Radiobot-CFF: A Spoken Dialogue System for Military Training

Antonio Roque†, *Anton Leuski*†, *Vivek Rangarajan*‡,
Susan Robinson†, *Ashish Vaswani*†, *Shri Narayanan*‡, *David Traum*†

†Institute for Creative Technologies, ‡Speech Analysis and Interpretation Laboratory
University of Southern California, Los Angeles, USA
contact:traum@ict.usc.edu

Abstract

We describe a spoken dialogue system which can engage in Call For Fire (CFF) radio dialogues to help train soldiers in proper procedures for requesting artillery fire missions. We describe the domain, an information-state dialogue manager with a novel system of interactive information components, and provide evaluation results.

Index Terms: spoken dialogue systems.

1. Introduction

Spoken dialogue technology has great promise for allowing more natural communication and interaction with automated resources. There is often a trade-off, however, between the quality of the performance (speech recognition, understanding, task success), and the naturalness of the interaction. Commercial systems often achieve high success by limiting the style and nature of interaction to something not much more natural than voice menus. On the other hand, research systems that strive for greater flexibility in dialogue interaction often suffer from issues of reliability and robustness against a range of user types. Sometimes the domain itself constrains the type of interaction allowed. Military radio dialogues are often in an intermediate stage between fully natural dialogue, and simple retrieval tasks. Specialized Protocols are already used to increase communication reliability yet there are still fairly open aspects, placing challenges on traditional dialogue system technology.

There are many military simulation systems available for training various aspects of skills from logistics, to command decisions, to tactics, techniques, and procedures at a local level, however the simulation exercises usually require much effort on the parts of human operators to simulate the radio traffic that is required in these exercises. Our **Radiobots** project has as its aim producing automated radio operators that can engage in military radio communication for training tasks, allowing fewer operators to achieve the same training effect.

After some small scale tests, our current testbed, Radiobot-CFF, is a system that can act as the radio operator in a simulated Fire Direction Center (FDC) and take calls from a forward observer (FO) for artillery fire in training exercises. Radiobot-CFF operates in three modes (which can be changed at any time during the dialogue):

- a fully-automated mode, in which the radiobot engages directly in dialogue with an FO, and sends effects to a simulator, without human operator intervention.
- a semi-automated mode, in which the radiobot fills in forms and makes suggestions of what it should say, but an operator has an ability to override the recognition, make changes if necessary, and select responses.
- a passive mode, in which the radiobot monitors the dialogue and saves state, for operator memory or after-action review, but otherwise does not participate in the dialogue.

Radiobot-CFF has been integrated with military simulators in the Urban Terrain Module (UTM) of the Joint Fires and Effects Trainer System (JFETS), whose purpose is to train U.S. Army soldiers in conducting Call For Fire radio dialogues. The JFETS-UTM training environment, located in the Army base of Fort Sill, Oklahoma, is meant to be immersive. The soldier trainees are located in a room that has been built to resemble an apartment in the Middle East, with one window open to show a view of a city below. The window is actually a rear-projected computer display, and the soldiers can view close-ups of the city with binoculars that have computer displays at the end of them. Sounds of the city - and of explosions - are audible, and the temperature of the room can be increased to that typical of the Middle East.

In this paper we describe the domain, design and preliminary evaluation of the radiobot with soldier trainees in a training simulation. In the next section, we describe the domain and setting in a little more detail. In section 3, we describe the dialogue model used for this domain. In section 4, we describe the system architecture and major modules. In section 5 we describe preliminary evaluation results.

2. Domain

Calls For Fire (CFFs) are requests for artillery fire missions from a Forward Observer (FO) team to a Fire Direction Center (FDC). The elements of a CFF are defined in a military manual, [1], which also specifies the order of messages, how confirmations and corrections are handled, and how the dialogue proceeds after the artillery shots are fired. These specifications are not fully explicit, however, and some choices are left to the participants, e.g., when to confirm that a shot has landed. Furthermore, there are often slight variations based on a unit's standard operating procedures. Finally, because this is a training environment, the trainees may make mistakes in protocol, which still must be understood.

Our analysis of CFF dialogues reveals that there are generally three distinct phases to the dialogue. We can see this looking at Figure 1, a sample CFF taken from our evaluation, with out radiobot acting as FDC for a subject FO. In the first phase (utterances 1-6 in Figure 1) the FOs identify themselves and important information about the desired call, including their coordinates, the kind of fire they are requesting, the location of the target, and the kind of target. Each turn by the FO is confirmed by repetition, and any errors in transmission are immediately corrected.

