JESTER: Joke Entertainment System with Trivia-Emitted Responses

Diana David, Huang Yang, Zion Adika, Hanoch Freund
Zeynep Altinbas, Murtaza Karim, Matthew Marge, Manish Muttreja, Thomas Nasti

Stony Brook University
Computer Science Department
Stony Brook, NY 11794-4400
ddavid@ic.sunysb.edu, yliu@cs.sunysb.edu

Abstract

The Stony Brook Robot Design Team has focused on three main areas of research in the creation of JESTER (The Joke Entertainment System with Trivia-Emitted Responses), our new robot created for the American Association of Artificial Intelligence's (AAAI) Open Interaction Event competition, including navigation, computer vision and user interface / human interaction. The purpose is to create an intelligent machine to entertain people at the AAAI conference in San Jose, California.

Introduction

At the 2004 American Association of Artificial Intelligence (AAAI) Robot Competition and Exhibition, robots had the option to enter four different events including the open interaction event, the robot exhibition, the robot rescue and the robot challenge.

The open interaction event involves creating a robot focused on artificial intelligence with the purpose of entertaining people. A strong emphasis was also placed on human interaction with the robot. Teams participating in the robot exhibition event are able to show interesting and unique research in the field of robotics that is perhaps not covered by any of the other events. For the robot rescue event, AAAI creates a scenario where a disaster has occurred and thus left dummy-people trapped. Robots then search through the rubble to locate potential victims they could save. The robot challenge event participants create robots capable of finding their way to the conference registration desk, registering themselves, finding the room they are meant to present in and then giving a presentation on themselves and the technologies used in their creation.

The Stony Brook Robot Design Team created JESTER (The Joke Entertainment System with Trivia-Emitted Responses) for the AAAI open interaction event and robot exhibition event. The team was divided into three subgroups: data server / human interaction, navigation and computer vision. The data server / human interaction group devised a way to carry on a seemingly intelligent conversation, created a user-friendly interface for users to interact with, developed a joke server and produced a trivia game which kept track of scores to determine winners each hour. The navigation group focused on the movement of JESTER, the finite state machine controlling JESTER and the infrared sensors to avoid obstacles. The computer vision group worked closely with the navigation group to differentiate between an object and people through a motion detection procedure and to detect the color of potential user's clothing.

Background

As JESTER's main goal is to entertain, the human interaction and conversation system are two key components in differentiating JESTER from other robots. Chatterbots serve a wide range of functions. Richard Wallace, the founder of Alicebot, states: "Business uses include online sales representatives and help desks, and advertising… Yet perhaps the biggest markets are the Entertainment markets" (Wallace, 2003). Thus, a chatterbot fits nicely into the objective of creating an entertainment robot. Joseph Weizenbaum created one of the earliest, and still currently popular, chatterbot-type systems named ELIZA. Weizenbaum worked on developing the system during the early 1960's while residing at the MIT Artificial Intelligence Lab. ELIZA was created with the concept of a psychiatrist in mind and is based on a straightforward pattern recognition algorithm. ELIZA was the first program to transform personal pronouns so that if a user entered a sentence such as "Show me how much you love me", the program could respond with "You want me to show you how much I love you?"

ELIZA was successful in convincing people that she understood them and could give meaningful advice. When Weizenbaum's secretary found out that all of her conversations with ELIZA were stored in a log, she was incensed because she had told ELIZA so many personal issues. Weizenbaum soon decided that ELIZA was a

threat, and initially he was going to shut down the program, but instead he decided to write the book *Computer Power and Human Reason.*

From ELIZA, Dr. Richard Wallace found inspiration to begin the Alice Foundation, a non-profit foundation in 1995. The knowledge stored within all A.L.I.C.E. (Artificial Linguistic Internet Computer Entity) systems is stored in the form of AIML (Artificial Intelligence Markup Language). AIML is an XML-compliant language and thus is simple to learn, is easily expanded upon and has many tools that support it. "Alice and it's implementation language AIML is based on the notion that while human thinking is quite complex, it might be just "good enough" to simulate thinking by providing "enough" response patterns to potential inquiries" (Wallace, 2003). Alice and its knowledge base in AIML have proven their approach successful through winning the famous Loebner contest multiple times, where Chatterbots try to convince judges that they are speaking with actual people.

