

## Highlights

### **Speech Generation for Indigenous Language Education**

Aidan Pine, Erica Cooper, David Guzmán, Eric Joanis, Anna Kazantseva, Ross Krekoski, Roland Kuhn, Samuel Larkin, Patrick Littell, Delaney Lothian, Akwiratékha' Martin, Korin Richmond, Marc Tessier, Cassia Valentini-Botinhao, Dan Wells, Junichi Yamagishi

- We detail the variety of challenges facing low-resource speech synthesis and provide points of reference and possible solutions for future projects
- We describe the release of the EveryVoice TTS Toolkit designed specifically for low-resource TTS
- To motivate the release of the EveryVoice TTS Toolkit, we compare the implementations of 10 selected requirements for low-resource TTS between EveryVoice TTS and six other existing toolkits

# Speech Generation for Indigenous Language Education

Aidan Pine<sup>a</sup>, Erica Cooper<sup>b</sup>, David Guzmán<sup>a</sup>, Eric Joanis<sup>a</sup>, Anna Kazantseva<sup>a</sup>, Ross Krekoski<sup>d</sup>, Roland Kuhn<sup>a</sup>, Samuel Larkin<sup>a</sup>, Patrick Littell<sup>a</sup>, Delaney Lothian<sup>a</sup>, Akwiratékha’ Martin<sup>a</sup>, Korin Richmond<sup>c</sup>, Marc Tessier<sup>a</sup>, Cassia Valentini-Botinhao<sup>c</sup>, Dan Wells<sup>c</sup>, Junichi Yamagishi<sup>b</sup>

<sup>a</sup>*National Research Council, 1200 Montréal Rd., Ottawa, K1A 0R6, Ontario, Canada*

<sup>b</sup>*National Institute of Informatics, 2 Chome-1-2 Hitotsubashi, Tokyo, 101-0003, Chiyoda City, Japan*

<sup>c</sup>*University of Edinburgh, 10 Crichton Street, Edinburgh, EH8 9AB, Scotland, UK*

<sup>d</sup>*University nuhelot’jine thaiyots’j nistameyimâkanak Blue Quills, 3 Airport Rd N, St. Paul, T0A 3A0, Alberta, Canada*

---

## Abstract

The vast majority of the world’s languages are unable to follow in the footsteps of existing resource-intensive pathways to building text-to-speech (TTS) systems. But, as the quality of contemporary speech synthesis grows, so too does the interest from many of these underserved language communities in adopting TTS for a variety of real-world applications. The goal of this paper is to provide signposts and points of reference for future low-resource speech synthesis efforts, with insights drawn from the Speech Generation for Indigenous Language Education (SGILE) project. Funded and coordinated by the National Research Council of Canada (NRC), this multi-year, multi-partner project has the goal of producing high-quality text-to-speech systems that support the teaching of Indigenous languages in a variety of educational contexts. We provide background information and motivation for the project, as well as details about our approach and project structure, including results from a multi-day requirements-gathering session. We discuss some of our key challenges, including building models with appropriate controls for educators, improving model data efficiency, and strategies for low-resource transfer learning and evaluation. Finally, we provide a detailed survey of existing speech synthesis software and introduce EveryVoice TTS, a toolkit designed specifically for low-resource speech synthesis.

*Keywords:* speech synthesis, text-to-speech, low-resource languages, Indigenous languages, language education, language revitalization

---

## 1. Introduction

There are approximately 70 Indigenous languages spoken in Canada from 10 distinct language families. As a consequence of the residential school system and other policies of cultural suppression, the majority of these languages now have fewer than 500 fluent speakers remaining, most of them elderly. Despite this, Indigenous people have resisted colonial policies and continued speaking their languages, with interest from students and parents in Indigenous language education continuing to grow. Teachers are often overwhelmed by the number of students, and the trend towards online education means many students who have not previously had access to language classes now do. Supporting these growing cohorts of students comes with unique challenges in languages with few fluent first-language speakers, and teachers are particularly concerned with providing their students with opportunities to hear the language outside of class. While there is no replacement for a speaker of an Indigenous language, there are possible applications for speech synthesis (text-to-speech) to supplement existing text-based tools like verb conjugators, dictionaries, and phrasebooks. To this end, the National Research Council of Canada (NRC) has partnered with the Onkwawenna Kentyohkwa Kanyen'kéha immersion school, WSÁNEĆ School Board, University nuhelot'ine thaiyots'i nistameyimâkanak Blue Quills, the National Institute of Informatics in Japan (NII), and the University of Edinburgh (UoE) to research and develop state-of-the-art text-to-speech (TTS) systems and techniques for Indigenous languages in Canada.

### 1.1. Research Significance

There are two main intended impacts for this paper. First, we provide details and reflections on the practical, methodological, and technical challenges related to conducting speech synthesis research for extremely low-resource languages in educational contexts. Second, we present a Python library designed to address some of these challenges, named the EveryVoice TTS Toolkit.

We introduce EveryVoice TTS by comparing it to existing neural speech synthesis toolkits with a focus on how well each toolkit supports our use case

(§5). Specifically, the low-resource language communities we are concerned with may have very few fluent speakers, often elderly people with many other pressing responsibilities and limited time to spend recording audio. Data efficiency – techniques for creating TTS systems that generate high-quality speech even if they’ve only been trained on a small amount of speech – is a very minor consideration when one is dealing with high-resource languages. In our context, however, it is crucial. We have designed EveryVoice TTS to be easily adapted to new languages and new datasets, with specific considerations for making the development process more user-friendly for technical users without specific expertise in TTS research.

While some of the discussion in this paper is specific to certain educational contexts, the goal of applying TTS models to augment text-based educational tools is a general one and aspects of our approach will be relevant to a wider audience. Specifically, we believe many of the challenges and proposed solutions discussed in this paper will be relevant to other ‘low-resource language’ contexts since many of the technical issues faced when developing speech synthesis systems for Indigenous languages can be coarsely generalized as problems related to the ‘*low-resource*’ nature of the languages: limited speakers means limited data, limited eligible participants for evaluation, and limited prior work.

### *1.2. A Roadmap for Readers*

Since this paper focuses on making TTS more accessible to deploy in new domains, it speaks to multiple audiences: from one direction, TTS experts who may be unfamiliar with the challenges of developing Indigenous language technologies, and from other directions, developers or managers working for Indigenous language organizations who may be unfamiliar with the challenges and potential perils of TTS.

In consequence, the paper is quite long, and we provide the following roadmap to help guide readers with different perspectives and priorities.

**Speech Synthesis Researchers.** For speech synthesis researchers, we recommend starting with §2 for a description of the motivation that community organizations have in developing TTS systems as well as an example of the linguistic variation that exists among the Indigenous languages in our project. Proceed to §2.3 for a brief description of prior work in TTS for Indigenous languages in Canada and a cautionary note against carrying out research in Indigenous language TTS without engaging with the language community.

Finally, §4 details four of the main challenges to low-resource TTS in an educational context. We hope that the description of these challenges encourages future research in these areas.

**Technologists and Developers.** Developers and technologists interested in applying TTS to a new language or use case need to be aware of possible ethical issues. Thus, we recommend starting with §2.3 and the ‘*Imagine somebody misusing TTS technology*’ paragraph in §Appendix A, which give a cautionary overview to the ways that TTS technology could be misused. Then, if you are involved in gathering/recording data, we recommend you proceed to read §3.2 and §3.3; otherwise, you can skip them. Finally, we recommend reading §5 for a description of EveryVoice TTS as well as Appendix B for a comparison with other toolkits.

**Program Managers.** For program managers, administrators, or other non-technical readers interested in starting or designing a project related to speech synthesis, we recommend reading the introduction to §2 for background context and information about ethical engagement. We then recommend reading §3 and §3.1 which discuss our methods and approach to project organization and defining project goals. The rest of §3 may be of interest if you are involved in data collection aspects of the project, and §3.3.3 may be of particular interest when estimating a budget for recordings. Finally, §6 itemizes key take-aways from the paper.

## 2. Motivation, Ethical Engagement, and Context

The language *revitalization* efforts taking place today within Indigenous language communities in Canada are in response to over a century of colonial language policy aimed at *devitalizing* Indigenous languages. Accordingly, the decision to speak an Indigenous language in Canada is often seen as a political act; one that asserts broader goals of self-determination and community building (Brinklow et al., 2019; Pine & Turin, 2017). However, instead of affirming these goals, many academic and industry-led projects undermine them by compromising community data sovereignty and by creating inequitable collaborations primarily focused on alienating Indigenous communities from their data (Junker, 2024). Put another way, while previous colonial efforts in Canada oppressed Indigenous language communities

as a means of alienating them from their land and natural resources, contemporary colonial efforts often disingenuously support Indigenous languages as a means of obtaining their data. While the solution to many so-called ‘low-resource’ problems in natural language processing (NLP) are approached solely by obtaining more data, there is a growing chorus of voices within the NLP community that identify more structural and systemic problems with current approaches to collaborations between NLP researchers and Indigenous communities (Bird, 2020, 2022; Brinklow et al., 2019; Schwartz, 2022). Researchers must make ethical engagement with communities their top priority. Thus, we believe it is important to describe who the partners in this collaboration are, how our collaboration came to be, and how it has evolved.

