# Probabilistic Grounding of Situated Speech using Plan Recognition and Reference Resolution

Peter Gorniak
MIT Media Laboratory
20 Ames St.
Cambridge, MA, USA
pgorniak@media.mit.edu

Deb Roy
MIT Media Laboratory
20 Ames St.
Cambridge, MA, USA
dkroy@media.mit.edu

## ABSTRACT

Situated, spontaneous speech may be ambiguous along acoustic, lexical, grammatical and semantic dimensions. To understand such a seemingly difficult signal, we propose to model the ambiguity inherent in acoustic signals and in lexical and grammatical choices using compact, probabilistic representations of multiple hypotheses. To resolve semantic ambiguities we propose a situation model that captures aspects of the physical context of an utterance as well as the speaker's intentions, in our case represented by recognized plans. In a single, coherent Framework for Understanding Situated Speech (FUSS) we show how these two influences, acting on an ambiguous representation of the speech signal, complement each other to disambiguate form and content of situated speech. This method produces promising results in a game playing environment and leaves room for other types of situation models.

**Categories and Subject Descriptors:** I.2.7 [Artificial Intelligence]: Natural Language Processing
**General Terms:** Human Factors, Algorithms
**Keywords:** situated, speech, language, grounding, understanding, plan recognition.

## 1. INTRODUCTION

Naturally occurring speech is ambiguous in form (how it was said) and in content (what was intended by the speaker). This paper presents a Framework for Understanding Situated Speech (FUSS) that handles aspects of both kinds of ambiguity, and integrates them to produce disambiguated interpretations of situated speech acts. We first explain both kinds of ambiguity and sketch the related components of the FUSS and how they interact. The remainder of the paper presents an experimental platform for collecting speech together with the embedding situation, and provides technical details on the FUSS components. We conclude with an example showing how the FUSS resolves the meaning of a specific utterance, and provide a preliminary evaluation that suggests that the framework understands referential speech across a larger dataset.

This paper makes two main contributions. First, we describe a new framework (FUSS), that maintains the ambiguity of form inherent in the acoustic speech signal and syntactic parsing, while allowing for grounding of language in complex situation models. We believe that while semantic interpretation and composition should be driven by syntax, it is important to acknowledge that understanding is central, and thus that the semantic interpretation step may well involve perception, situational modelling, plan recognition and acting on the world. FUSS therefore allows for arbitrary grounding and composition actions during the parsing process whenever a possible grammatical constituent is completed, as long as the results can be expressed as probabilities over a situational model. Secondly, we show how the FUSS can be used to perform simple binding of referring constituents to physical objects in the situational model, and how such binding can be disambiguated successfully by taking into account the speaker's plans. To our knowledge this is the first instance of speech understanding taking into account both the physical and the intentional situation of an utterance in a general framework.


*ICMI'05,* October 4–6, 2005, Trento, Italy.


### 1.1 Ambiguity of Form

Speech is an acoustically, lexically and grammatically noisy signal. A speaker may convey the same message with wildly different audio signals, select from an immense set of possible words, and put these words together in an infinite number of ways. We believe that in many cases of spontaneous speech it is impossible to confidently transcribe the words a speaker used based solely on the audio signal - and that no human listener does this. Instead, we believe that much of the message is disambiguated using the shared situation of speaker and listener. The handling of form ambiguity in our FUSS therefore does not so much focus on resolving the ambiguity, but rather on compactly capturing and ranking the possible grammatical parse trees for later disambiguation through context. The ranking should be informed by acoustic, lexical and grammatical knowledge.

The FUSS uses a Hidden Markov Model [14] based speech recognizer to produce confusion networks, which concisely capture a large number of different hypotheses and allow for easy computation of the probability of a hypothesis based on the speech recognizer's acoustic and language models [11]. The framework then feeds these confusion networks into a probabilistic context free grammar parser, which combines the probabilities the speech recognizer assigns with those associated with a set of grammatical rules. It in turn generates a probabilistically ranked list of parse trees that assign grammatical interpretations to the confusion networks. In this way, the FUSS carries through ambiguity arising from each interpretation step to the next, offering a set of parse trees to

the subsequent interpretation stages that take into account probabilistic and structural information from all previous steps.