Alicebots are also based on the research of Professor George Kingsley Zipf. Applying Zipf's Law to A.L.I.C.E., it is thought that a particular set of user response's are most common, followed by a set which are less common, followed by still another set which is almost never used. There are in fact roughly 1800 first words of a response sent to Alice 95% of the time. However, from here the number of second words dramatically drops to approximately two overall. There are some exceptions of course, but on average after the first word there is approximately two branches, and with each successive word the "average branching factor decreases" (Wallace, 2003). Thus, essentially 6,000 patterns match 95% of what user's actually input.

While there are now a wide variety of different Alicebots, the majority of them still have an unrealistic avatar. Perhaps the largest drawback of using one of the existing Alicebot and avatar packages is their lack of emotion displayed. AIML has no tag designated for emotion, but since AIML is easily extended, an emotion tag could be added, provided the avatar interface supports a way to display emotions.

Joseph Bates notes that both a Disney animator's goal and an AI researcher's goal is to: "[build] apparently autonomous entities that people, especially the creators, would genuinely care about" (Bates, 1994). Bates' notes that perhaps due to "the communities of which the [AI] researchers were part," they opted to use "reasoning, problem solving, learning via concept formation, and other qualities apparently central to the capacity we call intelligence," while animators simply had 2-dimensional hand drawings made frame by frame to create motion in which to portray a real character (Bates, 1994). Bates further goes on to state, "that while scientists may have more effectively recreated scientists, it is the artists who have come closest to understanding and perhaps capturing the essence of humanity… that AI researchers, ultimately seek" (Bates, 1994). AI researches often overlook a crucial factor that animators picked up on, which is the importance of unambiguous emotions being conveyed at appropriate times. By expressing emotions, avatars can portray that they have feelings and the ability to care. Emotions can add the next dimension to chatter bots, bringing them one step closer to simulating a real human.

SARA: Stony Brook Robot Avatar

A variety of technologies were researched before deciding to use Haptek Technology to create JESTER's user interface. The 3-D avatar that comes with Haptek Technology is easily integrated into C/C++ code through Microsoft's Visual C++ environment via an ActiveX component, thus simplifying the integration of SARA with the hardware control software for JESTER. Additionally, the graphics used in the Haptek avatars is phenomenal in terms of being realistic as show in Figure 1:
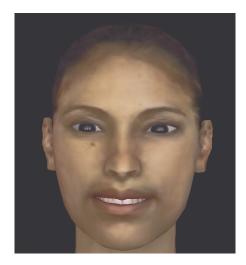

Figure 1: Haptek Avatar chosen for SARA

A program called PeoplePutty makes it possible to create customized 3-D avatars and specialized emotions for the characters. So rather than being restricted to a constant set of avatars, an avatar fit for the given system can be created. Additionally, changing the avatar is simple even after the system is created and running.

However, the biggest factor in selecting Haptek Technology stems from the fact that their avatars have the ability to express emotion. When an emotion is set for SARA, her face is constantly moving and changing so for example when she is happy, she does not just have a constant and static smile. See Figure 2 for some examples of her emotions.

Figure 2: SARA's Various Emotions

Once the avatar was selected, an A.L.I.C.E. engine was needed to integrate with the avatar. The avatar was programmed to use Microsoft's Speech SDK text-to-

speech engine. Thus, the interface needed an intelligent conversation system to feed it appropriate responses to user inputs. Unfortunately, there are no good tested implementations of ALICE that are compatible with Microsoft's Visual C++ environment, which is what the rest of the Stony Brook robot is programmed in for hardware compatibility reasons. Therefore, the Java version of A.L.I.C.E. called Program D was used. The avatar and interface still resided in MS Visual C++, since it was easier to use Haptek Technology in this environment and this was the environment necessary for the interface to be compatible with the robot's actual hardware. Therefore, the Java program and Visual C++ interface needed to communicate through socket communication. The interface sends requests and listens on port 8000 while the Java A.L.I.C.E. code listens for requests and sends replies back through port 8000.

