Multi-Modal Annotation of Quest Games in Second Life

Sharon Gower Small, Jennifer Stromer-Galley and Tomek Strzalkowski
ILS Institute
State University of New York at Albany
Albany, NY 12222
small@albany.edu, jstromer@albany.edu, tomek@albany.edu

Abstract

We describe an annotation tool developed to assist in the creation of multimodal action-communication corpora from on-line massively multi-player games, or MMGs. MMGs typically involve groups of players (5-30) who control their avatars[1], perform various activities (questing, competing, fighting, etc.) and communicate via chat or speech using assumed screen names. We collected a corpus of 48 group quests in Second Life that jointly involved 206 players who generated over 30,000 messages in quasi-synchronous chat during approximately 140 hours of recorded action. Multiple levels of coordinated annotation of this corpus (dialogue, movements, touch, gaze, wear, etc) are required in order to support development of automated predictors of selected real-life social and demographic characteristics of the players. The annotation tool presented in this paper was developed to enable efficient and accurate annotation of all dimensions simultaneously.

1 Introduction

The aim of our project is to predict the real world characteristics of players of massively-multiplayer online games, such as Second Life (SL). We sought to predict actual player attributes like age or education levels, and personality traits including leadership or conformity. Our task was to do so using only the behaviors, communication, and interaction among the players produced during game play. To do so, we logged all players' avatar movements, "touch events" (putting on or taking off clothing items, for example), and their public chat messages (i.e., messages that can be seen by all players in the group). Given the complex nature of interpreting chat in an online game environment, we required a tool that would allow annotators to have a synchronized view of both the event action as well as the chat utterances. This would allow our annotators to correlate the events and the chat by marking them simultaneously. More importantly, being able to view game events enables more accurate chat annotation; and conversely, viewing chat utterances helps to interpret the significance of certain events in the game, e.g., one avatar following another. For example, an exclamation of: "I can't do it!" could be simply a response (rejection) to a request from another player; however, when the game action is viewed and the speaker is seen attempting to enter a building without success, another interpretation may arise (an assertion, a call for help, etc.).

The Real World (RW) characteristics of SL players (and other on-line games) may be inferred to varying degrees from the appearance of their avatars, the behaviors they engage in, as well as from their on-line chat communications. For example, the avatar gender generally matches the gender of the owner; on the other hand, vocabulary choices in chat are rather poor predictors of a player's age, even though such correlation is generally seen in real life conversation.

Second Life[2] was the chosen platform because of the ease of creating objects, controlling the play environment, and collecting players' movement, chat, and other behaviors. We generated a corpus of chat and movement data from 48 quests comprised of 206 participants who generated over 30,000

[1] All avatar names seen in this paper have been changed to protect players' identities.

[2] An online Virtual World developed and launched in 2003, by Linden Lab, San Francisco, CA. http://secondlife.com

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 171–179,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

messages and approximately 140 hours of recorded action. We required an annotation tool to help us efficiently annotate dialogue acts and communication links in chat utterances as well as avatar movements from such a large corpus. Moreover, we required correlation between these two dimensions of chat and movement since movement and other actions may be both causes and effects of verbal communication. We developed a multimodal event and chat annotation tool (called RAT, the Relational Annotation Tool), which will simultaneously display a 2D rendering of all movement activity recorded during our Second Life studies, synchronized with the chat utterances. In this way both chat and movements can be annotated simultaneously: the avatar movement actions can be reviewed while making dialogue act annotations. This has the added advantage of allowing the annotator to see the relationships between chat, behavior, and location/movement. This paper will describe our annotation process and the RAT tool.

2 Related Work

Annotation tools have been built for a variety of purposes. The CSLU Toolkit (Sutton et al., 1998) is a suite of tools used for annotating spoken language. Similarly, the EMU System (Cassidy and Harrington, 2001) is a speech database management system that supports multi-level annotations. Systems have been created that allow users to readily build their own tools such as AGTK (Bird et al., 2001). The multi-modal tool DAT (Core and Allen, 1997) was developed to assist testing of the DAMSL annotation scheme. With DAT, annotators were able to listen to the actual dialogues as well as view the transcripts. While these tools are all highly effective for their respective tasks, ours is unique in its synchronized view of both event action and chat utterances.

