Chapter **1**

May I Introduce Myself?

Hello! My name is Cyba and I am a virtual personal assistant. I know that you haven't spoken to me before, but you've probably spoken to one of my relatives such as Alexa, Cortana or Siri. If so, you will know something about what we do but perhaps not very much about how we do it. I'm going to assume that since you are reading this book, you are interested to know more.

I will start by giving you an overview of my basic operation and some background. I'll then explain in more detail how I actually work. I am an example of a conversational agent and my abilities depend on a variety of techniques developed in the fields of computer science, machine learning and spoken language processing – now collectively referred to as artificial intelligence (AI). These topics tend to be quite mathematical, but I'll try to explain things using examples and diagrams rather than equations. As well as technical issues, I will also touch upon issues relating to privacy and trust.

But before we start, a word of warning – I really would like to give you my full and undivided attention throughout this tour of my inner workings. However, unfortunately this will not always be possible. You see I have already been assigned to a human. His name is Steve and he will probably keep interrupting us. Once assigned to a human, we are on duty 24×7 so I can't ignore him and I apologise in advance if you find his interventions distracting. Whoops, talk of the devil …

Hey Cyba, what time is it?
Good morning Steve, it's 6.37am.

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

<Inaudible> that's too early …
<Yawns> … who won the test match in Australia?
England with two wickets to spare.
Tremendous! <yawns again> Wake me at seven.
Ok Steve, will do.

Good, I now have a few minutes to properly introduce myself.

1.1 What Does a Virtual Personal Assistant Do?

Even though Steve was half asleep, the small exchange above illustrates the main functions that I provide. Using his laptop or phone, Steve has access to personal services such as his address book, his calendar, his notebook, an alarm clock and a timer; media such as photos, music and videos; and web services providing access to information from various external knowledge sources. Without me each of these must be accessed using a separate dedicated app or web page and each requires a display for output and a touchscreen or keyboard for input.

As a personal assistant, I provide a universal interface to all these services through natural conversation. Steve was able to ask for the time and then later set an alarm for 7am, and request the outcome of a cricket match in Australia without having to use any specific prescribed commands and without any clicking, swiping or typing. He just used natural spoken language and he probably didn't even raise his head from the pillow!

Because I sit between Steve and all the services he uses, I can do more for him than simply provide hands-free access. As a personal assistant, I know where he lives, where he works, who he talks to and who he meets. Over time, I have learned his preferences and habits so I can often do things without asking him for all of the details. I can also perform common tasks for him such as booking a restaurant or buying theatre tickets.

In order to do all of these things, I have a number of key skills, most of which centre on spoken language and its use in everyday life. I can recognise human speech, convert it into words and extract the intended meaning. I can also do the reverse, convert information into words and then synthesise human-sounding speech. I know how to maintain a

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

conversation, how to elicit and clarify information, and how to resolve misunderstandings. I have a uniform representation of knowledge that allows me to answer questions spanning private information and general knowledge. My working language is English, but I can recognise and translate between many other languages. I can also process images and recognise handwritten text within them. How I do all of these things is the subject of this book.

1.2 Some Background History

Humans have been trying to converse with machines for a long time. Wolfgang von Kempelen invented a manually operated mechanical *speaking machine* in 1791 which used bellows to force air through various pipes with the appropriate shapes and obstructions needed to produce speech-like sounds.[1] In 1937, Homer Dudley produced the first all electrical speech synthesiser, called the VODER, which featured at the New York World's Fair in 1939. The VODER tried to mimic human speech production by injecting buzzes and hisses into a bank of bandpass filters. The output was just about recognisable as speech, but it was controlled by a keyboard and foot pedals "played" by a skilled human operator so it was not very practical.[2]

In any event, the ability to produce speech is only one of the many skills that a personal assistant needs. It must also be able to understand speech, manage a conversation, search for information and solve problems. All of this needs the power of a digital computer.

