MAS.632 Speech Interfaces and Mobile Devices
Fall 2008

13

Toward More Robust Communication

This book has discussed a variety of speech technologies including digital audio coding, speech recognition, text-to-speech synthesis, and the telephone. Intermingled with chapters explaining the algorithms that provide these capabilities to computers have been chapters devoted to applications and interactive techniques whereby speech may be employed effectively in a computational environment. The emphasis has been on *interaction* and *responsiveness* of voice applications; the case studies expand on these obvious surface features of conversation. But, despite the themes of higher-level discourse presented in Chapter 9, no specific application is capable of conversing at the level of a four-year-old child even when limited to a single topic.

Major constraints to conversational interactions are the limited ability of current speech recognition technology and the marginal fluency of synthetic speech. Chapters 10 and 11 temporarily overlooked these limitations by discussing the role of computers as *facilitators* of unconstrained conversation among people over the telephone. This led to Chapter 12's discussion of the role of audio as data captured on the desktop from sources such as voice mail and recordings of a meeting or telephone call. In some ways these later chapters are disjoint from prior chapters in that they describe situations in which the computer makes no pretense at conversational ability, understanding nothing about the content of the audio it is routing or recording while it assists with setting up a conversation or archiving it. But from the end user's perspective, this disjunction may not be so apparent. We converse in diverse situations and for many purposes; we introduce strangers, initiate conversations, and often listen to others more than we speak. For the user, speech is about getting something done, and the tasks we already perform

among ourselves are more important in defining how we will utilize speech than the limitations of current technology.

This final chapter returns to the issue of increased computer participation in conversation and poses several domains in which computers woefully lag our conversational abilities beyond hearing words or stringing them together into spoken sentences. One component of improved computer support for conversations is a better appreciation of the interactive techniques we employ to ensure robust communication while talking. Another is tighter coupling between the acoustic and conceptual phases of language understanding. Finally, prosody is an essential element of human speech largely overlooked in computational speech systems. However, these are not presented here as unsolved problems but rather as encouraging avenues that may lead to more productive voice interaction between people and computers.

ROBUST COMMUNICATION

In our use of language we employ a variety of techniques to confirm that the other party understands our message. When the other party is a computer communicating via technology of marginal ability, such robustness is even more essential. Most current applications of speech technology are extremely brittle; if they succeed it is mostly due to the cooperation of the user and our remarkable ability to adapt to conversations under conditions of extreme noise or very thick accents. If conversational computers incorporate our own communication protocols, they can become much more capable partners in conversation. This section describes some aspects of our conversational ability that have been largely left out of the design of dialogue systems to date, although many of these ideas were briefly introduced in Chapter 9.

In a seminal and lengthy journal article, Hayes and Reddy note a number of important elements of conversational protocols and identify some additional behaviors that a conversational system might employ [Hayes and Reddy 1983]. They identify the principle of *implicit confirmation:* if the speaker believes that the listener received the message, then the speaker also believes that the listener understood the message in the absence of any response. This places a huge burden on the listener who must indicate any difficulties understanding the message at the proper time or else the speaker may continue and the conversation will rapidly break down. Such expectations imply that speech recognition applications, which may be thought of as a listener at the end of a very noisy communication channel, must always respond in a timely manner to recognition errors.

Clearly it is a violation of the listener's side of conversational protocol to detect that recognition failed and not say anything as this constitutes an explicit acknowledgment that the message was understood. What should be said in response? Humans concisely communicate what has been understood and what information is lacking through a process of progressive clarification. This may be accomplished by asking a series of short, specific questions rather than more general requests for the speaker to repeat or rephrase the original utterance. The

nature of questions that we ask strongly influences the speaker's response, and we should guide the conversation to a successful conclusion. By echoing portions of what has been thought to be understood, the listener also invites correction by the speaker. The principle of implicit confirmation suggests that an echoed statement can be assumed to be correct unless it is contested by the original speaker.