Although researchers studying online communication use either off-the shelf qualitative data analysis programs like Atlas.ti or NVivo, a few studies have annotated chat using custom-built tools. One approach uses computer-mediated discourse analysis approaches and the Dynamic Topic Analysis tool (Herring, 2003; Herring & Nix; 1997; Stromer-Galley & Martison, 2009), which allows annotators to track a specific phenomenon of online interaction in chat: topic shifts during an interaction. The Virtual Math Teams project (Stahl, 2009) created a tool that allowed for the simultaneous playback of messages posted to a quasi-synchronous discussion forum with whiteboard drawings that student math team members used to illustrate their ideas or visualize the math problem they were trying to solve (Çakir, 2009).

A different approach to data capture of complex human interaction is found in the AMI Meeting Corpus (Carletta, 2007). It captures participants' head movement information from individual head-mounted cameras, which allows for annotation of nodding (consent, agreement) or shaking (disagreement), as well as participants' locations within the room; however, no complex events involving series of movements or participant proximity are considered. We are unaware of any other tools that facilitate the simultaneous playback of multi-modes of communication and behavior.

3 Second Life Experiments

To generate player data, we rented an island in Second Life and developed an approximately two hour quest, the Case of the Missing Moonstone. In this quest, small groups of 4 to 5 players, who were previously unacquainted, work their way together through the clues and puzzles to solve a murder mystery. We recruited Second Life players in-game through advertising and setting up a shop that interested players could browse. We also used Facebook ads, which were remarkably effective.

The process of the quest experience for players started after they arrived in a starting area of the island (the quest was open only to players who were made temporary members of our island) where they met other players, browsed quest-appropriate clothing to adorn their avatars, and received information from one of the researchers. Once all players arrived, the main quest began, progressing through five geographic areas in the island. Players were accompanied by a “training sergeant”, a researcher using a robot avatar, that followed players through the quest and provided hints when groups became stymied along their investigation but otherwise had little interaction with the group.

The quest was designed for players to encounter obstacles that required coordinated action, such as all players standing on special buttons to activate a door, or the sharing of information between players, such as solutions to a word puzzle, in order to advance to the next area of the quest (Figure 1).

Slimy Roastbeef: *“who’s got the square gear?”*

Kenny Superstar: *“I do, but I’m stuck”*

Slimy Roastbeef: *“can you hand it to me?”*

Kenny Superstar: *“i don’t know how”*

Slimy Roastbeef: *“open your inventory, click and drag it onto me”*

Figure 1: Excerpt of dialogue during a coordination activity

Quest activities requiring coordination among the players were common and also necessary to ensure a sufficient degree of movement and message traffic to provide enough material to test our predictions, and to allow us to observe particular social characteristics of players. Players answered a survey before and then again after the quest, providing demographic and trait information and evaluating other members of their group on the characteristics of interest.

