Te Reo Māori Speech Technologies and NCEA examinations

Completed for NZQA – March 2021

by Paora Mato, Te Taka Keegan, Dayne Perkins-Gordon
The University of Waikato

1 Introduction

As part of their role, the New Zealand Qualifications Authority (NZQA) are responsible for ensuring the delivery, quality and credibility of New Zealand's secondary school educational qualifications. NZQA are also charged with managing the New Zealand Qualifications Framework (NQF), administering the secondary school assessment system, independent quality assurance of non-university tertiary education providers and qualifications recognition and standard-setting for some specified unit standards.[1]

This research is focused on particular aspects of the online delivery and assessment of the National Certificate of Educational Achievement (NCEA), which includes the educational achievement standards normally delivered in New Zealand secondary schools. In an effort to replicate how students already interact with their world, NZQA is supplementing the hand-written NCEA examinations with online digital options. Currently, both paper and digital examinations are able to be answered in either English or te reo Māori, if the exam allows it. Assistive technologies such as Spell Checking, assisted correction and Text-to-Speech are currently available only in the English language.

Current Text-To-Speech (TTS) functionalities in NCEA digital assessments are not available in te reo Māori, affecting translated examinations and any examinations that would use te reo Māori text and speech. Before TTS functionalities can be implemented accurately across all appropriate NCEA digital assessments, research needs to be completed to identify current speech technologies and to investigate digital sources of te reo Māori suitable for use in TTS functions.

This research, therefore, aims to identify current or potential systems that might be used to implement or inform development of te reo Māori TTS capability for NCEA digital assessment. Currently, candidates can be afforded text conversion to speech in the English language, although the functionality is still under investigation and currently not in use. The same functionality is not available for te reo Māori. In terms of equitable access and outcomes, the drive to provide language equity in the digital space is a paramount objective as a part of the NZQA Te Kōkiritanga strategy. Furthermore, research has identified that an extended learning experience can be realised by multi-sensory delivery – such as being able to read text while listening to it being read out loud. This capability would further aid the delivery and assessment of digital examination for te reo Māori.

1.1 Rationale (per NZQA):

As an official language of New Zealand, students are able to, and may be required to respond to their NCEA examinations in te reo Māori. The move to digital assessment provides an opportunity to improve NCEA provision and outcomes for all students, however, the current limitations of TTS functionality for te reo Māori are a hindrance to the NZQA equity goal.

Te Ao Māori is an important principle underpinning the vision to equitably serve Māori students while benefiting the wider assessment system. Digital assessment will ensure an

[1] See: https://www.nzqa.govt.nz/about-us/our-role/

equitable experience for Māori students if a 21st century approach is applied to its development, delivery practices and data analytics. Enabling students to use te reo Māori and mātauranga Māori in their assessments will support the NZQA vision of equitable NCEA outcomes.

1.2 Research aims:

This research investigates Speech Technology, with focus on TTS conversion for te reo Māori. Observations will be included that may be useful for the implementation of TTS capability for online NCEA te reo Māori examination. As mentioned previously, the capability for English language TTS already exists although the functionality is not in use currently. For less-resourced languages, such as te reo Māori, there are a myriad of issues that need to be addressed in order to develop similar capability that provides a relative balance for students undertaking te reo Māori examinations online and who also present from a variety of ethnicities. The design and deployment of digital tools that will support the education of future generations, therefore, will need to consider equitable outcomes as a central objective.

This research therefore aims to:

1. provide an overview of Speech Technologies, including:
    a. Artificial Intelligence;
    b. Machine Learning and Natural Language Processing;
    c. Speech recognition and synthesis;
    d. Deep Learning and Neural Networks.
2. investigate current TTS technologies and discuss how they might be applied to te reo Māori;
3. investigate current TTS projects and initiatives;
4. identify the TTS approaches that can be used by NZQA for the NCEA online te reo Māori examinations.

2 Research Overview

This project has investigated Speech Technologies with a focus on TTS technology that can be applied to the online assessment of, and in, te reo Māori. While much of this research will revolve around the technologies themselves, some thought as part of the ongoing research outcomes should be devoted to the identification and/or potential acquisition and use of appropriate data. The discussion regarding Māori data necessitates the development of levels of relationships or partnering with the holders, owners, managers and guardians of such data. In particular cases the discussions should also include the technologies that are already being built and deployed by those stakeholders. Our efforts to progress such discussions have been somewhat hampered by national conditions in terms of the COVID-19 pandemic and, more importantly, the inability to discuss some matters face-to-face as Māori are wont to do. We had hoped to include findings from collaboration with actual initiatives and requested a reporting extension with the aim of doing so. Unfortunately, that was thwarted largely by the residual effects of the COVID-19 lockdowns.

Many of the relevant issues regarding TTS capability are dependent on a variety of machine learning requisites. Most notably, the need for suitable volumes of appropriate data types as

mentioned, but, more importantly, the necessity to ensure issues of ownership, security and the governance of such data can be maintained. For Māori, this viewpoint, most commonly referred to as data sovereignty, arises from a history of exploitation and diminished control, especially in terms of Indigenous artefacts and knowledge. However, without such data, initiatives such as TTS conversion for te reo Māori would be much more difficult to realise. Simplistically, current speech synthesis models require sufficient levels of suitable data. In terms of te reo Māori, the acquisition and utility of relevant data requires the formation of relationships and/or partnerships with Māori knowledge holders or data 'owners'. Furthermore, data storage systems must somehow secure and privatise those knowledge systems, artefacts and taonga in a way where the data can actually be interrogated or mined but not remain bare for public or incidental access.

3 Other Considerations

There are further considerations in New Zealand regarding TTS for te reo. In terms of the NCEA online assessments for instance, how is the quality of assessment measured? How is that quality defined? How are issues like dialect or language differences accounted for? How do you mirror colloquialism? Therefore, we offer in the context of this research that there are a number of considerations that need to be addressed in conjunction with the philosophy of development before we consider the technology itself.

More specifically:

1. the acquisition of suitable volumes of appropriate data and data types;
2. data and technical sovereignty;
3. quality of speech technology for this use case
    a. the accuracy of online assessment;
    b. the measures that inform that accuracy;
    c. language issues such as language accuracy/quality, dialect and colloquialism.

Current TTS software quality favours the major languages. The investigation and subsequent deployment of language initiatives is relatively obscure for less-resourced languages such as te reo Māori. In spite of that, current Indigenous initiatives tend to focus on the value of connection and expression and, perhaps, less on the technology that is required to stay connected. In this sense, one could readily appreciate the difference between a spiritual, 'Māori' outlook versus a purely 'education'-driven system. Notwithstanding the need to attain some levels of online assessment parity, the value differences and regional mindsets of candidates should surely be a consideration at some point in terms of online assessment – perhaps an addendum to the discussion of quality. This may also beg some questions regarding identity development, tikanga and nationhood as part of the education delivery prior to NCEA examination online.

The Crúbadán Project[2] and The Human Language Project[3] aim to accumulate data for under-resourced languages and list resources for over 2000 and 3433 languages respectively. While

[2] See: http://crubadan.org

[3] See: https://hlp.taus.net

gathering sufficient volumes of data in the appropriate formats for te reo Māori is a task that is currently onerous, given the identified potential sources of Māori-language corpora, the undertaking is not insurmountable. Issues of data and technical sovereignty should proceed with some haste since the settling of the issues involved within will, at many levels, determine the use and/or acquisition of the identified (and yet-to-be identified) data. The accuracy of any deployed speech technologies will largely depend on the quality of the input data and the relevance or 'Māoriness' of the speech models. Accuracy and quality measures should be taken into account to cover aspects of Māori-language use, especially for online examination and assessment, and, while some mention may be made of that, this report will remain largely focussed on the technology.

4 Speech Technologies

Speech technology refers to the digital recognition (speech-to-text, speech translation) of the human voice, the digital reproduction of the human voice (text-to-speech and speech synthesis) and the processing of speech data. Such technologies are gaining increasing significance, replicating the importance of spoken language especially in areas where the spoken word is progressively supplementing keyboards and graphical interactive interface systems and, in some cases, entirely replacing these and similar peripheral inputs into technical systems.[i,ii] Furthermore, the dependence on being able to recognise and process text and images when using digital devices or performing human-computer interactions becomes much more important for those who are visually impaired or who are illiterate. For this reason alone, speech technologies are valuable for supporting human-to-human and human-to-machine communication. Historically, speech technology initiatives proved rather clunky and largely unusable. The advent of big data and big data processing, and the various forms of Artificial Intelligence (AI), including Machine Learning and Deep Learning, has progressed a range of current speech technologies that include:

- Speech synthesis including text-to-speech conversion;
- Speech recognition including speech-to-text conversion;
- Text and speech translation;
- Speaker recognition and verification;
- Multimodal interaction; and,
- Interactive voice response.[ii,iii]

The progress in these technologies, in recent history, has been described as phenomenal.[4]

Speech technologies aim to synthesise human speech from arbitrary text or an image. Speech synthesis is, therefore, the artificial production of human speech using models that convert data into the spoken voice. In many cases speech signals have been reproduced that have very high intelligibility but still lack suitable levels of sound quality and naturalness.[iv] Section 5 describes a variety of speech technologies that are achieving levels of accuracy with varying degrees of success. As described in Section 4.8, this most often occurs using customised speech synthesis APIs (Application Programming Interfaces). Speech APIs can be viewed as middleware, or

[4] See: https://www.globalme.net/blog/the-present-future-of-speech-recognition/

interfaces, that sit between speech engines, performing recognition and synthesis, and the applications that deliver the final outputs to the users. Conversely, speech recognition converts spoken word into text or into other forms of recognisable inputs for further processing. One example of further processing includes the analysis of the text to understand context and/or intent. Using machine learning models based on large volumes of training data and previous input/output examples, the 'educated guesses' regarding content and meaning can be highly accurate.[5] Speaker recognition and verification is used in some systems to verify the identity of the speaker, which, again, can be used to replace other methods of verification such as eye-activation, face recognition and manual password input. Multimodal interfaces enable multiple methods of interaction by a user with a system. These types of interfaces are able to process two or more combined modes of input by a user. These might include touch, pen, manual gesture and speech.

The accuracy of speech technology output can be affected by a multitude of factors. Models that use huge volumes of high quality corpora can still be affected by particular language variations, localised dialects or accents for instance, the impact of background or ambient noise, voice changes due to illness or emotion, or the way humans change pitch or speaking speed for instance to suit the current environment. This means engineers must build reliable processing systems that are able to recognise and analyse a wide range of factors that might impact on the accurate recognition of text or speech.

4.1 Artificial Intelligence, Machine Learning and Deep Learning

Artificial Intelligence (AI) is the science of training technologies and machines to emulate human behaviour. It is possible, for instance, to apply a sticker to a bottle of milk that signals when the milk is reduced to a certain level or to change the lighting and play specific types of music when the technology in your house detects your mood.

Machine Learning (ML) is a branch of AI that looks for patterns in data in order to formulate conclusions about that data. ML is a method of data analysis that automates analytical model build and is based on the premise that systems can learn from data, recognise patterns and generate decisions unsupervised or with minimal human intervention. The ML algorithms are trained on the available data and are able to adapt to and draw conclusions from new data based on that previous data training.[v]

Similarly, Deep Learning is a type of machine learning where neural networks, algorithms inspired by the human brain, learn from large amounts of data. Deep learning algorithms train systems to recognise patterns by setting up basic parameters around data and then by repeatedly performing a task. This repeated processing gradually improves the outcome due to the deep layers of the network that allow progressive learning.[vi] One of the foundations of AI, deep learning models, can be used to train technologies to identify images, make predictions, recognise speech and vocalise text.

[5] See: https://www.globalme.net/blog/how-does-speech-recognition-technology-work/

4.2 Natural Language Processing

Natural language processing (NLP) is a field of artificial intelligence that aims to establish human interaction/communication with computers. In spite of some significant gains, computers still struggle to comprehend many facets of language that are difficult to characterise formally,[vii] such as pragmatics which deal with interactional intent and meaning[6]. For the largest part, language processing successes are achieved for popular languages like English and other languages that have text corpora of hundreds of millions of words. However, those languages represent only about 20-30 of the approximately 7000 languages in the world.[iii] Other languages lack large volumes of language corpora and are referred to as low resource languages. Common NLP models require large amounts of training data and/or sophisticated language-specific engineering which are not available for most languages and in many cases there are not enough linguistically trained speakers who are capable of building multiple language models. Two main approaches to NLP in the low resource setting, where the volumes of language data and knowledge are insufficient for traditional approaches, are:

1. a traditional approach that focuses on collecting more data for a particular language;
2. approaches that apply transfer learning.[ii]

Transfer learning and shared language models have become one of the cornerstones of NLP. The transfer learning approach asserts that there are enough commonalities between some languages that could be exploited so that a language model that was deployed for one language was actually derived from a model for a different language. The process of transferring resources and models from resource-rich sources to resource-poor target languages, to inform the build of a language model for another language, is referred to as cross-lingual transfer learning.[i] Both Microsoft and Google have used Neural Networks to demonstrate forms of transfer learning – where relatively large corpora of a particular language could be used to create speech technologies for languages that were similar but with low volumes of their own resource. Successful developments include style transfer between Indonesian languages which are closely related. Using a large corpus from Standard Indonesian, and a small crowd-sourced corpus for Javanese and Sudanese, both low resource languages, TTS capability was produced for both Sudanese and Javanese.[viii] Such techniques lend credence to the transfer learning models.

