CHAPTER 1

Introduction

From human prehistory to the new media of the future, speech communication has been and will be the dominant mode of human social bonding and information exchange. The spoken word is now extended, through technological mediation such as telephony, movies, radio, television, and the Internet. This trend reflects the primacy of spoken communication in human psychology.

In addition to human-human interaction, this human preference for spoken language communication finds a reflection in human-machine interaction as well. Most computers currently utilize a *graphical user interface* (GUI), based on graphically represented interface objects and functions such as windows, icons, menus, and pointers. Most computer operating systems and applications also depend on a user's keyboard strokes and mouse clicks, with a display monitor for feedback. Today's computers lack the fundamental human abilities to speak, listen, understand, and learn. Speech, supported by other natural modalities, will be one of the primary means of interfacing with computers. And, even before speech-based interaction reaches full maturity, applications in home, mobile, and office segments are incorporating spoken language technology to change the way we live and work.

A spoken language system needs to have both speech recognition and speech synthesis capabilities. However, those two components by themselves are not sufficient to build a useful spoken language system. An understanding and dialog component is required to manage interactions with the user; and domain knowledge must be provided to guide the system's interpretation of speech and allow it to determine the appropriate action. For all these components, significant challenges exist, including robustness, flexibility, ease of integration, and engineering efficiency. The goal of building commercially viable spoken language systems has long attracted the attention of scientists and engineers all over the world. The purpose of this book is to share our working experience in developing advanced spoken language processing systems with both our colleagues and newcomers. We devote many chapters to systematically introducing fundamental theories and to highlighting what works well based on numerous lessons we learned in developing Microsoft's spoken language systems.

1.1. MOTIVATIONS

What motivates the integration of spoken language as the primary interface modality? We present a number of scenarios, roughly in order of expected degree of technical challenges and expected time to full deployment.

1.1.1. Spoken Language Interface

There are generally two categories of users who can benefit from adoption of speech as a control modality in parallel with others, such as the mouse, keyboard, touch-screen, and joystick. For novice users, functions that are conceptually simple should be directly accessible. For example, raising the voice output volume under software control on the desktop speakers, a conceptually simple operation, in some GUI systems of today requires opening one or more windows or menus, and manipulating sliders, check-boxes or other graphical elements. This requires some knowledge of the system's interface conventions and structures. For the novice user, to be able to say *raise the volume* would be more direct and natural. For expert users, the GUI paradigm is sometimes perceived as an obstacle or nuisance and shortcuts are sought. Frequently these shortcuts allow the power user's hands to remain on the keyboard or mouse while mixing content creation with system commands. For example, an operator of a graphic design system for CAD/CAM might wish to specify a text formatting command while keeping the pointer device in position over a selected screen element.

Speech has the potential to accomplish these functions more powerfully than keyboard and mouse clicks. Speech becomes more powerful when supplemented by information streams encoding other dynamic aspects of user and system status, which can be resolved by the semantic component of a complete multi-modal interface. We expect such multimodal interactions to proceed based on more complete user modeling, including speech, visual orientation, natural and device-based gestures, and facial expression, and these will be coordinated with detailed system profiles of typical user tasks and activity patterns.

In some situations you must rely on speech as an input or output medium. For example, with wearable computers, it may be impossible to incorporate a large keyboard. When driving, safety is compromised by any visual distraction, and hands are required for controlling the vehicle. The ultimate speech-only device, the telephone, is far more widespread than the PC. Certain manual tasks may also require full visual attention to the focus of the work. Finally, spoken language interfaces offer obvious benefits for individuals challenged with a variety of physical disabilities, such as loss of sight or limitations in physical motion and motor skills. Chapter 18 contains detailed discussion on spoken language applications.

1.1.2. Speech-to-speech Translation

Speech-to-speech translation has been depicted for decades in science fiction stories. Imagine questioning a Chinese-speaking conversational partner by speaking English into an unobtrusive device, and hearing real-time replies you can understand. This scenario, like the spoken language interface, requires both speech recognition and speech synthesis technology. In addition, sophisticated multilingual spoken language understanding is needed. This highlights the need for tightly coupled advances in speech recognition, synthesis, and understanding systems, a point emphasized throughout this book.

1.1.3. Knowledge Partners

The ability of computers to process spoken language as proficient as humans will be a landmark to signal the arrival of truly intelligent machines. Alan Turing [29] introduced his famous *Turing test*. He suggested a game, in which a computer's use of language would form the criterion for intelligence. If the machine could win the game, it would be judged intelligent. In Turing's game, you play the role of an interrogator. By asking a series of questions via a teletype, you must determine the identity of the other two participants: a machine and a person. The task of the machine is to fool you into believing it is a person by responding as a person to your questions. The task of the other person is to convince you the other participant is the machine. The critical issue for Turing was that using language as humans do is sufficient as an operational test for intelligence.