After the Java intelligent conversation system was communicating with the Visual C++ user interface, the element of emotion was added. With over 40,000 categories in the current AIML file, to add emotions for every response would be too tedious for a single person. So the most common topics were focused on with the idea that the emotion tag would be an optional add-on as an AIML tag in the future. Thus, since it is an open source project, people can continue to add emotions as the project grows. If the following tags are found anywhere within the response returned from Program D, the corresponding emotion is assigned to the avatar in addition to her verbally responding with the response provided by the Program D system. Valid emotions are: Angry, Psycho, BrokenHearted, Bully, Calm, Happy, Mellow, PonderingMysteries, Sad, Shy, ShyAndInLove, Skeptic, and the default is a neutral emotion if no emotion is specified.

After editing one version of the AIML library, or the Alicebot brain, there were two versions: one with emotions and one without. A group of 25 undergraduate and graduate students were asked to interact with SARA with the AIML set enhanced with emotions and another group of 25 students were asked to interact with the version of SARA that had no emotions. There was no time constraint given to the users, thus they could interact with her for as little or as much time as they wanted. When they were finished, they were all asked the simple question, "Did you enjoy your overall experience?" The user responses to this question in addition to the data collected on how long each user elected to interact with the system were used to judge the effect of adding emotion to the system.

The users interacting with the intelligent system that was not enhanced by emotion spent an average of 7.12 minutes talking to SARA. The user that spent the least amount of time stayed for 4.7 minutes and the user with the maximum time interacted for 13.3 minutes. Of the 25 users that interacted with the plain intelligent system, 21 said they

enjoyed their experience, which is to say that 84% reported a positive experience.

The users interacting with the emotionally enhanced SARA spent an average of 11.59 minutes talking to the robot. The user that spent the least amount of time stayed for 6.2 minutes while the maximum time spent by any user was 16.7 minutes. 23 of the 25 users that interacted with the emotional intelligent system said they enjoyed their experience, thus 92% reported a positive experience in this case.

The actual data collected can be seen in the chart shown in Figure 3. In both groups of users there are some users that spent longer and some users that spent less time, which could be attributed to their personal attention spans as well as how busy their personal schedule was. However, the majority of users for the non-emotion SARA spent around the 7 minutes and the users for the emotionally enhanced SARA stayed close to 11.5 minutes, which is a significant difference. Users spent approximately 38.6% more time interacting with the emotional interface than with the regular interface.
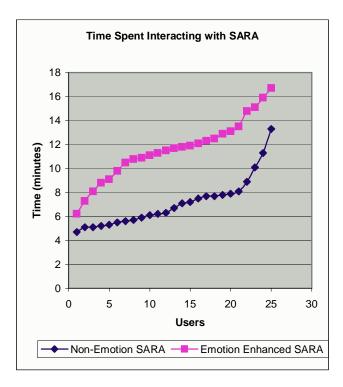

Figure 3: Data Collected From
Users Interacting With SARA

Trivia and Joke Server

Currently there are around 300 jokes and 200 trivia questions in JESTER's database. Jokes are used to capture the attention of people as JESTER moves about and it is used as another form of entertainment. The trivia system has an independent, user-friendly graphical user interface from SARA. Once the main program running JESTER invokes the trivia system, the trivia interface is displayed. The user may quit the trivia system at any time or may complete five questions to try and compete for the highest score for the hour. The highest score each hour is rewarded with a prize from a secret compartment in JESTER's body. Users are asked to enter their name so that the system can keep track of who has the highest score.