## 1.2 Ambiguity of Content

Imagine two people working together to solve a problem. One of them says "Can you help me with this?". Likely, the other person will understand what is asked of them. You, the reader, however, cannot know the full meaning of this utterance. You do not know what problem they are working on, which part of the problem they are currently tackling, what each person's abilities are, what they have said and done so far, and what they plan to do next. In short, you need to be informed of the situation embedding the utterance to understand it, because people rarely specify the full situation explicitly in what they say. We can express different meanings with the same words by leveraging shared context. For example, "can you help me with this?" may be a request to jointly lift a heavy object in one situation, and an order to connect a laptop to a projector in another. Barwise and Perry call this the *efficiency of language* [2].

Most speech we would like machines to understand, be it for interacting with mobile devices, for autonomous robots or for desktop speech interfaces, is efficient and situated in this manner when the human speaker can speak naturally. Current systems that attempt to understand the user via natural language often force a more explicit style of language than is natural (type "the place I visited yesterday" into Google to see what a powerful but non-situated natural language processing engine will do).

In the work presented here, we are specifically interested in models that capture two aspects of the situation: the physical context, including location of the speaker and objects nearby, as well as the functional aspect of what the speaker is trying to achieve with his or her utterance. Both aspects go some way towards disambiguating an efficient utterance, but only integrating them into a single understanding process will leads towards human level flexibility in natural language understanding. The FUSS presented here is a first step towards integrating a situation model that includes both the physical context and the functional context (by modelling the users plans) into speech understanding. Such a situation model must be dynamically coupled with and grounded in the world the speaker acts in, meaning that it must be able to make predictions about how the situation will evolve over time, which in turn can be verified against the model's predictions. For our current focus, we will use 'grounding' to denote the process of using such dynamically predictive situation models to understand and produce language. Grounding thus links language to the world and the world to language via a situation model.

## 1.3 Related Work

Work that takes into account the situation during speech understanding has until now limited itself to the immediate physical (usually visual) context [15],[16]. Aside from speech, some language understanding work involves aspects of grounded situation models, for example Narayanan's interpretation of news stories using an action representation [12], and our own grounding of spatial language in visual scenes [7]. All of these handle some aspects of language efficiency by maintaining situational models. In contrast to dealing solely with visual scenes or abstract action models, we here particularly focus on language efficiency that occurs when speakers share a common history and common future plans, making anaphora and reliance on shared intentions a common occurrence. Of these related works, only Schuler also uses a speech recognizer as input (as opposed to transcribed speech or text), but does not maintain ambiguities all the way down to semantic interpretation as we do here. Roy and Mukherjee integrate visual priming directly into the speech understanding process, whereas we utilize speech confusion networks as a compact representation of ambiguous speech recognizer output. Confusion networks and probabilistic Earley parsing are well known in the speech recognition literature [11, 17], and stochastic context free grammars have recently been proposed for plan recognition in other domains [3, 13]. Plans and intentions have long been recognized as important elements of natural language understanding, but related work is restricted to non-situated language systems that assume all necessary information is available from the words themselves [4, 1].

# 2. DOMAIN AND DATA COLLECTION

Current day multi-user graphical role playing games provide a rich interaction environment that includes rooms and exterior areas, everyday objects like chairs, doors and chests, possessions, character traits and other players' avatars. All of these can be acted upon by a player, be it through taking direct action on the world or through speaking with other players. We are using a commercial game, Neverwinter Nights (*http://nwn.bioware.com*), that includes an editor allowing the creation of custom game worlds.