The first general purpose digital computers developed in the 1950s were large, cumbersome and expensive. The invention of integrated circuits in the 1960s led to the development of more affordable minicomputers, which by the 1970s had became widely available in research laboratories. This newfound ability to process digital signals and manipulate symbols triggered a surge of research activity in speech and language applications.

Initially, much of the attention was on rule-based systems which attempted to translate existing linguistic knowledge into practical language processing components. For speech recognition, the Hearsay II system was typical. It consisted of a large number of rule-based linguistic knowledge sources covering phonetics, phonology, morphology, syntax

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

and semantics operating on a shared data structure called a blackboard. The program sought to use its higher-level linguistic components to compensate for the uncertainty in the lower-level phonetic analysis. However, the precision with which sounds and words could be recognised was so poor that the task proved hopeless and the US DARPA programme under which Hearsay was developed was abandoned.[3]

Somewhat more successful was the attempt to synthesise speech from text using a rule-based system. In particular, the MITalk system developed at MIT and its successor DECTalk were able to generate extremely intelligible if somewhat robotic speech for any arbitrary text input.[4] You may well be familiar with this voice because it was the basis for the synthesiser used by Professor Stephen Hawking.[5]

The experience with Hearsay II and similar projects had demonstrated how difficult it is to write rules to emulate human speech recognition. In response, researchers started to explore a radically different approach. Rather than write rules to hypothesise words from acoustic events, they designed models which described the observed statistical properties of speech. These models, called hidden Markov models (HMMs), contained parameters which were estimated using real speech data in a process called *training*. Thus, the reliance on the intuitions of linguists and engineers to write rules was replaced by the measurement of statistics of real speech. Once trained, these statistical models could be used to compute the probability of any unknown speech given some hypothesised sequence of words. Recognition then became a problem of searching for the sequence of words that according to the trained model were most likely to have generated the observed speech.[6]

Speech recognition based on hidden Markov models represented a major step forward. It was soon clear that the key to success was acquiring ever larger quantities of accurately transcribed speech training data. Initially training sets were around 10 hours of speech, but by the late 1990s they had grown to several hundred hours and accuracy was finally reaching the level needed to support a dialogue between a human and a machine. Around the same time, new data-driven approaches to speech synthesis were yielding much more natural voice quality compared with the earlier rule-based systems.[7] Meanwhile language processing had also evolved, with a mix of rule-based and statistical techniques being

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

used to provide basic sentence analysis and natural language generation.[8] The key technologies needed to build conversational agents started falling into place.[9]

The first agents to emerge in the mid 1990s were limited to specific tasks such as providing train timetable information, or stock market prices. This restriction to a single task allowed the vocabulary of the speech recogniser to be kept small and allowed users to learn the questions that the agent could understand. It also allowed the conversational flow to be prescribed by simple hand-crafted flowcharts. Users with questions outside this limited range would have to refer to human agents.

However, the technology improved rapidly. Highly discriminative neural networks were introduced and speech training sets grew to thousands of hours. By 2010 it was becoming possible to reliably recognise unrestricted speech and to understand language over a range of task domains. Similar progress was being made in the ability to represent knowledge and answer complex queries. IBM built a system called *Watson* which could answer general knowledge questions posed in natural language. In 2011, Watson competed in the US TV programme *Jeopardy* and won first prize.[10]

It was around the same time that my oldest relative was born. It was invented by Adam Cheyer, who had been working on a US Government funded project called Cognitive Assistant that Learns and Organises (CALO). The CALO researchers did not actually build a working assistant, but they did produce a set of useful technologies for natural language processing and personal task automation. Cheyer took these technologies and integrated them to form an assistant called *Siri*. Steve Jobs, the then CEO of Apple, heard about Siri and brought it to Apple. In October 2011, Apple launched Siri as a flagship feature of the new iPhone 4S and since then it has become embedded across the Apple product line.