### *2.1. Project Motivation and Indigenous Partners*

The original motivation for pursuing TTS research came from open-response feedback held during a user evaluation study of the Kawennón:nis verb conjugator educational software project in 2018 (Kazantseva et al., 2018). In these user evaluation sessions, held in person at the Onkwawenna Kentyohkwa Kanyen’kéha immersion school, participants repeatedly expressed a desire to be able to hear the conjugations produced by the tool. However, the domain of the generative text model underlying Kawennón:nis (at that time 120 000+ unique forms) was simply too large to be feasibly recorded. This question of finding the most efficient means to supplement a text-based tool like Kawennón:nis with audio is what catalyzed this research effort.

In 2021, members from the National Research Council of Canada co-designed a project called Speech Generation for Indigenous Language Education (SGILE) in an application to the National Research Council of Canada’s Small Teams funding initiative, which supports NRC researchers in partnership with external organizations. The project proposal was co-designed with collaborators from organizations with significant expertise in either speech synthesis or Indigenous language education: the University of Edinburgh and National Institute of Informatics in Japan (NII, unfunded partner), the previously mentioned Onkwawenna Kentyohkwa Kanyen’kéha immersion school (§2.1.1), the WSÁNEĆ School Board (§2.1.2), and University nuhelot’jne thaiyots’j nistameyimâkanak Blue Quills (§2.1.3). The project obtained funding in 2022 and will run until 2025. The three Indigenous partners represent three unrelated language families and communities, with different phonological properties and educational programs. We include Fig. 1, which visualizes

the overlap of phoneme<sup>1</sup> inventories for each language, to illustrate the extent of the phonological differences between these languages and to dispel the common assumption that most Indigenous languages in Canada are similar to one another.

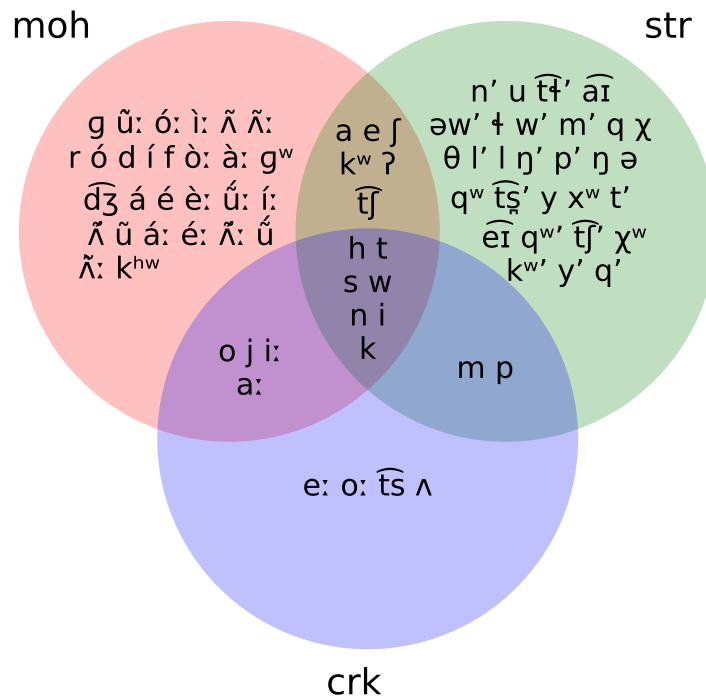


Figure 1: Phone modelling inventory overlap between Kanyen'kéha (moh, §2.1.1), SENĆOŦEN (str, §2.1.2), and nēhiyawēwin (crk, §2.1.3)

### 2.1.1. Onkwawenna Kentyohkwa Kanyen'kéha Immersion School

Onkwawenna Kentyohkwa (Our Language Society) is a community-based organization that teaches Kanyen'kéha (the ‘Mohawk’ language) to adults on the Six Nations Grand River Territory near Brantford, Ontario. It is one of the longest-running adult immersion program for any Indigenous language in

<sup>1</sup>Phonemes are the minimal phonological units capable of producing contrasts between words.

the country and bases its curriculum around the ‘root word method’ (Green & Maracle, 2018).

Kanyen’kéha is an Iroquoian language spoken in Quebec, Ontario, and New York State. The language is highly polysynthetic, with long words often translating to entire sentences as seen below from Kazantseva et al. (2018).

- (1) *tetsyonkyathahahkwahnónhne*  
 te-ts-yonky-at-hahahkw-hnón-hne  
 DUAL-REP-3SG.N/1.DU.INCL-SREFL-walk-PURP-PPFV  
 ‘the two of us went for a walk again’

The phoneme inventory for Kanyen’kéha (moh) has 12 consonants and 24 vowels, including 6 lengthened vowels, four nasalized vowels, and the high- and low-tone variants of each vowel.

### 2.1.2. *WSÁNEĆ School Board*

The WSÁNEĆ School Board operates a variety of accredited immersion programs for the SENĆOFEN language, spoken by the WSÁNEĆ people on the island colonially known as Vancouver Island off the coast of British Columbia. It provides world-renowned programming to preschool, kindergarten, primary school, and adult education for learners. The community has a long history of language activism, with the language’s orthography being developed by the late Dave Elliott in 1978 (WSÁNEĆ School Board, 2023).

The SENĆOFEN language has a large inventory of consonant phonemes, with 36 consonants compared to 7 vowels (including two diphthongs). Words often contain complex consonant clusters as in the examples below from Montler (2018):

- |  |  |
|--|--|
| <p>(2) <i>XDQETEN TÁ,</i><br/>       χt’k<sup>w</sup>’átəŋ t̪’éʔ<br/>       ‘It was carved again.’</p> | <p>(3) <i>STLTPÁLKEN.</i><br/>       st̪’lt̪’pelqən<br/>       ‘little feathers’</p> |
|--|--|

### 2.1.3. *University nuhelot’ine thaiyots’i nistameyimâkanak Blue Quills (UnBQ)*

UnBQ was the first university in Canada to be First Nations owned and operated. The university is jointly owned by seven First Nation band governments. Its mission is to “address the spiritual, emotional, physical and



mental needs of the seven member First Nations through the delivery of quality education programs” (UnBQ, 2023).

The university is located in Northern Alberta in Treaty 6 territory. It is located in the same building as a former residential school where, in 1970, parents and some Indigenous staff members of the school protested and hosted a sit-in at the school to oppose plans to have the school amalgamated into the public school system. Following weeks of protest, the school was transferred to the Native Education Council and is now an accredited Indigenous-run university, specializing in programs in Social Work, Early Childhood Development, Community Environmental Technology, nêhiyawêwin and Denesųłine languages, and other post-secondary programming with an Indigenous focus.

The nêhiyawêwin language (also known as Plains or Y-dialect Cree) is an Algonquian language spoken in Saskatchewan, Alberta, Manitoba, and Montana. Its phoneme inventory is relatively small, comprising 10 consonants and 7 vowels. The language has both an alphabetic and syllabary-based standard orthography.

## *2.2. Funding that supports equitable cross-institutional collaboration*

Our project is intentionally titled Speech Generation for Indigenous Language *Education*. The goal is not to develop speech synthesis systems independently of their use case; rather, it is to develop speech synthesis systems which braid into existing community goals and workflows. Given the sordid history of research within Indigenous communities (Medin & Bang, 2014; Mosby, 2013; Smith, 2023) as well as the contemporary prevalence of data extraction and exploitation in Indigenous communities (Junker, 2024), there is little appetite for research projects which do not align with community goals (Kuhn et al., 2020; Le Ferrand et al., 2022).

Part of the way our project ensures alignment with community goals is structural. Often, research grants are only available to academic organizations, which in turn provide funds to Indigenous community organizations and peoples as ‘participants’. There is a large body of work across academic disciplines that discusses ethical research collaborations (Bird, 2022; Brinklow, 2021; Czaykowska-Higgins, 2009; Hermes & Engman, 2017; Schwartz, 2022; Smith, 2023) and challenges research that is structured to valorise the expertise of academic researchers while excluding, ignoring, and devaluing the expertise of Indigenous peoples.

The focus of the NRC Ideation Fund’s ‘Small Teams Initiative’, which funded our project, is to connect NRC teams with external collaborators

who possess complementary capabilities and expertise. Crucially for our project, the fund does not restrict qualifying organizations to universities or research organizations. This allowed our project to be co-developed with Indigenous organizations with expertise in language education from the outset and resulted in each of the partners being funded independently by the NRC’s National Program Office. Each collaborator defined their own set of goals in the application and is in control of their own budget, finances, hiring, deliverables, and work practices for the duration of the project.

While the funding structure described here is not a comprehensive solution to the enduring impacts of colonization that continue to affect research collaborations between settler and Indigenous peoples, it is an important step in creating an equitable environment for research collaborations.

### *2.3. Ethical Issues of TTS Research without Community Engagement*

There is little prior work on speech synthesis for the languages described, and indeed for many Indigenous languages. Statistical parametric speech synthesizers exist for nêhiyawêwin (Harrigan et al., 2019) and Kanyen’kéha (Saunders, 2008). Preliminary neural systems have also been developed for Kanyen’kéha and SENĆOŦEN (Pine, Wells, et al., 2022). Neural systems have also been created for languages in the same language families as nêhiyawêwin (Algonquian), and Kanyen’kéha (Iroquoian) (Conrad, 2020; Hammerly et al., 2023; Pratap et al., 2024).