It is worth noting, however, that this still presents a problem for te reo Māori because there are few models that have yet to be created for Protopolynesian, or even Polynesian, languages like te reo Māori.

4.3 Part of Speech Tagging

Part of Speech tagging (POS tagging) is generally performed as a prerequisite to simplify common NLP tasks. Simplistically, POS tagging is the process of automatically marking up or labelling words in a text with a corresponding part of speech tag, such as verb, adverb, noun, pronoun etc. Simple mark-up is based on the definition of the word and how it relates to the others words in the same phrase, sentence or paragraph. In practice, POS tagging is actually

[6] See: https://all-about-linguistics.group.shef.ac.uk/branches-of-linguistics/pragmatics/what-is-pragmatics/

much more complicated because the same word could have multiple meanings and multiple parts of speech within the same sentence. For example:

"They refuse to permit us to obtain the refuse permit".[ix]

Refuse and permit are used twice each, both as verbs in the first instance and then as nouns. Additionally, both words are pronounced differently in each of their different contexts. Current POS taggers are able to identify with high accuracy which version of a word is being used so that the text can be pronounced accurately. For this reason, TTS systems are usually preceded by POS tagging.[vii]

One example of how POS tagging might be used is the popular music app "Shazam". Shazam is a mobile app that is used to identify songs as they are playing. Shazam uses POS tagging to recognise the lyrics in the song and then lists the title and artist of the song on your device.[x]

4.4 Text-to-Speech

Text-to-Speech (TTS) technology is a form of assistive technology that converts digital text into spoken text – the main goal being the production or synthesis of naturally sounding speech when provided with forms of text.[xi] This means people are able to listen to the text being spoken while they are reading it. Many TTS tools also highlight the words as they are read out aloud, allowing each word to be seen and heard at the same time. Because the speech is computer-generated, the reading speed can normally be varied, the voice pitch can be altered and such things as gender and ethnicity can be readily imbued. One of the current goals of TTS technology is not just to simply convert text to speech but also to make the speech sound like humans of different ages and genders – including computer-generated voices that sound like children speaking.[xii] Optical character recognition (OCR) technology can also be embedded within TTS tools, allowing images to be converted to text and then read out loud. In much the same way human speech production translates a text or a concept and uses a range of muscle movements, intonations and other processes to produce speech signal, TTS processes and speech synthesis aim to digitally mimic this same process.[x] Some technologies report that they are able to mimic the human voice accurately enough to make distinguishing whether a computer or a person is talking much more difficult.[xiii]

The TTS process is composed of two parts. Firstly, two major tasks are performed. Raw text that may include numbers and abbreviations are converted into the equivalent of words or written text. Once this text normalisation is completed, each word is assigned a phonetic transcription and then the text is divided and marked into units such as phrases, clauses and sentences. These units are referred to as prosody information which, together with the phonetic transcription, form a symbolic linguistic representation of the raw text as the output of the first stage. The second stage converts that output into sound. The speech synthesis process generates raw audio data as, for example, a base64-encoded string. These strings are able to be marked-up using Speech Synthesis Markup Language (SSML) as a process that allows the insertion of pauses, acronym pronunciations, pitch, speaking rate or other additional details

into the audio data created by TTS.[7] The base64-encoded string must be decoded into an audio file that is then able to be played by most applications. Many platforms and operating systems have such decoding tools that can convert base64 text into playable media files such as .wav, .mp3, or .mp4 audio.

4.5 Neural Networks and Deep Learning

Throughout the evolution of TTS conversion, the two main methods of speech synthesis have been Concatenative TTS and Parametric TTS.

Concatenative TTS stitches high-quality audio recordings together to form speech. The recordings are normally scripted and then tagged and segmented by linguistic units to form a large database or corpus. A TTS engine then searches the corpus for the speech units that match the input text, concatenates those units (joins them together) to synthesise speech as an audio file. This technique utilises a large database of short speech fragments, recorded from a single speaker. These types of systems[8] tend to be quite time consuming, requiring large volumes of input audio and high amounts of data hard-coding. Furthermore, text components are mapped to a particular voice fragment, which are then recombined to form complete utterances on synthesis. This makes it difficult to modify the voice (for example switching to a different speaker, or altering the emphasis or emotion of the speech) without recording a whole new database. At times, when a different style of voice is required, a new database of voices is needed which then limits the scalability of a concatenative approach.[xiv] Concatenation Synthesis tends to produce intelligible, lifelike speech audio.

Parametric TTS is a more statistical method than Concatenative TTS where all the information required to generate the data is stored in the parameters of the model. Parametric models use recorded human voice and a function with a set of parameters that can change the voice. Initially text is processed to extract linguistic features such as phonemes or duration. Some inherent characteristics of human speech, such as homomorphic signals, frequency and noise variation, are then extracted, hand engineered and combined with linguistic features to form the input into signal processing algorithms, mathematical models known as Vocoders.[9] Essentially the vocoder analyses and synthesises the human voice signal and estimates the parameters of the speech, like phase, speech rate and intonation, to produce synthesised speech.[xi] Approximations of the parameters that make up speech are then used to train speech-generation models.[xiii] Statistical Parametric Speech Synthesis brings together a number of hardcoded mathematical parameters and models to generate speech. This makes it much easier to modify the resulting voice, and requires very little voice recording to develop a unique voice. However, the resulting audio is often less intelligible, and contains artifacts like static. Although Parametric Synthesis can sound quite robotic, it is capable of producing an emotional voice (with some work) but also has much potential in terms of speaker adoption and speaker interpolation.

[7] See: https://cloud.google.com/text-to-speech/docs/basics

[8] For instance eSpeak and Orca - see: Section 5.6 & 0

[9] See: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio

Formant synthesis creates a synthesised speech output that is created using additive synthesis, a technique that adds tone quality by adding sine waves together, and an acoustic model (physical modelling synthesis) where the waveform sound is computer-generated using mathematical models and sound simulation algorithms. This type of speech synthesis does not use human speech samples at runtime but rather parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech, a method sometimes referred to as *rules-based synthesis*. Many systems based on formant synthesis technology generate artificial, robotic-sounding speech that would never be mistaken for human speech. However, formant-synthesised speech can be reliably intelligible, even at very high speeds, avoiding the acoustic glitches that commonly plague concatenative systems. High-speed synthesised speech can be used by the visually impaired, for instance, to quickly navigate computers using a screen reader. Because they do not have a database, formant synthesisers are usually smaller programs than concatenative systems and can therefore be used in embedded systems, where memory and microprocessor power are especially limited.

Innovations in data collection and processing have heralded the Deep Learning revolution, adding fresh perspectives to the areas of technology-generated speech. This has enabled a new emphasis – a focus shift from human-generated speech features to entirely machine-developed strictures. With deep learning technologies, it is now possible to create naturally-sounding speech that includes pitch, rate, pronunciation and inflection. Deep learning or deep neural networks is another variation of statistical synthesis (i.e. Parametric TTS) that is able to model complex context dependencies. Neural networks involve a series of algorithms that recognise relationships within vast amounts of data by mimicking how the human brain works.[xv] Each network is like a web and contains interconnected nodes called perceptrons which are designed to simulate the ability of the brain to distinguish relationships by recognising and discriminating. The outputs are added to the network which can then be used as inputs by other nodes in multiple layers. A feature of the network is the hidden fine-tuning of input weightings to a point where the neural network's margin of error becomes minimal.[xv] Neural networks have driven the recent surge in AI, enabling projects like self-driving cars, smart homes and speech bots that are practically indistinguishable from people.[vi]

Because current speech features are generally based on a human understanding of speech, they may not necessarily be correct. In the deep neural approach speech features are designed by machines without human intervention. The relationships between input text and their acoustic realisations are modelled by deep learning techniques to create acoustic features using analogous relationships, pattern recognition and connection, connected layering and most - probable parameter generation.[xvi] In this sense, deep learning is an artificial intelligence function that makes decisions by processing data and creating patterns in much the same way as a human brain.

Despite deep learning's many practical successes, there's still much it can't do. Using hierarchical forms of neural networks that are connected together like a web, neural networks are brain-inspired but are not really like the brain. The intelligence that deep learning gives computers can be exceptional at narrowly defined tasks – play this particular game, recognise

these particular sounds – but is currently not adaptable and versatile like human intelligence. Deep Learning speech synthesis techniques still require research and development.[vi]

4.6 Speech Synthesis

Speech synthesis is the technology-generated simulation of human speech. The ability to synthetically generate speech is important for user interfaces and has specific applications for people who may have an impeded ability to recognise text or to converse using speech. Developers such as Microsoft report that they have achieved breakthroughs such as human parity in conversational speech recognition and human parity in machine translation.[xvii] The quality of Speech synthesis has improved hugely with the application of deep learning networks with products like Deep Speech[xviii,10] and Wavenet.[11]

Neural TTS has significantly enhanced the listening experience and AI interactions with the human-like natural prosody – matching patterns of stress and intonation, and clear articulation of words.[xvii] Microsoft reports a milestone in text-to-speech synthesis with a voice production system, using deep neural networks, that makes the voices of computers almost indistinguishable from the recorded voices of people. Deep neural networks are used to overcome the limits of traditional TTS systems in prosody and in synthesizing the units of speech into a computer voice. Traditional TTS systems can result in a muffled, unclear voice synthesis as the models attempt to break down prosody into separate linguistic analyses and the acoustic predictions are governed by independent models. In comparison, neural models produce a more fluid and natural-sounding voice because they perform prosody prediction and voice synthesis simultaneously.[xvii]

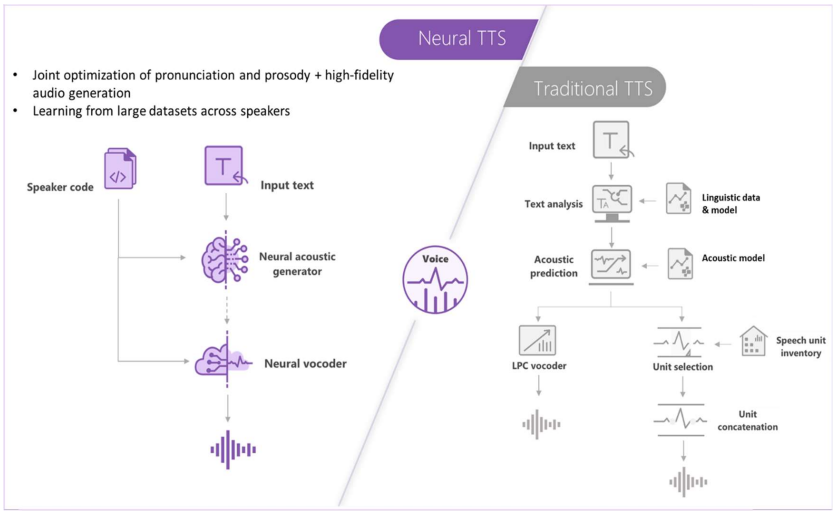


Figure 9 - Comparison of Text to Speech Approaches[xvii]

[10] See: https://hacks.mozilla.org/2019/12/deepspeech-0-6-mozillas-speech-to-text-engine/
[11] See Section 0

Neural TTS is a new method of generating text-to-speech where the system does prosody partition and voice synthesis simultaneously, so the outputs are of a much higher quality, in terms of accuracy and credibility, when compared to traditional methods of conversion.