The ultimate use of spoken language is to pass the Turing test in allowing future extremely intelligent systems to interact with human beings as knowledge partners in all aspects of life. This has been a staple of science fiction, but its day will come. Such systems require reasoning capabilities and extensive world knowledge embedded in sophisticated search, communication, and inference tools that are beyond the scope of this book. We expect that spoken language technologies described in this book will form the essential enabling mechanism to pass the Turing test.

1.2. SPOKEN LANGUAGE SYSTEM ARCHITECTURE

Spoken language processing refers to technologies related to speech recognition, text-to-speech, and spoken language understanding. A spoken language system has at least one of the following three subsystems: a speech recognition system that converts speech into words, a text-to-speech system that conveys spoken information, and a spoken language understanding system that maps words into actions and that plans system-initiated actions

There is considerable overlap in the fundamental technologies for these three subareas. Manually created rules have been developed for spoken language systems with limited success. But, in recent decades, data-driven statistical approaches have achieved encouraging results, which are usually based on modeling the speech signal using well-defined statistical algorithms that can automatically extract knowledge from the data. The data-driven approach can be viewed fundamentally as a pattern recognition problem. In fact, speech recognition, text-to-speech conversion, and spoken language understanding can all be regarded as pattern recognition problems. The patterns are either recognized during the runtime operation of the system or identified during system construction to form the basis of runtime generative models such as prosodic templates needed for text to speech synthesis. While we use and advocate a statistical approach, we by no means exclude the knowledge engineering approach from consideration. If we have a good set of rules in a given problem area, there is no need to use a statistical approach at all. The problem is that, at time of this writing, we do not have enough knowledge to produce a complete set of high-quality rules. As scientific and theoretical generalizations are made from data collected to construct data-driven systems, better rules may be constructed. Therefore, the rule-based and statistical approaches are best viewed as complementary.

1.2.1. Automatic Speech Recognition

A source-channel mathematical model described in Chapter 3 is often used to formulate speech recognition problems. As illustrated in Figure 1.1, the speaker's mind decides the source word sequence $\mathbf{W}$ that is delivered through his/her text generator. The source is passed through a noisy communication channel that consists of the speaker's vocal apparatus to produce the speech waveform and the speech signal processing component of the speech recognizer. Finally, the speech decoder aims to decode the acoustic signal $\mathbf{X}$ into a word sequence $\hat{\mathbf{W}}$, which is hopefully close to the original word sequence $\mathbf{W}$.

A typical practical speech recognition system consists of basic components shown in the dotted box of Figure 1.2. Applications interface with the decoder to get recognition results that may be used to adapt other components in the system. *Acoustic models* include the representation of knowledge about acoustics, phonetics, microphone and environment variability, gender and dialect differences among speakers, etc. *Language models* refer to a system's knowledge of what constitutes a possible word, what words are likely to co-occur, and in what sequence. The semantics and functions related to an operation a user may wish to perform may also be necessary for the language model. Many uncertainties exist in these

areas, associated with speaker characteristics, speech style and rate, recognition of basic speech segments, possible words, likely words, unknown words, grammatical variation, noise interference, nonnative accents, and confidence scoring of results. A successful speech recognition system must contend with all of these uncertainties. But that is only the beginning. The acoustic uncertainties of the different accents and speaking styles of individual speakers are compounded by the lexical and grammatical complexity and variations of spoken language, which are all represented in the language model.

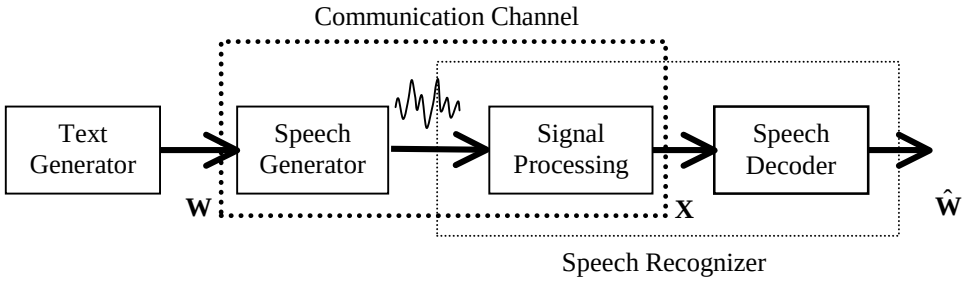

Figure 1.1 A source-channel model for a speech recognition system [15].

