1 Introduction

This is a book about getting computers to read out loud. It is therefore about three things: the process of reading, the process of speaking, and the issues involved in getting computers (as opposed to humans) to do this. This field of study is known both as **speech synthesis**, that is the "synthetic" (computer) generation of speech, and as **text-to-speech** or **TTS**; the process of converting written text into speech. It complements other language technologies such as **speech recognition**, which aims to convert speech into text, and **machine translation**, which converts writing or speech in one language into writing or speech in another.

I am assuming that most readers have heard some synthetic speech in their life. We experience this in a number of situations; some telephone information systems have automated speech response, speech synthesis is often used as an aid to the disabled, and Professor Stephen Hawking has probably contributed more than anyone else to the direct exposure of (one particular type of) synthetic speech. The *idea* of artificially generated speech has of course been around for a long time – hardly any science-fiction film is complete without a talking computer of some sort. In fact science fiction has had an interesting effect on the field and our impressions of it. Sometimes (less technically aware) people believe that perfect speech synthesis exists because they "heard it on *Star Trek*".[1] Often makers of science-fiction films fake the synthesis by using an actor, although usually some processing is added to the voice to make it sound "computerised". Some actually use real speech-synthesis systems, but interestingly these are usually not state-of-the-art systems, since these sound too natural, and may mislead the viewer.[2] One of the genuine attempts to predict how synthetic voices will sound is the computer HAL in the film *2001: A Space Odyssey* [265]. The fact that this computer spoke with a calm and near-humanlike-voice gave rise to the sense of genuine intelligence in the machine. While many parts of this film were wide of the mark (especially the ability of HAL to understand, rather than just recognise, human speech), the makers of the film just about got it right in predicting how good computer voices would be in the year in question.

Speech synthesis has progressed remarkably in recent years, and it is no longer the case that state-of-the-art systems sound overtly mechanical and robotic. That said, it

[1] Younger readers please substitute the in-vogue science-fiction series of the day.

[2] In much the same way, when someone types the wrong password on a computer, the screen starts flashing and saying "access denied". Some even go so far as to have a siren sounding. Those of us who use computers know this never happens, but in a sense we go along with the exaggeration as it adds to the drama.

is normally fairly easy to tell that it is a computer talking rather than a human, so substantial progress has still to be made. When assessing a computer's ability to speak, one fluctuates between two judgments. On the one hand, it is tempting to paraphrase Dr Johnson's famous remark [61] "Sir, a talking computer is like a dog's walking on his hind legs. It is not done well; but you are surprised to find it done at all." Indeed, even as an experienced text-to-speech researcher who has listened to more synthetic speech than could be healthy in one life, I find that sometimes I am genuinely surprised and thrilled in a naive way that here we have a talking computer: "like wow! it talks!". On the other hand, it is also possible to have the impression that computers are quite dreadful at the job of speaking; they make frequent mistakes, drone on, and just sound plain *wrong* in many cases. These impressions are all part of the mysteries and complexities of speech.

1.1 What are text-to-speech systems for?

Text-to-speech systems have an enormous range of applications. Their first real use was in reading systems for the blind, where a system would read some text from a book and convert it into speech. These early systems of course sounded very mechanical, but their adoption by blind people was hardly surprising because the other options of reading braille or having a real person do the reading were often not available. Today, quite sophisticated systems exist that facilitate human–computer interaction for the blind, in which the TTS can help the user navigate around a windows system.

The mainstream adoption of TTS has been severely limited by its quality. Apart from users who have little choice (as is the case with blind people), people's reaction to old-style TTS is not particularly positive. While people may be somewhat impressed and quite happy to listen to a few sentences, in general the novelty of this soon wears off. In recent years, the considerable advances in quality have changed the situation such that TTS systems are more common in a number of applications. Probably the main use of TTS today is in call-centre automation, where a user calls to pay an electricity bill or book some travel and conducts the entire transaction through an automatic dialogue system. Beyond this, TTS systems have been used for reading news stories, weather reports, travel directions and a wide variety of other applications.

While this book concentrates on the practical, engineering aspects of text-to-speech, it is worth commenting that research in this field has contributed an enormous amount to our general understanding of language. Often this has been in the form of "negative" evidence, meaning that when a theory thought to be true was implemented in a TTS system it was shown to be false; in fact, as we shall see, many linguistic theories have fallen when rigorously tested in speech systems. More positively, TTS systems have made good testing grounds for many models and theories, and TTS systems are certainly interesting in their own terms, without reference to application or use.