4.7 Speech-to-Text

Speech-to-Text technologies involve recognition of spoken word followed by conversion into some form of text. Audio recognition software sends data to a Speech-to-Text API which performs recognition on that data and returns text-based results when the audio has been processed. With this type of technology users are able to dictate notes, texts, email, fill out forms, write documents, search the internet and control smart devices. Speech can now be automatically entered into many text locations – on a web page or a digital note for example. Web browsing and website control can also be actioned using voice commands.[12]

Speech-to-Text API synchronous recognition can process up to one minute of speech audio data and is regarded as the simplest method for performing recognition on speech audio data. This type of audio request processing is called synchronous because the Speech-to-Text API must return a response before processing the next request. Speech-to-Text typically processes audio faster than realtime, in most cases processing twice as fast as the spoken word.[13] If the audio quality is poor however, recognition requests can take significantly longer to process.[14]

Asynchronous Speech Recognition also sends audio data to the Speech-to-Text API before initiating a long running operation during which users are able to periodically poll for recognition results. One advantage of using asynchronous requests is that the process is able to convert audio data of any duration up to 8 hours.[16]

Streaming Recognition recognises audio data that is provided as bi-directional streams. Streaming requests are designed and deployed for real-time recognition which includes capturing live audio, from a microphone for instance. Conversion of streaming audio data can also provide interim results, such as transcription to occur while either the audio is still playing or while someone is still speaking.[24]

In summary, speech technology enables people to see and hear written text. Digital recognition (speech-to-text, speech translation), digital reproduction of the human voice (text-to-speech and speech synthesis) and the processing of speech data form the basis of speech technologies which are gaining increasing significance in technology-capable environments where the spoken word is progressively usurping traditional methods of manual input into technical systems.

Additionally, speech technologies are becoming a valuable capability for supporting human-to-human and human-to-machine communication, particularly for those who are visually impaired or are illiterate. Speech technology initiatives have evolved from the early prototypes

[12] See: https://chrome.google.com/webstore/detail/speech-recognition-anywhe/kdnnmhpmcakdilnofmllgcigkibjonof?hl=en
[13] See: https://cloud.google.com/speech-to-text for usable demo (Google)
[14] See: https://cloud.google.com/speech-to-text/docs/basics

which were initially quite clunky and stuttered. The prevalence of various forms of Artificial Intelligence (AI), are underpinned by the advent of big data and big data processing. Furthermore, the application of Machine Learning and Deep Learning models has progressed a range of current speech technologies that include:

- Speech recognition;
- Speech synthesis;
- Text and speech translation;
- Speaker recognition and verification;
- Multimodal interaction; and,
- Interactive voice response.[ii,iii]

The recent surge in accuracy and availability in these technologies is extraordinary.[15] The artificial production of human speech can now be completed by models that recognises arbitrary text or an image and is then able to convert that data into the spoken voice. In many cases speech signals have been reproduced that have very high intelligibility but still lack suitable levels of sound quality and naturalness.[xix] In other cases high voice quality and naturalness have been reported. The following section (Section 5) describes a variety of speech technologies that are achieving levels of accuracy with varying degrees of success.

The accuracy of speech technology output can be affected by a multitude of factors that include particular language variations, localised dialects or accents, the impact of background or ambient noise, voice changes due to illness or emotion, or the way humans change pitch or speaking speed for instance to suit the current discussion tone or environment. This means engineers must build reliable processing systems that are able to recognise and analyse a wide range of factors that might impact on the accurate recognition of text or speech and the accurate representation of synthesised speech.

In terms of the education sector, speech technologies have a myriad of uses that include the support of:

1. People who struggle to read text and who struggle to read text with comprehension;
2. Aural learners and people whose learning is best supported by a multi-sensory delivery.

Such multi-sensory learning experiences have been shown to improve word recognition and to embed information more accurately through systems of seeing and hearing.

[15] See: https://www.globalme.net/blog/the-present-future-of-speech-recognition/

5 Current Speech Technologies

A selection of current speech technologies have been identified and described.

Table 1: Comparative overview for selected technologies (Source: Carl Stephens - University of Waikato).

Technology	Google TTS	Microsoft TTS	Amazon Polly	Natural Reader	AMAI	eSpeak
Neural Voice	Yes	Yes	Yes	No	Yes	No
Languages Supported	40+	50+	29	16	Unknown	100+
Custom Voice	Yes	Yes	Yes	No	Yes	Yes
Speed Adjustment	Yes	Yes	Yes	Yes	Yes	Yes
Pitch Adjustment	Yes	Yes	Yes	No	Unknown	Yes
SSML Support	Yes	Yes	Yes	No	Unknown	Yes (Partial)
Reo Māori Support	No	No	No	No	No	Yes (Partial)
REST API	Yes	Yes	Yes	No	Yes	No
Multi-Language SDK	Yes	Yes	Yes	No	No	Yes
Open-source	No	No	No	No	No	Yes
Offline Usage	No	See below	No	Yes (Desktop app)	Yes	Yes
Self-Deployable	No	Yes	No	No	Yes	Yes

5.1 Google

Google Cloud Text-to-Speech provides developers with over 220 voices[16], of both neural (WaveNet)[23] and standard (Basic) types and in both male and female tones for over 40 languages. Google TTS applies DeepMind's[17] research in WaveNet and Google's neural networks to synthesize natural-sounding speech that is able to mimic lifelike interactions across multiple applications and devices.[18] Google TTS is mainly available for Android devices but can also be used for selected Google apps on iDevices.[19]

Capabilities:

- Customised voice for different audio profiles, e.g. Speakers vs. Earbuds
- Speed and pitch adjustment
- SSML support
- REST/gRPC API and multi-language SDK
- Custom voice generation for an existing language
- Audio returned in MP3 or LINEAR16 (WAV) format if specified, otherwise as a base64 string

Technologies:

- WaveNet - Google's deep learning model for audio. Based on DeepMind research. Voices produced using WaveNet are 'neural voices'. This technique can generate speech audio up to 20 times faster than real time, and is deployed online in Google Assistant[20]
- Basic - Makes use of acoustic modelling in statistical parametric text-to-speech (speech synthesis)[21] using a state-of-the-art long short-term memory recurrent neural network. Additionally, the development of a Uniform Multilingual Multi-Speaker Acoustic Model[22] means Statistical Parametric Speech Synthesis can be deployed for Low-Resourced Languages. Since the acquisition of training data is expensive and often difficult to source for low-resourced languages, this acoustic modelling approach utilises a long short-term memory (LSTM) recurrent neural network (RNN) aimed at partially addressing the language data scarcity problem. The salient feature of this approach is that, once constructed, the resulting system does not need retraining to cope with the previously unseen (low resource) languages due to language and speaker-agnostic model topology and universal linguistic feature set. Tests have shown that when small amounts of training data are available, pooling the data sometimes improves the overall intelligibility and naturalness. Furthermore, having a multilingual system with no prior exposure to a language is sometimes better than building a single-speaker system from a small dataset for that language.

The WaveNet model of generating audio is capable of producing speech that is very close to the human voice. Reportedly, users find the WaveNet-generated voices to be warmer and more human-like than other synthetic voices.[16] The WaveNet models are trained using the voice samples of actual people speaking, enabling the models to produce synthetic speech with

[16] For a full list of supported voices and languages, see:: https://cloud.google.com/text-to-speech/docs/voices
[17] See: https://deepmind.com
[18] See: https://cloud.google.com/text-to-speech
[19] See: https://play.google.com/store/apps/details?id=com.google.android.tts&hl=en_NZ&gl=US
[20] See: https://assistant.google.com
[21] See: https://research.google/pubs/pub42624/
[22] See: https://research.google/pubs/pub46142/

emphasis and inflection on syllables, phonemes, and words that sound more human-like.[16] The recordings are fed into a neural network for generating raw audio waves, where each audio sample is then trained on the previous audio sample. The essence of the WaveNet model is that each predicted voice sample, that has been conditioned by the previous sample, is then inserted back into the network to facilitate the prediction of the next voice sample.[xiii]

The usual speech synthesis paradigm employed by most other TTS systems is bypassed in the WaveNet models which create raw audio from scratch by directly modelling the raw waveform of the audio signal, one sample at a time. During training the network can identify the underlying structure of the recording, such as what tones follow each other and how realistic waveforms look. As well as yielding more natural-sounding speech, using raw waveforms means that WaveNet can model any kind of audio, including music.

Experimental comparisons between WaveNet determined at least a 50% improvement over other existing technologies.[23] When compared with two earlier technologies, Concatenative and Parametric, the Wavenet synthesised voice produced significantly better quality and was also able to learn the characteristics of different voices. Reportedly, the model was also able to reproduce dialectical differences and distinguish between first language and second language speakers.[24]

5.2 Microsoft

Microsoft Text-to-Speech (Azure Cognitive Services)[25] is available either through the Microsoft cloud service (Azure) or in a more limited capacity as a self-deployable container. The service provides more than 200 voices, including both neural and standard types, in male and female tones, for over 50 languages.[26]

Capabilities:

- Speed and pitch adjustment
- SSML support
- REST API and multi-language SDK
- Custom voice generation for an existing language
- Self-deployable
- Long audio creation service for speech longer than 10 minutes
- Selectable speaking style and emotional tones, e.g. 'Newscast' or 'Customer Service' voice styles.

Technologies:

- Neural Voice; Neural TTS (discussed in Section 4.6).
- Standard Voices - created using a mix of Statistical Parametric Synthesis and/or Concatenation Synthesis techniques (discussed in Section4.5).

The **Speech Application Programming Interface** or **SAPI** is an API developed by Microsoft to allow the use of speech recognition and speech synthesis within Windows applications. A number of versions of the API have been released, and provided either as part of a Speech Software Development Kit (SDK) or as part of the Windows Operating System (OS) itself.

[23] See: https://deepmind.com/blog/wavenet-generative-model-raw-audio/

[24] See: https://venturebeat.com/2019/02/21/google-cloud-text-to-speech-adds-31-wavenet-voices-7-languages-and-dialects/

[25] See: https://azure.microsoft.com/en-us/services/cognitive-services/text-to-speech/

[26] See: https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/language-support#text-to-speech

SAPI is used by a number of applications that include Microsoft Office, Microsoft Agent and Microsoft Speech Server.

In general, all versions of the API have been designed so that software developers can write an application to perform speech recognition and synthesis using a standard set of interfaces, accessible from a variety of programming languages. Third-party companies are also able to produce their own Speech Recognition and TTS engines or adapt existing engines to work with the SAPI as long as those engines conform to the interfaces that have been defined by Microsoft for their own engines.

The **Microsoft TTS voices** are speech synthesizers provided for use with applications that use the Microsoft Speech API (SAPI) or the Microsoft Speech Server Platform. Client, server, and mobile versions of Microsoft TTS voices are available.[27]

5.3 Amazon

Amazon Polly, from Amazon Web Services (AWS), is a cloud service that uses deep learning technologies to convert text into synthesised natural speech. Polly can verbalise sentences to help users see and follow the text while simultaneously hearing it read aloud. Polly has two types of voices, Standard and Neural. The standard voice concatenates words or recorded speech to form a sentence, while the neural voice is centred around sounding as human as possible by using different frequencies and tones in the voice. Both male and female voices are available for use, including children's voices. Polly is also available in two distinct speaking styles allowing narration, similar to a news reader's voice, and a more conversational style that is ideal for two-way communication.[28]

Pronunciation lexicons give additional control over how Amazon Polly pronounces words for the selected language. This enables users to customise the pronunciation and intonation of words using the provided phonetic alphabet.[29]

Basic Info: Available through their cloud service, Amazon Web Services. Compared to the other services, it appears to offer more niche features that reduce the reliance on third-party tools to further process the speech audio, but lacks in the number of languages supported. Provides both neural and standard voices, in male and female tones, for 29 languages.[30]

Capabilities:

- Speed and pitch adjustment
- SSML support
- REST API and multi-language SDK
- "Brand Voice" - Custom voice generation for an existing language
- Can return an additional stream of metadata that provides information about when particular sentences, words and sounds are being pronounced
- Can stream audio back in multiple formats/sampling rates (MP3, Vorbis, raw PCM)
- "Time Driven Prosody" - ability to specify a maximum time in which the text must be spoken
- Custom Lexicons - able to define custom pronunciation for certain words
- Selectable speaking style, e.g. 'Newscaster' or 'Conversational' voice style.

Technologies:

[27] See: https://azure.microsoft.com/en-us/services/cognitive-services/text-to-speech/

[28] See: https://aws.amazon.com/polly/

[29] See: https://docs.aws.amazon.com/polly/latest/dg/managing-lexicons.html

[30] See: https://docs.aws.amazon.com/polly/latest/dg/voicelist.html

- Neural Voice; Neural TTS (discussed in Section).
- Standard Voices - created using a mix of Statistical Parametric Synthesis and/or Concatenation Synthesis techniques (discussed in Section 4.5).

5.4 NaturalSoft Ltd

Natural Reader[31] is a consumer-oriented product that barely caters to those who need bulk, programmatically-driven speech synthesis but provides a powerful, easy to use product for individuals who need TTS capabilities. Natural Reader provides a free interface to the TTS voices built into computers and phones and offers paid access to two tiers of higher quality voices – the top tier being neurally generated. Natural Reader has personal, commercial and educational offerings.

Capabilities:

- OCR reading
- Desktop application
- Web application, with mobile support
- Dyslexia font
- Allows custom pronunciations to be defined for individual words
- Allows documents in various formats to be uploaded and synthesised.