The speech signal is processed in the signal processing module that extracts salient feature vectors for the decoder. The decoder uses both acoustic and language models to generate the word sequence that has the maximum posterior probability for the input feature vectors. It can also provide information needed for the adaptation component to modify either the acoustic or language models so that improved performance can be obtained.

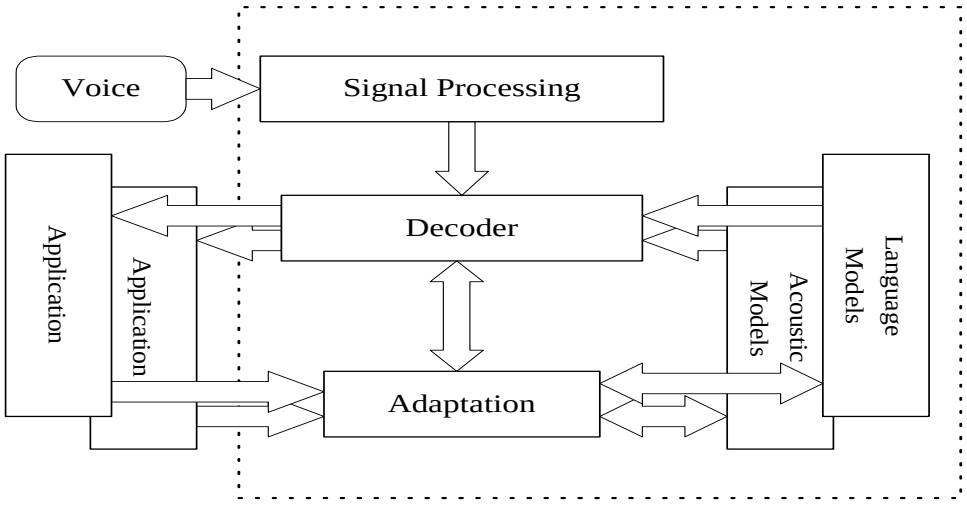

Figure 1.2 Basic system architecture of a speech recognition system [12].

1.2.2. Text-to-Speech Conversion

The term *text-to-speech*, often abbreviated as TTS, is easily understood. The task of a text-to-speech system can be viewed as speech recognition in reverse – a process of building a machinery system that can generate human-like speech from any text input to mimic human speakers. TTS is sometimes called *speech synthesis*, particularly in the engineering community.

The conversion of words in written form into speech is nontrivial. Even if we can store a huge dictionary for most common words in English; the TTS system still needs to deal with millions of names and acronyms. Moreover, in order to sound natural, the intonation of the sentences must be appropriately generated.

The development of TTS synthesis can be traced back to the 1930s when Dudley's *Voder*, developed by Bell Laboratories, was demonstrated at the World's Fair [18]. Taking the advantage of increasing computation power and storage technology, TTS researchers have been able to generate high quality commercial multilingual text-to-speech systems, although the quality is inferior to human speech for general-purpose applications.

The basic components in a TTS system are shown in Figure 1.3. The text analysis component normalizes the text to the appropriate form so that it becomes speakable. The input can be either raw text or tagged. These tags can be used to assist text, phonetic, and prosodic analysis. The phonetic analysis component converts the processed text into the corresponding phonetic sequence, which is followed by prosodic analysis to attach appropriate pitch and duration information to the phonetic sequence. Finally, the speech synthesis component takes the parameters from the fully tagged phonetic sequence to generate the corresponding speech waveform.

Various applications have different degrees of knowledge about the structure and content of the text that they wish to speak so some of the basic components shown in Figure 1.3 can be skipped. For example, some applications may have certain broad requirements such as rate and pitch. These requirements can be indicated with simple command tags appropriately located in the text. Many TTS systems provide a set of markups (tags), so the text producer can better express their semantic intention. An application may know a lot about the structure and content of the text to be spoken to greatly improve speech output quality. For engines providing such support, the *text analysis* phase can be skipped, in whole or in part. If the system developer knows the orthographic form, the phonetic analysis module can be skipped as well. The prosodic analysis module assigns a numeric duration to every phonetic symbol and calculates an appropriate pitch contour for the utterance or paragraph. In some cases, an application may have prosodic contours precalculated by some other process. This situation might arise when TTS is being used primarily for compression, or the prosody is *transplanted* from a real speaker's utterance. In these cases, the quantitative prosodic controls can be treated as special tagged field and sent directly along with the phonetic stream to speech synthesis for voice rendition.

