Susan J. Boyce

# NATURAL SPOKEN DIALOGUE SYSTEMS *for* TELEPHONY APPLICATIONS

*A friendly "How may I help you?" replaces traditional menus for telephony services.*

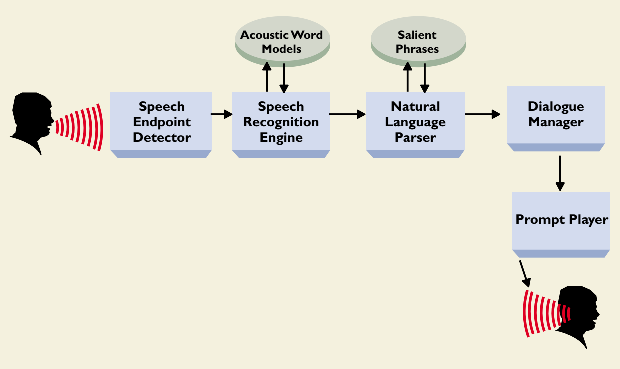
For more than a generation, science fiction writers have assumed that in the future people would talk to their computers and their computers would talk back to them. This seemingly simple method of communicating with machines is apparently preferred by the spacesuit-wearing set the galaxy over. But what really constitutes a natural-language dialogue with a computer? Why isn't it a reality yet? And how close are we to achieving natural conversation with computers?

Natural-language speech recognition refers to computer systems that recognize and act on unconstrained speech. That is, the user need not know a predefined set of command words in order to use the system successfully. This is not to say that in order to qualify as a natural-language speech-recognition device, a computer has to incorporate a typical human's complete range of spoken-language understanding. Natural-language devices in most cases are being designed to carry out specific tasks, accepting inputs only from a specified range of topics. An example is a natural-language computer that aids people making travel arrangements. The user might say something like, "I'd like a flight to LA on November 15th, returning the next day." This hypothetical computer might get all or part of this command right and continue the dialogue to refine these plans. However, if the user asks, "Do you think the Yankees will win on Saturday?," the computer is likely to make its best attempt at recognizing the input and produce something nonsensical in return. Limited to current technology, natural spoken-language systems are likely to respond to fluently spoken dialogue only if it falls within their specialized task domains.

A second component of a natural spoken-language recognition system is the ability to gracefully handle breakdowns in recognition [1]. A spoken natural dialogue system has to be able to identify when it fails to understand the user (an inevitable turn of events) while carrying on a conversation that clears up the misunderstanding and realize when it is incapable of recovering from the misunderstanding (see Figure 1).

If it becomes possible to build computers that respond to fluent natural language and gracefully recover from errors, the users' view of the computer as a social entity is likely to change. The act of using natural speech as the input mechanism makes the computer seem more human-like. (I discuss the

**Figure 1. General components of a spoken natural dialogue system.**



degree to which designers of spoken dialogue systems should imbue their systems with personality and human-like traits later in the article.) Although individual natural-language applications differ in terms of architecture, some components are common to most spoken natural-language systems, including:

- A continuous speech-recognition engine;
- Acoustic word models;
- A natural-language parser;
- A dialogue manager; and
- A prompt generator, using text-to-speech technology or recorded prompts.

The process of spoken-language recognition requires the speech detector indicate when the speaker has started and when the speaker has stopped talking. The word models and the speech-recognition engine then perform a word-by-word transcription of the spoken input. The natural-language parser then attempts to map meaning onto words and groups of words. The dialogue manager takes the resulting interpretation and determines what is known and not known in order to further the dialogue by playing an appropriate response to the speaker.

## Spoken Dialogue Systems versus Alternatives for Telephony Applications

We are all familiar with touch-tone-based interactive voice-response phone services, and most of us find them inadequate and frustrating to use. A major drawback is that they require users to listen to and remember an artificial mapping between numbers on the keypad and actions (for example, "press 1 for checking" and "press 2 for savings"). In addition, in order to keep users from being overwhelmed by options, high-level descriptions of groups of functions are used at the main menus to represent the myriad functions the service is trying to perform. The use of high-level descriptions of group functions often makes it very difficult for users to determine which menu option to choose.

Speech-recognition systems that recognize individual keywords were once viewed as an advance, because users did not have to be taught the mapping between keys and action and could instead be instructed to use one of several spoken keywords. For example, the prompt "Please say collect, calling-card, third-number, person-to-person, or operator now" is used successfully in the phone network today. But there are several disadvantages to this kind of system: The keyword options have to be listed for users; users still have to wend their way through a hierarchical menu structure; and it is difficult to achieve word-recognition accuracy rivaling touch-tone accuracy.