Technologies:

Technology descriptions for NaturalSoft were not located. However, speech audio comparisons to other services indicate the following technologies may be used within the various voice tiers:

- Plus Voices; (see *Concatenation Synthesis* and *Parametric Synthesis* discussed in Section 4.6)
- Premium Voices - Created using Formant Synthesis. "Formant synthesis technique is a rule-based TTS technique. It produces speech segments by generating artificial signals based on a set of specified rules mimicking the formant structure and other spectral properties of natural speech. The synthesized speech is produced using an additive synthesis and an acoustic model. The acoustic model uses parameters like, voicing, fundamental frequency, noise levels, etc that varied over time. Formant-based systems can control all aspects of the output speech, producing a wide variety of emotions and different tone voices with the help of some prosodic and intonation modeling techniques."[xiii]
- Free Voices - Uses the TTS API built into the device in use.

5.5 AI Interaction Corp

AI Interaction Corp (AMAI)[32] is a 2019 start-up company that offers neural voices with an extremely flexible deployment/usage surface. Their public webpage is exceedingly light on any technical or detailed information about their technologies and features and signing up to the service requires that direct contact is made with the company. Despite this, the service has been included due to the non-market-standard focus on self-hosting and offline usage.

Capabilities:

- Self-hosting

[31] See: https://www.naturalreaders.com/index.html
[32] See: https://amai.io

- Model can be deployed on EDGE devices without internet access
- Custom-defined acronyms
- Different accents/emotions.

Technologies: Uses neural voices (see Section 4.5).

No further public documentation.

5.6 eSpeak

eSpeakNG is a compact, open-source, software speech synthesizer for platforms that include Linux and Windows. It uses a formant synthesis method[33] and much of the programming for eSpeakNG's language support uses rule files bolstered by feedback from native speakers. Due to its small size and support for over 100 languages and accents, it is included as the default speech synthesizer in the Non-Visual Desktop Access (NVDA), a free, open-source, portable screen reader for Windows, Android, Ubuntu and other Linux distributions.

Because eSpeak NG uses formant synthesis, many languages are able to be provided in small chunks. The speech is clear, and can be used at high speeds, but is not as natural or smooth as larger synthesizers which are based on human speech recordings.[34]

Capabilities:

- Includes different Voices, whose characteristics can be altered.
- Can produce speech output as a WAV file.
- SSML (Speech Synthesis Markup Language) is supported (not complete).
- HTML (HyperText Markup Language) is supported.
- Compact size. The program and its language data totals about a few Mbytes.
- eSpeak NG converts text to phonemes with pitch and length information.
- Can translate text into phoneme codes, so it could be adapted as a front end for another speech synthesis engine.
- Potential for other languages. Several are included in varying stages of progress.
- Help from native language speakers is welcome.
- Written in C.

The quality of the language voices varies greatly. In eSpeakNG's predecessor eSpeak, the initial versions of some languages were based on information found on Wikipedia and, consequently, have had more work or feedback from native speakers than others.

A small project was undertaken to test the suitability of eSpeak for te reo Māori. The results of that investigation are attached as Appendix 2 – Investigating eSpeak NG as a Tool for Developing a te reo Māori Text-to-Speech System.

[33] Discussed in Section 4.5

[34] See: https://github.com/espeak-ng/espeak-ng

Other technologies (not comparatively overviewed)

5.7 Nuance

Nuance[35] provides TTS technology that leverages neural network techniques to deliver human-like neural voices for 53 languages. The voices are specifically trained for specific use-case and dialogues. According to its developers, Nuance Vocalizer deliver "life-like voices that are trained on your use cases and dialogues, and speak your language as fluently as a live agent"[36].

Capabilities:

- Graceful blending of static and dynamic speech output
- Enhanced expressivity
- Improved multilingual support
- High-quality speech output
- Refined speech quality and accuracy through optimized text processing
- More comprehensive pronunciation dictionaries
- Complete voice refresh in many languages

Nuance Vocalizer uses advanced TTS technology based on recurrent neural networks.

5.8 Orca

Orca is a free and open-source, flexible, extensible screen reader from the GNOME project[37] for individuals who are blind or visually impaired. Using various combinations of speech synthesis and braille, Orca helps provide access to applications and toolkits that support a range of applications that include the GNOME desktop, Mozilla Firefox/Thunderbird, OpenOffice and GTK+[38], Qt[39] and Java Swing, a GUI widget toolkit for Java.

Orca, the default screen reader of the GNOME platform, is provided by default on a number of operating system distributions, including Solaris[40], Fedora[41], openSUSE[42] and Ubuntu[43] and follows the GNOME stable release cycles of approximately six-months.

5.9 Talkify

Talkify[44] provides 396 WaveNet-generated neural voices for 42 languages with a focus on providing premade integration tools for websites, ebook readers etc. Attributes such as pitch, volume, rate of speech, pauses are able to be adjusted and the ability to use a child voice for content aimed for children or to use a voice specific to a dialect is provided.

---

[35] See: https://www.nuance.com/omni-channel-customer-engagement/voice-and-ivr/text-to-speech.html

[36] See: https://www.nuance.com/omni-channel-customer-engagement/voice-and-ivr/text-to-speech/vocalizer.html

[37] See: https://www.gnome.org/about/

[38] See: https://developer.gnome.org/gtk3/stable/gtk-getting-started.html

[39] See: https://www.qt.io

[40] See: https://www.oracle.com/solaris/solaris11/

[41] See: https://getfedora.org/#:~:text=Fedora%20Server%20is%20a%20powerful,Learn%20more.&text=Fedora%20IoT%20provides%20a%20trusted,strong%20foundation%20for%20IoT%20ecosystems.

[42] See: https://software.opensuse.org/

[43] See: https://ubuntu.com/

[44] See: https://talkify.net/products/text-to-speech-voices

Most of the Talkify neural voices use WaveNet, a deep generative model of raw audio waveforms able to mimic any human voice (discussed in Section 4.6).

5.10 Mozilla Deep Speech

Mozilla DeepSpeech is an open source embedded (offline, on-device) deep learning-based automatic speech recognition engine (speech-to-text) engine which can run in real time on devices ranging from a Raspberry Pi 4 to high power GPU servers. The Mozilla Corporation aims to make DeepSpeech technology and trained speech models openly available to developers.[xvi] DeepSpeech v0.6 includes a host of performance optimizations, designed to make it easier for application developers to use the engine without having to fine tune their systems. Mozilla's new streaming decoder offers the largest improvement, which means DeepSpeech now offers consistent low latency and memory utilization, regardless of the length of the audio being transcribed. Application developers can obtain partial transcripts without worrying about big latency spikes.

DeepSpeech is composed of two main subsystems: an acoustic model and a decoder. The acoustic model is a deep neural network that receives audio features as inputs, and outputs character probabilities. The decoder uses a beam search algorithm to transform the character probabilities into textual transcripts that are then returned by the system.

5.11 Facebook

Facebook AI have recently introduced and open-sourced a new framework for self-supervised learning of representations from raw audio data known as wav2vec 2.0. The company claims that with just 10 minutes of transcribed speech and 53K hours of unlabeled speech, wav2vec 2.0 enables speech recognition models at a word error rate (WER) of 8.6 percent on noisy speech and 5.2 percent on clean speech on the standard LibriSpeech benchmark.[xx]

Neural network models have gained much traction over the last few years due to their applications across various sectors and the existence of vast quantities of labelled training data. Current systems require thousands of hours of transcribed speech to achieve acceptable output quality, which is quite challenging for low resource languages. To mitigate such challenges, researchers open-sourced the wave2vec framework which has the capability to make efficient development in Automatic Speech Recognition (ASR) for the low-resource languages.

The pretrained wav2vec 2.0 model learns basic speech units that are used to tackle a self-supervised task. The model is then trained to predict the correct speech unit for masked parts of the audio while learning what the speech units should be at the same time. The model's self-supervision method enables learning from unlabelled training data which, in turn, is able to facilitate speech recognition systems for many more languages, dialects, and domains that previously required vast amounts of transcribed audio data.[xxi] Facebook researchers have also developed a cross-lingual approach, dubbed XLSR, that can learn speech units common to several languages. This transfer learning approach means that low resource languages can benefit from languages for which more data is available.[xx]

Opacus is Facebook's recently open-sourced high-speed library for training specific models with Differential Privacy. The library, a computing framework for supporting machine learning algorithms, is claimed to be more scalable than existing state-of-the-art methods and can be used for applications such as computer vision and natural language processing.[45]

[45] See: https://ai.facebook.com/blog/introducing-opacus-a-high-speed-library-for-training-pytorch-models-with-differential-privacy/

5.12 Usage Costs of Major Speech Synthesis Systems

5.12.1 Google

Usage of the standard voices allows 4 million characters to be synthesised for free before a charge of 4USD per one million characters is incurred. Usage of the WaveNet voices allows one million characters to be synthesised for free before a charge of 16USD per one million characters is incurred. There is no separation between free and charged tiers, so care should be taken to ensure the free limit isn't unknowingly exceeded, incurring unwanted charges. Billing options can be defined before using the service.

5.12.2 Microsoft

Microsoft offers five million normal characters/0.5 million neural characters free per month. Past this point, for every one million characters spoken the charge is: 4USD for standard voices or 16USD for neural voices. There is a 100USD charge per one million characters for long neural audio creation.

5.12.3 Amazon

One year limited free tier. Otherwise, for every one million characters spoken the charge is 4USD for standard voices or 16USD for neural voices

5.12.4 Natural Soft Ltd

Prices below are for the commercial version of NaturalReader.
Single User Plan: 49USD / month.
Team Plan: Starts at 59USD / month for two users, scaling by 10USD / month for each additional user.
The education version is currently free to support the COVID-19 effort, however listed below are the normal prices.
5 users - $199/year
10 users - $299/year
30 users - $699/year
50 users - $899/year
50+ users - $18/user/year*
*maximum 2000 users; all prices in USD

6 Current Speech Initiatives

6.1 MAONZE Project

The Māori and New Zealand English (MAONZE) Project was established in 2004 to examine changes in pronunciation of te reo Māori and New Zealand English over time. Primarily funded by two grants from The Royal Society of New Zealand's Marsden Fund, the project is led by world renowned linguistic and Māori language experts including Catherine Watson and Peter Keegan from the University of Auckland, Margaret Maclagan and Jeanette King from the University of Canterbury and Ray Harlow an independent researcher formerly at the University of Waikato. Catherine Watson is an acoustic engineer with a long history in speech science research. She has developed synthetic voices (text-to-speech) for both New Zealand English and te reo Māori[46] and also has worked on developing speech recognition tools.

The research examines the pronunciation of early speakers, from different regions in New Zealand, with modern speakers of te reo Māori. The analysis compares the pronunciation of fluent native Māori speakers born in the late 19th century, using an archive of recordings of that generation of speakers, with modern speakers of Māori of different age groups. The analysis highlights how pronunciation of the Māori language has altered over time, adapting to on-going interaction with New Zealand English while retaining its own character.[xxii] The project outputs include 28 publications and 34 presentations on the MAONZE website.[47]

To date, changes in the pronunciation of vowels and the most common diphthongs have been completed, along with preliminary work on aspiration of plosives and changes in the rhythm of the language.[xxii] Dialectal differences and changes have also been tracked over time. The international implications are considerable given that no other indigenous language has been subject to such a longitudinal analysis which reveals change in pronunciation over time which provides foundation for revitalisation efforts.[xxiii]

The pronunciation of vowels in Māori has changed significantly over time, in terms of both quality and quantity (i.e., duration). Within the Māori speaking community sound change was not seen as a natural consequence of a living language, but rather as a break in intergenerational transmission. Māori pronunciation is not well taught, nor understood by many Māori-language teachers and the Māori sound system is very different to NZ English.

Prompted by a large number of second language speakers of te reo Māori, intent on improving their pronunciation, a Māori Pronunciation Aid (MPAi) was developed. MPAi language aid is currently a Windows-based app, with three components:

- listening/video (vowels, words);
- vowel/word recognition; and,
- visual display of user vowels via formant plot.

The app was mostly trialled in 2019 using twenty two mostly-fluent speakers of te reo Māori. Further trials are planned in 2021.

A Māori Pronunciation Aid (MPAi) (see Figure 1) is supported by and draws upon the speaker database collated by the MaoZNE team.