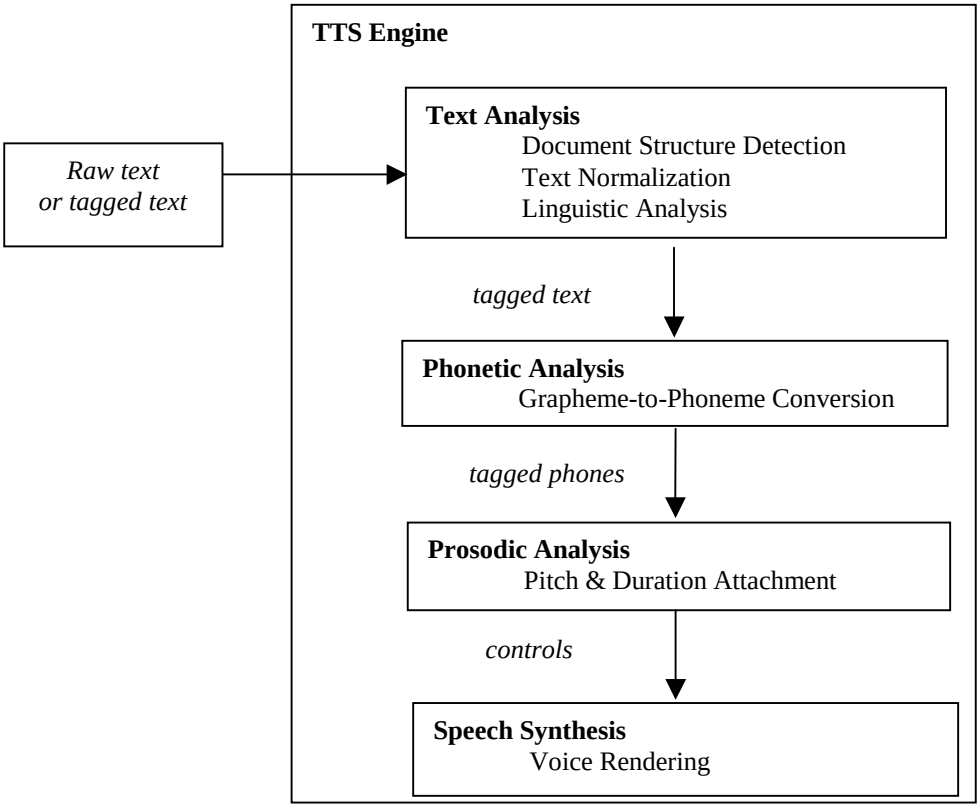

Figure 1.3 Basic system architecture of a TTS system.

1.2.3. Spoken Language Understanding

Whether a speaker is inquiring about flights to Seattle, reserving a table at a Pittsburgh restaurant, dictating an article in Chinese, or making a stock trade, a spoken language understanding system is needed to interpret utterances in context and carry out appropriate actions. lexical, syntactic, and semantic knowledge must be applied in a manner that permits cooperative interaction among the various levels of acoustic, phonetic, linguistic, and application knowledge in minimizing uncertainty. Knowledge of the characteristic vocabulary, typical syntactic patterns, and possible actions in any given application context for both interpretation of user utterances and planning system activity are the heart and soul of any spoken language understanding system.

A schematic of the typical spoken language understanding systems is shown in Figure 1.4. Such a system typically has a speech recognizer and a speech synthesizer for basic

speech input and output, *sentence interpretation* component to parse the speech recognition results into semantic forms, which often needs *discourse analysis* to track context and resolve ambiguities. *Dialog Manager* is the central component that communicates with applications and the spoken language understanding modules such as discourse analysis, sentence interpretation, and message generation.

While most components of the system may be partly or wholly generic, the dialog manager controls the flow of conversation tied to the action. The dialog manager is responsible for providing status needed for formulating responses, and maintaining the system's idea of the state of the discourse. The discourse state records the current transaction, dialog goals that motivated the current transaction, current objects in focus (temporary center of attention), the object history list for resolving dependent references, and other status information. The discourse information is crucial for semantic interpretation to interpret utterances in context. Various systems may alter the flow of information implied in Figure 1.4. For example, the dialog manager or the semantic interpretation module may be able to supply contextual discourse information or pragmatic inferences, as feedback to guide the recognizer's evaluation of hypotheses at the earliest level of search. Another optimization might be achieved by providing for shared grammatical resources between the *message generation* and *semantic interpretation* components.