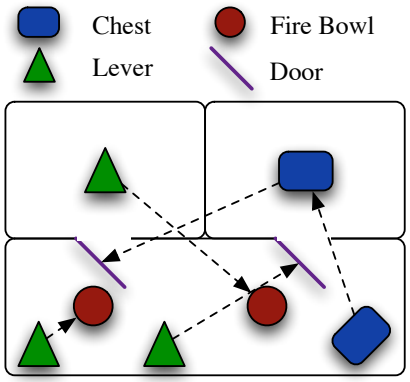


**Figure 1: A diagram of the map used for data collection, with dashed lines indicating dependencies between objects.**

For the work presented here, we designed a two player game module structured around a puzzle. To simplify dialogue aspects of the data, we only allow one of the players to speak. The other (played by the experimenter), is in the same real-world room as the first player (the study participant), but does not speak and does not act autonomously - he or she only does as instructed. In this way we restrict interaction to be similar to what commanding an intelligent but passive machine controlled character would be like. However, we do not restrict the language used in any way (except indirectly through the choice of puzzle), and the speaking study participant knows that a human being is listening to his or her commands. The game, puzzle and annotation methods are described in more detail in [6].

Figure 1 shows the map of the puzzle used for data collection. Both players' avatars start in the large room in the bottom half of the map. The coloured symbols in the map represent objects (explained in the map legend), whereas the dashed arrows indicate the dependencies between objects that must be followed to solve the puzzle. The overall goal is to light both fire bowls at the same time. The players were only told about this overall goal, without knowing how to accomplish it. One chest contains a key that unlocks the second chest, which in turn contains a key that unlocks one of the doors. One of the levers opens the door to the second chest, whereas the other two levers (one behind the second door) light a fire bowl each. The puzzle cannot be solved by a single player due to timing constraints: the right door on the map can be opened with one of the levers, but it closes again after a short time, making it impossible for the same person to pull the lever and run through the door. Similarly, each fire bowl extinguishes itself after a few seconds unless both are lit, making it impossible for a single person to light both quickly enough. Participants usually solved the puzzle within 15 minutes.

During data collection, we recorded player's in-game actions, and his or her speech using a head-worn microphone. We ran an utterance segmenter on the recorded audio, which produced 554 speech segments across 6 sessions [19]. We manually transcribed the utterances, yielding 200 utterances that were not pure noise, off topic (such as requests for help with playing the game), or commands (such as self-reporting by some players). The average length of these utterances is 7 words. For each session we built a closed vocabulary trigram language model for the speech recognizer using the transcripts from the other sessions. The confusion networks generated by the speech recognizer have on average 23 confusion sets (explained in Section 3.1). Most of the extra hypothesized word slots stem from silences and noise within or around the actual speech utterance. In this paper, we only concentrate on a specific subset of 90 utterances that contain noun phrases directly referring to a physical object in the game world, such as "activate the lever for me" or "can you come over here and pick this", but not "do that again" or "on the left".

To train a probabilistic context free grammar, we preparsed the transcripts with the Stanford Parser [10] using a standard grammar for written English that does not handle many aspects of the spontaneous speech in our utterances. We then corrected the produced parse trees manually and used the trees of five sessions to learn a grammar for the remaining one. Finally, we abstracted the event traces of each data collection session into a higher level description that only contains the crucial events such as object interactions and room changes. We hand-crafted a grammar that captures the sequence of events necessary to solve the puzzle in a hierarchical fashion, including multiple ways to solve the puzzle (e.g. opening a door to let the other character into a room vs. asking him to open the door). The grammar also includes sets of rules that have NOOP (a 'skip' symbol) as a symbol to handle exploration by the player. We then estimated probabilities for this grammar using rule counts from the sessions other than the one being tested.