Spoken dialogue systems, however, have several compelling advantages: Callers state their requests in their own language and do not have to listen to a list of commands; and the hierarchical design of the interface is flattened, so a caller may say, for example, "I want credit for the wrong number I dialed." This function would likely be buried under several submenus in a hierarchical system, but here, the caller, in a single conversational turn, navigates to the appropriate place in the interface.

Some of today's speech-recognition systems simulate natural dialogue by using a finite state grammar. Users' responses are modeled by constructing elaborate grammars, or the set of words recognizable at that particular point in the dialogue. These systems improve on single-keyword recognition devices, responding to a wider range of user inputs and allowing for a more-flexible, natural-sounding dialogue. However, handcrafting these grammars is time-consuming and error-prone. It is difficult to a priori anticipate every word users might actually use in that context for inclusion in the grammar.

Communicator, a U.S. Defense Advanced Research Projects Agency-funded project, represents a recent industrywide attempt to develop a more robust, flexible spoken natural-language dialogue system. Researchers from many companies and universities are collaborating on a plug-and-play architecture to create a next-generation conversational system. (For more about Communicator, see www.darpa.mil/ito/.)

## How AT&T's Spoken Dialogue System Works

Spoken natural-dialogue systems are considerably more complex to design and implement than spoken command-word systems and touch-tone systems. One might expect such a system would be hopelessly error-prone, since the processes of word recognition and natural-language parsing can both introduce errors.

How is it that word-recognition accuracy isn’t an insurmountable problem for natural spoken dialogue systems when it has been such a problem for command-word interfaces? The answer is that, unlike in spoken command-word interfaces, word-by-word accuracy isn’t as crucial.

For the How May I Help You? (HMIHY) experimental system, developed at AT&T Labs, even with word-error rates of 32%, correct classification of 97% can be achieved [9, 12]. The redundancy in user requests provides multiple pointers to the actual meaning of these requests.

In HMIHY, AT&T researchers have found a key in the concept of “saliency” in language. That is, certain words and groups of words in an utterance are more indicative of the meaning of the utterance than others. In many cases, a user’s utterance contains multiple “salient phrases” pointing to a single interpretation of the utterance. Therefore, it is not necessary to get every single word of an utterance correct. As long as enough words are recognized to trigger the correct salient phrase, the proper meaning is assigned to the utterance. For example, a caller might use HMIHY to try to get credit for a wrong number dialed by accident, as in the following dialogue:

*The basic problem is that if the system sounds too much like a human, the user can reasonably expect it to understand like a human.*

System: How may I help you?
Caller: Yes, I just made a call to a wrong telephone number.

The caller’s request contains several salient phrases. Phrases like “wrong telephone number” are obvious cues to the meaning of the utterance. But there are others, such as “I just made,” that are less obviously related to the meaning of the request, yet still turn out to be highly predictive of a particular interpretation (in this case, credit for a wrong number dialed). If the speech recognizer recognizes something like “Yes, I just made a call to a long telephone number,” confusion between “long” and “wrong” isn’t detrimental to the interpretation of the meaning of the request, since the phrase “I just made” also points to the correct interpretation. This redundancy gives spoken natural-language systems their most valuable advantage.

As can be seen from the dialogue, this system was designed to automate certain kinds of requests callers routinely make to phone company operators thousands of times every day. The HMIHY system is designed to classify callers’ responses to the prompt “How may I help you?” into one of 15 different kinds of requests. For example, if a caller says, “I want to place a call and reverse the charges,” the appropriate action for the system is to handle the call as a collect call. If the request is, “I don’t understand this charge on my bill,” the call should be routed to the appropriate customer-service agent.

The first phase of the system’s design process involves gathering information about how callers express their requests to fellow humans and how most human agents elicit clarifying information [6]. By learning how callers express requests and operators respond, my colleagues and I at AT&T Labs collected important data for algorithm and technology development, along with important information for designing the HMIHY user interface. By closely matching the wording of the system prompts to the words used by human operators, we thought we could achieve a greater degree of naturalness.

However, one of the by-products of this strategy was that the system ended up sounding very human-like, since we modeled the prompts on those used by human operators. Did such mimicry improve the system’s ability to handle callers’ requests? How would callers react to such an automated system? There has been much debate among human-computer-interaction researchers about whether or not making computer systems “sound” human-like is a good idea [11]. The primary argument against anthropomorphic systems has been they create unrealistic user expectations, leading to errors and then disappointment with the system. Even some pre-1990 human-factors research indicated this is often a natural reaction, and that there can be further negative reaction to a user interface that is too cutesy or talkative.