3.1 Data Collection

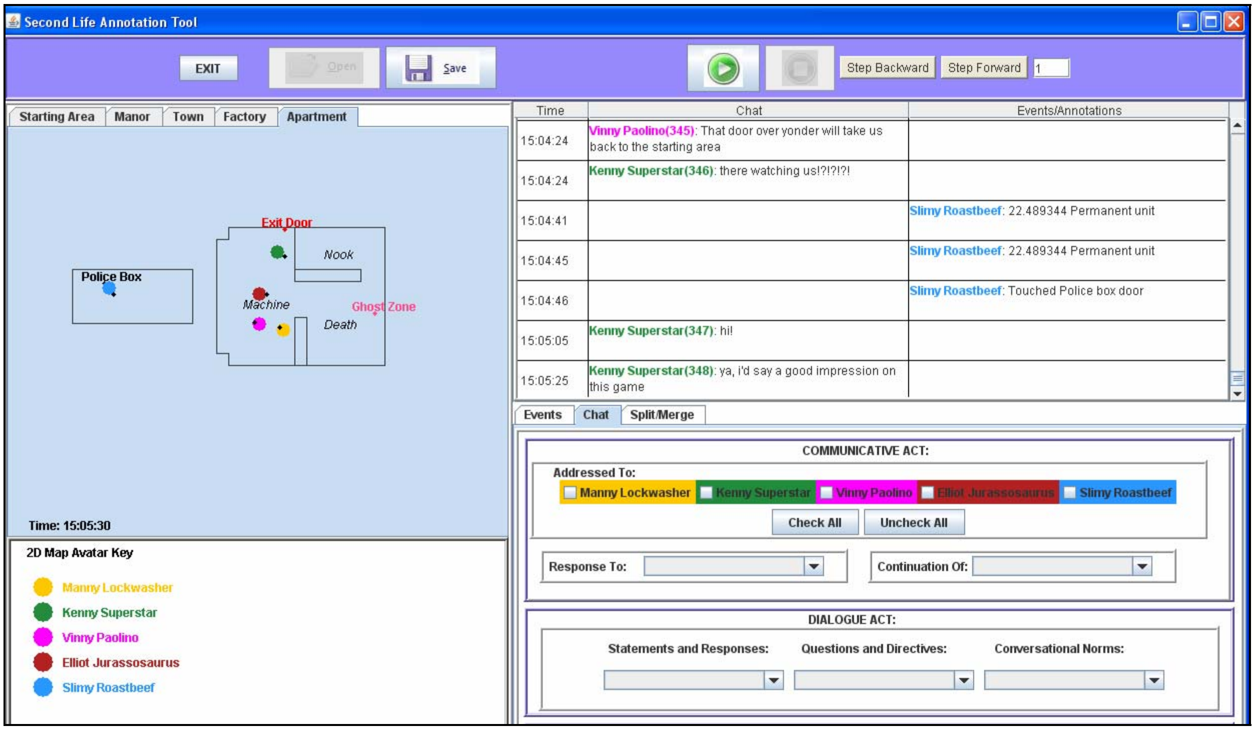
We recorded all players’ avatar movements as they purposefully moved avatars through the virtual spaces of the game environment, their public chat, and their “touch events”, which are the actions that bring objects out of player inventories, pick up objects to put in their inventories, or to put objects, such as hats or clothes, onto the avatars, and the like. We followed Yee and Bailenson’s (2008) technical approach for logging player behavior. To get a sense of the volume of data generated, 206 players generated over 30,000 messages into the group’s public chat from the 48 sessions. We compiled approximately 140 hours of recorded action. The avatar logger was implemented to record each avatar’s location through their (x,y,z) coordinates, recorded at two second intervals. This information was later used to render the avatar’s position on our 2D representation of the action (section 4.1).

4 RAT

The Relational Annotation Tool (RAT) was built to assist in annotating the massive collection of data collected during the Second Life experiments. A tool was needed that would allow annotators to see the textual transcripts of the chat while at the same time view a 2D representation of the action. Additionally, we had a textual transcript for a select set of events: touch an object, stand on an object, attach an object, etc., that we needed to make available to the annotator for review.

These tool characteristics were needed for several reasons. First, in order to fully understand the communication and interaction occurring between players in the game environment and accurately annotate those messages, we needed annotators to have as much information about the context as possible. The 2D map coupled with the events information made it easier to understand. For example, in the quest, players in a specific zone, encounter a dead, maimed body. As annotators assigned codes to the chat, they would sometimes encounter exclamations, such as “ew” or “gross”. Annotators would use the 2D map and the location of the exclaiming avatar to determine if the exclamation was a result of their location (in the zone with the dead body) or because of something said or done by another player. Location of avatars on the 2D map synchronized with chat was also helpful for annotators when attempting to disambiguate communicative links. For example, in one subzone, mad scribblings are written on a wall. If player A says “You see that scribbling on the wall?” the annotator needs to use the 2D map to see who the player is speaking to. If player A and player C are both standing in that subzone, then the annotator can make a reasonable assumption that player A is directing the question to player C, and not player B who is located in a different subzone. Second, we annotated coordinated avatar movement actions (such as following each other into a building or into a room), and the only way to readily identify such complex events was through the 2D map of avatar movements.

The overall RAT interface, Figure 2, allows the annotator to simultaneously view all modes of representation. There are three distinct panels in this interface. The left hand panel is the 2D representation of the action (section 4.1). The upper right hand panel displays the chat and event transcripts (section 4.2), while the lower right hand portion is reserved for the three annotator sub-panels (section 4.3).



Time: 15:05:30

Time	Chat	Events/Annotations
15:04:24	Vinny Paolino(345): That door over yonder will take us back to the starting area	
15:04:24	Kenny Superstar(346): there watching us!?!?!?!	
15:04:41		Slimy Roastbeef: 22.489344 Permanent unit
15:04:45		Slimy Roastbeef: 22.489344 Permanent unit
15:04:46		Slimy Roastbeef: Touched Police box door
15:05:05	Kenny Superstar(347): hi!	
15:05:25	Kenny Superstar(348): ya, i'd say a good impression on this game	

Figure 2: RAT interface

4.1 The 2D Game Representation

The 2D representation was the most challenging of the panels to implement. We needed to find the proper level of abstraction for the action, while maintaining its usefulness for the annotator. Too complex a representation would cause cognitive overload for the annotator, thus potentially deteriorating the speed and quality of the annotations. Conversely, an overly abstract representation would not be of significant value in the annotation process.