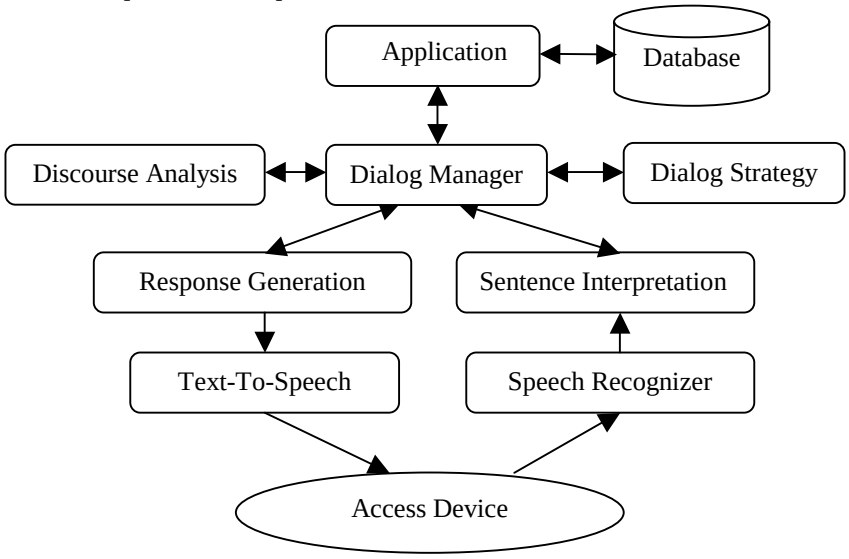

Figure 1.4 Basic system architecture of a spoken language understanding system.

1.3. BOOK ORGANIZATION

We attempt to present a comprehensive introduction to spoken language processing, which includes not only fundamentals but also a practical guide to build a working system that requires knowledge in speech signal processing, recognition, text-to-speech, spoken language understating, and application integration. Since there is considerable overlap in the fundamental spoken language processing technologies, we have devoted Part I to the foundations needed. Part I contains background on speech production and perception, probability and information theory, and pattern recognition. Parts II, III, IV, and V include chapters on speech processing, speech recognition, speech synthesis, and spoken language systems, respectively. A reader with sufficient background can skip Part I, referring back to it later as needed. For example, the discussion of speech recognition in Part III relies on the pattern recognition algorithms presented in Part I. Algorithms that are used in several chapters within Part III are also included in Parts I and II. Since the field is still evolving, at the end of each chapter we provide a historical perspective and list further readings to facilitate future research.

1.3.1. Part I: Fundamental Theory

Chapters 2 to 4 provide readers with a basic theoretic foundation to better understand techniques that are widely used in modern spoken language systems. These theories include the essence of linguistics, phonetics, probability theory, information theory, and pattern recognition. These chapters prepare you fully to understand the rest of the book.

Chapter 2 discusses the basic structure of spoken language including speech science, phonetics, and linguistics. Chapter 3 covers probability theory and information theory, which form the foundation of modern pattern recognition. Many important algorithms and principles in pattern recognition and speech coding are derived based on these theories. Chapter 4 introduces basic pattern recognition, including decision theory, estimation theory, and a number of algorithms widely used in speech recognition. Pattern recognition forms the core of most of the algorithms used in spoken language processing.

1.3.2. Part II: Speech Processing

Part II provides you with necessary speech signal processing knowledge that is critical to spoken language processing. Most of what discuss here is traditionally the subject of electrical engineering.

Chapters 5 and 6 focus on how to extract useful information from the speech signal. The basic principles of digital signal processing are reviewed and a number of useful representations for the speech signal are discussed. Chapter 7 covers how to compress these representations for efficient transmission and storage.

1.3.3. Part III: Speech Recognition

Chapters 8 to 13 provide you with an in-depth look at modern speech recognition systems. We highlight techniques that have been proven to work well in building real systems and explain in detail how and why these techniques work from both theoretic and practical perspectives.

Chapter 8 introduces hidden Markov models, the most prominent technique used in modern speech recognition systems. Chapters 9 and 11 deal with acoustic modeling and language modeling respectively. Because environment robustness is critical to the success of practical systems, we devote Chapter 10 to discussing how to make systems less affected by environment noises. Chapters 12 and 13 deal in detail how to efficiently implement the decoder for speech recognition. Chapter 12 discusses a number of basic search algorithms, and Chapter 13 covers large vocabulary speech recognition. Throughout our discussion, Microsoft's Whisper speech recognizer is used as a case study to illustrate the methods introduced in these chapters.

