Automated Chat Generator

Bryan Williams[1], David C. Uthus[2], and David W. Aha[1]

[1]Navy Center for Applied Research in Artificial Intelligence;
Naval Research Laboratory (Code 5514); Washington, DC 20375
[2]NRC/NRL Postdoctoral Fellow
Washington, DC 20375
first.surname@nrl.navy.mil | david.uthus.ctr@nrl.navy.mil

10 August 2012

Abstract. This document summarizes work towards the development of an automated chat generator, whose components are summarized in Figure 3. Our goal is that this will automatically generate chat that (slightly) resembles Navy Combat Information Center (CIC) chat, such that the resulting data can be used in a 5514 project that concerns automated chat highlighting and chat summarization.

1. User interface

The user interface for the chat generator is quite simple, especially since the program is run from a command line. A simple Graphical User Interface (GUI) is displayed to the user showing the relative positions of a chat scenario's ships and aircraft; see Figure 1 for an example. The user can specify the mission type, number of events, time ratio, and number of US ships by defining these details in a text file entitled "userinput.txt". The user does have the option of specifying none, some, or all of these state elements by

defining some values and inputting "Random" for others, which generates a value between two reasonable bounds (for the latter three variables). The possible inputs for mission types are *Antiterrorism Force Protection* (AT/FP), *Non-combatant Evacuation Operation* (NEO), *Intelligence, Surveillance, and Reconnaissance* (ISR), and *Random*. The possible input for the number of events can be any positive integer or *Random*, and the permitted input for the time ratio, which refers to the ratio between the simulation time and real time, can be any positive integer or *Random*. The possible input for the number of US ships can be any positive integer or

Figure 1: A screenshot of the GUI displayed to the user during runtime. US ships are shown as green semicircles, and their radar range is denoted by the large green circle around them. Enemy vessels are denoted in red, with ships as downward-facing triangles and aircraft upward-facing.

Random, and further attributes of the ships (*name*, *X-coordinate*, *Y-coordinate*, *bearing*, *radar range*, and *speed*) can be specified in the subsequent lines (if the user did not enter "Random" for the number of ships). All of these ship-specific attributes can also be left to chance except for the *name*, which must be specified. Figure 2 shows a sample user input in userinput.txt. All of these state elements and their roles in the chat generator are further described in Section 3.

The first ship entered on the user input is given the role of lead ship. If no value is given because the number of US ships was randomized, the name of the lead ship defaults to *ALPHA*. The lead ship is the only one authorized to give commands, and the chat log generated is from the perspective of this ship.

```
Mission: Force Protection
Number of events: Random
Time Ratio: 8
Number of US ships: 5
ALPHA, 250, 250, Random, Random, Random
BRAVO, Random, Random, Random, 150, Random
CHARLIE, Random, 50, Random, Random, Random
DELTA, 100, Random, 0, Random, Random
ECHO, Random, Random, Random, Random, 30
```

Figure 2: A sample user input. The user can completely customize the simulator, specifying some values and leaving others to chance.

At the end of the program's run, the chat log is saved in a text file entitled "chats.txt". This file will be overwritten upon the next execution of the program. Each execution of the program generates a unique chat log.

2. Logic

The chat generator itself consists of an Event Generator and a State Updater. Each mission comes with a unique starting and final event which introduce and close out the mission. However, for the remainder of the program, events are chosen from a mission-specific list of possible events.

At the beginning of this stage of the program, each event in this list is given an equal probability of being chosen. However, once an event is chosen, its probability decreases by half of its current value and the "lost" probability is equally distributed among all other events. This process ensures that the probabilities still sum to one. In this way, the probabilities are constantly being redistributed among the events to maximize variation in events chosen. No hard limits have been set into place on this system because it is naturally self-regulating. The possible events consist of *Update*, *Weak ID* (where the track is unknown), *ID*, *Complication*, *Locate*, and *Follow*. All are described in more detail in Section 3.

The temporal spacing between the events is determined somewhat randomly. Pauses are classified as either short or long depending on whether the conversation break is between two
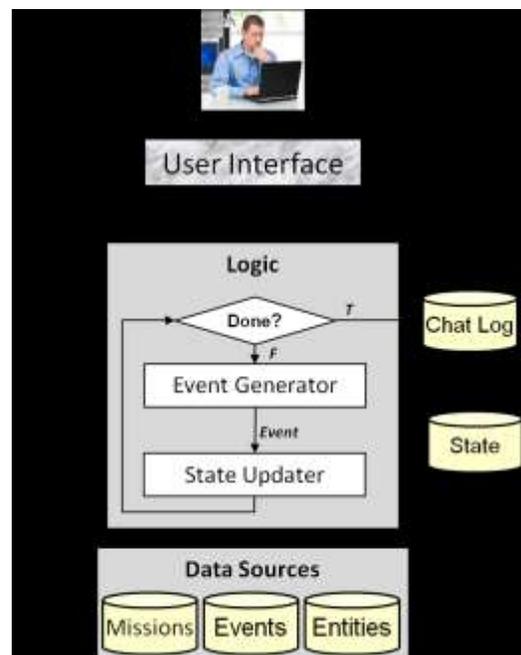

Figure 3: A diagram showing the framework of the logic for the chat generator.

speakers in an ongoing conversation or a break between conversation topics. Each pause category contains a range of possible time intervals, and the exact value within each category is chosen randomly.