# 3. HANDLING AMBIGUITY OF FORM

## 3.1 Speech Recognition

We use the Sphinx 4 speech recognizer as a front end for FUSS (*http://cmusphinx.sourceforge.net/sphinx4/*). We have augmented this speech recognizer with confusion network generation facilities. Sausages are compact representations of possible hypotheses [11]. Each confusion set spans exactly one word slot, containing all words that might have occurred over that period based on the speech recognizer's acoustic and language models. Each word hypothesis is associated with a corresponding posterior probability, where the posteriors of all possible hypotheses in one set sum to one. We call the confusion sets $C_0 ... C_n$. For the results reported here we used an efficient network construction algorithm based on the maximum a posteriori path [9]. The resulting source code is now publicly available as part of the Sphinx 4 distribution. Figure 2 shows part of a network from the data for the spoken utterance "Can you open the gate again." Nodes are shown in order of decreasing probability from top to bottom with the correct node highlighted in each confusion set. "<noop>" and "<sil>" are special words that stand for a possible word skip and a silence word, respectively. The example shows that the correct word is often not the one with the highest probability, and that confusion varies from a single word choice to more than 10 choices.

## 3.2 Probabilistic Parsing

We use a probabilistic Earley parser to syntactically parse confusion networks [5, 17]. This parser defines parsing states $S_i = i : {}_kX \rightarrow \lambda.\mu\,[\alpha, \gamma]$, indicating that before word $i$ of the utterance the parser has predicted and advanced grammar rule $X \rightarrow \lambda\mu$ beyond the (possibly empty) string of symbols $\lambda$ in the tail (position indicated by the dot), starting at word $k$ in the utterance. The state carries the forward probability $\alpha$, representing $P(C_0 ... C_{i-1}, S|G)$ the probability of the utterance up to position $i-1$ and the parser being in state $S_i$ at position $i$ given the probabilistic context free grammar $G$ being used for parsing. Similarly, $\gamma$ is the inner probability $P(C_k ... C_{i-1}, S_i|G)$. These quantities are analogous to the quantities of the same name defined for Hidden Markov Models [14], and the forward probability for most states is not technically a probability, see Stolcke [17]. An Earley parser starts with an initial state that produces a top level symbol (such as "S") for the grammar, and parsing is successful if the parser produces a final state for the same rule, with the tail completed. Both the forward an backward probability of this state correspond to $P(C_0 ... C_n|G$, i.e. the probability of the confusion network given the grammar.

We now discuss a few modifications we applied to the standard Earley algorithm, some covered by Stolcke [17]. As indicated above, we offer each word in a confusion set at position $i$ as a possible word in that position, and multiply a state's probabilities by the probability of the word in the confusion set. This incorporates the speech recognizer hypotheses directly into the parsing process and weighs them by the speech recognizer's acoustic and language model, effectively conditioning all probabilities produced during parsing on these models.

As speech is often noisy and grammatically incorrect, we also seed each parse position with an initial state, effectively causing the parser to work like a bottom-up parser so that it finds any grammatical substrings of the input. Currently, this modification is necessary because we only ground sentence fragments in the situation model, as opposed to complete sentences. We do not make this modification for our later use of the parser in plan recognition, because we are specifically interested in predicted probabilities in that case.