1.2 What should the goals of text-to-speech system development be?

One can legitimately ask, regardless of what application we want a talking computer for, is it really necessary that the quality needs to be high and that the voice needs to sound like a human? Wouldn't a mechanical-sounding voice suffice? Experience has shown that people are in fact very sensitive, not just to the words that are spoken, but to the *way* they are spoken. After only a short while, most people find highly mechanical voices irritating and discomforting to listen to. Furthermore, tests have shown that user satisfaction increases dramatically the more "natural" sounding the voice is. Experience (and particularly commercial experience) shows that users clearly want natural-sounding (that is human-like) systems.

Hence our goals in building a computer system capable of speaking are to build a system that first of all clearly gets across the message and secondly does this using a human-like voice. Within the research community, these goals are referred to as **intelligibility** and **naturalness**.

A further goal is that the system should be able to take any written input; that is, if we build an English text-to-speech system, it should be capable of reading any English sentence given to it. With this in mind, it is worth making a few distinctions about computer speech in general. It is of course possible simply to record some speech, store it on a computer and play it back. We do this all the time; our answering machine replays a message we have recorded, the radio plays interviews that were previously recorded and so on. This is of course simply a process of playing back what was originally recorded. The idea behind text-to-speech is to "play back" messages that weren't originally recorded. One step away from simple playback is to record a number of common words or phrases and recombine them, and this technique is frequently used in telephone dialogue services. Sometimes the result is acceptable, sometimes not, since often the artificially joined speech sounds stilted and jumpy. This allows a certain degree of flexibility, but falls short of open-ended flexibility. Text-to-speech, on the other hand, has the goal of being able to speak anything, regardless of whether the desired message was originally spoken or not.

As we shall see in Chapter 13, there are various techniques for actually generating the speech. These generally fall into two camps, which we can call bottom-up and concatenative. In the bottom-up approach, we generate a speech signal "from scratch", using our knowledge of how the speech-production system works. We artificially create a basic signal and then modify it, in much the same way as the larynx produces a basic signal that is then modified by the mouth in real human speech. In the concatenative approach, there is no bottom-up signal creation per se; rather we record some real speech, cut this up into small pieces, and then recombine these to form "new" speech. Sometimes one hears the comment that concatenative techniques aren't "real" speech synthesis in that we aren't generating the signal from scratch. This point is debatable, but it turns out that at present concatenative techniques outperform other techniques, and for this reason concatenative techniques currently dominate in commercial applications.

1.3 The engineering approach

In this book, we take what is known as an **engineering approach** to the text-to-speech problem. The term "engineering" is often used to mean that systems are simply bolted together, with no underlying theory or methodology. Engineering is of course much more than this, and it should be clear that great feats of engineering such as the Brooklyn bridge were not simply the result of some engineers waking up one morning and banging some bolts together. So by "engineering" we mean that we are tackling this problem in the best traditions of other engineering; these include working with the materials available and building a practical system that doesn't, for instance, take days to produce a single sentence. Furthermore, we don't use the term engineering to mean that this field is only relevant or accessible to those with (traditional) engineering backgrounds or education. As we explain below, TTS is a field relevant to people from many different backgrounds.

One point of note is that we can contrast the engineering approach with the scientific approach. Our task is to build the best possible text-to-speech system and in doing so we will use any model, mathematics, data, theory or tool that serves our purpose. Our main job is to build an *artefact* and we will use any means possible to do so. All artefact creation can be called engineering, but *good* engineering involves more: often we wish to make good use of our resources (we don't want to use a hammer to crack a nut); we also in general want to base our system on solid principles. This is for several reasons. First, using solid (say mathematical) principles assures us that we are on well-tested ground; we can trust these principles and don't have to verify experimentally every step we take. Second, we are of course not building the last ever text-to-speech system; our system is one step in a continual development; by basing our system on solid principles we hope to help others to improve and build on our work. Finally, using solid principles has the advantage of helping us diagnose the system, for instance to help us find why some components do perhaps better than expected, and allows the principles on which these components are based to be used for other problems.