In the second phase of a CFF, (utterances 7-12) the FDC takes dialogue initiative and provides a Message To Observer, which describes the kind of fire that they have decided will be sent, and later messages that the fire has been shot, and that the artillery has stopped firing, or that according to their calculations the fire will land within five seconds. Each of these are confirmed by the FO.

In the third phase, (utterances 13-20) the FO regains dialogue initiative, and either reports an end to the mission, including the resulting damage to target, or calls for a repetition of the fire, possibly adjusting the location of the fire to compensate for errors in precision and targeting.

At any point in the CFF, the FO or FDC may initiate non-CFF dialogue, providing intelligence or requesting the status of a fire mission. Also, at any point after the first phase the FO may initiate another fire mission. The FO may in this way request multiple fire missions whose second- and third-phase messages are disambiguated with target number IDs that are assigned during the Message to Observer.

1	FO	steel one niner this is gator niner one adjust fire polar over
2	FDC	gator nine one this is steel one nine adjust fire polar out
3	FO	direction five niner four zero distance four eight zero over
4	FDC	direction five nine four zero distance four eight zero out
5	FO	one b m p in the open i c m in effect over
6	FDC	one b m p in the open i c m in effect out
7	FDC	message to observer kilo alpha high explosive four rounds adjust fire target number alpha bravo one zero zero zero over
8	FO	m t o kilo alpha four rounds target number alpha bravo one out
9	FDC	shot over
10	FO	shot out
11	FDC	splash over
12	FO	splash out
13	FO	right five zero fire for effect out over
14	FDC	right five zero fire for effect out
15	FDC	shot over
16	FO	shot out
17	FDC	rounds complete over
18	FO	rounds complete out
19	FO	end of mission one b m p suppressed zero casualties over
20	FDC	end of mission one b m p suppressed zero casualties out

Figure 1: Example Dialogue with Radiobot-CFF

3. Dialogue Model

Our dialogue model was based on analysis of human dialogues in the UTM-JFETS CFF training exercises. Transcripts were used to train a domain-specific language model for our speech recognizer. We also identified a set of dialogue moves and information-bearing parameters as part of an Information State [2] dialogue manager, and annotated the transcripts using this formalization.

Figure 2 shows the dialogue moves and parameters for the first utterance in Figure 1. The Identification move "steel one nine this is gator nine one" has as its two parameters the call signs identifying the FO and FDC, and the Warning Order move has as its two parameters the method of fire being requested and the method that will be used to locate the target.

Identification: steel one nine this is gator niner one
  `fdc_id:` steel one nine
  `fo_id:` gator niner one
Warning Order: adjust fire polar
  `method_of_fire:` adjust fire
  `method_of_location:` polar

Figure 2: Example Dialogue Moves and Parameters

The dialogue moves define a range of FO actions including those providing information, such as Identification, Warning Order, Target Location and Description, and Mission results; those confirming information, such as the Message to Observer and Shot confirmations; and those making requests, such as Radio Checks and Say Again repetition requests.

The dialogue parameters describe aspects of the information being conveyed by the dialogue moves. Each parameter has a dialogue move it is typically associated with: for example fdc_id and fo_id typically occur in the Identification move as in Figure 2, and direction and distance parameters typically occur in the Target Location move as in utterance 3 in Figure 1. However, this association is not strict. Some parameters may occur in multiple dialogue moves: for example, the number_of_enemies and target_type parameters occur in both Target Description and End of Mission dialogue moves (utterances 5 and 19 in Figure 1.)

4. System Implementation

The Radiobot-CFF dialogue system is composed of several pipelined components: Speech Recognition, Interpreter, Dialogue Manager, and Generator. The Radiobot communicates with a Human FO, and an artillery simulator, Firesim XXI, which collects CFF information and sends messages to the UTM simulator, which displays the shells landing in the simulated city.

4.1. Speech Recognizer and Interpreter

The Speech Recognition component is implemented using the SONIC speech recognition system [3] with custom language and acoustic models developed from human-controlled and prototype Radiobot training sessions. The speech recognizer receives audio input and converts it to text which is sent to the Interpreter component.

The Interpreter tokenizes the ASR output into a sequence of words and assigns two separate labels to each word in the sequence: a dialogue move and a parameter label. In order to have high performance in the dialogue, this tagging process has to be robust to ASR errors and speech disfluencies. We have opted to use a statistically-based approach for the interpreter module. For each individual word in the sequence we create a vector of features.