There were five distinct geographic areas on our Second Life Island: *Starting Area*, *Mansion*, *Town Center*, *Factory and Apartments*. An overview of the area in Second Life is displayed in Figure 3. We decided to represent each area separately as each group moves between the areas together, and it was therefore never necessary to display more than one area at a time. The 2D representation of the Mansion Area is displayed in Figure 4 below. Figure 5 is an exterior view of the actual Mansion in Second Life. Each area's fixed representation was rendered using Java Graphics, reading in the Second Life (x,y,z) coordinates from an XML data file. We represented the walls of the buildings as connected solid black lines with openings left for doorways. Key item locations were marked and labeled, e.g. *Kitten*, *maid*, *the Idol*, etc. Even though annotators visited the island to familiarize themselves with the layout, many mansion rooms were labeled to help the annotator recall the layout of the building, and minimize error of annotation based on flawed recall. Finally, the exact time of the action that is currently being represented is displayed in the lower left hand corner.

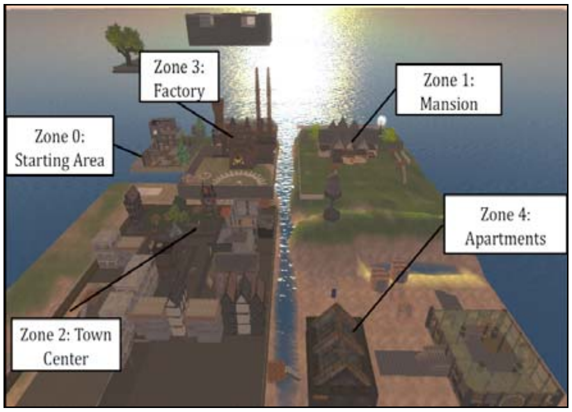


Figure 3: Second Life overview map

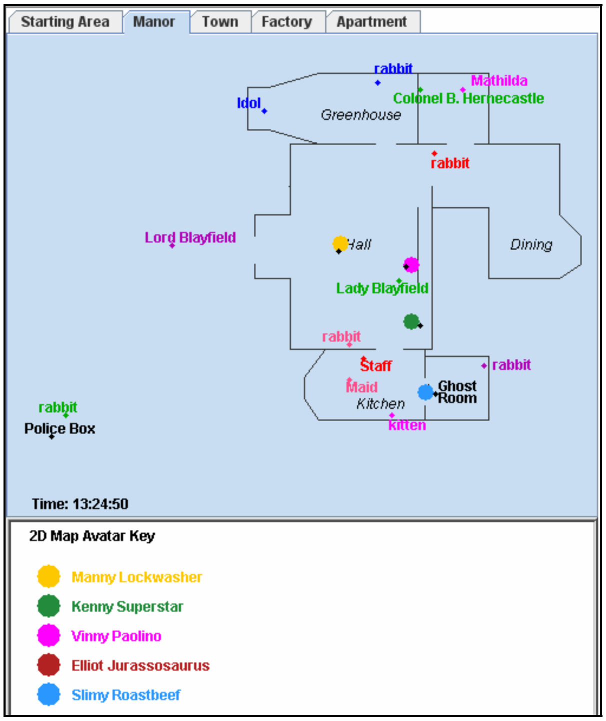


Figure 4: 2D representation of Second Life action inside the Mansion/Manor

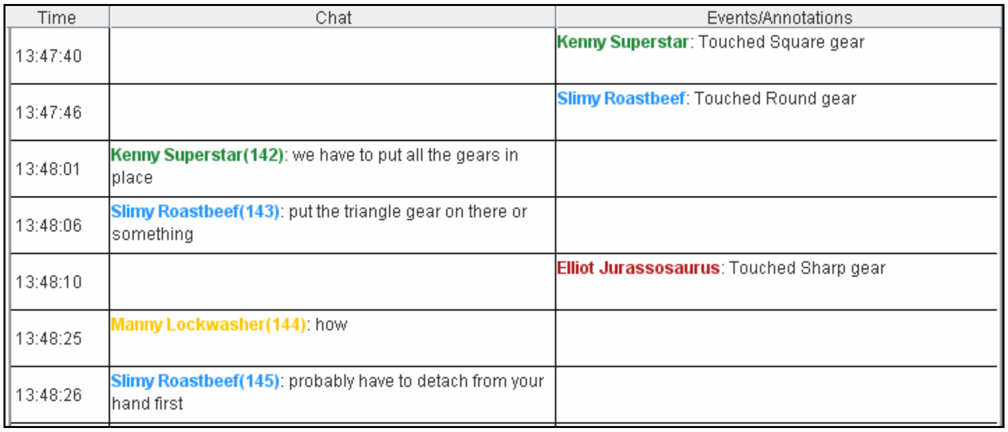
Figure 5: Second Life view of Mansion exterior