Commented [1]: you will need to add this to your references

Commented [2R1]: sweet

Commented [3]: Recheck original endnote

[46] See: https://aotearoavoices.nz/
[47] See: http://homepages.engineering.auckland.ac.nz/~cwat057/MAONZE/publications.html

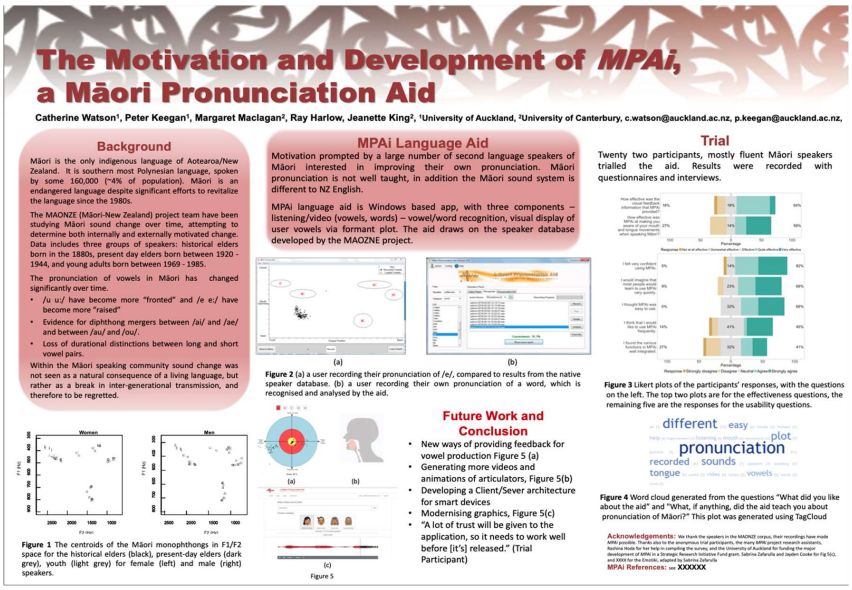


Figure 1: MPAi Poster[48]

The MAONZE team are collaborating with Te Hiku Media.[49] The benefits of aligning with a media specialist include:

- the availability of a Māori synthetic voice
- access to a broadcaster's (trained) voice
- the ability to use a range of voice types (robotics vs. schools/kids Māori voice)
- a choice of male and female voices

Because Te Hiku are a media organisation, the collaboration might also realise 'Black-box recordings', top-of-the-range recording tools and sound-proofed recording locations to ensure optimal sound clarity and quality.

6.2 Te Hiku Media

Te Hiku Media is a charitable media organisation, collectively belonging to the Far North iwi of Ngāti Kuri, Te Aupouri, Ngai Takoto, Te Rārawa and Ngāti Kahu. Te Hiku Media began in 1990 as an iwi radio station - Te Reo Irirangi o Te Hiku o Ika, but quickly realised that the role involved more than just broadcasting, so Te Hiku Media was formed. The station is an iwi communications hub for iwi radio, online TV and media services. Some key individuals in the organisation are Peter-Lucas Jones (CEO) and Keoni Mahelona (CTO).

Māori language revitalisation is a core focus of Te Hiku Media and, in doing so, Te Hiku are also using te reo Māori and technology to empower their people. The kaupapa of Te Hiku Media is best articulated through the vision and mission of the organization which was confirmed by a hui of kaumātua and kuia (elders), and other native speakers at Mahimaru

[48] Source: http://homepages.engineering.auckland.ac.nz/~cwat057/MAONZE/ISposter.pdf
[49] See Section 6.2

Marae in May 2013.[50] Their vision states: "He reo tuku iho, he reo ora'— living language transmitted inter-generationally". Their mission states: "Whakatōkia, poipoia kia matomato te reo Māori o ngā haukāinga o Te Hiku o Te Ika— Instil, nurture and proliferate the Māori Language unique to the homelands of Te Hiku o Te Ika (Northland)".

As mentioned previously Te Hiku Media began as a Māori radio station and in that role they began collecting and archiving recordings of Māori speakers. In this role they also realised they needed a storage facility that they could control and that would allow the 90% of their iwi that were not located on their home lands to have access to. So they built a website repository, called Whare Kōrero to where they began uploading material. With their live recording and live streaming they found they had a lot of material that needed to be digitised. This would allow for transcription, and then material enrichment through descriptions and hyperlinks. The subsequent resources were more valuable and more readily available online to their iwi members.

Transcribing voice and video is very time consuming. So in 2016 Te Hiku began to look to speech recognition tools that could assist with transcription. Given that initially they had little funding, they looked for an Open Source platform and decided on Deep Speech from Mozilla. DeepSpeech is an open-source implementation of Baidu's Deep Speech Neural Network Architecture (Cordoza, 2017). It is an architecture that uses machine learning techniques to provide speech recognition.

With assistance from a company called DragonFly Data Science they began working on an Automatic Speech Recognition System (ASR). An ASR requires three items; an alphabet, text to build a language model and utterances to build an acoustic model. The text and utterances need to be quality assured, but once you have done this you can use the model to further grow the corpus.

Te Hiku Media used a novel and innovative method to capture their language data. They built a website to streamline the process of collecting and reviewing data[51] and then ran a series of competitions where they encouraged Māori groups and individuals to sign on and contribute. The effective capture of 310 hours of speech data in 10 days as a result of intensive crowdsourcing was an unprecedented success.

Te Hiku Media have been successful in securing funding from a number of sources including the Ka Hao Fund, the National Science Challenge for Technology (SfTI), the Data Science Platform Fund and the Strategic Science Investment Fund. This funding support is rare for non-University based projects and has allowed them to form a partnership with Dragonfly, and with two key individuals there, Finlay Thompson the CEO, and Caleb Moses a Māori data scientist.

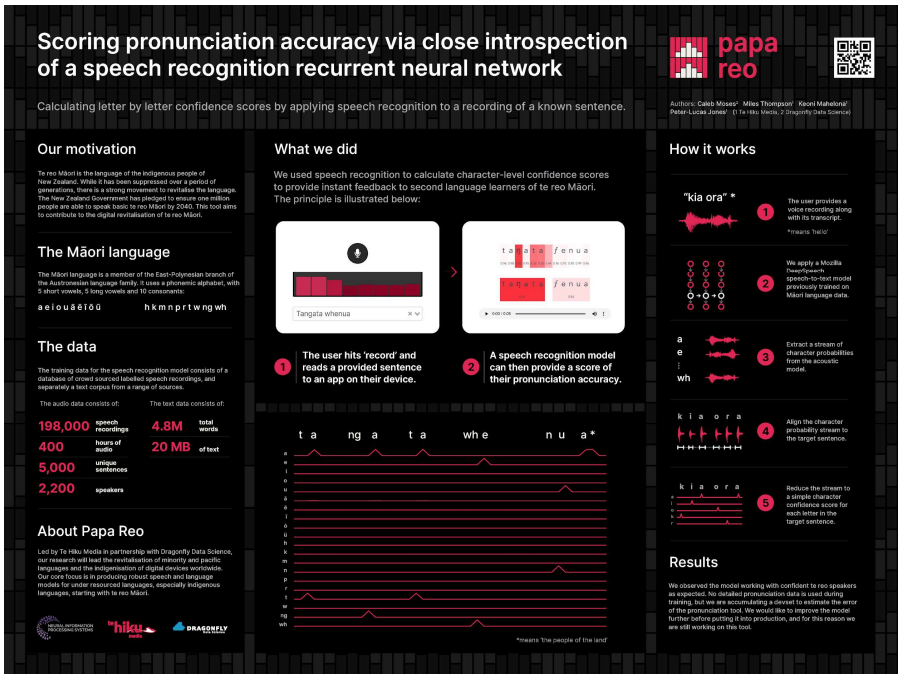
One of the reasons why they were so successful in capturing so much Māori language data was because of the Māori communities' trust they had built, in their work as a radio station and a Māori media platform for their iwi. Te Hiku Media take this trust very seriously and are staunch advocates for Māori Sovereignty over Māori data. For the language data they hold they do not consider that they are owners, rather they are kaitiaki (guardians), and as such have created a new license called the 'Kaitiaki Licence.[52] This license states that the data and the tools created from that data will be managed under tikanga Māori, will be held in guardianship rather than ownership, its use will respect the mana from the people from whom it descends with any financial benefits returning to those peoples.

[50] https://tehiku.nz/about/

[51] https://koreroMāori.com/

[52] https://github.com/TeHikuMedia/Kaitiakitanga-License

With the language data that Te Hiku Media collected they were able to bootstrap a speech recognition model for te reo Māori, the first of its kind using deep learning techniques. This model was created in 2017 and is explained in the following poster, presented at the 34th Conference on Neural Information Processing System, Dec 2020.[53]



The Te Hiku Media Māori language model has led to development of a number of other tools. Kaituhi is a web app they built to help transcribe their archives. It is a web based tool, they can be easily shared so multi people can work on transcriptions, but something also that is integrated back to their speech recognition system.[54] They have a real time Speech Pronunciation Tool which gives live feedback to speakers of te reo Māori language to improve pronunciation. Te Hiku Media are also using Mozzilla's Deep Speech to build a preliminary Speech Synthesis Tool.

[53] https://papareo.nz/docs/PapaReo_NeurIPS2020_Poster.pdf
[54] https://kaituhi.nz/

6.3 User-Friendly Deep Learning (UFDL)

The User-Friendly Deep Learning project is based within the Department of Computer Science at the University of Waikato. To address the shortage in New Zealand of deep learning specialists, the project aims to create new applications of machine learning that increases productivity and yields better decision-making. Access to deep learning technology for industry and research institutions wishing to develop their own applications is compromised by the need for expert knowledge in deep learning architectures and algorithms, and the difficulty of developing deep learning models for new domains outside standard application areas such as object recognition, text classification, and speech recognition. The UFDL project aims to enable domain experts to apply deep learning without involving a machine learning expert and without requiring any programming, while minimising the amount of data labelling required. Using a carefully designed software and an interactive graphical user interface (GUI), one objective is to engage the end-user in the deep learning process in a manner that automates model selection and parameter tuning, and does not require any programming. This will enable access to deep learning technology for a much wider sector of the economy without the involvement of machine learning experts and allow end-users to build predictive models directly without involving deep learning experts. The overarching objective is to yield more accurate solutions in shorter time frames.[55]

It is the intention of UFDL to make available Deep Learning tools with a Text-to-Speech interface in te reo Māori. Investigations have been initiated to determine the most appropriate and applicable technologies. Some early testing has begun with the Mary TTS system, using resources kindly shared by the MAONZE project.

MaryTTS (Modular Architecture for Research on speech Synthesis Text-to-Speech) is an open-source, multilingual Text-to-Speech Synthesis platform. Written in Java, MaryTTS arose from a collaborative project between the Speech and Language Technology Laboratory[56] of the German Research Center for Artificial Intelligence[57] and the Institute of Phonetics[58] at Saarland University[59]. The latest version, MaryTTS 5.2, supports German, British and American English, French, Italian, Luxembourgish, Russian, Swedish, Telugu, and Turkish.

Mary TTS has undergone significant development by the MAONZE project for te reo Māori.[60] Using some of their resources, including a pronunciation dictionary and their reo Māori datasets the practicability of building a TTS system was tested by Ethan McKee-Harris, University of Waikato. The results of that investigation are attached as Appendix 3 – Investigating Mary TTS as a Tool for Developing a Te Reo Māori Text-to-Speech System.

[55] Source: https://waikato-ufdl.github.io
[56] See: https://www.dfki.de/en/web/research/research-departments/speech-and-language-technology/
[57] See: https://www.dfki.de/web/
[58] See: http://www.coli.uni-saarland.de/groups/WB/Phonetics/
[59] See: https://www.uni-saarland.de/start.html
[60] See: Section 6.1

7 Digital Assistants

A Digital Assistant, also referred to as a predictive chatbot, is an advanced computer program that simulates a conversation with the people who use it, typically over the internet. Digital assistants use advanced AI, NLP, natural language understanding, and ML to continuously learn, providing a personalised, conversational experience, but more importantly, recalling historical information such as purchase preferences for instance, to create data models that identify and refine patterns of behaviour as data is added. By learning a user's history, preferences, and other information, digital assistants can answer complex questions, provide recommendations, make predictions, and even initiate conversations.[61]

Digital assistants can accurately predict human behaviour, initiate a conversation, answer queries and carry out tasks based on voice commands. Progress remains slow but recent breakthroughs hint at the advent of ambient computing. While it may seem remarkable that digital assistants can whisper now, the real point perhaps is that they can whisper back in much the same way as your friend might also lower their voice when you start speaking quietly or conspiratorially from across the table. Some assistants can also remind you to do things that you would normally do, like lock the door when you go out or turn the light off when you go to bed.

7.1 Google

Google Assistant is an AI-powered virtual assistant developed by Google, mainly available on mobile and smart home devices, that can engage in two-way conversations. The Assistant has been further extended to support a large variety of devices, including cars and third party smart home appliances. The functionality of the Assistant can also be enhanced by third-party developers.