Most of the existing efforts have been done on a small scale by projects that, appropriately, engage with the language communities in question (Hammerly et al., 2023; Pine, Wells, et al., 2022). By contrast, Pratap et al. (2024) describes work on Meta’s ‘Massively Multilingual Speech’ project, which provides TTS models for over 1 100 languages. In that work, data is collected from Bible translation resources, but the authors do not state whether they had permission from the publishers to use the data for this purpose. It is also unstated whether the authors were able to obtain permission from each of the speakers to train systems to model their likeness. In subsequent personal communication with the authors<sup>2</sup>, it was reported that they indeed did not consult with the communities or speakers in question due to the scale of the project, and are under the belief that because the data was obtained from public sources, it can be used for non-commercial purposes (despite the

---

<sup>2</sup>Personal communication to Roland Kuhn, August 4 2023.

terms which appear to disallow it (Faith Comes By Hearing, 2021)). With the collected data, the authors train separate VITS (Kim et al., 2021) models for each language, and release the checkpoints after 100 000 steps. The released models were evaluated using a combination of automatic metrics (Mel-Cepstral Distortion and Word Error Rates from ASR systems) and listening tests, although the listening tests did ‘not require raters to be able to speak the respective language’ (Pratap et al., 2024, p. 31).

Releasing weakly evaluated models trained on data obtained without the explicit permission of the publisher or speakers poses a variety of potential harms that were not discussed by Pratap et al. (2024), such as the generation of offensive content, embarrassment due to pronunciation errors, and unauthorized use of someone’s likeness possibly after that person has passed away. Beyond these potential harms, research conducted with unclear data permissions has negative effects on reproducibility. As an example in the parallel case of research on face recognition, many of the datasets originally assembled have become inaccessible due to legal and ethical concerns, as well as a lack of individual consent from data subjects (Boutros et al., 2023).

The ethical issues related to possible misuse of TTS technology elicited animated discussion during the brainstorming part of the kickoff meeting for the project (§Appendix A). We believe that future efforts for TTS for Indigenous languages should engage meaningfully with the language communities in question, and follow established guidelines surrounding ownership of data (Schnarch, 2004). Approaches to mitigating risks related to data misuse and maintaining data sovereignty are also discussed in Appendix C.1.2.

#### *2.4. Repeatability*

With over 70 Indigenous languages spoken in Canada, our hope is that text-to-speech technology could become available to any community that wants it. Throughout the project we have tried to create a repeatable recipe for others to follow. This is partially accounted for through the diverse group of collaborators, which helps ensure our project outcomes do not ‘overfit’ to any one particular language or educational context, while providing room for adaptation to community-specific goals. Not only are the three languages involved in this project from three separate language families and highly dissimilar phonologically and orthographically, the educational contexts where these languages are taught are also quite different, ranging from early childhood education, to university education, and adult immersion. We hope that this diversity leads to more reproducible and widely applicable outcomes.

However, we caution the reader in assuming that just because a language they are working with can be described as ‘low-resource’, that our findings will be directly relevant. Categorizing languages based on whether or not data-driven probabilistic algorithms can be applied to them is a coarse method that, while convenient and frequently seen in literature, does not account for other important factors such as community population, literacy rates, community language goals, and domains of language use. The reductive nature of the term also promotes a simplistic view of both the problem and solution for the languages it describes; a view which takes for granted the benefits of technology and ‘problematizes complex socio-political situations purely in terms of missing data’ (Bird, 2022, pg. 7818). Despite its technical convenience in describing a set of languages for which certain algorithms could be applied, using the term in a decontextualized way risks disenfranchising language communities (Bird, 2020, 2022; Brinklow, 2021). Accordingly, we caution the reader not to assume that our motivations, methods, or findings extend to other language communities directly. We therefore encourage the reader to follow our repeatable recipe with a grain of salt, and to expect to make necessary adaptations to the recipe to suit their context.

### 3. Methodology & Project Structure

To ensure continued alignment between partners throughout the project, we have created a project structure which enables them to continue to co-develop and maintain a shared, cohesive vision of the desired outcomes.

The day-to-day operations of the project are split into three main streams of work (Figure 2). The first is the ‘text stream’ (§3.2). The activities of this stream vary between collaborators, from verifying existing text resources such as dictionaries or descriptive grammars (Montler, 2018), to developing generative text tools such as verb conjugators that will serve as the domains for synthesis. The second, the ‘recording stream’ (§3.3), is responsible for operating recording equipment, performing the recordings, and ensuring quality in the recording process. Lastly, the ‘modelling stream’ is tasked with building the speech synthesis models to address the challenges in §4. From time to time, the TTS system for a particular language will be evaluated by the relevant collaborators, often motivating improvements to the system. This section describes the text stream and the recording stream. The modelling stream and evaluation pose unique challenges, discussed in detail in §4. In

practice, given the relatively small size of the team, individual members often belong to multiple streams, but we have found the division helpful for determining goals and meeting structures.

### 3.1. Defining objectives and a shared vision

To formally begin the project, we held a kick-off meeting on 22–25 August 2022 on the traditional territory of the WSÁNEĆ people (§2.1.2). Each of

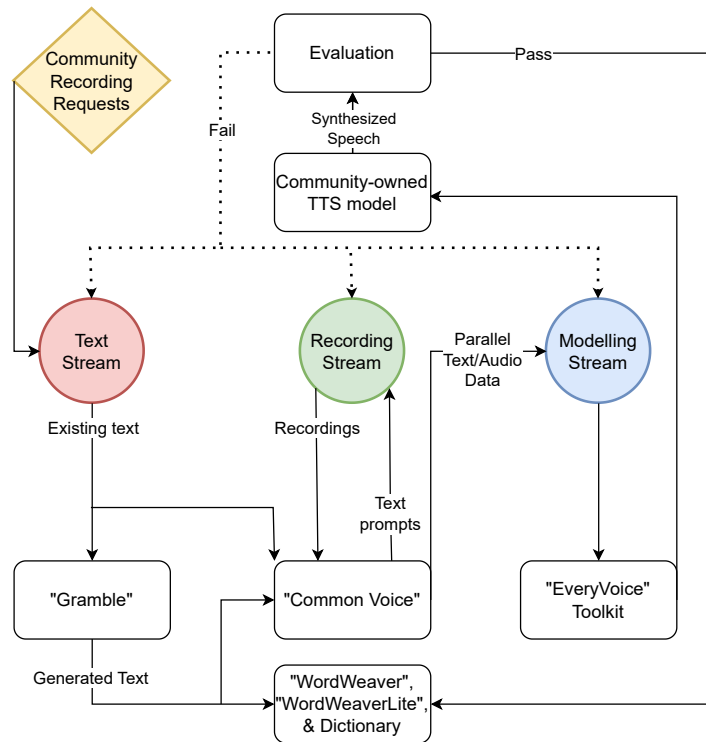


Figure 2: Flow chart depicting work flow of the SGILE project across the three streams: text (§3.2), recording (§3.3), and modelling (§4). Existing text is used as prompts for recordings and also as reference material for creating grammar models (via programming language ‘Gramble’, §3.2.2). The text stream is responsible for uploading prompts (either generated from ‘Gramble’ or cleaned from existing text) to CommonVoice (§3.3.2). Text that is uploaded to CommonVoice serves as prompts for the recording stream. Recordings created in CommonVoice by the recording stream are retrieved by the modelling stream to create a TTS model (via the ‘EveryVoice’ toolkit, §5.2 & B.7). The model is then evaluated and, following positive assessment, is incorporated into the target educational application (§3.2.1). If the model does not pass evaluation, the errors are analyzed and another iteration of text processing, recording, and modelling begins.

the organizations participating in the project had one or more representatives who came from around the world to meet in person and build relationships and skills. In addition to several workshops on recording techniques and grammar modelling, we held a brainstorming session over two days to elicit design ideas and discuss key themes, concerns, and expectations.

To facilitate the brainstorming session, collaborators were grouped with others from outside their institutions to discuss and answer each of three questions. All collaborators in attendance hold teaching or administrative positions at their organizations. These discussions were facilitated by NRC staff. We intend to collect similar information from students of the stakeholder institutions and beginner learners of the stakeholder languages further on in the project, through participant-based studies.

Collaborators who were focused primarily on technical aspects of speech synthesis modelling were instructed not to comment on whether a particular idea was technically feasible or not. This unrestricted brainstorming allowed the conversation to focus on goals and concerns, shifting feasibility to a later stage of discussion.

The brainstorming sessions were conducted with the following prompts: Imagine a tool that can speak your language:

- What does it sound like?
- Where are the people using the tool? What are they doing with it?
- Imagine somebody misusing the technology, what are some things that could go wrong?

After the questions had been discussed, the recorded ideas were analyzed by NRC staff later in the day to find underlying themes. These results were then reviewed by the collaborators in attendance to ensure accurate interpretations of the discussions. Summaries of these findings are found in Appendix A.

Later in the kick-off meeting, the basic structure of the neural speech synthesis model was sketched out during meetings between attendees representing NRC, UoE, and NII. Certain technical challenges were identified, as discussed in detail in §4.