Avatar location was recorded in our log files as an (x,y,z) coordinate at a two second interval. Avatars were represented in our 2D panel as moving solid color circles, using the x and y coordinates. A color coded avatar key was displayed below the 2D representation. This key related the full name of every avatar to its colored circle representation. The z coordinate was used to determine if the avatar was on the second floor of a building. If the z value indicated an avatar was on a second floor, their icon was modified to include the number “2” for the duration of their time on the second floor. Also logged was the avatar’s degree of rotation. Using this we were able to represent which direction the avatar was looking by a small black dot on their colored circle.

As the annotators stepped through the chat and event annotation, the action would move forward, in synchronized step in the 2D map. In this way at any given time the annotator could see the avatar action corresponding to the chat and event transcripts appearing in the right panels. The annotator had the option to step forward or backward through the data at any step interval, where each step corresponded to a two second increment or decrement, to provide maximum flexibility to the annotator in viewing and reviewing the actions and communications to be annotated. Additionally, “Play” and “Stop” buttons were added to the tool so the annotator may simply watch the action play forward rather than manually stepping through.

4.2 The Chat & Event Panel

Avatar utterances along with logged Second Life events were displayed in the Chat and Event Panel (Figure 6). Utterances and events were each displayed in their own column. Time was recorded for every utterance and event, and this was displayed in the first column of the Chat and Event Panel. All avatar names in the utterances and events were color coded, where the colors corresponded to the avatar color used in the 2D panel. This panel was synchronized with the 2D Representation panel and as the annotator stepped through the game action on the 2D display, the associated utterances and events populated the Chat and Event panel.



Time	Chat	Events/Annotations
13:47:40		Kenny Superstar: Touched Square gear
13:47:46		Slimy Roastbeef: Touched Round gear
13:48:01	Kenny Superstar(142): we have to put all the gears in place	
13:48:06	Slimy Roastbeef(143): put the triangle gear on there or something	
13:48:10		Elliot Jurassosaurus: Touched Sharp gear
13:48:25	Manny Lockwasher(144): how	
13:48:26	Slimy Roastbeef(145): probably have to detach from your hand first	

Figure 6: Chat & Event Panel

4.3 The Annotator Panels

The Annotator Panels (Figures 7 and 10) contains all features needed for the annotator to quickly annotate the events and dialogue. Annotators could choose from a number of categories to label each dialogue utterance. Coding categories included communicative links, dialogue acts, and selected multi-avatar actions. In the following we briefly outline each of these. A more detailed description of the chat annotation scheme is available in (Shaikh et al., 2010).

4.3.1 Communicative Links

One of the challenges in multi-party dialogue is to establish which user an utterance is directed towards. Users do not typically add addressing information in their utterances, which leads to ambiguity while creating a communication link between users. With this annotation level, we asked the annotators to determine whether each utterance was addressed to some user, in which case they were asked to mark which specific user it was addressed to; was in response to another prior utterance by a different user, which required marking the specific utterance responded to; or a continuation of the user's own prior utterance.

Communicative link annotation allows for accurate mapping of dialogue dynamics in the multi-party setting, and is a critical component of tracking such social phenomena as disagreements and leadership.

4.3.2 Dialogue Acts

We developed a hierarchy of 19 dialogue acts for annotating the functional aspect of the utterance in the discussion. The tagset we adopted is loosely based on DAMSL (Allen & Core, 1997) and SWBD (Jurafsky et al., 1997), but greatly reduced and also tuned significantly towards dialogue pragmatics and away from more surface characteristics of utterances. In particular, we ask our annotators what is the pragmatic function of each utterance within the dialogue, a decision that often depends upon how earlier utterances were classified. Thus augmented, DA tags become an important source of evidence for detecting language uses and such social phenomena as conformity. Examples of dialogue act tags include Assertion-Opinion, Acknowledge, Information-Request, and Confirmation-Request.