The general logic behind the simulator, as outlined in Figure 3, consists of the Event Generator picking an event, the event generating chat messages and updating the chat log, and the event communicating what changes need to be made regarding the state to the State Updater, which has direct access to all the entities and their attributes.

The State Updater also updates the GUI to reflect the movement of the ships. For instance, if the event chosen is a command which tells one ship to follow an enemy ship, the State Updater ensures that the ship's position over time reflects this command by getting closer and closer to the enemy ship's position over time. The Event Generator interacts heavily with the Events database and the State Updater interacts heavily with the Entity database; both are described in more detail in Section 3.

The conversation starters and repliers are chosen somewhat randomly. Certain events are restricted by convention, but others are free to be started by any officer. For instance, only the lead TAO can give orders (issue a Follow command), but any ship can ID another.

The output consists of a randomized chat log such that each execution of the Automated Chat Generator yields a unique chat log. The format of the chat messages is shown in Figure 4.
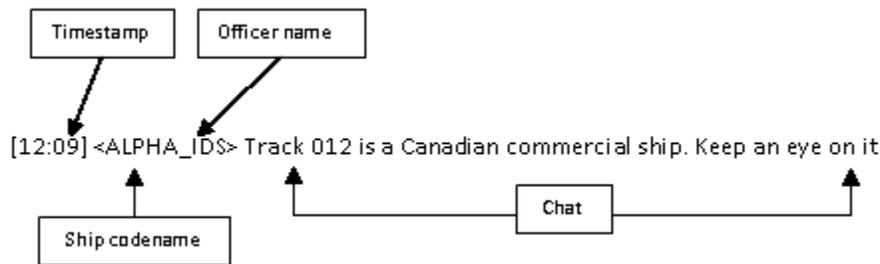

Figure 4: An example chat message with its component parts labeled.

3. Data Sources

The entities, which are objects of the state, can be of type aircraft or ship. Every entity has a variety of attributes, as shown in Figure 5. The State Updater can access these attributes and modify them according to what events are selected by the Event Generator. Every US entity also comes with at least one officer who serves as the connection between the actions on the GUI and the chat messages produced. There are two types of officers: Tactical Action Officers (TAO) and Identification Supervisors (IDS). Every ship has officers of both types, but only the two officers on the lead ship make an appearance

Entities	Events	Missions
• Country	•Update	•Antiterrorism Force Protection (AT/FP)
• Location	•ID	
• Type	oTrack is known	•Intelligence, Surveillance, Reconnaissance (ISR)
o Ship	oTrack is unknown	
o Aircraft	•Complication	•Non-Combatant Evacuation Operations (NEO)
• Bearing	•Locate	
• Radar Range	•Follow	
• Speed		

Figure 5: The three types of data sources used by the Automated Chat Generator.

in the chat room because the interaction between the IDS and TAO of non-lead ships are not shown. Therefore, all chats are sent by TAOs except those chats sent by the IDS of the lead ship.

There are three possible mission types: *Antiterrorism Force Protection* (AT/FP), which consists of a general defense operation, *Intelligence, Surveillance, Reconnaissance* (ISR), which consists of a general surveillance operation, and *Non-Combatant Evacuation Operation* (NEO), which consists of a general rescue operation. Each mission type has its own unique starting and final events, but the entire set of intermediate events consists of *Update*, *Weak ID* (where the track is unknown), *ID*, *Complication*, *Locate*, and *Follow*. The *Update* event consists of the TAO requesting updates from other officers and their replies. The *ID* event consists of an appropriate IDS reporting the presence of a foreign ship or aircraft. In this event, the track can be known or unknown. When the track is unknown, the IDS reports that they have received intelligence that a ship or aircraft from a particular country is in the area, but the exact location in unknown. When the track is known, the enemy appears on the radar as the chat is generated. The *Complication* event consists of one of the officers commenting on a difficulty that has arisen in the mission. Possible complications include an oil spill, rough waters, strong winds, and a thunderstorm. The *Follow* event consists of the TAO on the lead ship ordering another randomly chosen ship to follow one of the enemy ships.