Navigation

JESTER's navigation system uses both infrared sensors and computer vision to avoid obstacles and distinguish between people and other objects. There are six infrared sensors used in JESTER. Three sensors line the north, northeast and northwest of JESTER's front, and another three line the south, southeast and southwest of the front. JESTER never walks backwards; hence infrared sensors are not needed along the back. When any of these sensors detect an object, the robot base stops moving and lets the computer vision determine the next step. The computer vision runs a motion detection routine to differentiate between a non-moving object and a moving object, assumed to be a human. If a human is found, a color detection routine is run to determine what color clothing the person is wearing, and this color information is then used to obtain the attention of the person in front of JESTER.

Computer Vision

The two main tasks for ButlerBot's vision system are to detect human activity and to determine colors. The whole vision system is written in the Java programming language. As an independent command-based system, it acts as a server that responds to a list of commands that it receives through the network. To achieve the purpose of successfully communicating with any other programs that might be written in different programming languages, such as C/C++, the system uses the socket-programming standard provided by Java. It is crucial that the vision software communicates via sockets as the hardware control libraries for JESTER and the interface are created in Microsoft Visual C++. Once the system is running and has established a connection with the main driver for the robot, it waits to receive appropriate commands. The following table in figure 4 shows the usage and responses of the commands that the vision system accepts

Start motiondetection

Usage: First, it initializes the capture device if it has not been initialized. Then it runs the motion detection module.

Responds: "OK" if the motion detection module initialized successfully "TRUE" if motion has been detected

Start colordetection

Usage: First, it initializes the capture device if it has not been initialized. Then it runs the color detection module.

Responds: "OK" if the color detection module initialized successfully *Color name* if the color has been detected. Valid color name includes: *(Color names would be based on the needs)*
-0 WHITE
-1 YELLOW
-2 ORANGE
-3 BLACK
-4 GREEN
-5 PURPLE
-6 BLUE
-7 RED

Stopvision

Usage: "Soft" stops the motion or color detection modules so that there is no need to initialize the device before using them next time.

Responds: "OK" if the motion or color detection module has been stopped

Stop

Usage: "Hard" stops the motion or color detection modules so that the device need to be initialized before using them next time

Responds: "OK" if the motion or color detection module has been stopped

Quit

Usage: Shut down the server, i.e. exit the application

Responds: Connection lost

Figure 4: Commands for Vision System

A motion detection system, modified from Konrad Rzeszutek's Motion Detection Toolkit, determines if there is motion in front of the robot, and thus distinguishes between facing into a crowd and facing a wall. The Java Media Frame Library is used to capture data from a camera mounted on top of JESTER. For the best performance, the Logitech Pro 4000 was selected as the image capture device.

After configuring the camera, the program compares two sequential images, the reference image and the input image. The program compares these two images by observing whenever a pixel has changed significantly based on a threshold variable. An internal black-white-red image is then generated to highlight clusters in the area where a change has occurred. Next, the program eliminates clusters that are very small, and thus deemed insignificant, in the black-white-red image while keeping a count of the number of large clusters. Ultimately, the only clusters left on the black-white-red image are those that are big enough to be considered a moving person or a group of moving people. In the final step, the program compares the count result to a specific threshold to determine if the input image has a large enough amount of motion to be considered significant.

Another important aspect of JESTER's computer vision system is its ability to detect colors. The color detection system is modified from Lawrie Griffiths's Lejos Vision System used for color detection. Instead of taking two pictures, as in the motion detection routine, the program only takes one picture at a time and scans it pixel by pixel for the colors that JESTER is asked to find. A sRGB threshold for colors is set as a constant so that a specific color within a threshold boundary is given a label such as red, blue or yellow. Then, the program calculates the percentage of each color and compares it to the preset values. The routine continually takes ten pictures and if it detects the same color multiple times, the person in front of JESTER is assumed to be wearing clothing of the detected color.