Using the augmented DA tagset also presents a fairly challenging task to our annotators, who need to be trained for many hours before an acceptable rate of inter-annotator agreement is achieved. For this reason, we consider our current DA tagging as a work in progress.

4.3.3 Zone coding

Each of the five main areas had a corresponding set of subzones. A subzone is a building, a room within a building, or any other identifiable area within the playable spaces of the quest, e.g. the *Mansion* has the subzones: *Hall*, *Dining Room*, *Kitchen*, *Outside*, *Ghost Room*, etc. The subzone was determined based on the avatar(s) (x,y,z) coordinates and the known subzone boundaries. This additional piece of data allowed for statistical analysis at different levels: avatar, dialogue unit, and subzone.

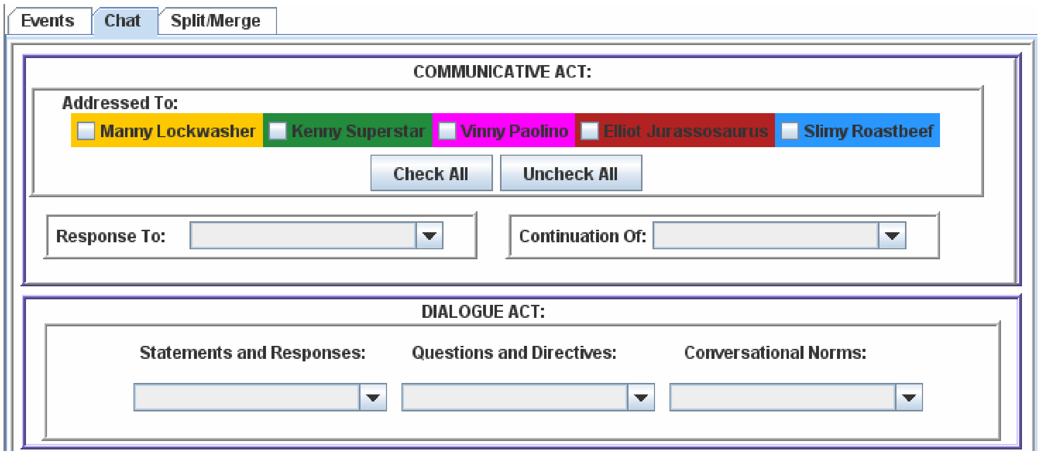


Figure 7: Chat Annotation Sub-Panel

4.3.4 Multi-avatar events

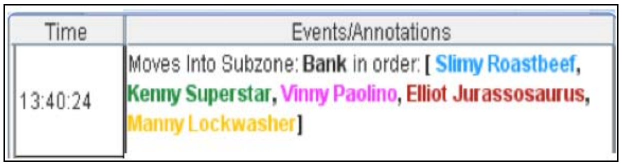
As mentioned, in addition to chat we also were interested in having the annotators record composite events involving multiple avatars over a span of time and space. While the design of the RAT tool will support annotation of any event of interest with only slight modifications, for our purposes, we were interested in annotating two types of events that we considered significant for our research hypotheses. The first type of event was the multi-avatar entry (or exit) into a sub-zone, including the order in which the avatars moved.

Figure 8 shows an example of a “Moves into Subzone” annotation as displayed in the Chat & Event Panel. Figure 9 shows the corresponding series of progressive moments in time portraying entry into the Bank subzone as represented in RAT. In the annotation, each avatar name is recorded in order of its entry into the subzone (here, the Bank). Additionally, we record the subzone name and the time the event is completed[3].

The second type of event we annotated was the “follow X” event, i.e., when one or more avatars appeared to be following one another within a subzone. These two types of events were of particular interest because we hypothesized that players who are leaders are likely to enter first into a subzone and be followed around once inside.

In addition, support for annotation of other types of composite events can be added as needed; for example, group forming and splitting, or certain joint activities involving objects, etc. were fairly common in quests and may be significant for some analyses (although not for our hypotheses).

For each type of event, an annotation subpanel is created to facilitate speedy markup while minimizing opportunities for error (Figure 10). A “Moves Into Subzone” event is annotated by recording the ordinal (1, 2, 3, etc.) for each avatar. Similarly, a “Follows” event is coded as avatar group “A” follows group “B’, where each group will contain one or more avatars.