A feature vector consists of binary features indicating whether a particular word occurs in the close proximity to the word being labeled. Specifically, consider the ith word in the sequence. Let $f(j, w)$ be the binary feature indicating that the word w occurs in position j. Then the feature vector v_i is

$$f(j, w) | j = i - 2, ..., i + 2, \forall w \quad (1)$$

Given such a sequence of feature vectors, a Conditional Random Field (CRF) [4, 5] tagger produces a sequence of labels. The tagger was trained using hand-annotated transcribed utterances. Our current version has 1,800 utterances in the training set. We have two separate and independent taggers assigning the dialogue move and parameter labels.

4.2. Dialogue Manager

The Dialogue Manager uses the Information State approach [2] to define a set of relevant information on the status of a dialogue and rules that recognize updates in that information and an ability to produce appropriate utterances given an information state. The dialogue moves and parameters produced by the interpreter component are used by the dialogue manager to update the information state. Other rules are used to determine when to send messages to the simulator and when to generate utterances to the FO. For example, in Figure 1, Utterance 2 is a confirmation of the information received in Utterance 1. Utterance 2 is produced by using two templates that operate on the dialogue information shown in Figure 2: one that reverses the identifying call signs (after doing some scenario-dependent information checking) and one that confirms the Warning Order information. Also the 'adjust fire' information in Figure 2, would be sent to the simulator. The Dialogue Manager tracks what fire mission-related information it has sent to the simulator so far, and once it has received enough information, sends a message to the simulator indicating that a fire mission is to be shot.

As shown in Figure 3, the Information State tracks a variety of information needed to deliver an artillery fire, to respond to the FO, and for input processing. Three key parts of the total Information State are shown. The first seven components are used to determine whether Radiobots has enough information to send a fire. To send a fire, the system must have received a warning order, a target location (which can either be a grid, or a polar direction and distance if the Observer Coordinates is known), and a target description. In the Information State shown, only the target description is missing for the shot to be fired.

```
has warning order? true
has target location? true
    has grid location? false
    has polar direction? true
    has polar distance? true
    has polar observer coord? true
has target descr? false

method of control: adjust fire
method of fire: adjust fire
method of engagement: none given
target type: -
grid value: -
direction value: 5940
distance value: 480
observer coordinate value: 45603595

target number "0"
phase: Info-Gathering
missions active: 0
last method of fire: adjust
```

Figure 3: Information State excerpt after turn 3 in Figure 1

The next eight components in Figure 3 show the details of the requested fire that Radiobots has collected so far: the method of control and fire that were given as part of the Warning Order, the values for the target location, and the scenario-specific observer coordinates. This is the information that will be sent to the simulator.

The final four components are used in disambiguating and handling problematic input. Other components not shown handle information used during fire mission adjustments, during end of mission damage reports, or while processing the update rules.

An innovative aspect of this dialogue manager is that certain of the information state components are editable by a GUI, which enables various levels of operator intervention as described in section 4.4. This allows a greater degree of flexibility in using the system as a training device: rather than being constrained by a dialogue system that the trainer cannot influence, the trainer is free to intervene when a learning opportunity presents itself. In this way, the Radiobot can handle the routine dialogue tasks, freeing up the trainer to focus on such learning opportunities, or assessment, or potentially the supervision of multiple simultaneous training sessions.

4.3. Generator and Simulator

The generator uses templates which construct a conforming text string from an information specification. Once the appropriate output is determined, it is sent to the user in the form of sound clips (in the case this is a known and pre-recorded message), or using a speech synthesizer (if there are no appropriate available sound clips).

The simulator involved is FireSim XXI[1], an artillery fire simulator responsible for realistically modelling friendly and enemy forces, logistics, firing platforms, munitions, and more, for military analysis. In this context it has been adapted to communicate with the graphical and audio components of JFETS-UTM, which shows the results of the explosion, such as the destruction of the enemy and surrounding environment.

4.4. Interactivity

Radiobots allows human intervention through a GUI, which a trainer can use to control various parts of the Dialogue Manager, the Generator, and the Simulator. The trainer can choose to run Radiobots in Fully-Automated, Semi-Automated, or Manual mode, and can switch between these in mid-session.

In Fully- and Semi-Automated modes, Information State components of the requested fire, such as the Warning Order and Target Location values are automatically entered into the GUI, where the trainer can edit them before they are sent to the simulator. Outgoing radio messages to the trainee are similarly editable, appearing as text that the trainer can change, completely discard, or create from scratch.