### *3.2. Text Data Collection & Generation*

One of the main challenges for text-to-speech systems is supporting general-purpose synthesis, i.e., creating models where the eventual domain of synthe-

sis is not known in advance. However, due to the possible opportunities for misuse envisioned during the kick-off meeting, there was not a strong desire from collaborators to build a general-purpose interface capable of synthesizing arbitrary text. Instead, the primary application for our speech synthesis models will be to supplement text-based tools such as dictionaries and verb conjugators with audio.

Defining the text domain in advance is helpful because it enables us to better mitigate issues of domain mismatch between the data we create and the speech we intend to synthesize. For example, many Indigenous-language text corpora consist mostly of narrative monologues (histories, legends, etc.) and biblical translations, which have a bias towards particular persons, moods, tenses, and so on (e.g. third person indicative past). By specifying the domain for synthesis in advance, we can mitigate some of these challenges, focus our recording efforts, and create task-specific test sets.

### *3.2.1. Existing Text Resources*

As mentioned in the introduction, the initial idea for augmenting text-based educational applications with TTS came from user studies of an online verb conjugator called Kawennón:nis (lit. *‘the Wordmaker’* in Kanyen’kéha) (Kazantseva et al., 2018). Kawennón:nis is a collaborative effort between the NRC and educators from two separate Kanyen’kéha communities: Ohswéken and Kahnawà:ke. The original idea for Kawennón:nis came from Owenatekha Brian Maracle, a collaborator on our current project and an experienced teacher at Onkwawenna Kentyohkwa in Ohswéken in 2016 who wanted to create software for modelling verbal paradigms in Kanyen’kéha, which are very complex and are crucial to becoming proficient in the language.

The interface for Kawennón:nis was built using open-source software called WordWeaver which is a language-agnostic application for visualizing and interacting with inflectional verb paradigms. The underlying language models for the Ohswéken and Kahnawà:ke versions contain approximately 120 000 and two million conjugations respectively.

While the SENĆOFEN community does not have a version of WordWeaver for their language, they do have existing text resources that could be supplemented with audio. The SENĆOFEN dictionary (Montler, 2018) and grammar are rich resources replete with tens of thousands of words and example sentences. Instead of recording each of these words and sentences individually, the goal of ‘breathing life into the dictionary and grammar’ through supplemental synthesized audio was immediately proposed by the

WSÁNEĆ collaborators.

### 3.2.2. *Text-based grammar modelling*

The WSÁNEĆ School Board and University nuhelot’ine thaiyots’j nistameyimâkanak Blue Quills do not have an existing generative text tool like Kawennón:nis, but are interested in developing such tools. To this end, we are building new grammar models of important categories and the derivations/inflections that teachers identify as educationally important. This ensures that our TTS systems will have relevant text to generate speech from. It also allows us to generate text outside the training data for evaluation.

To accomplish this goal, and to improve collaboration between language experts and developers, we have been building an intuitive, layperson-readable programming system, named Gramble (Littell et al., 2024). Gramble sits roughly in the same niche as finite-state based generation tools, such as Xerox Finite-State Tool (XFST) (Beesley & Karttunen, 2003).<sup>3</sup> It uses a spreadsheet-like interface (Fig. 3) that, when used in conjunction with cloud-based spreadsheet software like Google Sheets, enables live, online pair-programming, so that linguist-programmers and teachers can more directly work together in modelling their languages. Working in this way has greatly reduced our turnaround time in building new paradigm generators.

Grammar modelling is done in close collaboration with teachers from our partnered organizations, with aid from existing text-based language resources (e.g., dictionaries, grammars) created or approved by them. (Y. Lu et al., 2024) describes a collaboration of this kind involving Gramble and an Indigenous language community not discussed in the current paper: speakers of Oneida (an Iroquoian language closely related to Kanyen’kéha).

### 3.2.3. *Grapheme-to-Phoneme (G2P) Conversion*

A challenge for many TTS systems is being able to derive a pronunciation form from the language’s orthography. Many of the world’s writing systems contain insufficient information for deriving a corresponding pronunciation form, either due to their nature (e.g. logographic systems), or because pronunciations have diverged significantly from their spellings over

---

<sup>3</sup>Despite recent advances in neural language models, they are not able to reliably generate forms they have never seen. In practice, very-low-resource natural language generation is likely to remain handwritten and rule-based. We want to ensure that students are learning actual forms and not forms guessed from inadequate training data.



subject/concept/gloss	gloss	subject_translation/concept_translation	subject_base
1SG	-	I	niya
2SG	-	You	kiya
3SG	-	S/he	wiya
1PL	-	Plural I	niyanān
2I	-	Inclusive You & I (plural/singular)	kiyanaw
2PL	-	Plural You	kiyanāwāw
3PL	-	Plural She/He	wiyawāw
3'	-	4th Person	John okosisa
O	-	It (inanimate)	ōma
OP	-	They (inanimate)	anihi
O'	-	His/her object (inanimate)	John omaskisina
O'P	-	His/her objects (inanimate)	John omaskisina

Figure 3: A screenshot of pronoun definitions for nêhiyawêwin in the ‘Gramble’ tabular programming language. By design, source files are meant to look familiar to people with experience using spreadsheets, so that non-programmers can understand and maintain them. The language is not *just* a spreadsheet, however; it is a declarative programming language combining many of the capabilities of XFST, LEXC, and SQL.

time, as with English. In these cases, the standard approach is to build a pronunciation dictionary which is often either created entirely by hand or involves a significant amount of human intervention to handle exceptions.

Nearly all Indigenous languages spoken in Canada, including the languages involved in this project, have relatively newly developed writing systems. While this means that the languages’ orthographic forms do not deviate greatly from their pronunciation forms, it also means that in many cases there is not an agreed-upon standard, and that even when a standard is set, it is often not used unanimously across the language community (Hinton, 2014). Handling this level of variation presents a challenge to developing natural language processing tools. This problem is further compounded by the fact that many Indigenous languages are comprised of multiple dialects.

Our project has partially circumvented the complexity of multiple dialects and orthographies because there is a standard orthography that has been chosen in the educational organizations we are partnered with (even if that standard is not unanimously used in the community), and our primary use case for speech synthesis is to generate speech for existing text-based tools like dictionaries and verb conjugators (§3.2) which themselves are implemented in specific orthographies and dialects. This dialectal and orthographic consistency means that rule-based approaches to G2P are feasible, and indeed they already exist for the three languages in question as for many other Canadian Indigenous languages (Pine, Littell, et al., 2022).

One danger of this approach, however, is that the adoption of a particular dialect or orthography to the exclusion of others can be inferred to be a form of linguistic or orthographic prescriptivism (Perkins, 2020), even if it is not the intention to prescribe the use of a particular dialect or writing system.

### *3.3. Audio Data Collection & Recording*

Collecting training data is almost an afterthought in languages with millions of speakers: one simply hires a third-party voice actor, or (more likely) uses one of many pre-existing datasets with appropriate licensing for speech synthesis. For Indigenous language organizations, however, collecting training data comes with an enormous opportunity cost, since fluent speakers are their most precious and limited resource. While it might be possible for many organizations to commit one of their teachers to make 20 hours of speech recordings to train a TTS system (possibly meaning 200+ hours outside of the classroom, see §3.3.3), if the resulting system only practically provides a few hours of additional educational value for students, this would not be a good strategic investment. Accordingly, the following sections describe estimates for the amount of time required to create recordings as well as some of our efforts to increase efficiency.

#### *3.3.1. Recording Method*

As part of the kick-off meeting (§3.1), the first author taught a 1-day recording workshop to guide personnel through operating the recording equipment used in the project. For recording, we use either a Zoom H6 recorder for remote recording or a Scarlet Solo as an interface and Audio Technica AT2020 large diaphragm condenser microphones mounted on a shock mount and fitted with a pop filter, along with Neewer NW-12 portable tabletop isolation shields. We record at 96 kHz/16 bit to lossless raw WAV format. This quality is higher than required by speech synthesis, but is recommended for archival quality by the Indigitization project (Bickel & Dupont, 2018), the project from which many of our workshop materials were sourced. Workshop participants were encouraged to maintain consistent volume, speaking rate, and 20 cm distance from the microphone throughout the recording sessions.

Our recording prompts represent sentences that have value beyond just the TTS project. That is, instead of selecting prompts that represent phonetic balance in a corpus as in Veaux et al. (2013), we are recording unfiltered sentences from dictionaries or stories, histories and legends. This may mean that our data is more repetitive, less efficient or that we eventually need to

supplement our dataset with examples containing particularly rare phonemes or phoneme sequences, but since recordings of this quality are rare for the languages described, we wanted to have them be useful beyond the project.

### 3.3.2. *Common Voice*

Unifying the text, recording, and modelling streams of the project is a logistically difficult task. Personnel within each stream are spread across three continents and five time zones, so most of our work is accomplished asynchronously. Many speech synthesis projects use local software (Clark & Bakos, 2015; Draxler & Jänsch, 2004), but managing the transfer of data between the three streams would be error-prone and require a high amount of coordination. To streamline the transfer of data between all parties involved in the project, we forked Mozilla’s Common Voice web application (CVWA) (Ardila et al., 2020; Common Voice, 2022) to adapt it to our needs.

Common Voice is a crowdsourcing project developed by the Mozilla Foundation that allows for volunteers to donate recordings of their voices to an open-source speech database. The interface also allows volunteers to donate their time by validating recordings. There are large datasets for over 100 languages in the latest released version (Version 13.0); as of writing, the English corpus contains 3 209 hours (2 429 validated) of utterance-aligned speech.