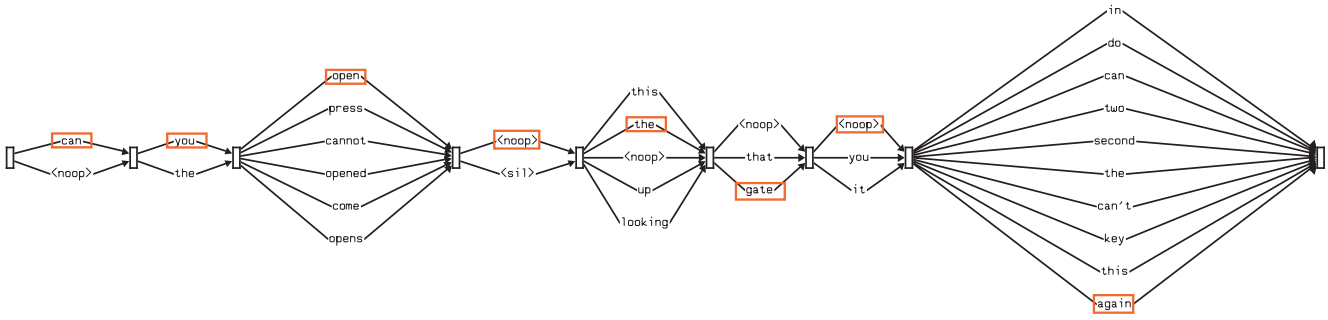


**Figure 2: A part of a confusion network produced by Sphinx 4 for the utterance "Can you open the gate again".**

At the end of a parse we search through the states produced for the most probable top level state that covers the largest portion of the confusion network. When performing semantic grounding, we also require such a state to have valid grounding probabilities (see below).

Finally, we automatically augment the grammar by splitting each rule N → t, where $t$ is a terminal, into three rules using a new non-terminal NOOP: the original rule, N → NOOP t and N → t NOOP. We add rules NOOP → <noop> and NOOP → NOOP NOOP, and estimate the probabilities of all these rules by counting the number of individual and pairs of <noop> symbols along the best paths of all confusion networks. In effect, this allows every terminal to be replaced with any number of skips preceding or following the terminal.

# 4. HANDLING AMBIGUITY OF CONTENT

## 4.1 Aspects of a Situation Model

A situation model for language grounding must capture many aspects of the context of an utterance. The physical context is important, as it provides possible referents such as objects as well as spatial features such as distance to disambiguate expressions such as "to the left of" or "near". As important as the physical context is the intentional context, which includes the speakers' plans and goals. An utterance like "can you help me with this?" can likely not be understood without both of these aspects of a situation model.

The situation model we use for the game setting includes information about the physical objects within the game, such as their location and type (whether the object is, for example, a chest or a lever). On the intentional level, the model currently consists of a set hierarchical plan fragments as provided by a probabilistic context free grammar parser parsing the event trace of the game session up to the current time. These plan fragments let us capture the past and predict the future of the current game session and indicate the goals held by a player on several levels of abstraction. Actions and intentions thus range from the physical action of activating a lever, to the intention of gaining access to the chest that is behind the door the lever opens, to the goal of having both fires lit at the same time by any means necessary.

## 4.2 Semantic Grounding

Similar to other work on incremental semantic interpretation during context free parsing [8], we let syntactic parsing drive semantic interpretation and derive a (possibly incomplete) interpretation for each syntactic constituent completed by the parser. As opposed to other approaches that force the situation model to be uniform and symbolic [16], the FUSS presented here makes no strong requirements of the situation model, except that its binding to any given constituent be expressible as a probability distribution (we currently work with discrete distributions, but an extension to continuous ones should be possible, just as it has been for other probabilistic models). Of course, using a context free grammar parser as the driver for semantic composition does make strong assumptions about the independence of semantic grounding and composition operations as well as the incremental nature of language understanding.

The parser driven semantic grounding and composition strategy discussed here derives from our prior work on visual grounding for referring expression [7]. We enrich the lexicon to specify the words' bindings to a situational model and their compositional behaviour. The grammar stays unchanged. Each time the parser completes a state with a rule that contains in its tail a terminal symbol, it calls on the bindings stored in the lexicon to ground this word in the situation model. For example, a word like "lever" would at this point bind itself to the levers in the world by producing a probability distribution over possible referents in which levers are more likely than other objects. In the virtual world of computer games we know a priori which objects are levers, but in many real world applications such information might be estimated from camera images or touch sensors and will include a degree of uncertainty. The word's binding is assigned as a grounding for the head of the rule of the completed state. In the case that a lexical item specifies an argument structure that cannot be satisfied yet (e.g. for "activate", which needs an argument specifying what to activate), the parser simply copies the potential binding without attempting to ground the word in the situation model at this stage. Lexical items can have any number of potential bindings and argument structures specified, and each one will be considered by the parser.