1.3.4. Part IV: Text-to-Speech Systems

In Chapters 14 through 16, we discuss proven techniques in building text-to-speech systems. The synthesis system consists of major components found in speech recognition systems, except that they are in the reverse order.

Chapters 14 covers the analysis of written documents and the text needed to support spoken rendition, including the interpretation of audio markup commands, interpretation of numbers and other symbols, and conversion from orthographic to phonetic symbols. Chapter 15 focuses on the generation of pitch and duration controls for linguistic and emotional effect. Chapter 16 discusses the implementation of the synthetic voice, and presents algorithms to manipulate a limited voice data set to support a wide variety of pitch and duration controls required by the text analysis. We highlight the importance of trainable synthesis, with Microsoft's Whistler TTS system as an example.

1.3.5. Part V: Spoken Language Systems

As discussed in Section 1.1, spoken language applications motivate spoken language R&D. The central component is the spoken language understanding system. Since it is closely related to applications, we group it together with application and interface design.

Chapter 17 covers spoken language understanding. The output of the recognizer requires interpretation and action in a particular application context. This chapter details useful strategies for dialog management, and the coordination of all the speech and system resources to accomplish a task for a user. Chapter 18 concludes the book with a discussion of important principles for building spoken language interfaces and applications, including general human interface design goals, and interaction with nonspeech interface modalities in

specific application contexts. Microsoft's MiPad is used as a case study to illustrate a number of issues in developing spoken language applications.

1.4. TARGET AUDIENCES

This book can serve a variety of audiences:

Integration engineers: Software engineers who want to build spoken language systems, but who do not want to learn all about speech technology internals, will find plentiful relevant material, including application design and software interfaces. Anyone with a professional interest in aspects of speech applications, integration, and interfaces can also achieve enough understanding of how the core technologies work, to allow them to take full advantage of state-of-the-art capabilities.

Speech technology engineers: Engineers and researchers working on various subspecialties within the speech field will find this book a useful guide to understanding related technologies in sufficient depth to help them gain insight on where their own approaches overlap with, or diverge from, their neighbors' common practice.

Graduate students: This book can serve as a primary textbook in a graduate or advanced undergraduate speech analysis or language engineering course. It can serve as a supplementary textbook in some applied linguistics, digital signal processing, computer science, artificial intelligence, and possibly psycholinguistics course.

Linguists: As the practice of linguistics increasingly shifts to empirical analysis of real-world data, students and professional practitioners alike should find a comprehensive introduction to the technical foundations of computer processing of spoken language helpful. The book can be read at different levels and through different paths, for readers with differing technical skills and background knowledge.

Speech Scientists: Researchers engaged in professional work on issues related to normal or pathological speech may find this complete exposition of the state-of-the-art in computer modeling of generation and perception of speech interesting.

Business planners: Increasingly, business and management functions require some level of insight into the vocabulary and common practices of technology development. While not the primary audience, managers, marketers and others with planning responsibilities and sufficient technical background will find portions of this book useful in evaluating competing proposals, and in making *buy-or-develop* business decisions related to the speech technology components.

1.5. HISTORICAL PERSPECTIVE AND FURTHER READING

Spoken language processing is a diverse field that relies on knowledge of language at the levels of signal processing, acoustics, phonology, phonetics, syntax, semantics, pragmatics, and discourse. The foundations of spoken language processing lie in computer science, electrical engineering, linguistics, and psychology. In the 1970s an ambitious speech understand-

ing project was funded by DARPA, which led to many seminal systems and technologies [17]. A number of human language technology projects funded by DARPA in the 1980s and '90s further accelerated the progress, as evidenced by many papers published in *The Proceedings of the DARPA Speech and Natural Language/Human Language Workshop*. The field is still rapidly progressing and there are a number of excellent review articles and introductory books. We provide a brief list here. More detailed references can be found within each chapter of this book. Gold and Morgan's *Speech and Audio Signal Processing* [10] has a strong historical perspective on spoken language processing.

Hyde [14] and Reddy [24] provided an excellent review of early speech recognition work in the 1970s. Some of the principles are still applicable to today's speech recognition research. Waibel and Lee assembled many seminal papers in *Readings in Speech Recognition Speech Recognition* [31]. There are a number of excellent books on modern speech recognition [1, 13, 15, 22, 23].