Our fork of the CVWA allows us to have a single, centralized location for uploading, archiving, recording, and retrieving all text and speech data related to the project. However, we were not able to simply direct our users to the platform for several reasons. Most importantly, as discussed in §Appendix A, maintaining community-restricted access to and control of the data is absolutely vital. While Common Voice’s goal of a radically open-source speech dataset is commendable for English, it is not appropriate in our context. To address this, we put our version behind a web gateway with a strict allowlist that only permits project members to access the site. Furthermore, the CVWA compresses all recordings using an MP3 encoding and uses a 48 kHz sampling rate by default. Ardila et al. (2020) state that using a lossy encoding for their data was a decision made to improve browser compatibility. Presumably, since the CVWA is deployed internationally, a compressed version of the audio also reduces network transfer issues. We instead record and store a lossless version of the audio suitable for archiving.

Our version of the CVWA is implemented with a continuous deployment pipeline that rebuilds and deploys the app when either the application code is changed or text data is uploaded. The resulting workflow is that personnel

from the text stream upload plain text files with verified text to the GitHub repository storing the code, which triggers a new build of the application, and personnel in the recording stream are subsequently notified that there are more utterances to record. Finally, the modelling stream personnel can retrieve all data for specific languages or speakers from the same location when recording is complete.

### 3.3.3. *Setting Realistic Estimates for Recording*

It is difficult to estimate how much time is required to create a dataset of recorded speech, due to numerous challenges that vary between recording contexts. The challenges in recording that we have experienced can be coarsely categorized as being related to recording environment, software, speaker availability or experience with voice acting.

The speakers of the languages involved in this project are generally already working in a variety of capacities, with many competing demands on their time, meaning that their availability is often a significant limiting factor. Additionally, some of our initial recording environments were located in schools (in part to reduce the travel time for speakers). However, because the school was not a dedicated and permanent recording environment, there was extra time spent setting up and tearing down the equipment for each session. There were also periodic noises from inside or outside the school that would require that we temporarily stop recording, and since it was the primary workplace of the speaker, there would also be occasional interruptions or competing interests for their time.

Recording in non-dedicated recording environments using *SpeechRecorder* (Clark & Bakos, 2015) to record sentence-by-sentence, we were obtaining an average of approximately 3.5 minutes of recordings per 1.5 hour session (studio time/recording time ratio of 25:1). By contrast, when we switched to using our version of the Common Voice web application (CVWA) and to longer 4 hour sessions at a permanent recording studio off-site, we began averaging 30 minutes of recordings per 4 hour session, slightly more than 3 times our previous rate. This is partly due to reduced interruptions in the off-site recording studio, but we speculate it is also a result of the gamification features in CVWA, such as progress bars with daily targets for recordings and positive messaging following uploads, which seem to encourage speakers.

We now use a rough, conservative 10:1 estimate for future recordings assuming we are able to use CVWA, the speaker has participated in an introductory recording/voice acting workshop, and the recording environment

is free of disruptions. In other words, under the aforementioned conditions, we estimate it will take 100 hours in the recording booth to create 10 hours of raw data (including silence), with additional time needed for verifying and correcting recordings, and post-processing to create the dataset.

### *3.4. Evaluation*

In addition to the many challenges involved in gathering data and building TTS systems in low-resource contexts, evaluating the resulting speech is also extremely difficult. We discuss the theoretical aspects of this challenge in greater depth in §4.4, but we provide a quick synopsis of the evaluation methodology for our project in this section as well.

Since the total number of speakers of some of the languages we are working with is less than the number of participants required for statistically significant subjective listening tests (Wester et al., 2015), we cannot rely on them for routine evaluation in our project. Even for languages with larger numbers of speakers, we must be judicious when choosing when to spend valuable time performing large-scale subjective evaluations.

For our project, we only intend to evaluate our systems using larger-scale subjective MOS-style listening tests at the very end of the project, prior to publicly implementing them. In the interim in order to help triage our models, we conduct our evaluations in a more targeted way, through qualitative interviews with language speakers who are directly involved in the project. Those of us who are speakers of these languages have come up with a test set of words and sentences that are representative of the domain we wish to synthesize. We then synthesize audio using the EveryVoice TTS toolkit (§5) and generate a time-aligned ‘ReadAlong’ of the synthesized speech (Littell et al., 2022; Pine et al., 2023) which then allows project member evaluators who speak the languages in question to annotate errors specific to individual words or phrases. These errors are then analyzed by the members of the project involved in TTS modelling to help guide future data collection or changes to the TTS modelling design (see Figure 2 for a diagram of this process).

## **4. Four Challenges for Low-Resource Educational TTS**

The following subsections describe some of the main challenges for modelling and evaluating low-resource educational TTS, with a variety of possible

preliminary strategies for addressing them. We have written this section with the hopes that it will catalyze increased future research efforts in these areas.

#### *4.1. Controllability & Pedagogical TTS*

Speech synthesis has been shown to be an effective educational aid in computer-aided language learning (CALL), helping to improve learner abilities in a wide range of tasks such as listening comprehension, writing skills, pronunciation, and discerning differences in accent (Bione et al., 2017; Liakin et al., 2017; Lim, 2022). Despite these positive results, most existing studies on TTS in the classroom employed off-the-shelf TTS systems that were not tailored to an educational context. Unlike a typical non-educational TTS use case (say, in turn-by-turn navigation), our primary goal is not just to be intelligible to people that are already fluent in the language, but to provide pronunciation help for a non-fluent audience. Targeting pedagogical-quality TTS in the way that was requested by the Indigenous partners in the project (see §Appendix A) leads to increased demands on the controllability and naturalness of the synthesized speech.

In this paper, we will use the term ‘pedagogical speech’ in a narrow sense, as being one of two types of speech with educational value. Pedagogical speech in the language classroom represents a particular style which is slower and more carefully enunciated than typical speech. This can involve a slower pace, the separation of syllables, the hyper-articulation of difficult or easy-to-miss sounds (e.g. glottal stops), and the restoration of reduced or dropped segments (e.g. short vowels). This does not imply that more rapid, natural-sounding speech has no place in the classroom. A typical language learner might first be exposed to pedagogical speech, in order to master listening and pronunciation skills; as the learner gains expertise and confidence, they would be encouraged to become comfortable with more natural speech. Ideally, our systems should be able to produce both pedagogical and natural speech.

There are existing efforts to address inference-time controls over style, for example, in the synthesis of Lombard speech (Hu et al., 2021) and emotion (Kosgi et al., 2022). An alternative approach uses Global Style Tokens (Y. Wang et al., 2018), embeddings learned jointly during training using an additional reference encoder alongside the text encoder in a neural TTS system, which during training may learn to represent factors not explicitly accounted for by the text. This typically accounts for the suprasegmental elements of prosody such as pitch contours, as well as utterance-level factors such as speaking rate, emotion or background noise. In other work carried

out by members of our project, Nishihara et al. (In Submission) present a PCA-based technique for controlling these multiple characteristics of hyper-articulated speech by moving along a single dimension, providing a practical way for users to control the degree of hyperarticulation at inference time.

Beyond stylistic manipulation in synthesized speech, controllability is important in an educational setting for making corrections. We anticipate possible pronunciation errors with suprasegmental features such as stress and tone. Orthographies vary in terms of whether these features are marked, and even where they are marked in the languages we are working with, not all writers use them reliably. It is therefore likely that our systems will generate at least some outputs with incorrect or unnatural prosody, leading to a question of how, practically, we can correct this. Recently, models such as FastSpeech 2 (Ren et al., 2021) and FastPitch (Łańcucki, 2021) have added explicit pitch, energy, and duration prediction modules, rather than relying on implicit prosody prediction like other architectures such as Tacotron (Shen et al., 2018; Y. Wang et al., 2017). This also allows for fine-grained control of these parameters at the frame or phoneme level when synthesizing speech with these models.

Instead of asking the user to use a command-line or graphical user interface to adjust prosodic features in speech, Aylett et al. (2019) investigate the use of ‘voice puppetry’, in which humans give recorded corrections to supply a target prosodic contour, and the output of the TTS is adjusted to match it. This kind of feedback, used to adjust the manipulable representations of the models mentioned above, could be an intuitive way for non-experts to control and adjust prosody when necessary. Such a system could be used, for example, to set initial values for pitch and duration sliders across an entire utterance simply by speaking aloud, which could then be followed by additional fine adjustments as necessary.

#### *4.2. Data Efficiency*

The availability and quality of transcribed speech data are major concerns for training TTS models. While our project is well-resourced enough to be able to create more than enough training data, albeit with some challenges (§3.3.3), this is not the case for many language communities. One pillar of our project, therefore, is to reduce the amount of data needed to train a modern TTS system: not simply turning Indigenous language TTS into a theoretical possibility, but to the point where training such a system is a sound strategic decision for resource-limited organizations. From an educational perspective,

that is one of the core value propositions of TTS: the ability to create a very large (even infinite) collection of recordings by recording only a subset of them. The more we can reduce training data requirements, the better this promise is fulfilled.