Every time the parser completes a higher level state that has non-terminals in its tail, it searches through the bindings of each symbol in the tail, looking for pending compositions that use all symbols in the tail of the rule. For example, a rule such as NP → DT NN that might be completed after the fragment "this lever" would find an argument structure for "this" that expects one argument to its right. As a first pass, we have implemented an egocentric distance based interpretation of "this", that takes the bindings from "lever" and scores them according to Tenenbaum's word generalization model [18]. This example also shows how the dynamic

bindings to the situation model allow for the interpretation of efficient utterances such as "this chest", which refers to different objects given different player locations.

Using the situation model bindings of words, the parser now effectively calculates $P(R|C_{k...i-1})$, the probability that a segment of the confusion network refers to an entity $R$ in the situation model. Note that while our examples so far have equated model entities with physical (or at least in-game physical) objects, they can be arbitrarily complex constructs in the model such as the action of a certain player activating a certain lever, past or future actions of players or players' beliefs. We can now calculate

$$P(C_{k...i-1}|R) = P(R|C_{k...i-1})P(C_{k...i-1})/P(R)$$

where we calculate the prior term from the confusion network path fragments used during semantic interpretation of this segment. With appropriate independence assumptions this can be used to combine parsing probabilities with reference probabilities:

$$P(C_{k...i-1}, S_i|R, G) = P(C_{k...i-1}|R)P(C_{k...i-1}, S_i|G)$$

, which with one more application of Bayes' theorem yields the sought after

$$P(R|C_{k...i-1}, S_i, G) = \frac{P(C_{k...i-1}, S_i|R, G)P(R)}{P(C_{k...i-1}, S_i|G)}$$

namely the probability that a fragment of the confusion network produced at a certain point in the parse refers to an entity in the situation model, incorporating all available information. If $S_i$ is the final state corresponding to the initial seed state, this represents the probability of a fragment of the confusion network referring to a specific entity. We will show a use for non-uniform reference priors in the next section.

## 4.3 Plan Recognition

The puzzle described in Section 2 encourages the use of efficient language: there are few objects with easily identifiable functionality that are used repeatedly and towards a known goal. Interestingly, due to the tight spatial arrangement and the fact that players often have to be in different rooms leading to lack of visual contact between the avatars, there were relatively few egocentric spatial descriptions with respect to either character (such as "the lever next to you"). In fact, despite there being at least two possible referents for every simple noun, participants often underspecified as in "pull the lever". Our hypothesis for how successful communication is possible in spite of ambiguous speech utterance relies on player's knowledge of past actions and future plans. After a short time, players know the function of the different objects and incorporate them into partial plans. When engaged in such a shared plan fragment, there is no need to spatially disambiguate object references, because they already are disambiguated due to plan knowledge.

To test this hypothesis, and to show that the FUSS presented here not only successfully resolves object references given a noisy speech signal, but can also incorporate priors due to more sophisticated situation modelling, we re-use the Earley parser described above in its predictive mode to recognize a player's plan.

Due to the predictive nature of the Earley parser it is possible to estimate the probability of a symbol being parsed at the next step by summing the forward probabilities of all states with a dot to the left of that symbol in the current parsing slot. During plan recognition, this lets us predict which objects the player will likely want the other character interact with next, namely those that are involved in actions estimated as likely in the next steps of the plans currently in progress. We use these predictions, summed and normalized across objects, as priors $P(R)$ for semantic parser.

| | **Accuracy** |
|---|---|
| Full Understanding | 50/90 (56%) |
| Physical Baseline | 27/90 (30%) |
| Plan Recognition Baseline | 21/90 (23%) |
| Random Baseline | 1/7 (14%) |