Users primarily interact with the Google Assistant using their own voice, although keyboard input is also supported. Google Assistant is able to search the Internet, schedule events and alarms, adjust hardware settings on the user's device, and show information from the user's Google account. Google has also announced that the Assistant will be able to identify objects and gather visual information through the device's camera, support purchasing products and sending money, and identify songs[62]. From asking for phrases to be translated into another language, to converting weight units and currency, Google Assistant not only answers correctly, but also gives additional context and cites website sources which is perhaps less astounding given that it's backed by Google's powerful search technology.

Google previewed new technology that makes speech recognition strikingly more responsive, suggesting voice control could soon be seamless enough to be irresistible. As of late 2017, Google boasted a 95% word accuracy rate for U.S. English; the highest out of all the voice-assistants currently out there. This translates to a 4.9% word error rate – making Google the first of the group to fall below the 5% threshold. At its annual developer conference in Mountain View, Google boasted of shrinking its speech recognition software to 1/25th of its

[61] Source: https://www.oracle.com/chatbots/what-is-a-digital-assistant/
[62] See: https://assistant.google.com/

prior size. CEO Sundar Pichai called that a “milestone” because it means software that traditionally lives in Google’s cloud servers can be installed in Pixel smartphones Google will launch later this year, allowing the devices to respond to a person’s voice much more quickly.[63]

Google recently announced an update to its Assistant, which works across smartphones and Google Home devices, that was supposed to make it more conversational. For a while now you’ve been able to ask the Assistant a question, like “How tall is Eiffel Tower?”, and immediately ask it a follow-up question without having to say the Eiffel Tower again. Google has extended the Assistant’s memory so that, following a question or a command, it will continue to listen for 8 seconds afterwards, so you don’t have to keep repeating the Wake phrase. Google also gave its Assistant the ability to do some chores for you—things like screen your calls on an Android phone, or (in a feature called Duplex, which rolled out recently) hold telephone conversations with an actual human to book a table at a restaurant or an appointment at the salon.[36]

7.2 Siri

Apple’s Siri was the first digital voice assistant to be created by a mainstream technology company in 2011 and has since been integrated on all iPhones, iPads, the AppleWatch, the HomePod, Mac computers and Apple TV. Siri is even being used, via your mobile device, as the key user interface in Apple’s CarPlay system for automobiles as well as the wireless AirPod earbuds. The release of the development tool SiriKit lets third-party companies integrate with Siri and HomePod, Apple’s attempt at an intelligent speaker, Siri’s abilities become even more robust.[64]

Siri is the digital virtual assistant that is part of Apple Inc.'s iOS, iPadOS, watchOS, macOS, and tvOS operating systems. The assistant uses voice queries, gesture based control, focus-tracking and a natural-language user interface to answer questions, make recommendations and perform actions by delegating requests to a set of Internet services. Siri uses advanced machine learning software that, over time, adapts to each individual users' language uses, searches and preferences – individualising the overall experience and connection based on the stored results.

Siri supports a wide range of user commands that includes completing phone actions, checking basic information, scheduling events and reminders, handling device settings, Internet searches, navigation, finding entertainment information and is able to engage with iOS-integrated apps. Later iOS releases opened up limited third-party access to Siri, including third-party messaging apps, payments, ride-sharing and Internet calling apps.

7.3 Cortana

Cortana is the virtual digital assistant developed by the Microsoft Corporation and uses the Bing search engine to perform tasks such as setting reminders and answering questions. Editions of Cortana are currently available in English, Portuguese, French, German, Italian,

[63] See: https://www.wired.com/story/voice-assistants-ambient-computing/
[64] See: https://www.globalme.net/blog/the-present-future-of-speech-recognition/

Spanish, Chinese and Japanese, depending on the software platform and the region in which it is used.

In 2016 a research group from Microsoft Artificial Intelligence and Research announced an error rate of 5.9%, subsequently surpassed in late 2017 with a conversational speech recognition system error rate of 5.1%At that time this was better than the Google error rate and puts its accuracy on par with professional human transcribers who have advantages like the ability to listen to text several times.

While percentages and accuracy-rates are important, Cortana differentiates itself from other voice-assistants by actually being based upon real, human personal assistants.

7.4 Amazon Alexa

Alexa, developed by Amazon, was first used with the Amazon Echo smart speakers and is capable of a range of functions such as controlling smart applications, setting alarms, playing music, voice interaction and providing real time information like weather and traffic conditions. These functions are able to be extended by installing additional functionality made available by third-party developers.[65]

Housed inside Amazon's very popular Amazon Echo smart speaker as well as the newly released Echo Show (a voice-controlled tablet) and Echo Spot (a voice-controlled alarm clock), Alexa is currently one of the most popular voice-assistants. Whereas Apple focuses on perfecting Siri's ability to do a small handful of things versus expanding its areas of expertise, Amazon wagers that the voice assistant with the most "skills, will gain a loyal following, even if it sometimes makes mistakes and takes more effort to use".

Although Alexa's word recognition rate is regarded by some users as being a shade behind other voice platforms, Alexa adapts to your voice over time, offsetting any issues it may have with your particular accent or dialect. Additionally, Amazon's Alexa Skills Kit (ASK) is perhaps what has propelled Alexa forward as a bonafide platform. ASK allows third-party developers to create apps and tap into the power of Alexa without ever needing native support. With over 30,000 skills and growing, Alexa certainly outperforms Siri, Google Voice and Cortana combined in terms of third-party integration. With the incentive to "Add Voice to Your Big Idea and Reach More Customers" and the ability to build for free in the cloud ("no coding knowledge required"). it is little wonder developers are rushing to put content on the Skills platform.Similar to Google Assistant, Alexa has a recently released "Follow up" feature, in which you can ask Alexa, say, the weather in a particular city, and then ask about a restaurant in that same city without having to identify the city again.

7.5 Dragon Assistant and Dragon Naturally Speaking

Though Nuance hasn't come out with a smart home speaker, their **Dragon Assistant** and **Dragon Naturally Speaking** systems have been used as the speech recognition backbone for other tech companies. Although much of Nuance's voice-recognition technology is centered

[65] See: https://www.amazon.com/b?ie=UTF8&node=17934671011

around in-car speech systems; bringing embedded dictation capabilities and conversational infotainment to the car, the Nuance Communications[66] philosophy is to just be able to talk to a device without touching it. The device will be constantly listening for trigger words that execute an action — pop up a calendar, or ready a text message, or use a browser to fulfil a search. A further development involves deeper levels of understanding, where the aim is to not only recognize speech, but also to extract the meaning and intent of what has been said, enabling voice driven systems as a whole to react in an intelligent way that is appropriate to the user's needs.

Summary

Despite efforts to make virtual assistants human-sounding, they still require us, the *real* humans in the equation, to talk to them like robots. Basically, they sometimes fail to understand natural language despite using advanced natural language processing. There are many user testimonials that continue to voice frustrations that their devices are difficult to talk to or don't listen to them or don't understand what they are saying. However, good voice control presents just as many ethical problems as it does moments of ease. Virtual assistants are entering our lives just as we're becoming more aware of the insidious data-sharing practiced by some of the world's biggest tech companies. The advent of interactive websites and the prevalence of large social media platforms have allowed organisations to collect, use and share data gleaned from users who have been actively imparting shopping queries, future destinations, romantic interests and even their innermost thoughts. Now the voice control systems from Amazon, Google, Apple, Microsoft, and Facebook are collecting and analysing our spoken words.

Given what digital assistants are able to do and what they let us not do, it appears that privacy concerns are not a huge deterrent for current or potential users of voice-controlled assistants. Users are willing to put privacy aside for a little bit of convenience. And according to IDC's research, privacy isn't even the leading inhibitor to using a smart assistant; the majority of survey respondents (more than 31 percent) said they just "have no use for them."[67]

[66] See: https://www.nuance.com/index.html
[67] See: https://www.wired.com/story/voice-assistants-ambient-computing/

8 Analysis and Conclusions

Speech technologies are gaining increasing significance as the spoken word and digital conversation is increasingly becoming the preferred method of interaction with one's technology. The ability for technology to process input and then generate intelligent speech supports a raft of initiatives that range from mundane action, dimming lights for instance, to assistive technologies for those who are sight-impaired, to navigation and translation requirements and also to support education in online and offline environments.

Speech synthesis is extremely challenging due to the high levels of technical requirements that are needed to deliver accurate, high-quality, naturally-sounding audio from text. Advanced machine learning methods and the acquisition of suitable amounts of 'training' data enables high-quality, naturally-sounding voice synthesis but sourcing adequate amounts of training data can be problematic for low resource languages such as te reo Māori. However, there is an expectation that the accuracy for te reo Māori will improve as the technology improves and is able to be deployed on smaller sets of Māori language corpora.

While we have reviewed a variety of text-to-speech and some speech-to-text technologies, it appears that current speech synthesis technologies repurposed for te reo Māori would suffer from a dearth of suitable data and data types. Additionally, issues of scale would suggest that developing and maintaining bespoke systems from scratch would be onerous and prone to early obsolescence when compared to the larger models already available. Using the more popular speech technologies that are described in Section 5 might seem more sensible, especially in terms of currency and version updates, but issues of data management and guardianship would surface especially in terms of data protection and privacy and in terms of ensuring Māori language data remains 'onshore'. As mentioned in Section 4, to undertake Natural Language Processing of te reo Māori significant reo Māori data must be obtained. Perhaps investigating the Te Hiku Media model in some depth may yield solutions where offshore technologies can be used in ways that ensure data sovereignty remains with Māori.

The Te Hiku model is based on using Mozilla DeepSpeech but other options such as the recently open-sourced high-speed library for training specific models with Differential Privacy by Facebook AI Research should also be investigated as possible solutions.

9 NZQA - summary

9.1 NZQA Current Position

In 2018 NZQA entered into a contract with SoNET Solutions to deliver NZCEA examinations online. SoNET Systems is a Melbourne based software company that primarily offers two products; iCase and Assessment Master. Assessment Master is the online assessment software solution that is used in NCEA online exams. NZQA has the flexibility to create, alter and mark online exams through this system. The system provides spell checking and text to speech facilities for the English language, but not for te reo Māori.

The SoNET software uses Amazon Polly, a cloud based service, to deliver their Text To Speech (TTS) facilities. This currently delivers life-like speech for English, with the facility offering male and female voices in either an American, an English or an Australian accent. As stated in Section 5.3 Amazon Polly has a lot of features that can be adjusted to alter the sound of the voice and it offers both standard voice types and neural voice types.

As Amazon Polly is a web based service it receives requests, in terms of words of sets of words from a client host website. The client host website's back end application is responsible for generating the requests which include parameters such as language, ID of voice, gender of voice and either the Neural or Standard voice type. These are configured by the host and are a part of their AWS profile configuration. Amazon Polly receives all the information in this request and then returns a MP3 file that is played by the client browser for the user to hear.

In an ideal world Amazon Polly would have a Neural Māori voice that would be able to be adjusted to provide MP3 voice files of different characters e.g. an elderly Māori woman, a younger Māori adult, a Māori child. However as shown in the table at the beginning of Section 5, this facility, a Māori voice, is currently not available in Amazon Polly or any of the other large commercially supported Neural TTS systems.

Amazon Polly does offer the facility to build a 'Brand' voice[68] but rather than a new language, this is more of a flavour on an existing language. Facilities also exist to alter the pronunciation of words using specified lexicons,[69] and the encoding of intonation and pauses using Speech Synthesis Markup Language (SSML) tags[70]. But again, these facilities allow for the 'flavouring' of a language rather than the generation of a completely new language.

A recent news article[71] suggests that the Bank of New Zealand has customised Amazon Polly to speak Māori, but on closer investigation it is apparent that it is just a 'Brand' voice that has been customised.

[68] https://aws.amazon.com/blogs/machine-learning/build-a-unique-brand-voice-with-amazon-polly/
[69] https://docs.aws.amazon.com/polly/latest/dg/managing-lexicons.html
[70] https://docs.aws.amazon.com/polly/latest/dg/supportedtags.html
[71] https://www.zdnet.com/article/bank-of-nz-to-launch-text-to-kiwi-voice-service/

9.2 NZQA - possible solutions

The Assessment Master software used by NZQA to deliver NCEA exams online does not appear to be able to offer a current solution of text-to-speech for te reo Māori. NZQA could enter into negotiations with SoNET Solutions to make this available but, given the underlying technology, this would appear problematic.

A short term solution could involve a Māori language speaker recording the texts of the exams, and then making these sound files available to the Assessment Master software. This would enable relevant textual parts of the assessment material to be read aloud until automated TTS systems are built and deployed.

We have investigated two concatenative systems to test their suitability for te reo Māori. Concatanative systems are possible for testing because they are usually open source, and there is supporting documentation to assist in development. Both eSpeak and Mary TTS were tested to see if they could perform as a solution for NZQA. Both systems have challenges and issues that would necessitate further development, but could be a future solution. The Mary TTS system is under further development by the MAONZE project.