Perhaps due to the prevalence of autoregressive, attention-based TTS systems such as Tacotron 2, there seems to be a perception that the data requirements of such systems (i.e. tens of hours of speech) are representative of neural TTS in general. However, this requirement comes in part from the need to learn robust alignments between input symbols and much longer sequences of acoustic features, the failure of which constitutes a major class of errors for attention-based TTS (Valentini-Botinhao & King, 2021). Such attention failures are more likely to occur with smaller training corpora, resulting in completely unintelligible TTS output; we have previously found the cut-off to be somewhere between 5 and 10 hours in experiments on English using Tacotron 2 (Pine, Wells, et al., 2022).

By contrast, non-autoregressive systems such as FastSpeech (Ren et al., 2021; Ren et al., 2019) and FastPitch (Łańcucki, 2021) instead either incorporate an explicit duration predictor learned from forced alignments or use an alternative alignment module with a strict monotonic prior (Badlani et al., 2022). These approaches provide a more stable basis for training the decoder module to predict acoustic features, as the alignment portion of the learning process typically converges much more quickly than traditional attention-based methods. Badlani et al. (2022) found that replacing the attention module in a Tacotron 2 system with a monotonic alignment framework reduces overall convergence time. In previous work, we were able to train intelligible TTS systems from scratch using a modified FastSpeech 2 architecture with as little as 15 minutes of English speech, 25 minutes of SENĆOŦEN and 3.5 hours of Kanyen'kéha (Pine, Wells, et al., 2022).

Our call to researchers working on novel neural architectures for TTS is to consider the data efficiency of the models they develop, and to report the results of their systems on limited amounts of data as in Pine, Wells, et al. (2022) and Kharitonov et al. (2023).

#### *4.3. Cross-lingual Transfer Learning*

The data efficiency problem can also be approached by making use of existing data in other languages more effectively. A common technique is to train on combined corpora including data from multiple languages simultaneously, as in Demirsahin et al. (2018), Gutkin et al. (2018), Korte et al. (2020),



and Y. Zhang et al. (2019). By contrast, transfer learning approaches first train a source model in a well-resourced setting – for example a multi-speaker TTS model trained on hundreds of hours of transcribed English speech – then fine-tune it using a smaller amount of target-domain data, as in Y.-J. Chen et al. (2019), Latorre et al. (2021), and Wells and Richmond (2021).

In both cases, it is necessary to unify the input vocabulary of the model, which for TTS usually comprises either the set of characters used to write the target languages, or their combined phone inventories. The simplest approach is to take the union of input symbols across all target languages, although this may fail to address certain problems when working with a diverse set of languages. Y. Zhang et al. (2019) noted that this could lead to very large input spaces if using character inputs from multiple different writing systems. For phone inputs, symbols from the International Phonetic Alphabet (IPA) are useful since they are intended to provide a universal phonetic representation across languages. However, problems of exploding vocabulary size persist depending on how broad or narrow the transcriptions are, among other language-specific choices in phonetic description (Demir-sahin et al., 2018). This has implications for both multilingual training and fine-tuning approaches: if language-specific symbol inventories do not overlap, the potential benefits from shared encoder training will be limited, since certain parts of the model (namely the embeddings for symbols unseen in source languages) will only ever be trained on limited target-language data. Since our target languages indeed represent a diverse group with limited phonemic overlap and different orthographic conventions, neither character nor phone inputs are particularly well suited.

Rather than take as input the specific phoneme inventories of each language, we consider features below the level of the phoneme. Articulatory features (binary features describing the configuration of the human vocal apparatus as a sound is being made) are a natural starting point: all possible human speech sounds can be differentiated using an inventory of roughly 20 features for place and manner of articulation, and there are existing software libraries that can perform this conversion (Mortensen et al., 2016). For example, the phone inventories shown in Fig. 1 on page 6, demonstrate that many phones are unique to a single language if taken as atomic IPA symbols, but differ only by a single phonological feature, for example, /t/ shared across Kanyen’kéha, SENĆOŦEN, and nêhiyawêwin and /t’/ unique to SENĆOŦEN. To make this point more concrete, there is a 17% overlap between phone sets in Kanyen’kéha (moh) and SENĆOŦEN (str) as seen

in Fig. 1, but there is a 79% overlap between the sets of articulatory features in each language (as calculated by our previous implementation (Pine, Wells, et al., 2022)) using 24 segmental features from PanPhon (Mortensen et al., 2016) plus an additional 7 features representing suprasegmental features inspired by W. S.-Y. Wang (1967). Using these features instead of language-specific phoneme inventories avoids the problem of distinct input vocabularies, and thus moves us substantially closer to the ideal of being able to easily fine-tune a pre-existing model on a new language. This approach has already been shown to help in training multilingual TTS models for low-resource languages of India (Gutkin et al., 2018), and to be effective for cross-lingual transfer learning as well (Lux & Vu, 2022; Staib et al., 2020a; Wells & Richmond, 2021).

In extending this work to Indigenous languages, with potentially very different sound inventories compared to English, future research should also consider using data from additional languages, which would allow for greater coverage over possible combinations of phonological features compared to pre-training on a single high-resource language, as previously investigated by Do et al. (2022) and Maniati et al. (2021).

#### 4.4. Evaluation

Evaluation of TTS systems relies heavily on listening tests, with fluent speakers of the target language being asked to make judgements about the quality of synthetic speech or to transcribe synthesized utterances, and with word error rates over their transcriptions being interpreted as a measure of intelligibility. This is a slow and expensive process, with recruitment and payment of participants presenting a significant bottleneck in the development of TTS systems even for languages such as English. If we add in the lack of availability of fluent speaker time for our target Indigenous languages, the task becomes even more daunting.

These costs have inspired the adoption of so-called objective measures of evaluation. These include simple metrics like spectral distance (e.g. Mel cepstral distortion and log spectral distance), framewise percentage of F0 voicing errors or F0 RMSE, and more complex metrics based on auditory models (e.g. PESQ (Rix et al., 2001), POLQA (Beerends et al., 2013), STOI (Taal et al., 2010)). Recent efforts in the automatic evaluation of synthetic speech have seen the rise of data-driven quality prediction models based on neural networks. The most prominent such models are AutoMOS (Patton et al., 2016), trained on a large set of proprietary data (47 320 data points),

and MOSNet (Lo et al., 2019), trained on data from the Voice Conversion Challenge (13 580 data points). However, experience shows that mean opinion score (MOS) prediction requires large amounts of training data and is particularly challenging for unseen speakers, listeners and systems (Huang et al., 2022). Other work focuses on intelligibility testing, for example, using automatic speech recognition (ASR) systems to transcribe synthetic speech rather than human listeners (J. Taylor & Richmond, 2021). ASR is its own data-intensive problem, however, limiting this approach in practice to high-resource languages such as English. The question of whether objective (or automatic) evaluation methods that do not require human participants can be applied to our particular context remains open, but we see some promising directions to follow. The limited time and availability of native-speaker evaluators means that they will not be able to rate every sample from every system we build. While we do not want to release any systems without some native-speaker evaluation and approval, we also need to be respectful of evaluators’ time. These evaluators are usually busy teachers and language professionals, and we must consider the opportunity cost that comes with asking them to evaluate potentially hundreds of experimental systems.

Recruiting student learners and linguists familiar with the target language may help alleviate this by serving as a first-pass screening to decide which samples most need native-speaker judgment. While not fluent speakers, we believe that they could bolster the existing participant resources available for listening test evaluations. For example, though they may not be able to make judgements on questions of overall naturalness without being speakers of the language, linguists can potentially provide a judgment on questions like “Does this utterance contain a lateral fricative?”. This could allow for more efficient iteration when developing particular features, for example, checking speech output against a candidate G2P system without using valuable fluent speaker time just to ensure each system is basically linguistically correct. Crucially, however, fluent speakers should not be replaced by learner or linguist evaluators. Rather, this first-pass evaluation could allow limited fluent speaker effort to then be prioritised on making final judgements about the suitability of the voice for the target application in light of their organization’s and community’s needs, which only they can speak to.

As a proxy for measuring intelligibility as well as spotting pronunciation errors, rather than using an ASR model (which itself would rely on huge language resources for training), phonological feature detectors as in Qamhan et al. (2021), could present a more efficient alternative. In this framework,

in order to quantify pronunciation errors, sequences of phonological features estimated for a synthetic utterance are compared to canonical features derived from the input text (or to those extracted from a natural reference signal). These detectors can in principle be trained with multilingual speech data, potentially obviating a lack of data in any given language of study for evaluating segmental features of synthetic speech. Finally, to alleviate the need for huge amounts of human scores to train automatic evaluation models, we may investigate the use of self-supervised speech representation models to better leverage any small amount of labelled data, following encouraging results on zero-shot and out-of-domain tasks by Cooper et al. (2022).

Importantly, future work on automatic methods to reduce the resource requirements for evaluation should only ever be used to triage models before human evaluation by fluent speakers, not replace such evaluation. All models must be thoroughly evaluated prior to public release, but more automated methods would be helpful in determining which models to present to the final stage of evaluation.

## 5. Motivation & Design of the EveryVoice TTS Toolkit

After determining some of the modelling requirements of a potential TTS system (§4), we set about determining the appropriate toolkit to include in our ‘repeatable recipe’ (§2.4) as well as for our own research and use in educational applications. We surveyed a range of existing TTS toolkits which we summarize in §5.1, and introduce our preliminary implementation of a new toolkit, titled EveryVoice TTS (§5.2) along with benchmark naturalness and intelligibility evaluation results (§5.3). We also direct the interested reader to Appendix B and Appendix C which discuss and compare important features for our use case between EveryVoice TTS and six other popular existing toolkits in greater detail.