In Fully-Automated mode the Dialogue Manager also decides when to send the messages to the simulator and to the trainee; in Semi-Automated mode the trainer makes this decision. In Manual mode the trainer enters all information and decides when to send it. In this way the trainer can have the Radiobot system handle as much or as little of the dialogue as is necessary.

5. Evaluation

To evaluate the system we used volunteers from the United States Field Artillery School at Ft Sill, Oklahoma. After initial briefings, trainees conducted two fire missions each. We tested them in

[1] http://sill-www.army.mil/blab/sims/FireSimXXI.htm

either Fully-Automated, Semi-Automated, or Manual conditions, tracking performance of the individual system components, task completion and similar performance measures, as well as user satisfaction measured by responses to a post-training survey.

Our goal was to produce a system that was usable by trainers in the Semi-Automated mode, and was able to handle most dialogues in the Fully-Automated mode, preferrably performing as well as the Manual control condition. As it turned out, the various components of our system performed well, and the system was comparable to the human control condition and successfully completed most fire missions.

5.1. System Performance

In offline studies of preliminary data, the ASR had a 2.3% WER for male American military speakers, the interpreter had a 93% accuracy for dialogue acts and 97% accuracy for dialogue parameters., and the dialogue manager had a 98% accuracy in updating the Information State appropriately.

During the actual evaluation, the combined Speech Recognition and Interpreter components achieved scores shown in Table 1. Over the course of an entire dialogue, the system's interpretation of an FO's utterance was compared to a hand-coded interpretation and evaluated in three ways. First, as a comparison between unique moves: for example, Figure 2 contains the two unique moves Identification and Warning Order. Second, as slot-value pair comparisons: for example, Figure 2 contains a pair (fdc-id, steel one nine). Finally, as triples, where Figure 2 contains a triple such as (ID, fdc-id, steel one nine).

Table 1: *Speech Recognition and Interpreter performance.*

Comparison Type	F-Score
Unique Move	0.91
Unique Parameter	0.88
Slot-Value Move	0.78
Slot-Value Parameter	0.76
Word-Move-Parameter Triple	0.80

5.2. Comprehension and Task Completion

The first two items in Table 2 show results from the post-training survey. These are median scores of on a 10-point scale to the question of "How well could you understand the radio operator?" (You Understood FDC?) and "How well do you think the radio operator understood you?" (FDC Understood You?), where Radiobots performed within 1-2 points of the Human control condition.

Table 2: *Comprehensibility.*

Survey Question	Human	Semi-Auto	Fully-Auto
You Understood FDC?	9	8	8
FDC Understood You?	9	8	7

The final question was to what extent the system would be able to handle a CFF dialogue without human intervention. As shown in the last item in Table 3, the human control condition achieved the baseline of 100% task completion rate. The Semi-Automated condition's 97.5% completion rate reflects one incomplete mission. The Fully-Automated condition's 85.9% completion rate reflects 10 fire missions that were not completed; of these, 5 required that a human set the system to Semi-Automated mode and take control of the session for at least one turn before the fire mission could be completed, and the other 5 were not completed because of system breakdowns, usually in the simulator or graphical environment.

Table 3: *Task Completion.*

Measure	Human	Semi-Auto	Fully-Auto
Task Completion	100%	97.5%	85.9%

6. Conclusions

We have presented Radiobots, a dialogue system for automating the dialogue and simulation components of a military training environment. We have also described an approach to describing and managing radio-based military dialogue, focused on a particular domain, and an information-state dialogue manager with a novel system of editable information components. Evaluation results show our system performs well at automating radio operations for military training tasks.

7. Acknowledgements

This work has been sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

We would like to thank Charles Hernandez and Janet Sutton of the Army Research Laboratory, and Bill Millspaugh and the Depth & Simultaneous Attack Battle Lab in Fort Sill, Oklahoma, for their efforts on this project.

8. References

[1] Department of the Army. 2001. Tactics, techniques and procedures for observed fire and fire support at battalion task force and below. Technical Report FM 3- 09.30 (6-30), Department of the Army.

[2] Staffan Larsson and David Traum. 2000. Information state and dialogue management in the TRINDI dialogue move engine toolkit. Natural Language Engineering, 6:323340, September. Special Issue on Spoken Language Dialogue System Engineering.

[3] Bryan Pellom. 2001. Sonic: The university of colorado continuous speech recognizer. Technical Report TRCSLR-2001-01, University of Colorado.

[4] F. Sha and F. Pereira. 2003. Shallow parsing with conditional random fields.

[5] Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.