Speech synthesis has also been approached from a more scientific aspect. Researchers who pursue this approach are not interested in building systems for their own sake, but rather as models that will shine light on human speech and language abilities. Thus, the goals are different, and, for example, it is important in this approach to use techniques that are at least plausible possibilities for how humans would handle this task. A good example of the difference is in the concatenative waveform techniques which we will use predominantly; recording large numbers of audio waveforms, chopping them up and gluing them back together can produce very-high-quality speech. It is of course absurd to think that this is how humans do it. We bring this point up because speech synthesis is often used (or was certainly used in the past) as a testing ground for many theories of speech and language. As a leading proponent of the scientific viewpoint states, so long as the two approaches are not confused, no harm should arise (Huckvale [225]).

1.4 Overview of the book

I must confess to generally hating sections entitled "how to read this book" and so on. I feel that, if I bought it, I should be able to read it any way I damn well please! Nevertheless, I feel some guidelines may be useful.

This book is what one might call an *extended text book*. A normal text book has the job of explaining a field or subject to outsiders and this book certainly has that goal. I qualify this by using the term "extended" for two reasons. Firstly, the book contains some original work and is not simply a summary, collection or retelling of existing ideas. Secondly, the book aims to take the reader right up to the current state of the art. In reality this can never be fully achieved, but the book is genuinely intended to be an "all you ever need to know". More modestly, it can be thought of as "all that I know and can explain". In other words, this is it: I certainly couldn't write a second book that dealt with more advanced topics.

Despite these original sections, the book is certainly not a monograph. This point is worth reinforcing: because of my personal involvement in many aspects of TTS research over the last 15 years, and specifically because of my involvement in the development of many well-known TTS systems, including CHATR [53], Festival [55], and rVoice, many friends and colleagues have asked me whether this is a book about those systems or the techniques behind those systems. Let me clearly state that this is not the case; *Text-to-Speech Synthesis* is not a system book that describes one particular system; rather I aim for a general account that describes current techniques without reference to any particular single system or theory.

1.4.1 Viewpoints within the book

That said, this book aims to provide a single, coherent picture of text-to-speech, rather than simply a list of available techniques. While not being a book centred on any one system, it is certainly heavily influenced by the general philosophy that I have been using (and evolving) over the past years, and I think it is proper at this stage to say something about what this philosophy is and how it may differ from other views. In the broadest sense, I adopt what is probably the current mainstream view in TTS, namely that this is an engineering problem, which should be approached with the aim of producing the best possible system, rather than with the aim of investigating any particular linguistic or other theory.

Within the engineering view, I again have taken a more specialist view in posing the text-to-speech problem as one where we have a single integrated text-analysis component followed by a single integrated speech-synthesis component. I have called this the **common-form model** (this and other models are explained in Chapter 3). While the common-form model differs significantly from the usual "pipelined" models, most work that has been carried out in one framework can be used in the other without too much difficulty.

In addition to this, there are many parts that can be considered original (at least to the best of my knowledge) and in this sense the book diverges from being a pure text book at these points. Specifically, these parts are

1. the common-form model itself,
2. the formulation of text analysis as a decoding problem,
3. the idea that text analysis should be seen as a semiotic classification and verbalisation problem,
4. the model of reading aloud,
5. the general unit-selection framework and
6. the view that prosody is composed of the functionally separate systems of affective, augmentative and suprasegmental prosody.

With regard to the last topic, I should point out that my views on prosody diverge considerably from the mainstream. My view is that mainstream linguistics, and as a consequence much of speech technology, has simply got this area of language badly wrong. There is a vast, confusing and usually contradictory literature on prosody, and it has bothered me for years why several contradictory competing theories (of, say, intonation) exist, why no-one has been able to make use of prosody in speech-recognition and -understanding systems, and why all prosodic models that I have tested fall far short of the results their creators say we should expect. This has led me to propose a completely new model of prosody, which is explained in Chapters 3 and 6.

1.4.2 Readers' backgrounds

This book is intended for both an academic and a commercial audience. Text-to-speech or speech synthesis does not fall neatly into any one traditional academic discipline, so the level and amount of background knowledge will vary greatly depending on a particular reader's background. Most TTS researchers I know come from an electrical engineering, computer science or linguistics background. I have aimed the book at being directly accessible to readers with these backgrounds, but the book should in general be accessible to those from other fields.