### 5.1. Existing TTS Toolkits

There are many excellent toolkits for developing neural speech synthesis models (Gölge & The Coqui TTS Team, 2021; Hayashi et al., 2021; Kuchaiev et al., 2018; Lux et al., 2021; C. Wang et al., 2021; Watanabe et al., 2018; H. Zhang et al., 2022). However, it is not immediately obvious how to evaluate the existing toolkits with respect to our project, and whether we should adopt any of them for use in our repeatable recipe. The vast majority of existing toolkits prioritize research applications by implementing a wide array

of models and supporting many non-TTS tasks as well. For certain research applications this variety of models and tasks is advantageous, but it brings increased complexity which could be confusing for users who are only looking for a model to use for limited-resource TTS.

As one concrete example, the IMS Toucan toolkit (Lux et al., 2021) proposes a single main architecture, and is specifically geared towards low-resource applications. They implement a two-stage system comprised of a modified FastSpeech2 feature prediction network paired with either an Avocado (Bak et al., 2023) or BigVGAN (Lee et al., 2023) vocoder if the user is willing to compromise inference speed for naturalness (a decision that is documented and explained to the user). The decision to limit the toolkit in terms of available architectures provides a sensible baseline model for new users looking to build TTS for their own language, removing much of the initial guesswork compared to the larger selection of possible architectures included in other toolkits such as ESPnet (Watanabe et al., 2018), Coqui TTS (Gölge & The Coqui TTS Team, 2021) and NeMo (Kuchaiev et al., 2019). However, much of the hyperparameter configuration in IMS Toucan is hard-coded in the model implementation code itself, and setting up training pipelines for new languages involves modifying these modules directly. Alternatively, other toolkits such as NeMo and Coqui TTS provide powerful and comprehensive control of hyperparameter settings through external configuration files, providing a clear pathway for users to easily modify an initial baseline model once selected from the many on offer. An ideal toolkit for our repeatable recipe should combine the strengths of both of these approaches, by providing a baseline model tuned for low-resource TTS alongside strong configuration tooling, with explicit guidance, documentation and hyperparameter validation for users working with diverse languages and datasets.

### *5.2. EveryVoice TTS Toolkit*

To help address some of the above-mentioned task-specific challenges, we present the development of a new TTS toolkit, named the EveryVoice TTS toolkit. EveryVoice TTS has been designed to provide a unified speech synthesis toolkit specifically tailored to limited-data TTS applications. Currently, the model we have chosen is a two-stage system based on specific considerations for our use case primarily around phone-level controllability (§4.1) and data-efficiency (§4.2). The system is comprised of a feature prediction network based on FastSpeech2 (Ren et al., 2021), and a vocoder based on iSTFTNet (Kaneko et al., 2022). While so-called end-to-end systems

like VITS (Kim et al., 2021) are capable of producing impressively natural speech, in low-resource applications, two-stage systems are currently more data efficient since the vocoder is trained without text, lessening the burden of required transcribed data. As discussed in §4.1, FastSpeech2 predicts pitch, energy, and duration at the phone level, which makes it a suitable baseline model for experimenting with controllability. The model has also been shown to be data efficient, requiring as little as 15 minutes of data to produce intelligible speech (Pine, Wells, et al., 2022). To facilitate transfer learning (§4.3), we implement the possibility of training using phonological features as calculated by PanPhon (Mortensen et al., 2016) and  $G_i2P_i$  (Pine, Littell, et al., 2022), which provides out-of-the-box grapheme to phoneme conversion for dozens of Indigenous languages. Our choice of model architecture here should be understood as a sensible baseline for low-resource TTS, but we hope to continually adjust the model and default hyperparameters as research with low-resource TTS architectures advances.

Beyond the aforementioned decisions regarding modelling, EveryVoice TTS has been designed with adaptation to new languages and datasets in mind. This includes providing tools and guidance to assist users in creating and preprocessing their own datasets, for example by providing audio segmentation tools, removing silence, and detecting and removing outlier audio samples. We also provide a configuration wizard command line interface tool for supporting users in designing hyperparameter configuration files to suit their languages. For more implementation details and comparisons with other toolkits, we direct the interested reader to Appendix B and Appendix C.

### 5.3. Benchmark Evaluation

To help justify our choice of model architecture, we provide the results from an evaluation of the EveryVoice TTS architecture, with English as the test language. In order to limit our evaluation to a manageable size, we limit our comparison of EveryVoice TTS to being against models trained with the NeMo toolkit (Kuchaiev et al., 2019), despite the larger list of toolkits surveyed in Appendix B. NeMo was chosen because it appeared to satisfy the most requirements of any of the toolkits that we surveyed. Specifically, it offers production-ready implementations of data-efficient TTS models, robust tools for data preparation, and thorough guides and documentation, including introductory content to the field of speech synthesis.

We trained two feature prediction models with EveryVoice TTS and two other feature prediction models with a Fastpitch implementation from NeMo.

All models were trained using either the full LJ Speech (Ito & Johnson, 2017) dataset or a fixed 30-minute subset, with input texts converted to phone sequences. For the NeMo models, we synthesized audio samples from the generated Mel spectrograms using NeMo’s pre-trained ‘tts\_en\_hifigan’ vocoder checkpoint (NVIDIA, 2023) which was trained on the LJ Speech dataset as well as Mel spectrograms generated from FastPitch, Tacotron2 and TalkNet (Beliaev & Ginsburg, 2021). We generated audio samples for the EveryVoice TTS models using a pre-trained EveryVoice TTS vocoder checkpoint that was trained for 2 500 000 steps on ground-truth Mel spectrograms from the LJ Speech, LibriTTS (Zen et al., 2019), and VCTK (Yamagishi et al., 2019) datasets using the ‘C8C8I’ iSTFTNet model specification and default hyperparameters (Kaneko et al., 2022). Since the NeMo vocoder had been trained on generated Mel spectrograms as well, we also finetuned our pre-trained EveryVoice TTS vocoder checkpoint with the Mel spectrograms from our EveryVoice TTS feature prediction models. We then conducted a short (15 minute) listening test that compared the two NeMo models against our two EveryVoice TTS models using both the fine-tuned EveryVoice TTS vocoder and basic pre-trained checkpoint. We recruited 30 participants through Prolific, and presented each with 28 MOS-style questions where they were asked to rank each sample based on naturalness from 1 to 5.

Our MOS results are presented in Figure 4 and Table 1. For both dataset sizes, we find that EveryVoice TTS with a fine-tuned vocoder performs comparably to NeMo. We tested for significant differences between systems using the Mann-Whitney U test, with a Bonferroni correction applied to account for repeated pairwise comparisons, and found no significant difference between ratings for the EveryVoice 30m FT (finetuned) and NeMo 30m, nor between EveryVoice Full FT and NeMo Full ( $p = 0.05$ , corrected  $\alpha = p/21 = 0.0024$ ). There was also no significant difference between EveryVoice 30m and EveryVoice Full, using our pre-trained vocoder without fine-tuning. These results reflect the impact of training vocoders to synthesize natural speech from TTS-predicted Mel spectrograms. Artefacts introduced by our pre-trained vocoder, which has only seen Mel spectrograms derived from natural speech, apparently drown out any other differences between our EveryVoice models trained on differing amounts of data. Fine-tuning on predicted Mel spectrograms provides a significant increase in perceived quality, and further appears to allow listeners to discriminate between models trained on 30m and Full corpus subsets. The two NeMo models show clear differences between 30m and Full corpus subsets likely because their pretrained vocoder checkpoint,

trained on a mixture of natural and synthesized Mel spectrograms, does not produce the same distracting artefacts.

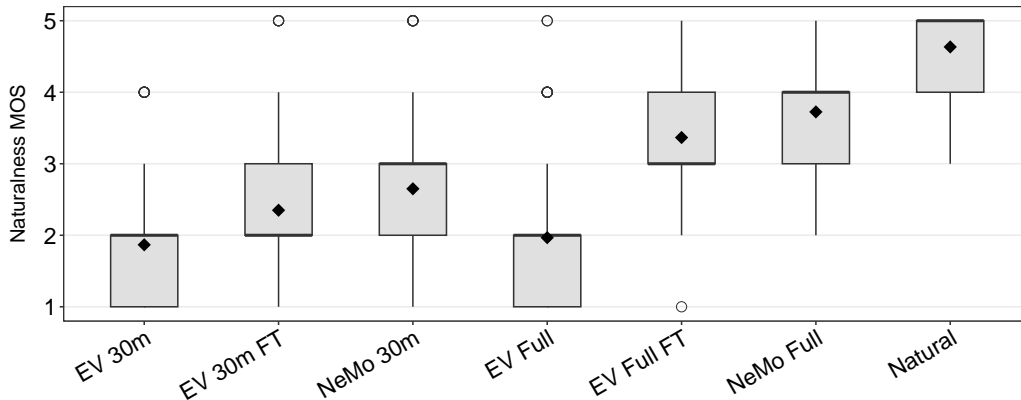


Figure 4: Naturalness MOS ratings for EveryVoice TTS (EV) and NeMo models trained on the full and 30 minute subset of the LJ Speech dataset. Diamonds indicate average MOS values per voice, and circles indicate outliers.