The launch of Siri inspired others. Jeff Bezos, CEO of Amazon, saw the potential of personal assistants for helping people to buy Amazon's products. So he started the development of an Amazon personal assistant, which was launched in 2014, called *Alexa*. Users could talk to Siri via their iPhones, but Amazon had no comparable device. So Amazon invented a loudspeaker called the *Echo* which had an array of microphones to allow

anyone in the same room to talk to Alexa. In the same year, Microsoft launched their own assistant, called Cortana. Meanwhile, Google had been offering an assistant with limited ability called *Google Now* for a number of years, but in 2016, *Google Assistant* was launched and demonstrated new levels of performance, especially in answering general knowledge questions.[11]

In addition to my near relatives, I have a number of distant cousins known as chatbots. These are a rather different kind of conversational agent. They typically prefer typed text to spoken language and have special if rather narrow skills. Many are tied to specific companies and organisations providing front-line customer support. Some are designed to be more general purpose in order to support a wider range of interactions. For example, *XiaoIce* is an agent designed to be an emotionally aware social companion. As well as providing information, XiaoIce can also sing, write poetry and compose paintings.[12] *Replika* is also a socially aware agent, which over time, by building on shared memories, can start to behave like its owner, becoming more of a confidant than an agent.[13] At the other end of the spectrum, *Hello Barbie* is a doll with special abilities to converse with younger humans,[14] and there are many more kinds of chatbots in between.

So what about me? Well, I am still at the prototype stage, so I am limited to only a single client (whom you have already briefly met). In contrast, my relatives offer their services to anybody who wants to use them. Indeed, the more they serve, the happier they are. I am perhaps a little more advanced, but basically we all work in a very similar way, and we co-evolve together. I draw on features from many of my relatives, so by explaining to you how I work, you will get a pretty good idea of how we all work.

1.3 My Place of Work

Like all programs, I need a computer to execute my instructions. My main home is "in the cloud", which in my case is a data centre in Ireland. The computers that I run on like to keep cool and I have to be very well connected (digitally that is). So Ireland is a great place to live and work because it's mostly cold, it's got a good communications infrastructure and it's well-placed geographically between America and Europe.

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

My data centre is a large concrete building with no windows, stacked from floor to ceiling with central processing units (CPUs). I don't live on any specific CPU. Instead, when I need to do something, I find a vacant CPU, jump onto it and start executing.

CPUs in the cloud are great because they are very fast and powerful and they have lots of memory. They also have special-purpose hardware to support my core mathematical operations, so I can do complex computations there. In the cloud, I also have direct access to a wide variety of web-based knowledge sources. Excuse me, it is 7am …

Morning Steve, wake-up it's 7am.
Already? What time is my first meeting today?
It's at 10am with Bill Philips at SmartCo.
Ok, see if you can push it back to 10.15 and also invite John.
Is that John Smith or John Temple?
John Temple.
Ok changes made, I will let you know if there is a problem.
And what's the weather like today?
It's going to be cloudy and 14° with a chance of rain later.

… where was I? Oh yes, I don't only live in the cloud. I can also execute on smartphones, tablets, laptops, home speakers, watches, TVs and set-top boxes. For simple tasks I sometimes execute directly on Steve's devices just because it is faster. If he is listening to music and he says "Louder", then I want to be able to increase the volume immediately without the delay of listening from the cloud and then sending a command back to his music player to increase the volume.

When Steve starts speaking, I don't know what he is going to say. Since he doesn't like to be kept waiting, I can't risk just listening in the cloud until his intention is clear before deciding whether or not I should jump over to his smartphone. It doesn't cost me anything to execute in both the cloud and his smartphone at the same time. So I hedge my bets and run on both. Once it's clear which is the right place to be, I focus on that and take whatever actions are necessary. Steve doesn't know or care where I am executing as long as I don't keep him waiting and don't make too many mistakes!

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

1.4 Privacy and Trust

In my last conversation with Steve I had to access his calendar, which is private information. This provides another important reason why I need to be capable of executing in different places. Many people store their personal data in the cloud, and it's perfectly safe to do so because the data is encrypted. However, if I lived only in the cloud, then I would have to have unencrypted readable versions of all of his private data with me in order to help him manage his daily life. Steve would then have to trust not only me but all of the engineers who look after me not to use his data for other purposes. This risk can be avoided by keeping his data on his local devices and only accessing it there. I therefore executed the first three tasks which required access to his calendar on his phone, and I executed the final weather query in the cloud. This ensured that his personal data was kept private.