We often describe an object partially, identifying only its salient features. Grice's maxims support this behavior; they indicate that we should speak only as much as is necessary, i.e., if a complete description is not necessary to identify an object, then we should not elaborate. Humans excel at understanding each other from fragmentary utterances, but these sentence fragments can wreck havoc on computational parsers and semantic analyzers. "Conceptual" parsers operate on the basis of keyword detection, e.g., by placing keywords into slots with appropriate roles in a frame. But simply detecting keywords may be insufficient for identifying their role even in a syntactically simple utterance such as "The lion roared at the elephant in the tiger's cage"; here we must identify the syntactic relationships between the animals if we are to infer their roles in the activity described.

A conversational speech system should have the ability to answer questions that may fall into several categories. Hypothetical questions like "If I were to go on Tuesday could I get a lower fare ticket?" are often a prelude to further transaction and establish the basis of the next request. Questions about ability like "Can you pass the salt?" are often indirect speech acts. But other times answering questions about ability is essential as speech systems are not omnipotent but perform in a limited domain. In a graphical user interface, the user can pull down menus and read options in the process of deciding how to make a choice. In a speech system, this is not possible: the application would need to continually recite all possible options because of the transient nature of speech. Instead, the user must be able to ask the system what its capabilities are, and it must be able to reply in a rational manner.

Combining clarifying protocols with the various aspects of discourse described in Chapter 9, we begin to appreciate the complexity of conversation. There may be a round of clarifying discourse to resolve ambiguities about what one party has just said. This contributes to subgoals—mutual understanding on an utterance-by-utterance basis—in the context of the discourse. But discourse is usually not about simply understanding each other; certainly in the case of a human speaking to a computer, the human wishes the computer to perform some action or service. The computer must track the discourse focus and understand the user's goals at each step while formulating its own subgoals to understand the utterance. This is very difficult, yet it provides a powerful basis for graceful conversational interaction.

SPEECH RECOGNITION AND ROBUST PARSING

One disturbing trend in speech recognition research is putting more and more linguistic knowledge into the word recognition process (this is good) but then using this knowledge to excessively constrain recognition, which leaves little

room for speaker or recognizer error. Linguistic constraints, most prominently rules of syntax or word combination probabilities, limit the perplexity of the recognition task and this results in both significantly improved accuracy and the potential for much larger recognition vocabularies. Such constraints are certainly necessary to understand fluent speech and humans use them as well. But once a language is completely specified, what happens when the human errs?

In the majority of current, experimental, large-vocabulary connected speech recognizers, omission of a single word is likely to result in no recognition result reported at all. If the recognizer is configured to hear seven or ten digit telephone numbers and the talker stops after six digits, the recognizer may report nothing or possibly only the occurrence of an error without any hint as to its nature. But consider the conversational techniques discussed earlier in this chapter as well as in Chapter 8. It is extremely unproductive for the application to remain mute or to utter unhelpfully "What was that?" A preferable alternative might be to echo the digits back to the user either in entirety or in part. Another might be to use back-channel style encouragement ("Uh-huh" spoken with a somewhat rising pitch) to indicate that the application is waiting for more digits and anticipates that the user has not yet completed his or her turn. Further misunderstandings might cause the application to explain what it thinks the user has requested and why the request is not yet complete, e.g., "I can dial the phone for you but you must specify at least seven digits for a valid number."

In the above scenario, the human made a clear mistake by speaking only six digits. But a cooperative system will not castigate the user, rather it will try to help complete the task. Perhaps the user accidently missed a digit or cannot decipher a digit in someone else's handwriting, or paused to obtain reassurance that the system was still listening and realized that a call was being placed. Such occurrences are not even limited to situations in which the human makes such an unequivocal mistake; spoken language is ripe with false starts, ill-formed grammatical constructs, and ellipsis. When these happen, is there any hope for an application using speech recognition to recover gracefully and cooperatively?