**Figure 2. Example dialogue (requesting an area code) for each prompt style tested.**

| | Casual | Formal |
|---|---|---|
| "I" | System: How may I help you?<br>User: I need the area code for Morristown, New Jersey.<br>System: Okay, you need me to give you an area code.<br>User: [either silence or] Yes.<br>System: City and State?<br>User: Morristown, New Jersey.<br>System:The area code is 201.<br>System: Anything else I can help you with?<br>User: No, thanks.<br>System: Thank you for using AT&T. | System: Please tell me how I can help you, now.<br>User: I need the area code for Morristown, New Jersey.<br>System: Do you need me to give you an area code?<br>User:Yes.<br>System: Please say the city and state for which you would like the area code, now.<br>User: Morristown, New Jersey.<br>System: The area code is 201.<br>System: If there is anything else I can help you with please tell me now.<br>User: No, thanks.<br>System: Thank you for using AT&T. |
| No "I" | System: What would you like?<br>User: I need the area code for Morristown, New Jersey.<br>System: Okay, you need to know an area code.<br>User: [either silence or] Yes.<br>System: City and State?<br>User: Morristown, New Jersey.<br>System:The area code is 201.<br>System: Is there anything else you need?<br>User: No, thanks.<br>System:Thank you for using AT&T. | System: Please state your request, now<br>User: I need the area code for Morristown, New Jersey.<br>System: Do you need to know an area code?<br>User: Yes.<br>System: Please say the city and state for which you would like the area code, now.<br>User: Morristown, New Jersey.<br>System: The area code is 201.<br>System: If you have an additional request, please say it now.<br>User: No, thanks.<br>System: Thank you for using AT&T. |

Other studies have involved screen-based systems in which the user types input on a keyboard. It is possible that since the abilities of natural spoken dialogue systems more closely match the abilities of humans that the negatives associated with anthropomorphic interfaces can be mitigated. Moreover, users may have been exposed to a much wider variety of automated services in the years since these early studies were done and that this exposure has resulted in a change in user perception about anthropomorphism.[1]

Therefore, the first simulation experiment (1996) for HMIHY was designed to determine which aspects of a system might make it seem more human-like, as well as determine if making a system seem more human-like has a positive or negative effect on users' satisfaction with it.

## How Human-like Should It Be?

A "Wizard of Oz" simulation of the HMIHY system was constructed to test users' reactions to different styles of interactive dialogue [7]. It simulated the system's speech-recognition and natural-language understanding components, though the users didn't know they were using only a simulated system, not the production system. The experimenters monitored the dialogue between users and the system, pressing buttons to play out the next computer response in the dialogue, thus testing and constructing different dialogue styles. For users, the experience appeared to involve speaking with a fully functional spoken dialogue system.

The simulation tested two aspects of anthropomorphism: whether or not the computer referred to itself as "I" (as in "How may I help you?"); and whether the language used by the computer was formal and traditional or more casual, like the language used by human operators (see Figure 2).

The results indicated that users prefer speaking to the versions of the system referring to themselves as "I" over those using the more traditional approach of avoiding "I." It appears that anthropomorphism, at least under some circumstances, isn't as bad as previously thought. Another interesting finding is that upon interviewing the simulation-using test subjects, most were unaware the computer referred to itself as "I" at all. Hence, it seemed the effect of "I" was positive but fairly subtle in that it wasn't very noticeable to users. The casual/formal manipulation yielded no significant differences in user satisfaction [3].

These results should not be interpreted as a license to make dialogue prompts wordy or chatty. When writing anthropomorphic versions of prompts, every attempt was made by HMIHY's designers to keep them from sounding cutesy, overly friendly, funny, or chatty. The moral seems to be that anthropomorphism used judiciously in a system designed for occasional use can enhance users' satisfaction, often without their being overly aware of the human-like quality of its prompts.

## Should It Have Personality?

In a different study of a voicemail system at AT&T Labs, we pushed the concept of anthropomorphism further, defining more extreme human-like interfaces. The application in this case was a voicemail system (never released) with which users would have to interact many times a day. The idea was that, given the success of the use of "I," perhaps we should take the next step in designing the computer to be human-like, *deliberately* imbuing it with "personality." That is, we designed the computer not only to behave in a human-like manner but to behave as a particular

[1] Anthropomorphism means making something non-human have human-like qualities.

human, with a name and human-like idiosyncrasies.