Parametric and, in particular, Neural TTS is clearly the technology of the future. The Mozilla Deep Speech and FaceBook AI systems are systems that warrant further investigation but were beyond the scope of this report. Both systems will require a large amount of language data to build the necessary language models. Herein lies the difficulty. Collecting the language data is an onerous task fraught with data sovereignty issues. Māori groups and organisations are unlikely to make their language data available if they do not have control of that data and if financial return to the holders of that data is not realised.

Te Hiku Media is an organisation that is in a unique position. They have sourced and collated their own Māori language data. That data is protected using Te Hiku Media's own kaitiaki license. A Māori language model has been built and subsequently deployed into a Speech-to-Text system. Early trials of the system have returned positive results that might then support further TTS development using the Te Hiku API.

9.3 Conclusion

A raft of TTS technologies have been reviewed and found to be suitable for te reo Māori TTS options but largely unusable because of a lack of high quality language data that can be represented in a variety of data types and will be appropriate for machine training and learning.

Current initiatives that were reviewed are still in the early stages of TTS development. This includes the Te Hiku API, eSpeak and Mary TTS. The latter both need further work and suitable volumes of high quality language data to enable the completion and deployment of accurate TTS systems for te reo using these applications. Interim manual systems may have to suffice until pertinent machine-based solutions are made available.

10 Appendices

10.1 Appendix 1 – Basic API Usage

10.1.1 Google

The following curl command demonstrates a JSON request to synthesise audio for the input text “I’ve added the event to your calendar.”. The output uses the female ‘Basic A’ voice for British English and is requested to be returned in an MP3 format.

```
curl -H "Authorization: Bearer "$(gcloud auth application-default print-access-token) -H
"Content-Type: application/json; charset=utf-8" --data "{
  'input':{
    'text':'I\'ve added the event to your calendar.'
  },
  'voice':{
    'languageCode':'en-gb',
    'name':'en-GB-Standard-A',
    'ssmlGender':'FEMALE'
  },
  'audioConfig':{
    'audioEncoding':'MP3'
  }
}" "https://texttospeech.googleapis.com/v1/text:synthesize"
```

10.1.2 Microsoft

The following curl command demonstrates an *SSML* request to synthesise audio for the input text “my voice is my passport verify me”, using the female ‘Aria’ voice for US English. More detailed information about the REST API usage is available at https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-text-to-speech

```
curl --location --request POST
'https://INSERT_REGION_HERE.tts.speech.microsoft.com/cognitiveservices/v1' \
--header 'Ocp-Apim-Subscription-Key: INSERT_SUBSCRIPTION_KEY_HERE' \
--header 'Content-Type: application/ssml+xml' \
--header 'X-Microsoft-OutputFormat: audio-16khz-128kbitrate-mono-mp3' \
--header 'User-Agent: curl' \
--data-raw '<speak version=\"1.0\" xml:lang=\"en-US\">
    <voice xml:lang=\"en-US\" xml:gender=\"Female\" name=\"en-US-AriaRUS\">
        my voice is my passport verify me
    </voice>
</speak>' > output.wav
```

10.1.3 Amazon

Below a JSON request to synthesise neural audio for the input text “Hello world”, using the male ‘Russel’ voice for US English. More detailed information about the REST API usage is available at
https://docs.aws.amazon.com/polly/latest/dg/API_SynthesizeSpeech.html

```
{
  "Engine": "neural",
  "LanguageCode": "en-AU",
  "OutputFormat": "mp3",
  "Text": "Hello world",
  "TextType": "text",
  "VoiceId": "Russel"
}
```

10.2 Appendix 2 – Investigating eSpeak NG as a Tool for Developing a reo Māori Text-to-Speech System

Author: Carl Stephens | cs260@students.waikato.ac.nz

Supervisors: Te Taka Keegan | tetaka@waikato.ac.nz
David Bainbridge | davidb@waikato.ac.nz

Introduction

The goal of this investigation was to see whether the eSpeak-NG text-to-speech (TTS) system could be modified to be capable of synthesising te reo Māori from any valid text input. Furthermore, the modified package should be deployable on most modern unix-based systems without the need for elevated permissions.
I considered a 'deployable' circumstance to be where the end user had access to a C compiler that supports C99, and a C++ compiler that supports C++11. A one-time script could then be run to compile and install our eSpeak NG package, and any dependencies, under a custom prefix.

About eSpeak NG

Basics

- Written in C
- 123 languages and accents supported in the latest development version (including a barebones reo Māori implementation)
- Can produce speech output in WAV format
- Basic support for SSML (Speech Synthesis Markup Language)
- Compact - the compiled program (including data) is only a few megabytes in size
- Can translate text in phoneme codes, enabling it to be used as a frontend for another speech synthesis engine
- Released under the GPL v3 (or greater) license

History

The eSpeak project was originally known as **speak**, written for Acorn/RISC_OS computers starting in 1995 by Jonathan S. Duddington. In 2007, Jonathan re-wrote and enhanced the project to create the **eSpeak** engine, which is still hosted on Sourceforge at http://espeak.sourceforge.net.

In 2010, Reece H. Dunn took up a maintainer role and shifted the codebase to Github, along with porting the build system to autotools in 2012. In late 2015, Dunn's port was forked to create the **eSpeak NG** project. eSpeak NG is a significant departure from the original eSpeak project, with the intention of cleaning up the existing codebase, adding new features, and adding to and improving the supported languages. The eSpeak NG project is hosted at https://github.com/espeak-ng/espeak-ng

Source: https://github.com/espeak-ng/espeak-ng#history. Retrieved 15/01/2021.

Synthesis Technology

eSpeak NG uses **Formant Synthesis** to create speech audio. This technique uses *additive synthesis* (wherein multiple soundwaves are combined) to form speech audio from synthesised sound waves produced using an *acoustic model* (a mathematical model that uses multiple equations and parameters to accurately produce various features of speech).

Formant synthesis generates highly artificial, robotic speech. However, it is reliably intelligible at a large range of speeds, and as long as the acoustic model is sufficiently complete, formant synthesis engines can output a wide variety of prosodies and intonations, meaning that emotions and various tones of voice can be easily produced. Furthermore, unlike concatenative engines formant synthesis engines do not require a large database of pre-recorded sounds, and therefore are viable to use in embedded systems where microprocessor power and storage are limited.

Package Availability/Supported Platforms

eSpeak NG is available as:

- A command line program (Linux and Windows 8+) to speak text from a file or from stdin.
- A shared library version for use by other programs. (On Windows this is a DLL).
- A SAPI5 version for Windows, so it can be used with screen-readers and other programs that support the Windows SAPI5 interface.
- eSpeak NG has been ported to other platforms, including Solaris and Mac OSX.

Source: https://github.com/espeak-ng/espeak-ng. Retrieved 15/01/2021.

Authors note: It has also been ported to Android, versions 4.0 (Icecream Sandwich) and upwards.

Implementing te reo Māori

As luck may have it, a barebones reo Māori implementation was already included in eSpeak NG. I'll detail here the steps needed to add and modify a language, and the adjustments I made to the included reo Māori implementation.

Adding a New Language

Five principle files must be edited/added to implement a new language:

- The **makefile** must be edited to reference the data files you'll be adding
- The **phsource/phonemes** file must be edited to reference the phoneme tables used in the language you are adding. The phoneme definitions control what formulas and parameters are used in the acoustic model.
- A language file must be added that contains basic information about the language, e.g. it's spoken speed, BCP-47 language tag and other attributes which affect the tone and quality of the output voice sound
- A language rules file must be added, which contains rules that govern how letters/spelling translates to phonemes
- A word list file must be added, which contains pronunciations for numbers, letters, symbols, and words with exceptional pronunciations. It also gives attributes such as "unstressed" and "pause" to some common words

Optionally a file which contains custom phoneme definitions can also be added, and linked within the master phonemes file.

Modifications to the reo Māori Implementation

I made rather minor changes to the existing implementation. Further changes would have required the input from someone with significantly more experience with reo Māori linguistics than myself. However, I:

- Reduced the speed to 80% of the eSpeak default

- Reduced the pitch slightly, as it made the language more intelligible
- Increased the strength of voiced consonants, as ‘n’s in particular were blending in with their surrounding phonemes
- Changed the phoneme sound used for ‘r’ to a voiced liquid trill, rather than a tap. This resulted in a much clearer pronunciation, despite a tapped ‘r being technically correct

In its current form, the speech output of the Māori language ranges from poorly to moderately intelligible. A particular issue is with some voiced consonants being absolutely indistinct.

Next Steps

To improve upon the reo Māori implementation, I believe an important step would be to properly define where stress should be placed on certain phonemes. This can be done through a combination of using advanced rules in the *mi_rules* file, and defining words with exceptional stresses in the *mi_list* file.

It is also possible that defining more accurate vowel sounds would improve the intelligibility of the speech output. Vowel sounds can be edited using the eSpeakEdit program. Note that while this program is currently being remastered by the eSpeak NG team, I have built a complete deployable package for the older version, similar to how I describe building a deployable package in the next section.

Creating a Deployable Package

To begin the process I traced the packages that *eSpeak-NG* was dependent on. I made sure to include even dependencies like the automake build system, to ensure that the user would only need a compiler installed to use our deployable package. The final dependency tree can be seen in Figure 1.

I only included two of eSpeak NG’s optional dependencies in the final package. The first was *pcaudiolib*, so that audio could be synthesised directly from the command line. *pcaudiolib* supports both *PulseAudio* and *ALSA* as a live audio backend; I chose to use PulseAudio as it has more features, such as supporting simultaneous audio output from multiple applications.

The other optional dependency I included was *sonic*, which provides a performant and quality algorithm for speeding up or slowing down speech. I decided not to include the dependencies that built the man page/documentation files, as they added unnecessary size to the package and increased build complexity. As a workaround, users can access the readily available online documentation, or the prebuilt pages could be included in the package.

```
### Dependency Tree

* [make-4.3](https://www.gnu.org/software/make/)
* [automake-1.16.2](https://www.gnu.org/software/automake/)
    * [autoconf-2.69](https://www.gnu.org/software/autoconf/autoconf.html)
        * [m4-1.4.18](https://www.gnu.org/software/m4/m4.html)
* [libtool-2.4.6](https://www.gnu.org/software/libtool/)
* [pkg-config-0.29.2](https://www.freedesktop.org/wiki/Software/pkg-config/)
* [pcaudiolib-1.1](https://github.com/espeak-ng/pcaudiolib/)
    * [pulseaudio-13.99.3](https://www.freedesktop.org/wiki/Software/PulseAudio/Download/)
        * [libsndfile-1.0.28](http://www.mega-nerd.com/libsndfile/)
        * [libatomic_ops-1.2](https://github.com/ivmai/libatomic_ops)
        * [speexdsp-1.2rc3](https://www.speex.org/downloads/)
        * [json-c-0.15-20200726](https://github.com/json-c/json-c)
            * [cmake-3.18.4](https://cmake.org/download/)
        * [gettext-0.21](https://www.gnu.org/software/gettext/)
* [sonic-67ed70f](https://github.com/waywardgeek/sonic)
```

Figure 1. eSpeak-NG dependency tree

To ease the process of compiling each package in the right order and to the designated installation prefix (prefix: the folders where the compiled app data is placed), along with managing the multitude of compilation flags and different build systems, I used a modified version of *cascade-make*, the build tool used in the University of Waikato's *Greenstone* software. Cascade-make also has the advantage that it is small, portable and only requires *bash* to run, therefore being easy to bundle with any package.

Cascade-make consists of a base library that contains functions to extract archives, run configuration scripts etc.. Each package then has a control script written for it, which sets up any required build flags and calls functions from the base library to successfully install the package. Finally, there is a master control file which determines the order in which the packages are built, and a one-time script that sets up environment variables (such as the prefix to install the package to) at the start of each development session.
My changes made to cascade-make consisted of making it a generic system, rather than customised for Greenstone, along with introducing new base functions to deal with the various compilation and installation methodologies that each package uses. Changes included:

- Implementing a function to run *autogen.sh* scripts
- Implementing functions to run *cmake* configure and compile steps
- Implementing functions to perform the *meson* and *ninja* configure/compile process
- Implementing functions to run a *Perl* project configuration
- Adding a usage printout
- Removing Greenstone-specific references and implementation details
- Creating a initialisation script to prepare a cascade-make build environment
- Adding a basic mechanism of checking whether cascade-make has already run a function like autogen, configure, or compile on a package to reduce unnecessary build time

Hence, our version of eSpeak NG can be distributed with all required dependencies, and easily installed on most modern UNIX systems with a C99/C++ 11 compatible compiler.