**Table 1: Understanding Results and Baselines**

# 5. RESULTS

Figure 3 shows the disambiguation of the confusion network in Figure 2. The relevant words of the network are at the top of the figure, followed by a few of the constituents the linguistic parser assigns to them (the full chart contains thousands of states). The parser finds the long and highly probably phrase shown here, and the physical binding of "gate" produces the highly skewed probability distribution on the left, where the two bars correspond to the two doors in the puzzle. At the bottom of the figure is another partial parse, this time of the event stream. The solidly outlined boxes correspond to the last few events and constituents found, whereas the boxes with dashed outlines are predicted constituents. Thus, the player has just asked for the first chest (chest 4) to be unlocked, and has retrieved the Chest Key from it. It stands to reason that he or she will now attempt to access the second chest to use this key (and acquire the Door Key in the process), and the plan parser properly predicts this. To do so, the player must enter the East room, and the parser thus predicts that he or she will next ask the other player to pull the lever that opens the door. Whether this will be expressed by referring to the lever or the door itself is arbitrary, thus the probability distribution produced by the plan recognizer at this stage is confused between the two objects as likely referents. Merging the two distributions as described above yields a clear target.

The speech recognizer has a 50% word error rate due to the spontaneous nature of the speech, and the very small training set for the language model. The oracle, the path through each confusion network that produces the fewest word errors, achieves about 23% word error rate. As can be expected from the small training set, the speech recognizer performs somewhat worse than currently state-of-the-art spontaneous speech recognition at 50%. However, because we are using the full confusion network during parsing the oracle, which yields a better word error rate than current spontaneous speech recognition results, is in many ways the more important recognizer performance measure.

As baselines, we predicted an utterance's referent using only the parsing and binding to the physical game world and using only the predictions made by the plan recognizer. If any prediction methods produced indistinguishable numbers for any referents, we counted the result according to a hypothetical random guess between the offered referents. Table 1 shows the percentages of correct referents achieved by the full understanding system, the two partial baselines, and a purely random baseline (choosing with equal likelihood amongst the 7 possible referents of three levers, two chest and two doors). Clearly, neither the primitive lin-

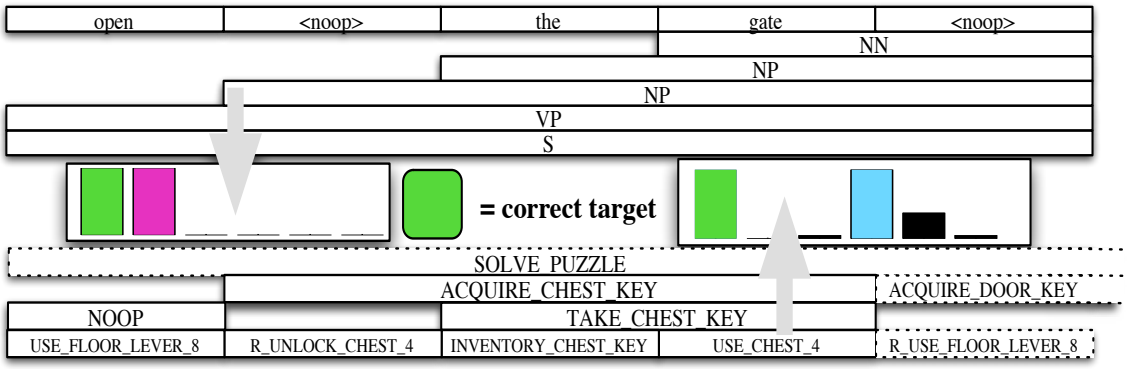


**Figure 3: An example linguistic and plan parse fragment showing disambiguation of the network in Figure 2.**

guistic and physical reference we used nor the guesses produced by plan recognition alone suffice to make good referent choices. Combining them shows a large improvement (about 50% improvement over either partial baseline) in correct referent determination, especially in the face of the very noisy speech signal used and the poverty of linguistic binding to the physical situation. This improvement supports our argument for the importance of combining functional and referential elements during speech understanding. Many remaining errors are due to mis-recognition by the speech recognizer and would be helped by a larger training set, some errors could be addressed with more sophisticated semantic composition while grounding in the physical world along the lines of our previous work [7] and some need a richer integration of the intentional model, such as those that refer to plan fragments.