Where does the state of the art speech recognition system stand today? A number of different recognition tasks can be used to compare the recognition error rate of people vs. machines. Table 1.1 shows five recognition tasks with vocabularies ranging from 10 to 5,000 words speaker-independent continuous speech recognition. The Wall Street Journal Dictation (WSJ) Task has 5000-word vocabulary as a continuous dictation application for the WSJ articles. In Table 1.1, the error rate for machines is based on state of the art speech recognizers such as systems described in Chapter 9, and the error rate of humans is based a range of subjects tested on the similar task. We can see the error rate of humans is at least 5 times smaller than machines except for the sentences that are generated from a trigrm language model, where the sentences have the perfect match between humans and machines so humans cannot use high-level knowledge that is not used in machines[1].

Table 1.1 Word error rate comparisons between human and machines on similar tasks.

Tasks	Vocabulary	Humans	Machines
Connected digits	10	0.009%	0.72%
Alphabet letters	26	1%	5%
Spontaneous telephone speech	2000	3.8%	36.7%
WSJ with clean speech	5000	0.9%	4.5%
WSJ with noisy speech (10-db SNR)	5000	1.1%	8.6%
Clean speech based on trigram sentences	20,000	7.6%	4.4%

We can see that humans are far more robust than machines for normal tasks. The error rate for machine spontaneous conversational telephone speech recognition is above 35%, more than a factor 10 higher than humans on the similar task. In addition, the error rate of humans does not increase as dramatic as machines when the environment becomes noisy (from quite to 10-db SNR environments on the WSJ task). The relative error rate of humans

[1] Some of these experiments were conducted at Microsoft with only a small number of human subjects (3-5 people), which is not statistically significant. Nevertheless, it sheds some interesting insight on the performance between humans and machines.

increases from 0.9% to 1.1% (1.2 times), while the error rate of CSR systems increases from 4.5% to 8.6% (1.9 times). One interesting experiment is that when we generated sentences using the WSJ trigram language model (cf Chapter 11), the difference between humans and machines disappears (the last row in Table 1.1). In fact, the error rate of humans is even higher than machines. This is because both humans and machines have the same high-level syntactic and semantic models. The test sentences are somewhat random to humans but perfect to machines that used the same trigram model for decoding. This experiment indicates humans make more effective use of semantic and syntactic constraints for improved speech recognition in meaningful conversation. In addition, machines don't have attention problems as humans on random sentences.

Fant [7] gave an excellent introduction to speech production. Early reviews of text-to-speech synthesis can be found in [3, 8, 9]. Sagisaka [26] and Carlson [6] provide more recent reviews of progress in speech synthesis. A more detailed treatment can be found in [19, 30].

Where does the state of the art text to speech system stand today? Unfortunately, like speech recognition, this is not a solved problem either. Although machine storage capabilities are improving, the quality remains a challenge for many researchers if we want to pass the Turing test.

Spoken language understanding is deeply rooted in speech recognition research. There are a number of good books on spoken language understanding [2, 5, 16]. Manning and Schutz [20] focuses on statistical methods for language understanding. Like Waibel and Lee, Grosz et al. assembled many foundational papers in *Readings in Natural Language Processing* [11]. More recent reviews of progress in spoken language understanding can be found in [25, 28]. Related spoken language interface design issues can be found in [4, 21, 27, 32].

In comparison to speech recognition and text to speech, spoken language understanding is further away from approaching the level of humans, especially for general-purpose spoken language applications.

A number of good conference proceedings and journals report the latest progress in the field. Major results on spoken language processing are presented at *the International Conference on Acoustics, Speech and Signal Processing (ICASSP), International Conference on Spoken Language Processing (ICSLP), Eurospeech Conference, the DARPA Speech and Human Language Technology Workshops*, and many workshops organized by *the European Speech Communications Associations (ESCA*) and *IEEE Signal Processing Society*. Journals include *IEEE Transactions on Speech and Audio Processing, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Computer Speech and Language, Speech Communications,* and *Journal of Acoustical Society of America (JASA).* Research results can also be found at computational linguistics conferences such as *the Association for Computational Linguistics (ACL), International Conference on Computational Linguistics (COLING),* and *Applied Natural Language Processing (ANLP).* The journals *Computational Linguistics* and *Natural Language Engineering* cover both theoretical and practical applications of language research. *Speech Recognition Update* published by TMA Associates is an excellent industry newsletter on spoken language applications.

REFERENCES

[1] Acero, A., *Acoustical and Environmental Robustness in Automatic Speech Recognition*, 1993, Boston, MA, Kluwer Academic Publishers.

[2] Allen, J., *Natural Language Understanding*, 2nd ed, 1995, Menlo Park CA, The Benjamin/Cummings Publishing Company.