One approach, similar to that used by Conversational Desktop and described in Chapter 9, is to define the recognizer's grammar so that all possible sentence fragments appear as valid utterances. When used during the recognition phase of discourse, such an under-constrained grammar would lead to reduced recognition accuracy due to increased perplexity. But humans are effective at conversing to attain a goal and, to the extent that we can engage in question-answering, we are capable of getting acceptable performance even in the face of recognition errors [Hunnicutt *et al.* 1992] so letting some errors occur and then negotiating with the user may be more productive than insisting on near-perfect recognition before reporting *any* results.

Two approaches from MIT's Spoken Language Systems Group illustrate practical hybrid solutions. One approach reported by [Seneff 1992] employs conventional parsing based on syntactic rules, but when a well-formed parse is not found, the parser switches to a more semantically oriented mode based on analysis of keywords in the context of the task domain. Additionally, conventional syntactic parsing can be improved by weighting word choices with probability

measures. Under this approach, a grammar is augmented with word-sequence probabilities based on analysis of a corpus of utterances spoken by naive subjects attempting to access the chosen task [Hirschman *et al.* 1991].

But in natural speech these probabilities are dynamic and depend on the current state of the conversation as well as the listener's expectations of what the talker may be up to. In a fully integrated conversational application, pragmatic information such as identification of a partially completed plan or detection of elements of a script could tune word probabilities based on expectations of what is likely to follow. Similarly a focus model could suggest heightened probabilities of words relevant to attributes of entities recently discussed in addition to resolving anaphoric references. Domain knowledge plays a role as well, from simplistic awareness of the number of digits in telephone numbers to knowledge of the acceleration and turning capabilities of various types of aircraft in an air traffic control scenario.

Part of the difficulty with flexible parsing is the excessive degree of isolation between the application, its discourse system, and the speech recognition component. For many current systems the recognizer is given a language model in whatever form it requires; it then listens to speech and returns a string of text to the application, which must parse it *again* to know how to interpret the words meaningfully. Not only has syntactic information been needlessly lost when reporting the recognized speech as a string of words, but it also may be detrimental to strip the representation of any remaining acoustic evidence such as the recognizer's degree of certainty or possible alternate choices of recognition results. How can partial recognition results be reported to the parser? Perhaps several noun phrases were identified but the verb was not, confounding classification of the nouns into the possible roles that might be expressed in a framed-based representation.

These observations are intended to suggest that despite this book's portrayal of the various layers of language understanding and generation as distinct entities, they still must be tightly woven into a coherent whole. Isolation of the word identification portion of discourse understanding into a well bounded "speech recognizer" component cannot in the long run support sophisticated conversational systems. Knowledge must be communicated easily across components, and analysis must be flexible and based on dynamic conversation constraints.

PROSODY

Prosody refers to the spoken style of discourse independent of lexical content, and it includes several aspects of how we speak. *Intonation* is the tune of an utterance: how we modulate F0 to change the pitch of our speech. Intonation operates at a sentence or phrase level; the rising tune of a yes-or-no question immediately differentiates it from the falling tune of a declarative statement. Intonation also helps to convey the stress of words and syllables within a sentence as stressed syllables are spoken with a pitch higher or lower than normal, and specific words are emphasized—an important aspect of communicating intent—by stressing

their stressed syllables even more. *Phrasing* is the breaking of speech into groups, how many words we squeeze into an utterance before stopping for breath, or how we may speak a few words much more slowly for emphasis. *Meter* is carried largely by the duration of individual syllables, stressed syllables being generally longer than unstressed ones. Among stressed syllables, some syllables are more stressed than others resulting in a meter scheme across an entire phrase. Syllabic stress is also reinforced by intonation, and intonation helps convey phrasing as well, so all these elements intermingle during speech production.

In a nutshell, prosody encompasses the majority of information lost when comparing an utterance to its transcription, and much of the richness of speech is conveyed by exactly this nonlexical means of expression.[1] Although a back alley in the field, intonation has been explored extensively by linguists attempting to categorize it and understand how it is systematically and predictably used in spoken language [Ladd 1978, Bolinger 1982]. Pierrehumbert and Hirschberg suggest a grammar relating pitch accents to aspects of meaning [Pierrehumbert and Hirschberg 1990]. In their interpretation, intonation indicates factors such as the salience of an utterance, the speaker's degree of involvement or belief in the facts being proposed, and paths of inference the speaker wishes to emphasize. Although prosody is what differentiates spoken from written language, few conversational speech systems have attempted to exploit its expressiveness.