Early on, we realized we would have to test more than one personality to do a fair job of exploring the concept of personality in spoken-language human-computer interfaces. We therefore developed five quite different versions of the system, each with a name, voice, and conversational style. Personalities ranged from formal butler ("Watson") to hip youth ("Mangohead"). The system's functionality was the same across all versions, but the style of the speech, voice talent, pacing of the interaction, and content of the recorded prompts varied dramatically among the versions.

We had 32 test subjects to experience the different versions, asking them how satisfied they would be with each as their voicemail system. The results indicated that there was interest in the concept of personality in computers but also how easy it is to design personalities that would be disliked by users. For example, some users loved Watson the butler, others found him stuffy and annoying. The personalities that were least extreme were much safer in that they had fewer negative reactions from users, though in some cases, fewer strongly positive reactions as well. These personalities were similar to the style and voice of the system tested in the HMIHY anthropomorphism study.

My colleagues and I concluded that anthropomorphism—when modeled on the speech of real human operators—is acceptable to users and can be beneficial in the design of a likable spoken dialogue system. However, anthropomorphism can be taken so far that some systems annoy and offend users.

## How Do Callers Know It's a Computer?

A frequent concern by some human-computer interaction researchers about anthropomorphic human-computer dialogues is that early in the interaction, users are likely to assume the system has greater abilities than it actually has, and therefore attempt to speak in a manner the system has little probability of understanding. Designing the right initial system greeting is necessary for establishing user expectations and helping users determine how to proceed.

The basic problem is that if the system sounds too much like a human, users can reasonably expect it to understand like a human, a feat machines are not yet capable of. At the other end of the spectrum are menu-driven command-word systems. With these systems, users may have the expectation that the words listed in the menus are a complete set of the words the system understands and that no other words can be used, making a natural-sounding dialogue difficult or impossible. The HMIHY system falls somewhere between these two extremes. The list of possible "commands" is too long to be presented in a menu, so a more open-ended prompt, such as "How may I help you?," seems appropriate. But from listening to many thousands of calls to real human operators, it became obvious to us at AT&T that if callers believe they are speaking to a human operator, their requests are often long and complicated. Therefore, the HMIHY designers wanted to prevent this system from being mistaken for a human operator; it was unreasonable to expect the system to be able to handle long complicated requests as well as a human operator understands them.

**Figure 3. Average number of words callers used to state their requests (as a function of the initial greeting).**

| Initial Greeting | Avg. No. of Words in Request |
|---|---|
| AT&T. How may I help you? | 12.99 |
| AT&T Automated Customer Service. How may I help you? | 12.43 |
| AT&T Automated Customer Service. This system listens to your speech and sends your call to the appropriate operator. How may I help you? | 10.52 |
| AT&T Automated Customer Service. This system listens to your speech and sends your call to the appropriate operator. How may I help you? (text-to-speech) | 8.47 |

Hence, the goal of the HMIHY user study was to come up with an initial greeting that lets callers know they are talking to a machine, not to a human. The expectation was that if users know they are talking to a machine, they bring what they know about machines to bear on the dialogue. Presumably, this expectation would alter user requests in some way, perhaps making them shorter [4, 8, 10].

The HMIHY simulation tested two ways of communicating to callers that they were speaking to a machine: One said explicitly it was a machine in the first announcement heard by the caller; the other used computer-generated text-to-speech output to make the system "sound" like a machine. These alternatives were tested using the Wizard of Oz method with a group of 545 callers to determine whether changes to the initial greeting influenced the length of the utterances callers used to state their requests.

Figure 3 includes the various versions of the initial tested greetings: The first is the phrase used by human operators; the second is the phrase "automated customer service" as a cue to the caller that the question comes from a machine, not a human; the third is a very wordy explanation explicitly telling callers they are speaking to a machine; and the fourth uses the same prompt, but instead of playing it as a sound file of recorded human speech, it plays it with text-to-speech output, which, though easily understood, has an unmistakable robotic quality.

The average number of words callers used to state their requests fell when the long version of the greeting was used and fell further when text-to-speech output was used. Thus, we were able to get callers to change the way they stated their requests by manipulating the content of the initial greeting. Unfortunately, the customer satisfaction measures indicated that callers hearing the long versions of the greetings and the text-to-speech versions were not very happy with the experience. Still unknown is how callers might have responded to a text-to-speech version of the simple HMIHY prompt.

A subsequent study of HMIHY interfaces was conducted at AT&T Labs in 1997 to further refine the wording of the system's initial greeting and test another alternative: playing an "audio logo"[2] or sound effect at the beginning of the greeting as a way of cuing callers they are communicating with an automated system. Playing an audio logo proved effective, achieving results like those with the long automated prompt (from the previous study) but requiring much less time, since the shorter version "How may I help you" could be used [3].