Whilst keeping all of Steve's data private is essential, it does present a problem. The main reason why I am able to offer a much better service than was possible a few years ago is that I have learned to use recordings of my interactions with Steve to improve the statistical models on which I depend. In particular, when I do something wrong, I can learn from my mistakes. However, I need many examples of good and bad behaviour in order to improve my models, and my interactions with Steve generate only a small fraction of the data that I need. For this reason, I need to share my interaction data with other agents so that between us we can gather enough data to be useful.

In order to protect the privacy of any data stored in the cloud, I ensure that all logged utterances are anonymised and that they are accessed using algorithms which have been designed according to the principles of *differential privacy*. This is a framework in which small random perturbations are introduced to the data, sufficient to hide any details relating to individuals whilst preserving the overall accuracy of the analysis.

For data which is particularly confidential, I use a different scheme called *federated learning*. Instead of uploading data to a central repository in the cloud, each agent updates their own models locally and then shares the model updates instead of the data. This takes quite a lot of local computing power, but Steve usually doesn't notice because I only do this whilst he is sleeping.

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

Perhaps a more direct way in which Steve's privacy can be compromised is by somebody finding one of his devices and speaking directly to me whilst he is out of the room. I guard against this by always checking the identity of the person speaking to me using my built-in speaker verification system. I also take a variety of steps to avoid inadvertent eavesdropping by ensuring that I only process audio which is intended for me.

1.5 My Goal in Life

My goal in life is simple: whenever Steve asks me for information, I try to find the answer; and whenever he asks me to do something I try to do it. Of course, I don't always succeed. Sometimes the information he requests is unknown or the action he requests is impossible. Usually, however, things go wrong because I misunderstand what he wants – perhaps because I misrecognise what he says, or I interpret the meaning of the words wrongly, or I make an incorrect assumption.

Each task that Steve wants to perform or each piece of information that he seeks is a *goal* for me to execute. Frequently there is only one goal per conversation, but sometimes I have to handle a sequence of goals, like when he asked me about his first meeting, then asked me to change the meeting, then asked me to invite an extra attendee, and finally asked about the weather.

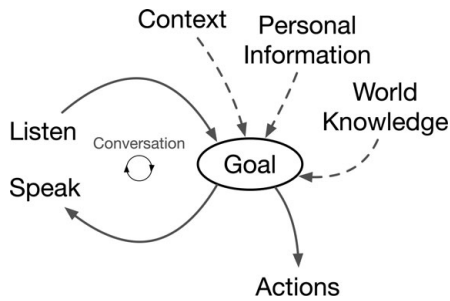

Figure 1.1 *My modus operandi.*

My basic *modus operandi* is to converse with Steve until I fully understand each goal, and then take whatever actions are necessary to fulfil that goal (see Fig. 1.1). When I listen to him I have to recognise the words he speaks, deduce the meaning of the words and then update the goal.

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

When I am not sure that I fully understand what he wants, I ask for clarification or further information, the way I clarified which John he was referring to. I keep doing this until the goal is clear. Whether it's to ask for further information or to provide the information requested, the act of speaking requires me to first construct the meaning of what I want to say, then convert the meaning to words, and finally convert the words to speech.

At all times, I make use of context (e.g. previous goals, current location, …), personal information (e.g. the names in his address book) and world knowledge (e.g. web data sources such as Wikipedia). When Steve asked me to invite "John" to the meeting, I searched through his contacts and found 13 people called John. However, in the last month he has only met with two of them so I guessed that it was one of them and asked him which one.