I assume that all readers are computer literate and have some experience in programming. To this extent, concepts such as algorithm, variable, loop and so on are assumed. Some areas of TTS are mathematical, and here I have assumed that the entry level is that of an advanced high-school or first-year university course in maths. While some of the mathematical concepts are quite advanced, these are explained in full starting with the entry-level knowledge. For those readers with little mathematical knowledge (or inclination!), don't worry; many areas of TTS do not require much maths. Even for those areas which do, I believe a significant understanding can still be achieved by reading about the general principles, studying the graphs and, above all, trying the algorithms in practice. Digital filters can seem like a complicated and abstract subject to many; but I have seen few people fail to grasp its basics when give the opportunity to play around with filters in a GUI package.

My commercial experience made it clear that it was difficult to find software developers with any knowledge of TTS. It was seen as too specialist a topic and even for those who were interested in the field, there was no satisfactory introduction. I hope this book will help solve this problem, and I have aimed it at being accessible to software engineers (regardless of academic background) who wish to learn more about this area. While this book does not give a step-by-step recipe for building a TTS system, it does go significantly beyond the theory, and tries to impart a feel for the subject and to pass on some of the "folklore" that is necessary for successful development. I believe the book covers enough ground that a good software engineer should not have too much difficulty with implementation.

1.4.3 Background and specialist sections

The book contains a significant amount of background material. This is included for two reasons. Firstly, as just explained, I wanted to make the book seem complete to any reader outside the field of speech technology. I believe it is to the readers' benefit to have introductory sections on phonology or signal processing in a single book, rather than having to resort to the alternative of pointing the reader to other works.

There is a second reason, however, which is that I believe that the traditional approach to explaining TTS is too disjointed. Of course TTS draws upon many other disciplines, but the differences between these, to me, are often overstated. Too often, it is believed that only "an engineer" (that is someone who has a degree in engineering) can understand the signal processing, only "a linguist" (again, a degree in linguistics) can understand the phonetics and so on. I believe that this view is very unhelpful; it is ridiculous to believe that someone with the ability to master signal processing isn't able to understand phonetics and vice versa. I have attempted to bridge these gaps by providing a significant amount of background material, but in doing so have tried to make this firstly genuinely accessible and secondly focused on the area of text-to-speech. I have therefore covered topics found in introductory texts in engineering and linguistics, but tried to do so in a novel way that makes the subject matter more accessible to readers with different backgrounds. It is difficult to judge potential readers' exposure to the fundamentals of probability as this is now taught quite extensively. For this reason, I have assumed a knowledge of this in the body of the book, and have included a reference section on this topic in the appendix.

The book is written in English and mostly uses English examples. I decided to write the book and focus on one language rather than make constant references to the specific techniques or variations that would be required for every language of interest to us. Many newcomers (and indeed many in the field who don't subscribe to the data-driven view) believe that the differences between languages are quite substantial and that what works for English is unlikely to work for French, Finnish, or Chinese. While languages obviously do differ, in today's modern synthesisers these differences can nearly all be modelled by training and using appropriate data; the same core engine suffices in all cases. Hence concentrating on English does not mean that we are building a system that will work on only one language.

2 Communication and language

Before delving into the details of how to perform text-to-speech conversion, we will first examine some of the fundamentals of communication in general. This chapter looks at the various ways in which people communicate and how communication varies depending on the situation and the means which are used. From this we can develop a general model, which will then help us specify the text-to-speech problem more exactly in the following chapter.

2.1 Types of communication

We experience the world though our senses and we can think of this as a process of gaining **information**. We share this ability with most other animals: if an animal hears running water it can infer that there is a stream nearby; if it sees a ripe fruit it can infer that there is food available. This ability to extract information from the world via the senses is a great advantage in the survival of any species. Animals can, however, cause information to be created: many animals make noises, such as barks or roars, or gestures such as flapping or head nodding, which are intended to be interpreted by other animals. We call the process of *deliberate creation* of information with the *intention that it be interpreted* **communication**.

The prerequisites for communication are an ability to create information in one being, an ability to transmit this information and an ability to perceive the created information by another being. All three of these prerequisites strongly influence the nature of communication; for example, animals that live in darkness or are blind would be unlikely to use a visual system. Despite these restrictions, it is clear that there are still many possible ways to make use of the possibilities of creation, medium and perception to communicate. We will now examine the three fundamental communication techniques that form the basis for human communication.