Time	Events/Annotations
13:40:24	Moves Into Subzone: Bank in order: [Slimy Roastbeef, Kenny Superstar, Vinny Paolino, Elliot Jurassosaurus, Manny Lockwasher]

Figure 8: The corresponding annotation for Figure 9 event, as displayed in the Chat & Event Panel

5 The Annotation Process

To annotate the large volume of data generated from the Second Life quests, we developed an annotation guide that defined and described the annotation categories and decision rules annotators were to follow in categorizing the data units (following previous projects (Shaikh et al., 2010). Two students were hired and trained for approximately 60 hours, during which time they learned how to use the annotation tool and the categories and rules for the annotation process. After establishing a satisfactory level of interrater reliability (average Krippendorff’s alpha of all measures was <0.8. Krippendorff’s alpha accounts for the probability of

[3] We are also able to record the start time of any event but for our purposes we were only concerned with the end time.

chance agreement and is therefore a conservative measure of agreement), the two students then annotated the 48 groups over a four-month period. It took approximately 230 hours to annotate the sessions, and they assigned over 39,000 dialogue act tags. Annotators spent roughly 7 hours marking up the movements and chat messages per 2.5 hour quest session.

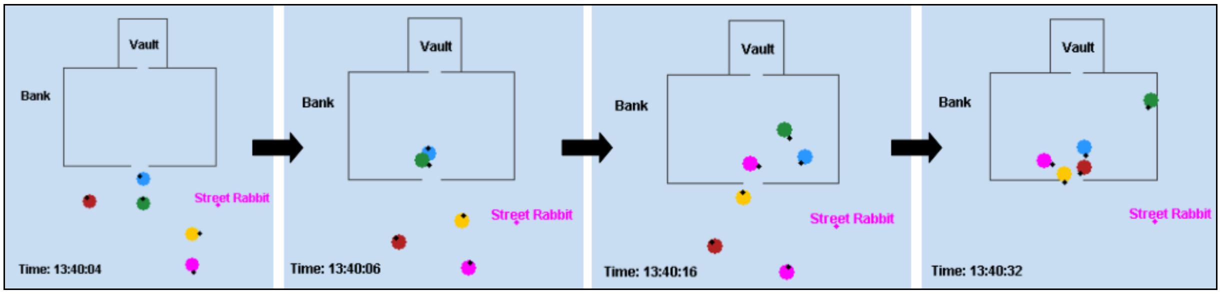
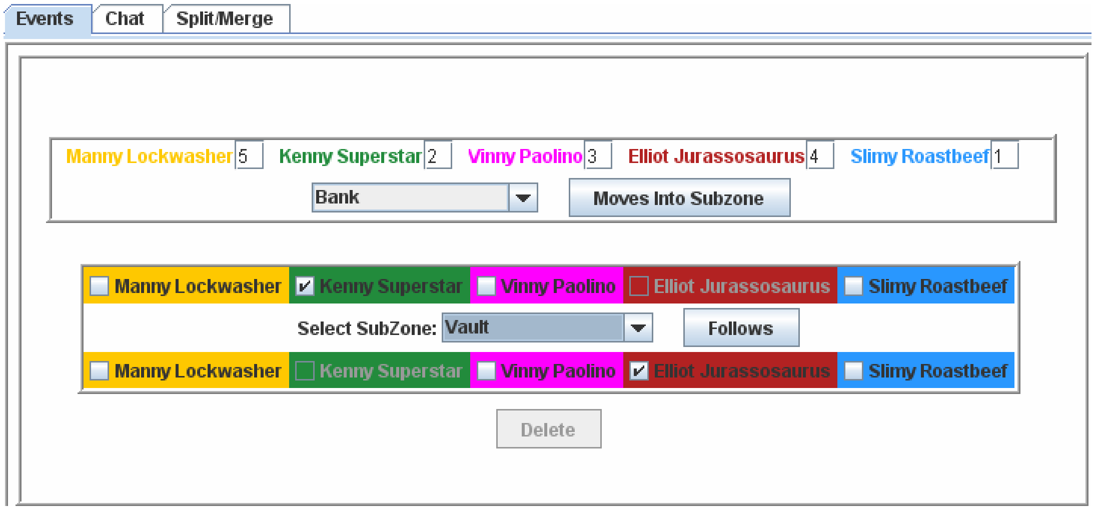


Figure 9: A series of progressive moments in time portraying avatar entry into the Bank subzone



