresa, Italy. July 2005.
- [5] Fleischman, M. B. and Roy, D. Intentional Context in Situated Language Learning. Ninth Conference on Computational Natural Language Learning, Ann Arbor, MI. June 2005.
- [6] Ge, R. and Mooney, R.J. A Statistical Semantic Parser that Integrates Syntax and Semantics Proceedings of the Ninth Conference on Computational Natural Language Learning, Ann Arbor, MI, pp. 9--16, June 2005.
- [7] Gildea, D. and Jurafsky, D.. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3) 245-288 14, 2002.
- [8] Gorniak, P. and Roy, D.. Probabilistic Grounding of Situated Speech using Plan Recognition and Reference Resolution. *ICMI*, 2005.
- [9] Moore, R. Improving IBM Word Alignment Model 1. in Proc. of 42nd ACL, Barcelona, Spain., 2004.
- [10] Pynadath, D. Probabilistic Grammars for Plan Recognition. Ph.D. Thesis, University of Michigan, 1999.
- [11] Reiter, E. and Roy, D. (in press). Connecting Language to the World. Special issue of *Artificial Intelligence*.
- [12] Roy, D. "Grounding Words in Perception and Action: Insights from Computational Models". *Trends in Cognitive Science*, 2005.
- [13] Traum, D., Rickel, J., Gratch, J. and Marsella, S. "Negotiation over Tasks in Hybrid Human-Agent Teams for Simulation-Based Training." in 2nd International Conference on Autonomous Agents and Multiagent Systems, Melbourne, Australia, July 2003.
- [14] Swartout, W., Gratch, J., Hill, R.W. Jr., Hovy, E., Lindheim, R., Marsella, S., Rickel, J., Traum, D.R. *Simulation in Hollywood: Integrating Graphics, Sound, Story and Character for Immersive Simulation*. Multimodal Intelligent Information Presentation (Oliviero Stock and Massimo Zancanaro Eds), 2005.
- [15] Zettlemoyer, L. and Collins, M.. Learning to Map Sentences to Logical Form: Structured Classification with Probabilistic Categorical Grammars. In proceedings of *UAI* 2005.
- [16] Brown, P. F. Della Pietra, V. J. Della Pietra S. A. & Mercer., R. L. The Mathematics of Statistical Machine Translation: Parameter Estimation, *Computational Linguistics* 19(2), 1993.
- [17] Miller, S., R. Bobrow, R. Ingria, and R. Schwartz. 1994. Hidden Understanding Models of Natural Language, Proceedings of the Association of Computational Linguistics (ACL) conference, pp. 25-32.
- [18] Horvitz, E., Breese, J., Heckerman, D., Hovel, D, and Rommelse., K. The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In Proc. of the 14th Conference on *UAI*, 1998.
- [19] DE Appelt, ME Pollack. Weighted abduction for plan ascription In Proceedings of *User Modelling and User-Adapted Interaction*, 1992.
- [20] Carberry, S. *Plan Recognition in Natural Language Dialogue*. MIT Press, Cambridge MA. 1990.