Conclusions

JESTER was awarded the Open Interaction Event Audience Favorite, and thus JESTER was clearly successful in entertaining the audience at the AAAI 2004 Robot Competition and Exhibition. In general, humans are most comfortable interacting with humans and thus the more human-like robots become, the more comfortable

people will be to interact with them. While carrying on an intelligent conversation is significant in trying to imitate a human, portraying emotions are vital as well. It is believed that people like cartoons that they can relate to, and that people best relate to cartoons with emotions that are clear and properly portrayed (Bates, 1994). Imagine watching a Disney movie in which the characters talk like people, but display no emotion. If cartoons without emotions do not capture people, then why make user interfaces without emotion?

Future Work

There is still much work and research needed to create a truly human-like avatar. One area of future research for JESTER's interface, SARA, will be to emotionally enhance her speech so that she sounds more like a human and less like a machine. Murtaza Bulut, Shrikanth Narayanan, and Ann Syrdal at AT&T Labs agree that through adding an emotional element to text-to-speech (TTS) engines, people will be able to interact with computers more naturally and will enjoy the experience more.

In the AT&T study, four different emotions were conveyed in short sentences that were of a neutral subject. The four emotional states they choose were happiness, sadness, anger and neutral. One flaw they noted was that "In most studies, human emotions are categorized into basic distinct categories such as anger, happiness, sadness, fear, disgust, surprise and neutral. Although this approach is correct, especially from a pragmatic sense, it can be regarded as a somewhat gross representation because it ignores the variations that exist within each emotion" (Bulut, 2002). Similarly, by selecting only a small set of possible emotions for the avatar to express, we leave out a great deal of variety that is seen in humans. Thus, in addition to adding emotion to the avatar's voice, adding a gradient and different levels of emotion to both the voice and the facial expressions will be added in the future.

One of the extensions that Bulut, Narayanan and Syrdal are interested in adding to their research is the "use of visual aids such as pictures, videos, sounds" as they believe it "may help… to express the required emotion more successfully" (Bulut, 2002). Thus, the SARA interface provides a wonderful graphical display of emotion that could be linked together with an emotional text-to-speech system, which would be mutually beneficial for both systems.

Acknowledgements

References

A.L.I.C.E. AI Foundation, Inc. "The A.L.I.C.E. Artificial Intelligence Foundation – AIML chat robot software." http://www.alicebot.org (15 Mar. 2004).

A.L.I.C.E. AI Foundation, Inc. "A.L.I.C.E. The Artificial Linguistic Internet Computer Eity." http://www.alicebot.org/documentation/gallery/ (15 Mar. 2004).

Bates, J. 1994. The Role of Emotion in Believable Agents. *Communications of the ACM, Special Issue on Agents.*

Bulut, M., Narayanan, S., Syrdal, A.: 2002. Expressive Speech Synthesis Using A Concatenative Synthesizer. *ICSLP 2002.*

Evolution Robotics, http://www.evolution.com

Futurama: Using Java Technology to Build Robots That Can See, Hear Speak, and Move, http://java.sun.com/developer/technicalArticles/Programming/robotics/

Haptek, Inc. "Haptek, Inc." http://www.haptek.com

Image Capture From Webcams using the Java[tm] Media Framework API, http://java.sun.com/dev/evangcentral/totallytech/jmf.html

Java[tm] Technology Robotics Developers Kit, http://www.sun.com/developers/evangcentral/totallytech/robo-sdk.html

Konrad Rzeszutek's Projects, http://darnok.dhs.org/~konrad/download.html

LeJOS Official Website, http://lejos.sourceforge.net/

Lejos Vision System, http://homepage.ntlworld.com/lawrie.griffiths/vision.html

Wallace, R., Tomabechi, H., Aimless, D.: 2003. Chatterbots Go Native: Considerations for an Ecosystem Fostering the Development of Artificial Life Forms in a Human World, http://www.pandorabots.com/pandora/pics/chatterbotsgonative.doc