Each event involves 2-3 chats. An event is generated from a set of possible chats, with different sets for conversation initiators and replies. Some events also have a third set of possible chats for conversation initiator responses. These chats differ in wording, but convey the same message, which drastically increases the variety of chat logs that can be generated.

4. Current Status and Future Work

This Automated Chat Generator is currently capable of taking valid user input, constructing the state, displaying the state in a GUI, generating chat, and writing it to a text file. An example of an entire chat conversation can be found in the appendix. However, a variety of extensions and modifications could be made to generate more comprehensive chat logs. For example, a prominent change would be to integrate WordNet[1] with the chat generator. WordNet is a simple lexical resource that organizes nouns, adjectives, adverbs, and verbs into sets of cognitive synonyms (synsets). WordNet could be utilized so that when a specific chat is chosen, some of its words could be (carefully) selected and replaced with one of their synonyms, allowing for further variation between chats while still preserving meaning. This would drastically increase the robustness of each event's set of possible chats.

Another important extension addresses more long-term goals, chat highlighting. Ideally, the user should be able to input a word or small list of keywords that are of particular importance. The program should take these words, generate similar words, and highlight all of these within the chat log. For instance, if the user chose the word "aircraft", "aircraft" should be highlighted in the output chat log along with other words such as "jet" and even specific names of aircraft like "C-130 Hercules". Small progress

---

[1] http://wordnet.princeton.edu/

has been made towards this goal, as the program contains commented stubs of methods that would support the highlighting process.

The differences between missions could also be further enhanced. Currently, while each mission does have unique starting and final events, they share all other events, hampering variety. The program makes it very easy to create additional events, so customizing each mission's set of possible events so that there is only a small overlap between missions is a matter of creativity, not possibility.

The aircraft aspect of the simulator could use some fleshing out. Currently, although the program supports both types, only enemy aircraft, not US aircraft, make an appearance. The US ships are able to ID enemy aircraft and see them on their radar but not interact with them in any other manner. Ideally, US aircraft would be present from the beginning of the mission, would generate chats, and could interact with enemy aircraft through ID and Follow events.

Finally, more work could also be done in increasing the realism of the chat messages by involving misspellings, abbreviations, acronyms, and other typical online chat behavior. Right now, the program does support interruptions (one officer starting a conversation about a new event before another one has finished, with the original conversation finishing shortly thereafter), but implementing other chat practices would significantly enhance authenticity.

5. Acknowledgements

6. Appendix

Below is a flowchart outlining the logic behind all of the classes used in the Automated Chat Generator. The arrow connection can mean one class simply uses another or can indicate a subclass-superclass relationship.
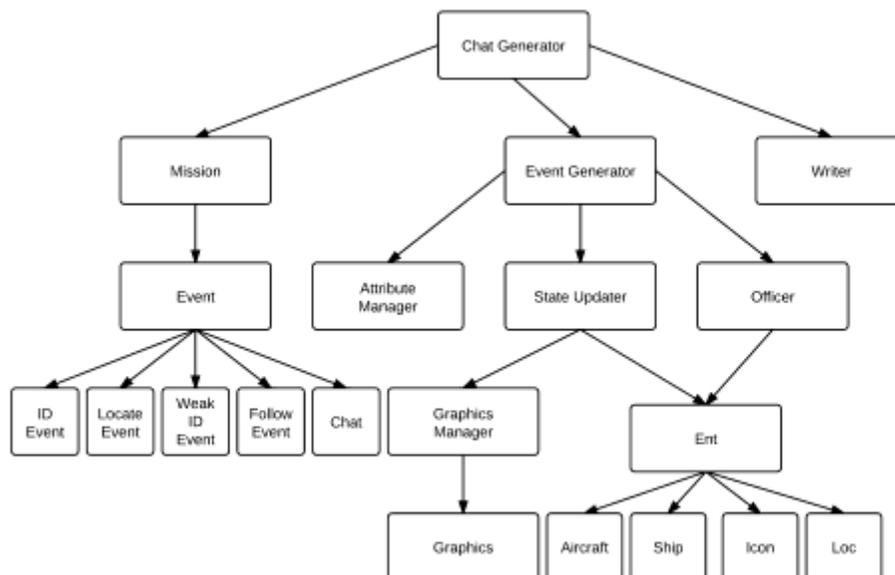
Below is a sample of user input and chat output.