We notice the prosody of synthetic speech mostly by its absence or occasional misplaced syllabic stress. In Chapter 5 intonation was discussed in two contexts: differentiating alternate syntactic forms of words spelled identically but stressed distinctly and differentiating declarative and interrogative sentences by overall pitch contours. This analysis did not extend beyond individual sentences. The difficulty with applying prosodic cues to human-authored text lies chiefly with the fact that intonation conveys so many levels of meaning. Each sentence would need to be parsed to identify the syntactic role of some of the confusing forms, e.g., "live," "elaborate," and "conduct." But without understanding the semantic structure of a sentence, it is difficult to identify which word should be most stressed. And a sentence must be considered in a discourse context to correctly use prosody to convey the difference between given and new information (new is more heavily stressed).

Better use of prosodic cues for speech synthesis can be made by applications employing synthesis-from-concept techniques, i.e., generating utterances in a discourse based on internal models of the discourse and an associated task domain. Witten's Telephone Enquiry Service allowed application programmers to explicitly program intonation into text to be synthesized by marking it with special codes [Witten and Madams 1977]. Davis and Hirschberg used intonational cues to improve the expressiveness of the Direction Assistance program described in Chapter 6 [Davis and Hirschberg 1988]. They used intonational cues to convey given and new information and to cue listeners to shift the focus of discourse by

[1] Other information lost by the transcription are the speaker's identity (carried in part by the timbre or spectral characteristics of one's speech), accent, and emotional state.

increasing pitch range. Cahn explored the use of prosody and other acoustical cues to convey affect using synthesized speech; such prosodic cues included pitch range and speech rate as well as variations in how phonemes are realized and acoustical parameters of the synthesizer's vocal tract model [Cahn 1990]. Similar work was also reported by [Murray et al. 1988].

Attempts to employ prosodic cues for language understanding have likewise been limited. Lea proposed a wide ranging framework to take advantage of prosody during speech recognition, including, in part, the observation that stressed syllables are more phonetically invariant than unstressed ones (unstressed syllables are subject to reduction, e.g., to a schwa) [Lea 1980]. More recently, Waibel investigated the role of prosody in speech recognition suggesting that it could be used as a cue to word boundaries [Waibel 1988].

Grunt, described in Chapter 9, detected monosyllabic questions and responded to them according to its discourse model. Daly and Zue analyzed longer utterances' pitch contours in an attempt to differentiate questions of the sort which expect yes-or-no answers from Wh- questions [Daly and Zue 1990]. They achieved significant (though imperfect) results largely by looking at the final boundary tone or pitch excursion at the end of an utterance.

These are but small steps into the realm of intonation. The role of intonation in language is far from completely understood, and detection of prosody may be acoustically difficult especially in the absence of clear lexical analysis of an utterance. Nonetheless, prosody is an essential and powerful component of both speaking and listening and key for more natural dialog systems.

WHAT NEXT?

This brief chapter has suggested some ways in which we have only begun to tap into the richness and robustness of conversation as a potential means of interacting with computer systems. But this is meant to be an optimistic note not a pessimistic one. The very richness of the human voice and its pervasiveness across so much of our expression mean that any ability to exploit it has potential for rewards.

Although the speech technologies described in this book are feeble when compared with human capabilities, the case studies demonstrate that with careful matching of technology to task and careful crafting of interaction techniques successful voice applications are already a reality. Speech is so powerful that even applications of very limited ability can be extremely effective in specific situations.

Speech technologies are improving rapidly, assisted by ever-increasing computer power. Raw technologies are an enabling factor and guarantee success only with careful consideration of how and when to apply them. That modern technologies impinge on only the most basic aspects of our attempts to converse clearly shows the power that conversational computing systems are one day destined to achieve.