2.1.1 Affective communication

The most basic and common type of communication is **affective** communication, where we express a primary emotional state with external means. A good example of this is the expression of pain, where we might let out a yell or cry upon hurting ourselves. A defining characteristic of this type of communication is that the intensity of the external

Figure 2.1 A (British) road sign, indicating a slippery road and high chance of skidding.

form is clearly a function of the the intensity of feeling; the more intense the pain the louder the yell. Other primary mental states such as happiness, anger and sadness can be expressed in this way. This type of communication is one that is common to most higher animals. While the *ability* to express these affective states is common among animals, the precise means by which these are expressed is not universal or always obvious. A high-pitched yell, squeal or cry often means pain, but it is by no means obvious that a dog's wagging tail and a cat's purring are expressions of happiness.

2.1.2 Iconic communication

Though fundamental and powerful, affective communication is severely limited in the range of things it can be used to express. Happiness can readily be conveyed, but other simple mental states such as hunger or tiredness are significantly more difficult to convey. To express more complex messages, we can make use of a second communication mechanism known as **iconic** communication. An iconic system is one where the created **form** of the communication somehow resembles the intended **meaning**. We use the term "form" here in a technical sense that allows us to discuss the common properties of communication systems: in acoustic communication, "form" can be thought of as a type of sound; in visual communication form might be types of hand signals, facial expressions and so on. For example, it is common to communicate tiredness iconically by the "sleeping gesture", whereby someone closes her eyes, puts her hands together and places her head sideways on her hands. The person isn't really asleep – she is using a gesture that (crudely) mimics sleep to indicate tiredness. Another good example of iconic communication is the road sign shown in Figure 2.1. In this case, the form is the diagram, and the meaning is *slippery road*, and the fact that the form visually resembles what can happen when a road is slippery means that this communication is iconic. Note that, just as with the sleep example, the form isn't a particularly accurate picture of a car, road, skid or so on; the idea is to communicate the essence of the meaning and little else.

Anecdotally, we sometimes think of pre-human communication also working like this, and, in such a system, the presence of a sabre-tooth tiger might be communicated by imitating its growl, or by acting like a tiger and so on. Such systems have a certain advantage of transparency, in that, if the intended recipient does not know what you mean, a certain amount of mimicry or acting may get the point across. In essence, this is just like the road-sign example, in that the form and meaning have a certain direct connection. When travelling in a country where we don't speak the language, we often resort to miming some action with the hope that the meaning will be conveyed.

Iconic systems have several drawbacks, though. One is that, while it may be easy to imitate a tiger, it is less easy to imitate more abstract notions such as "nervous" or "idea". More importantly though, iconic systems can suffer from a lack of precision: when a first caveman imitates the tiger, a second caveman might not get this reference exactly – he might be sitting there thinking "well, it could be a tiger, or perhaps a lion, or maybe a large dog". By the time the first caveman has mimed his way through this and the action of "creeping up behind", both have probably departed to the great cave in the sky. While useful and important, iconic communication clearly has its limits.

2.1.3 Symbolic communication

In contrast to iconic and affective communication, we also have **symbolic** communication in which we give up the idea that the form must indicate the meaning. Rather we use a series of correspondences between form and meaning, in which the relationship is not direct. In symbolic systems, a tiger might be indicated by waving the left hand, a lion by waving the right. There is no reason why these forms should indicate what they do; it is merely a matter of convention. The advantage is that it is easier to devise a system where the forms are clear and distinct from one another – less confusion will arise. The disadvantage is that the fact that left-arm-wave means tiger, whereas right-arm-wave means lion, has to be *learned*; if you don't know, seeing the movement in isolation won't give a clue to the meaning. Despite the disadvantage of needing to learn, using conventions rather than icons can be hugely advantageous in that the conventions can be relatively brief; one noise or gesture may represent the tiger and this need not be acted out carefully each time. This brevity and clarity of form leads to a swiftness and precision seldom possible with iconic communication.

Once a convention-based communication system is used, it soon becomes clear that it is the notion of **contrast** in form that is the key to success. To put it another way, once the form no longer needs to resemble the meaning, the communication system gains benefit from making the forms as distinct from one another as possible. To show this point, consider the following experiment.

Eight subjects were grouped into pairs, and each pair was asked, in isolation, to design a communication system based on colour cards. The premise was that the subjects were in a noisy pub, with one of the pair at the bar while the other was sitting down at a table. We said that there were four basic concepts to communicate: "I would like a drink of water", "I would like some food", "I would like a beer" and "I would like to listen to some music". Each pair was given a set of 100 differently coloured cards arranged