Conclusion

This investigation has proved that it is possible to utilise eSpeak NG as a tool for developing a reo Māori TTS system. For the system to reliably produce intelligible Māori speech, it will need considerable further development by someone with considerable experience in reo Māori linguistics. Note that this person does not necessarily need programming knowledge, although they will certainly need to have a level of familiarity with the command line, and know the steps required to compile the changes they make to the language.

I also achieved the goal of creating a portable, self-installing package that can be deployed on most modern UNIX systems without the need for elevated privileges.

10.3 Appendix 3 – Investigating MaryTTS as a Tool for Developing a Te Reo Māori Text-To-Speech System

Author: Ethan McKee-Harris | em134@students.waikato.ac.nz

Supervisors: Te Taka Keegan | tetaka@waikato.ac.nz

David Bainbridge | davidb@waikato.ac.nz

Introduction

The initial brief for this project set out a couple of main goals. The primary of which was to see whether MaryTTS could be used to build a Māori voice that was capable of synthesizing Te Reo Māori from text input with very high accuracy. Furthermore, the end system should be deployable on most modern Unix-based systems without the need for elevated permissions, think sudo.

Further refining what can be considered a 'deployable system', I consider it to be a system that has access to both JDK 8 and Apache Maven 3.6.3. I also assume that the tools required for the Māori voice, such as the built voice files, are provided and the end-user does not need to build these themselves beyond a maven install.

About MaryTTS

Basics

- Written in Java.
- 9 languages with 41 accent variations are officially published, with numerous third-party languages built.
- Released under the GNU Lesser General Public License v3.0
- Features a built-in front-facing API that exposes the text-to-speech backend.

History

The Text-To-Speech system was originally developed as a collaborative project of DFKI's Language Technology Lab and the Institute of Phonetics at Saarland University for research purposes as an in-house Text-To-Speech component, before becoming a fully open-source TTS platform. After a spate of research grants from various areas, such as the German Research Council (DFG), MaryTTS continued to grow and expand into a fully-fledged suite of tools that could be used to both create and use languages for Text-To-Speech purposes. Due to several departures, however, the software package is now maintained by the Multimodal Speech Processing Group in the Cluster of Excellence MMCI and DFKI using DFG grants.[1, 2]

Mary stands for "Modular Architecture for Research on speech sYnthesis."

Synthesis Technology

At the core of the MaryTTS system, stands Feature Extraction. Because MaryTTS does not only build one type of language model; rather through the usage of third-party tools such as

the HTS engine for a statistical parametric synthesis approach, or a set of Gradle plugins for your standard unit selection voice; a key part of the voice building process is transforming speech data into a feature representation. It is then this feature representation that allows the use of machine learning techniques to train models that can predict various linguistic items required for a language.[3]

Following on from feature representation, the end-user then has the choice to either create a Unit Selection based language model or a Statistical Parametric Synthesis language model. The key difference between these two choices can be summed up in a few sentences. When attempting to create a Unit Selection based voice, you will be given a voice that can sound very natural, however, will often suffer from audible glitches when you have to synthesize things that are 'out-of-domain'.[4] Another issue is the size of the language model, concerning our original constraints for deployability, having to have the entire set of audio training data bundled with your language. On the flip side, training a voice using a Statistical Parametric Synthesis approach using an HMM, or Hidden Markov Model-based form of language creation can lead to your voice being 'buzzy' and 'unnatural', however, these forms of voice models do offer far more flexibility as well as a much smaller memory footprint. Not everything is perfect however, building an HMM-based voice comes with a high technical overhead, and the HTS java port has fallen far behind HTS development.[5]

Implementing Te Reo Māori

There are several key steps required to build a new language, these are fully detailed at https://github.com/marytts/marytts-wiki in the New-Language-Support files as well as the related one for the type of voice model you choose to build.

Before you start, there are a few things you will need:

- A **lexicon** for the language you wish to build,
- A decent understanding of command-line usage, with an understanding of how to debug and fix the execution of a script when it fails,
- A small amount of linguistic knowledge, enough to translate a decent percentage of your language into IPA (The International Phonetic Alphabet),
- Access to either an internet archive of Wikipedia in your language or a corpus of data to train on.

The voice building process can be summed up as follows:

1. Build the MaryTTS 5.x system, this is their current stable release,
2. Build and compile Mysql version 5.7, see reference 6 for further details if you are having issues with regards to being able to set up this system on your machine,
3. Download and extract the Māori Wikipedia, see reference 7 for further details,
4. Use Mary's built-in tools to split up your Wikipedia dump into useable 'data chunks',
5. Check your data, most likely you will need to run our tool to clean up the data MaryTTS has 'cleaned' as it fails to properly clean the data, see reference 8,
6. Follow steps 3 & 4 for creating a new language, followed by the item in reference 9,

7. The built-in script for step 6 appears to fail to run, so you should attempt to use reference 10 as well,
8. Follow the rest of the steps and you should have all of the required files to build a language of your choice before adding it into MaryTTS for distribution.

After you successfully build the required data, it is as simple as following the next guide for building either an HMM voice or a Unit-selection voice. After which you can follow the 'Publishing a MaryTTS voice' guide and deploy your language. From my understanding, after you complete this, you should be able to get your language included in the 'master' version of MaryTTS, so anyone who wishes to use your languages doesn't need to go through all of the hurdles listed in this section. Rather, they can simply download and install the language from git and be good to go.

The **full** overview of the steps/process followed by a MaryTTS to take text input and convert it into sound can be seen here: http://mary.dfki.de/pdf/mary-architecture_v2.pdf

Next Steps

A good next step for us to improve upon the usage of Te Reo Māori with MaryTTS would be to finish our implementation. By doing this it would allow us full control over settings, usage, and deployment. Currently, we are in the process of doing this, however, at the time of writing, I am yet to get a māori implementation working. Due to this, I am basing my assumptions for the final product on the output of other research work conducted in the text-to-speech environment that was focused on Māori.

Conclusion

This investigation has shown potential for the usage of MaryTTS as a text-to-speech system for the Māori language, however, it has failed to provide a built māori voice model. This is due to multiple reasons, however, the primary of which is that the part of MaryTTS responsible for building a language has been unmaintained for numerous years and has failed to keep up with modern technology. Couple this with the third-party software **required** to build a new voice, and you end up with software that needs last decades platform to run reliably without needing to jump through numerous technical hurdles.

This investigation did however prove that MaryTTS can run in an environment without the need for elevated privileges. This means should further development work be done, and a voice model is successfully built. The end-user should be able to fairly easily acquire both MaryTTS and the Māori voice model such that they can use it for their application.

References

1. http://mary.dfki.de/index.html
2. http://mary.dfki.de/documentation/history.html
3. https://arxiv.org/pdf/1712.04787.pdf, Section 4.2
4. https://arxiv.org/pdf/1712.04787.pdf, Section 4.3.1
5. https://arxiv.org/pdf/1712.04787.pdf, Section 4.3.2
6. https://github.com/ateaspace/data-capsule-tts/tree/mary-process/marytts-complete#working-with-mysql-before-step-22
7. https://github.com/ateaspace/data-capsule-tts/tree/mary-process/marytts-complete#Māori-wikipedia-step-1
8. https://github.com/ateaspace/data-capsule-tts/blob/mary-process/marytts-complete/README.md#cleaning-the-database-after-step-22https://github.com/ateaspace/data-capsule-tts/blob/mary-process/marytts-complete/README.md#running-the-featuremaker-before-step-5
9. https://github.com/ateaspace/data-capsule-tts/blob/mary-process/marytts-complete/README.md#database-selection-after-step-6

Further references, not directly referenced in this report that were used throughout compiling this report:

- https://github.com/marytts/marytts
- https://github.com/marytts/marytts-wiki
- https://www.researchgate.net/publication/221481103_Open_source_voice_creation_toolkit_for_the_MARY_TTS_Platform
- A white-paper was released to Waikato University about how they used MaryTTS as a Text-To-Speech system for Māori.

This list is by no means extensive, and other papers, stack overflow pages, and support were used liberally throughout this investigation, however, are not catalogued here.

11 EndNotes

[i] Schmidt, C.A. (2020). Speech Technologies. Retrieved from https://www.iais.fraunhofer.de/en/business-areas/speech-technologies.html
[ii] Doulaty, M. (2015). A Brief Introduction to Speech Technology. Retrieved from https://www.prescouter.com/2015/05/a-brief-introduction-to-speech-technology/
[iii] Rouse, M. (2020). Speech Technology. Retrieved from https://searchunifiedcommunications.techtarget.com/definition/speech-technology
[iv] Mustapha, O., Ibiyemi, T.S., & Osagie, S. (2016). Retrieved from https://www.researchgate.net/publication/312038601_Text-to-Speech_Synthesis_Using_Concatenative_Approach
[v] SAS. (2020). Machine Learning: What it is and why it matters. Retrieved from https://www.sas.com/en_nz/insights/analytics/machine-learning.html
[vi] Simonite, T. (2019). The Godfathers of the AI Boom Win Computing's Highest Honor. Retrieved from https://www.wired.com/story/godfathers-ai-boom-win-computings-highest-honor/
[vii] Sciforce. (2019). NLP for low resource settings. Retrieved from https://medium.com/sciforce/nlp-for-low-resource-settings-52e199779a79
[viii] Wibawa, J.A.E., Sarin, S., Li, C., Pipatsrisawat, K., Sodimana, K., Kjartansson, O., Gutkin, A., Jansche, M., & Ha, L. (2019). Building Open Javanese and Sundanese Corpora for Multilingual Text-to-Speech. Retrieved from https://www.aclweb.org/anthology/L18-1255.pdf
[ix] Godayal, D. (2018). An introduction to part-of-speech tagging and the Hidden Markov Model. Retrieved from https://www.freecodecamp.org/news/an-introduction-to-part-of-speech-tagging-and-the-hidden-markov-model-953d45338f24/
[x] Gajavalli, S.H. (2020). All the Deep Learning Breakthroughs in NLP. Retrieved from https://analyticsindiamag.com/all-the-deep-learning-breakthroughs-in-nlp/
[xi] Van den Ord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. Retrieved from https://arxiv.org/pdf/1609.03499.pdf
[xii] The Understood Team. (2020). Text-to-Speech Technology: What it is and how it works. Retrieved from https://www.understood.org/en/school-learning/assistive-technology/assistive-technologies-basics/text-to-speech-technology-what-it-is-and-how-it-works
[xiii] Sciforce. (2020). Text-to-Speech Synthesis: an Overview. Retrieved from https://medium.com/sciforce/text-to-speech-synthesis-an-overview-641c18fcd35f
[xiv] Mwiti, D. (2019). A 2019 Guide to Speech Synthesis with Deep Learning. Retrieved from https://heartbeat.fritz.ai/a-2019-guide-to-speech-synthesis-with-deep-learning-630afcafb9dd
[xv] Chen, J., & Boyle, M. (2020). Neural Networks. Retrieved from https://www.investopedia.com/terms/n/neuralnetwork.asp#:~:text=Neural%20networks%20are%20a%20series,fraud%20detection%20and%20risk%20assessment.
[xvi] Hargrave, M. (2020). Deep Learning. Retrieved from https://www.investopedia.com/terms/d/deep-learning.asp
[xvii] Huang, X. (2018). Microsoft's new neural text-to-speech service helps machines speak like people. Retrieved from https://azure.microsoft.com/en-us/blog/microsoft-s-new-neural-text-to-speech-service-helps-machines-speak-like-people/
[xviii] Morais, R. (2019). DeepSpeech 0.6: Mozilla's Speech-to-Text Engine Gets Fast, Lean, and Ubiquitous. Retrieved from https://hacks.mozilla.org/2019/12/deepspeech-0-6-mozillas-speech-to-text-engine/
[xix] Mustapha, O., Ibiyemi, T.S., & Osagie, S. (2016). Retrieved from https://www.researchgate.net/publication/312038601_Text-to-Speech_Synthesis_Using_Concatenative_Approach
[xx] Baevski, A., Conneau, A., & Auli, M. (2020). Wav2vec 2.0: Learning the structure of speech from raw audio. Retrieved from https://ai.facebook.com/blog/wav2vec-20-learning-the-structure-of-speech-from-raw-audio
[xxi] Choudhury, A. (2020). Facebook Is Giving Away This Speech Recognition Model For Free. Retrieved from https://analyticsindiamag.com/facebook-is-giving-away-this-speech-recognition-model-for-free/
[xxii] Watson, C.I., Maclagan, M., King, J., Harlow, R., & Keegan P.J. (2016). Sound change in Māori and the Influence of New Zealand English. *Journal of the International Phonetics Association. 46*(02) 185-218. DOI: http://dx.doi.org/10.1017/S0025100316000013
[xxiii] Watson, C. (2009). MAONZE. Retrieved from http://homepages.engineering.auckland.ac.nz/~cwat057/MAONZE/purpose.html