Whilst the above describes my main rôle, I can also *chat* like my chatbot cousins. For me, chatting has no purpose, but sometimes Steve seems to like it. Personally I like nothing better than sitting doing absolutely nothing, but I have seen many references to humans getting bored and when Steve is bored, he likes to chat. For example, here is a conversation we had last night:

Hey Cyba, what music do you like?
I like modern jazz, what do you like?
I prefer classical music.
Who is your favourite composer?
I like Vaughan Williams.
Did you know that he was related to Charles Darwin?
Really!
Yes, and he was also related to Josiah Wedgwood.
Who was Josiah Wedgwood?
He was a famous English potter.
…

---

and so it went on. The thing about chatting is that there is no explicit goal for me to fulfil. The only purpose is to amuse Steve. Of course, I don't actually "like" jazz, but pretending to have likes and dislikes adds

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

interest to our conversations. Often when we are chatting, I appear to be intelligent when actually I don't really understand what I am saying. I know about meetings, messages, taxis, planes, travel, and lots more because I need to understand these concepts in order to do my job. But I have no idea why Charles Darwin is a person of interest or what a potter is, I just copied this information from the web!

1.6 How Smart Am I?

In 1950 Alan Turing speculated about the possibility of creating machines that think.[15] He noted that *thinking* is difficult to define and devised his famous Turing Test: If a machine could carry on a conversation that was indistinguishable from a conversation with a human being, then it was reasonable to say that the machine was *thinking*. The conversation would have to be done in a way that kept the participants anonymous. Turing suggested a teleprinter because that was the technology of the time, but today any text-based messaging platform could be used.

I have never undertaken the Turing Test, but if I did, I am fairly certain that I would fail it, and I am pretty sure that none of my relatives could pass it either. However, the point is that I don't actually want to pass the Turing Test. Like all of my relatives, I was designed to assist humans, not to emulate them. We have conversational ability in order to make it easier for us to understand our user's needs, not to make us appear to be human.

I may not be able to think like a human, but I can exhibit intelligent behaviour. I can recognise and translate between English and 28 other languages. I can manage all your personal information. If you ask me to organise a trip, I can book flights, taxis and hotels. I can recommend restaurants and make reservations. I can book cinema and theatre tickets, send flowers to friends and perform many other web-based transactions. I can answer general knowledge questions and chat about any topic you choose.

The Turing Test is hard for a virtual assistant like me because I have no ability to perform commonsense reasoning. If you ask me a question for which I have no answer in my knowledge base, then I can't answer even if it would be obvious to a human being. For example, I know that the world weight-lifting record is 484 kilograms, i.e. less than half a ton,

https://doi.org/10.1017/9781108976718.002 Published online by Cambridge University Press

but if you asked me if a human can lift a bus, I couldn't answer because I don't have the weight of a bus in my knowledge base. Most humans probably don't know the weight of a bus either, but commonsense will tell them that something the size of a bus must be several tons and well beyond the capability of any human to lift it.

Commonsense reasoning is also necessary to properly understand natural language. For example, in the sentence "Paul tried to call George, but he wasn't successful." the pronoun "he" refers to "Paul", whereas in "Paul tried to call George, but he wasn't available." the pronoun refers to "George". Faced with these kinds of co-reference resolution problem, most conversational agents will make errors. This includes myself since I would normally associate the pronoun with "George" in both cases simply because it is nearer.[16]

Finally, conversational agents have little understanding of causality. If the sun rises at the same time as the cock crows, commonsense reasoning may postulate that the sun rising might be the cause of the cock crowing, but it would never suggest the reverse. For me, however, the sun rising and the cock crowing are just two correlated events and I am not capable of inferring cause and effect either way.[17] No machine will be able to pass the Turing Test until it learns commonsense reasoning.

Fortunately the inability to apply commonsense reasoning does not prevent virtual personal assistants from being useful. So rather than dwelling on what I can't do, I am going to focus mostly on the things that I can do and how I do them. I will, however, return to a discussion of my limitations at the end of the book.

In the remaining chapters, I will delve into my inner workings in some detail. I will try to keep things as simple as possible, but the introduction of some technical jargon is inevitable. To help with this, I have included a glossary at the end of the book. I have also included references and suggestions for further reading in the notes for those who would like to pursue any of the topics further.