[3] Allen, J., M.S. Hunnicutt, and D.H. Klatt, *From Text to Speech: the MITalk System*, 1987, Cambridge, UK, University Press.

[4] Balentine, B. and D. Morgan, *How to Build a Speech Recognition Application*, 1999, Enterprise Integration Group.

[5] Bernsen, N., H. Dybkjar, and L. Dybkjar, *Designing Interactive Speech Systems*, 1998, Springer.

[6] Carlson, R., "Models of Speech Synthesis" in *Voice Communications Between Humans and Machines. National Academy of Sciences*, D.B. Roe and J.G. Wilpon, eds. 1994, Washington, D.C., National Academy of Sciences.

[7] Fant, G., *Acoustic Theory of Speech Production*, 1970, The Hague, NL, Mouton.

[8] Flanagan, J., *Speech Analysis Synthesis and Perception*, 1972, New York, Springer-Verlag.

[9] Flanagan, J., "Voices Of Men And Machines," *Journal of Acoustical Society of America*, 1972, **51**, pp. 1375.

[10] Gold, B. and N. Morgan, *Speech and Audio Signal Processing: Processing and Perception of Speech and Music*, 2000, John Wiley and Sons.

[11] Grosz, B., F.S. Jones, and B.L. Webber, *Readings in Natural Language Processing*, 1986, Morgan Kaufmann, Los Altos, CA.

[12] Huang, X., *et al.*, "From Sphinx-II to Whisper - Make Speech Recognition Usable" in *Automatic Speech and Speaker Recognition*, C.H. Lee, F.K. Soong, and K.K. Paliwal, eds. 1996, Norwell, MA, Klewer Academic Publishers.

[13] Huang, X.D., Y. Ariki, and M.A. Jack, *Hidden Markov Models for Speech Recognition*, 1990, Edinburgh, U.K., Edinburgh University Press.

[14] Hyde, S.R., "Automatic Speech Recognition: Literature, Survey, And Discussion" in *Human Communication, A Unified Approach*, E.E. David and P.B. Denes, eds. 1972, McGraw Hill, New York.

[15] Jelinek, F., *Statistical Methods for Speech Recognition*, Language, Speech, and Communication, 1998, Cambridge, MA, MIT Press.

[16] Jurafsky, D. and J. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, 2000, Upper Saddle River, NJ, Prentice Hall.

[17] Klatt, D., "Review of the ARPA Speech Understanding Project," *Journal of Acoustical Society of America*, 1977, **62**(6), pp. 1324-1366.

[18] Klatt, D., "Review of Text-to-Speech Conversion for English," *Journal of Acoustical Society of America*, 1987, **82**, pp. 737-793.

[19] Kleijn, W.B. and K.K. Paliwal, *Speech Coding and Synthesis*, 1995, Amsterdam, Netherlands, Elsevier.

[20] Manning, C. and H. Schutze, *Foundations of Statistical Natural Language Processing*, 1999, MIT Press, Cambridge, USA.

[21] Markowitz, J., *Using Speech Recognition*, 1996, Prentice Hall.

[22] Mori, R.D., *Spoken Dialogues with Computers*, 1998, London, UK, Academic Press.

[23] Rabiner, L.R. and B.H. Juang, *Fundamentals of Speech Recognition*, May, 1993, Prentice-Hall.

[24] Reddy, D.R., "Speech Recognition by Machine: A Review," *IEEE Proc.*, 1976, **64**(4), pp. 502-531.

[25] Sadek, D. and R.D. Mori, "Dialogue Systems" in *Spoken Dialogues with Computers*, R.D. Mori, Editor 1998, London, UK, pp. 523-561, Academic Press.

[26] Sagisaka, Y., "Speech Synthesis from Text," *IEEE Communication Magazine*, 1990(1).

[27] Schmandt, C., *Voice Communication with Computers*, 1994, New York, NY, Van Nostrand Reinhold.

[28] Seneff, S., "The Use of Linguistic Hierarchies in Speech Understanding," *Int. Conf. on Spoken Language Processing*, 1998, Sydney, Australia.

[29] Turing, A.M., "Computing Machinery and Intelligence," *Mind*, 1950, **LIX**(236), pp. 433-460.

[30] van Santen, J., *et al.*, *Progress in Speech Synthesis*, 1997, New York, Springer-Verlag.

[31] Waibel, A.H. and K.F. Lee, *Readings in Speech Recognition*, 1990, San Mateo, CA, Morgan Kaufman Publishers.

[32] Weinschenk, S. and D. Barker, *Designing Effective Speech Interfaces*, 2000, John Wiley & Sons, Inc.