Manny Lockwasher 5 Kenny Superstar 2 Vinny Paolino 3 Elliot Jurassosaurus 4 Slimy Roastbeef 1

Bank ▾ Moves Into Subzone

□ Manny Lockwasher ☑ Kenny Superstar □ Vinny Paolino □ Elliot Jurassosaurus □ Slimy Roastbeef

Select SubZone: Vault ▾ Follows

□ Manny Lockwasher □ Kenny Superstar □ Vinny Paolino ☑ Elliot Jurassosaurus □ Slimy Roastbeef

Figure 10: Event Annotation Sub-Panel, currently showing the "Moves Into Subzone" event from figure 9, as well as: *"Kenny follows Elliot in Vault"*

5.1 The Annotated Corpus

The current version of the annotated corpus consists of thousands of tagged messages including: 4,294 action-directives, 17,129 assertion-opinions, 4,116 information requests, 471 confirmation requests, 394 offer-commits, 3,075 responses to information requests, 1,317 agree-accepts, 215 disagree-rejects, and 2,502 acknowledgements, from 30,535 pre-split utterances (31,801 post-split). We also assigned 4,546 following events.

6 Conclusion

In this paper we described the successful implementation and use of our multi-modal annotation tool, RAT. Our tool was used to accurately and simultaneously annotate over 30,000 messages and approximately 140 hours of action. For each hour spent annotating, our annotators were able to tag approximately 170 utterances as well as 36 minutes of action.

The annotators reported finding the tool highly functional and very efficient at helping them easily assign categories to the relevant data units, and that they could assign those categories without producing too many errors, such as accidentally assigning the wrong category or selecting the wrong avatar. The function allowing for the synchronized playback of the chat and movement data coupled with the 2D map increased comprehension of utterances and behavior of the players during the quest, improving validity and reliability of the results.

Acknowledgements

This research is part of an Air Force Research Laboratory sponsored study conducted by Colorado State University, Ohio University, the University at Albany, SUNY, and Lockheed Martin.

References

Steven Bird, Kazuaki Maeda, Xiaoyi Ma and Haejoong Lee. 2001. annotation tools based on the annotation graph API. In Proceedings of ACL/EACL 2001 Workshop on Sharing Tools and Resources for Research and Education.

M. P. Çakir. 2009. The organization of graphical, narrative and symbolic interactions. In Studying virtual math teams (pp. 99-140). New York, Springer.

J. Carletta. 2007. Unleashing the killer corpus: experiences in creating the multi-everything AMI Meeting Corpus. Language Resources and Evaluation Journal 41(2): 181-190.

Mark G. Core and James F. Allen. 1997. Coding dialogues with the DAMSL annotation scheme. In Proceedings of AAAI Fall 1997 Symposium.

Steve Cassidy and Jonathan Harrington. 2001. Multilevel annotation in the Emu speech database management system. Speech Communication, 33:61-77.

S. C. Herring. 2003. Dynamic topic analysis of synchronous chat. Paper presented at the New Research for New Media: Innovative Research Symposium. Minneapolis, MN.

S. C. Herring and Nix, C. G. 1997. Is “serious chat” an oxymoron? Pedagogical vs. social use of internet relay chat. Paper presented at the American Association of Applied Linguistics, Orlando, FL.

Samira Shaikh, Strzalkowski, T., Broadwell, A., Stromer-Galley, J., Taylor, S., and Webb, N. 2010. MPC: A Multi-party chat corpus for modeling social phenomena in discourse. *Proceedings of the Seventh Conference on International Language Resources and Evaluation.* Valletta, Malta: European Language Resources Association.

G. Stahl. 2009. The VMT vision. In G. Stahl, (Ed.), Studying virtual math teams (pp. 17-29). New York, Springer.

Stephen Sutton, Ronald Cole, Jacques De Villiers, Johan Schalkwyk, Pieter Vermeulen, Mike Macon, Yonghong Yan, Ed Kaiser, Brian Rundle, Khaldoun Shobaki, Paul Hosom, Alex Kain, Johan Wouters, Dominic Massaro, Michael Cohen. 1998. Universal Speech Tools: The CSLU toolkit. Proceedings of the 5th ICSLP, Australia.

Jennifer Stromer-Galley and Martinson, A. 2009. Coherence in political computer-mediated communication: Comparing topics in chat. *Discourse & Communication*, *3,* 195-216.

N. Yee and Bailenson, J. N. 2008. A method for longitudinal behavioral data collection in *Second Life*. *Presence, 17,* 594-596.