Input:

```
Mission: ISR
Number of events: 15
Time Ratio: 8
Number of US ships: 4
ALPHA, 250, 250, Random, Random, Random
BRAVO, Random, Random, Random, 150, Random
CHARLIE, Random, 50, Random, Random, Random
DELTA, 100, Random, 0, Random, Random
```

Output:

```
[00:05] <ALPHA_TAO> Our objective is simple ISR. Let's keep it simple
[00:11] <ALPHA_IDS> ok
[00:18] <BRAVO_TAO> ok
[00:22] <CHARLIE_TAO> c
[00:27] <DELTA_TAO> Will keep an eye out
[01:04] <BRAVO_TAO> Received intel from Sao Paulo. South Korean vessels are
operating in this region.
[01:11] <ALPHA_TAO> BRAVO_TAO, update me if you do get IDs
[02:17] <CHARLIE_TAO> New intelligence from Guantanamo says Norwegian aircraft
are in the region. Keep an eye out.
[02:26] <CHARLIE_TAO> we need to keep an eye out
[02:27] <ALPHA_TAO> CHARLIE_TAO, if you get the IDs be sure to update me
[04:13] <ALPHA_TAO> All stations, locations?
[04:21] <DELTA_TAO> (100, 097)
[04:27] <CHARLIE_TAO> I'm at (359, 056)
[04:32] <BRAVO_TAO> I'm at (371, 272)
[04:39] <ALPHA_TAO> c
[05:58] <ALPHA_TAO> Everyone, Please let me know how you're doing
[06:13] <DELTA_TAO> Just keeping an eye out
[06:17] <BRAVO_TAO> All fine. Let me know if you need anything
[06:21] <ALPHA_IDS> Everything's fine, sir
[06:23] <CHARLIE_TAO> All systems functioning fine sir
[08:06] <ALPHA_TAO> Nice. Be sure to report back with any new news
[08:15] <DELTA_TAO> c
[08:19] <CHARLIE_TAO> c
[08:21] <ALPHA_IDS> c
[08:24] <BRAVO_TAO> c
[08:27] <ALPHA_TAO> All, please send me an update on your location
[08:33] <CHARLIE_TAO> (359, 056)
[08:38] <BRAVO_TAO> Current position is (371, 272)
[08:43] <DELTA_TAO> (100, 097)
[08:48] <ALPHA_TAO> c
[10:26] <DELTA_TAO> Stay alert, a Polish military aircraft has been spotted.
IDed as Track 019 by DELTA
[10:32] <ALPHA_TAO> DELTA_TAO, c
[10:50] <BRAVO_TAO> Received intel from Naples. Israeli vessels are operating
in this region.
[11:05] <BRAVO_TAO> we need to keep an eye out
[11:08] <ALPHA_TAO> BRAVO_TAO, let me know if you do get IDs
[11:28] <ALPHA_TAO> All, requesting system status updates
[11:33] <ALPHA_TAO> mission going well?
[11:34] <DELTA_TAO> No problems here
[11:37] <BRAVO_TAO> Everything's fine, sir
[11:44] <ALPHA_IDS> All fine here
[11:51] <CHARLIE_TAO> All fine here
[12:53] <ALPHA_TAO> Excellent. Make sure to keep me updated
```

```
[13:07] <BRAVO_TAO> c
[13:10] <CHARLIE_TAO> c
[13:16] <ALPHA_IDS> c
[13:23] <DELTA_TAO> c
[13:28] <ALPHA_TAO> All, please let me know where you are
[13:38] <CHARLIE_TAO> My coordinates are (359, 056)
[13:41] <BRAVO_TAO> (371, 272)
[13:44] <DELTA_TAO> (100, 097)
[13:49] <ALPHA_TAO> All, requesting system status updates
[13:57] <ALPHA_TAO> c
[14:03] <ALPHA_IDS> No problems here
[14:09] <CHARLIE_TAO> All systems functioning fine sir
[14:15] <BRAVO_TAO> No problems here
[14:21] <DELTA_TAO> All systems functioning fine sir
[15:59] <ALPHA_TAO> Good, keep me up to date if anything changes
[16:10] <BRAVO_TAO> c
[16:17] <DELTA_TAO> c
[16:19] <ALPHA_IDS> c
[16:23] <CHARLIE_TAO> c
[16:30] <CHARLIE_TAO> waters are rough. Not doing us any favors
[16:37] <ALPHA_TAO> don't worry too much about it
[18:08] <DELTA_TAO> these weather conditions are far from favorable...
[18:14] <DELTA_TAO> we need to able to keep an eye on everyone
[18:18] <ALPHA_TAO> this is going to take a little extra effort from all of us
[18:41] <DELTA_TAO> Just received new intel. Irish planes in the area. Stay on
the lookout.
[18:51] <ALPHA_TAO> DELTA_TAO, keep me posted on the ID status
[20:22] <ALPHA_TAO> We've identified everything. Recon mission complete
[20:27] <DELTA_TAO> Nice
[20:34] <CHARLIE_TAO> c
[20:39] <ALPHA_IDS> Wasn't too bad
[20:43] <BRAVO_TAO> c
```