The first prompt is an important element in establishing user expectations. By making it state explicitly that the speaker is a machine, not a human, callers then stated their requests with shorter utterances, making it easier for HMIHY to understand and respond appropriately. In addition to the content of the prompt, the voice (text-to-speech or recorded human speech) and sound effects were also effective cues to users that they were talking to a machine.

## Future Systems

My colleagues and I continue to conduct user-interface studies to refine the HMIHY design, making it easy and enjoyable to use. However, additional hurdles must still be overcome before spoken dialogue systems replace the existing embedded touch-tone services. For example, all of the computations necessary for word recognition and natural-language understanding have to be done in real time for a spoken dialogue system to be usable. Therefore, one of our challenges is that, to achieve real-time speed, the processing hardware can be too costly. But it is only a matter of time before processors are fast and cheap enough to allow natural-language processing to be attractive in a business sense.

Another key to this technology is the right application. There will always be less-than-perfect accuracy in spoken dialogue systems, though algorithms will continue to improve. The trick is finding applications in which 100% accuracy is not needed. Moreover, such applications must accomplish tasks that are actually useful and can't be done easily any other way.

Many touch-tone systems today seek to automate too many tasks and therefore become hierarchical and difficult to use. Spoken dialogue systems represent a better technology in many of these situations by flattening the hierarchy. A spoken dialogue system that understands spoken input 85% of the time is likely to automate more calls than a complicated touch-tone system in which callers end up down the wrong path.

Spoken natural dialogue systems have the potential to change the nature of automated telephony services. When these systems are designed to do useful, targeted applications, and when the user interface is well designed, they will exceed touch-tone menus in terms of automating functions and satisfying users. C

[2] An audio logo is a sequence of notes or tones used by a company to brand its products, often in conjunction with a voice recording of the company name.

## References

1. Ballentine, B. and Morgan, D. *How to Build a Speech Recognition Application.* Enterprise Integration Group, Inc., San Ramon, Calif., 1999.
2. Boyce, S. Spoken natural language dialogue systems: User interface issues for the future. In *Human Factors and Voice Interactive Systems,* D. Gardner-Bonneau, Ed. Kluwer Academic Publishers, Boston, 1999, 37–61.
3. Boyce, S. Design of computing systems: Social and ergonomic considerations. In *Advances in Human Factors/Ergonomics, vol. 21B.* M. Smith, G. Salvendy, and R. Koubek, Eds. Elsevier, Amsterdam, 1997, 367–370.
4. Falzon, P. Human-computer interaction: Lessons from human-human communication. In *Cognitive Ergonomics: Understanding, Learning, and Designing Human-Computer Interaction,* P. Falzon, Ed. Academic Press, London, 1990, 51–66.
5. Franzke, M. and Marx, A. Is speech recognition usable? An exploration of the usability of a speech-based voice mail interface. *SIGCHI Bullet. 25* (1993), 49–51.
6. Gorin, A., Parker, B., Sachs, R., and Wilpon, J. "How may I help you?" In *Proceedings of Interactive Voice Technology for Telecommunications Applications (IVTTA)* (Basking Ridge, N.J., Sept. 30–Oct. 1). IEEE, Piscataway, N.J., 1996, 57–60.
7. Gould, J., Conti, J., and Hovanyecz, T. Composing letters with a simulated listening typewriter. *Commun. ACM 26,* 4 (Apr. 1983), 295–308.
8. Kennedy, A. and Wilkes, A. Dialogue with machines. *Cognit. 30,* 1 (1988), 37–72.
9. Riccardi, G. and Gorin, A. Spoken language adaptation and state in a natural spoken dialogue system. *IEEE Trans. Speech and Audio 8,* 1 (2000), 3–10.
10. Richards, M. and Underwood K. Talking to machines: How are people naturally inclined to speak? In *Contempory Ergonomics,* E. Megaw, Ed. Taylor & Francis, London, 1984, 62–67.
11. Shneiderman, B. *Designing the User Interface, 2nd Ed.* Addison-Wesley, Reading, Mass., 1992.
12. Wright, J., Gorin, A., and Abella, A. Spoken language understanding within dialogues using a graphical model of task structure. In *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP'98)* (Sydney, Australia, Nov. 30–Dec. 4). Australian Speech Science and Technology Association, Inc. (ASSTA), 1998, 2103–2106.

**Susan J. Boyce** (sjboyce@att.com) is a principal technical staff member at AT&T Labs Labs in Middletown, NJ.