## 6. CONCLUSION

We have presented FUSS, a framework that handles both ambiguity of form and ambiguity of content in understanding spontaneous speech. Using speech confusion networks and probabilistic parsing, we have shown that the framework can capture and concisely represent the ambiguities of form inherent in the speech signal. On the content side, we have argued for the importance of taking into account reference as well as functional and intentional meaning of speech. Within FUSS, we have demonstrated the beginnings of a situational model that touches both on the physical situation and the currently held plans of the speaker, and uses both to disambiguate efficient speech. Our preliminary results show an improvement in reference resolution when using these combined influences.

Beyond more linguistically and referentially sophisticated parsing and situation models, which are lacking in the work presented here (for example, many of the remaining utterances could be disambiguated by properly interpreting the spatial language used in them), there are many open questions in the plan recognition side of this work and its integration into the understanding process. For example, plan fragments often seem to serve not only as predictive tools, but also as direct speech reference as in "do that again". In other cases, players smoothly go from utterances like "pull the lever for me" to "open the door" to "hit me again" to "let me out" (all asking the other player to perform the same action), a progression touching on the physical and planning realms mentioned here, but also including aspects of spatial confinement and change of language due to shared experience and repetition. We hope to address many of these elements in making the next iteration of a FUSS.

## 7. REFERENCES


[1] J. Allen and R. Perrault. Analyzing intention in utterances. *Artificial Intelligence*, 15:143–178, 1980.

[2] J. Barwise and J. Perry. *Situations and Attitudes*. MIT Press, Cambridge, MA, 1983.

[3] A. F. Bobick and Y. A. Ivanov. Action recognition using probabilistic parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[4] P. Cohen and C. Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science*, 1979.

[5] J. Earley. An efficient context-free parsing algorithm. *Communications of the ACM*, 6(8):451–455, 1970.

[6] P. Gorniak and D. Roy. Speaking with your sidekick: Understanding situated speech in computer role playing games. In *Proceedings of Artificial Intelligence and Digital Entertainment*, 2005.

[7] P. J. Gorniak and D. Roy. Grounded semantic composition for visual scenes. *Journal of Artificial Intelligence Research*, 21:429–470, 2004.

[8] N. Haddock. Computational models of incremental semantic interpretation. *Language and Cognitive Processes*, 4:337–368, 1989.

[9] D. Hakkani-Tur and G. Riccardi. A general algorithm for word graph matrix decomposition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, 2003.

[10] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of teh 41st Meeting of the Association of Computational Linguistics*, 2003.

[11] L. Mangu, E. Brill, and A. Stolcke. Finding consensus among words: Lattice-based word error minimization. In *Proceedings of EUROSPEECH'99*, volume 1, pages 495–498, Budapest, 1999.

[12] S. Narayanan. *KARMA: Knowledge-based Action Representations for Metaphor and Aspect*. PhD thesis, University of California, Berkeley, 1997.

[13] D. V. Pynadath and M. P. Wellman. Probabilistic state-dependent grammars for plan recognition. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence, UAI2000*. Morgan Kaufmann Publishers, 2000.

[14] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, February 1989.

[15] D. Roy and N. Mukherjee. Towards situated speech understanding: Visual context priming of language models. *Computer Speech and Language*, 19(2):227–248, 2005.

[16] W. Schuler. Using model-theoretic semantic interpretation to guide statistical parsing and word recognition in a spoken language interface. In *Proceedings of the Association for Computational Linguistics*, 2003.

[17] A. Stolcke. An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. *Computational Linguistics*, 21(2):165–201, 1995.

[18] J. B. Tenenbaum and F. Xu. Word learning as bayesian inference. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, 2000.

[19] N. Yoshida. Utterance segmenation for spontaneous speech recognition. Master's thesis, Massachusetts Institute of Technology, 2002.