Voice	MOS $\uparrow$	WER (%) $\downarrow$
EveryVoice 30m	$1.87 \pm 0.16$	18.77
EveryVoice 30m FT	$2.35 \pm 0.19$	18.50
NeMo 30m	$2.65 \pm 0.20$	16.92
EveryVoice Full	$1.97 \pm 0.18$	8.72
EveryVoice Full FT	$3.37 \pm 0.17$	8.35
NeMo Full	$3.73 \pm 0.17$	7.96
Natural Speech	$4.63 \pm 0.11$	5.98

Table 1: Evaluation results for both EveryVoice TTS and NeMo models trained on the full and 30 minute subset of the LJ Speech dataset. MOS indicates mean opinion score results with 95% confidence intervals from our subjective listening test and WER (%) indicates the word error rates from our ASR-based intelligibility evaluation. ‘FT’ indicates that the vocoder used to synthesize samples was finetuned on generated Mel spectrograms from the EveryVoice TTS feature prediction network.

In addition to subjective naturalness ratings, we present an objective analysis of system intelligibility using word error rates (WER), as shown in Table 1. To word error rates, we employed Whisper (Radford et al., 2023), a robust general-purpose speech recognition model, using the ‘openai/whisper-base’ model (OpenAI, 2023). The evaluation was conducted on a test set from



the LJ Speech dataset, comprising 512 sentences which were synthesized with each model. For both dataset sizes, we find that EveryVoice TTS models are within 2% WER of the NeMo models. We also see that the WER for samples generated with a finetuned vocoder are within 0.5% of their base checkpoint equivalents, which further suggests that the gap between MOS scores for finetuned and base EveryVoice models is influenced by vocoder-related artefacts affecting naturalness and not model intelligibility.

These results indicate that EveryVoice can produce speech of comparable naturalness and intelligibility to an existing production-ready TTS toolkit. We speculate that the observed gap in performance is largely attributable to vocoder training strategies rather than gross architectural differences, but we intend to investigate this in greater detail in future work. Even with our initial baseline architecture, we believe that EveryVoice is a good choice for certain low-resource TTS use cases, due to its specific supports for new language configuration, misuse prevention, and dataset creation. Ultimately, the decision of which toolkit to use will depend largely on the availability of required features, and the degree of expertise held by the users of the toolkit.

## 6. Conclusion

In this paper, we have presented the motivation for the Speech Generation for Indigenous Language Education project, providing signposts and a road map for readers who are looking to research, develop, and/or manage TTS projects for low-resource languages (§1.2). To help situate and motivate our research, we described the sociolinguistic context of this research and prior work (§2). Ethical issues associated with some (but certainly not all) prior work on TTS for Indigenous languages are discussed in §2.3; such issues related to potential misuse of the technology were also an important part of early discussions among the collaborators (Appendix A), and motivate the discussion in Appendix C.1.2. Our methodology section (§3) describes in detail how our project is organized and how we co-developed the shared vision of the project with each of the collaborating organizations involved. We also detailed four key theoretical challenges for educational low-resource TTS (§4); controllability and ‘pedagogical’ style for TTS, data efficiency, cross-lingual transfer learning, and evaluation. For each of these sections, we describe prior work in the area and hope that the section motivates future research related to it.

We have partially addressed some of the technical challenges described in this paper through the adaptation of Mozilla’s Common Voice platform for gathering recordings, and the development of our own speech synthesis toolkit, EveryVoice TTS, following a survey of existing neural speech synthesis toolkits (findings of the survey are presented in Appendix B and Appendix C).

While our evaluation for the EveryVoice TTS toolkit (§5) shows modest but comparable results with the existing NeMo toolkit, we believe that the design of EveryVoice TTS fills an ecological niche different from that of many other excellent TTS toolkits: it is particularly suitable for low-resource languages with few fluent speakers because it is extremely data-efficient (it requires very little speech training data), and it has been designed to be user-friendly when adapting to a new language or dataset.

The following list provides a summary of recommendations drawn from our project, which we hope will assist similar limited-resource TTS efforts:

- ‘Low-resource’ does not mean no-resource, even when no training data is available. Consulting available documentation and community-held knowledge is key to addressing gaps in data (§1 and §2).
- Ensure that the project is led and designed by – or co-led and co-designed with – organizations that represent the language community which, ideally, will benefit from the speech synthesis technology. This structure will help determine the correct goals and requirements of the speech synthesis efforts from the outset (§2 and §3).
- Collaboratively define the requirements of the TTS system with community stakeholders. This will help set the priorities for all other aspects of the project including recording and gathering text, and selecting a model architecture (§2.2).
- Determine the domain for synthesis in advance. This will help identify whether there is adequate available text in this domain that can be synthesized. If an insufficient amount of text exists in the target language, consider building a generative text model for a particular pattern of the grammar which can create both targeted training and evaluation data as well as provide a useful educational application in the process (§3.2). Specifying the intended domain for synthesis in advance will also help guide recording efforts (§3.3).

- Where possible, use cloud-based software like Mozilla’s Common Voice for recording, since the streamlined and gamified process encourages speakers. It also reduces human error related to file labelling, storage and transfer (§3.3.2).
- Do not underestimate the amount of time needed for recording, particularly in an ad-hoc recording environment. It can easily take ten hours to produce a single hour of data (§3.3.3).
- Review available toolkits and assess which one contains the features necessary for your project as well as adequate support for your development team (§5 and Appendix C).
- Mitigate some of the potential harms associated with the misuse of TTS technology by adopting a conservative data and model access policy by default, and transparently discuss the issues with community stakeholders (§Appendix A and Appendix C.1.2).

## Acknowledgments

We would like to sincerely acknowledge and thank the many people who have supported this project including, but not limited to, our collaborators Owennatekha Brian Maracle and Rohahiyo Jordan Brant at Onkwawenna Kentyohkwa, Marilyn Shirt, Tina Wellman and Wayne Jackson at University nuhelot’ine thaiyots’i nistameyimâkanak Blue Quills, and SXEDŦELISIYE Renee Sampson, PENÁĆ David Underwood and Tye Swallow at the WSÁNEĆ School Board who have all contributed to the shared vision of this project. We would also like to thank Sonya Bird and the University of Victoria Speech Research Lab, and EM Lewis-Jong at Mozilla for support with recording and Common Voice respectively. Finally, we also thank Annie En-Shiun Lee, Gisele Arevalo, and Rosa McBee for proofreading and suggestions.

This work was supported in part by the National Research Council of Canada’s Ideation Fund: ‘Small teams – Big Ideas’ and the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1) and the University of Edinburgh, School of Informatics and School of Philosophy, Psychology & Language Sciences.

## References

- Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. M., & Weber, G. (2020). Common Voice: A massively-multilingual speech corpus. *Proc. 12th Conf. Language Resources and Evaluation*, 4218–4222.
- Aylett, M., Braude, D., Pidcock, C., & Potard, B. (2019). Voice puppetry: Exploring dramatic performance to develop speech synthesis. *Proc. SSW*, 117–120. <https://doi.org/10.21437/SSW.2019-21>
- Badlani, R., Łańcucki, A., Shih, K. J., Valle, R., Ping, W., & Catanzaro, B. (2022). One TTS alignment to rule them all. *Proc. ICASSP*, 6092–6096.
- Bak, T., Lee, J., Bae, H., Yang, J., Bae, J.-S., & Joo, Y.-S. (2023). Avocado: Generative adversarial network for artifact-free vocoder. *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*. <https://doi.org/10.1609/aaai.v37i11.26479>
- Beerends, J., Schmidmer, C., Berger, J., Obermann, M., Ullmann, R., Pomy, J., & Keyhl, M. (2013). Perceptual objective listening quality assessment (POLQA), the third generation ITU-T standard for end-to-end speech quality measurement Part I-temporal alignment. *J. Aud. Eng. Soc.*, 61, 366–384.
- Beesley, K. R., & Karttunen, L. (2003). *Finite state morphology*. CSLI Publications.
- Beliaev, S., & Ginsburg, B. (2021). TalkNet: Non-Autoregressive Depth-Wise Separable Convolutional Model for Speech Synthesis. *Proc. Interspeech 2021*, 3760–3764. <https://doi.org/10.21437/Interspeech.2021-1770>
- Bickel, R., & Dupont, S. (2018). Indigitization. *KULA: Knowledge Creation, Dissemination, and Preservation Studies*, 2(1), 11. <https://doi.org/10.5334/kula.56>
- Binkowski, M., Donahue, J., Dieleman, S., Clark, A., Elsen, E., Casagrande, N., Cobo, L. C., & Simonyan, K. (2020). High fidelity speech synthesis with adversarial networks. *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. <https://openreview.net/forum?id=r1gfQgSFDr>

- Bione, T., Grimshaw, J., & Cardoso, W. (2017). An evaluation of TTS as a pedagogical tool for pronunciation instruction: The ‘foreign’ language context. In K. Borthwick, L. Bradley, & S. Thouësny (Eds.), *CALL in a climate of change: Adapting to turbulent global conditions – short papers from EUROCALL 2017* (pp. 56–61). Research-publishing.net. <https://doi.org/10.14705/rpnet.2017.eurocall2017.689>
- Bird, S. (2020). Decolonising speech and language technology. *Proc. 28th Int. Conf. Computational Linguistics*, 3504–3519. <https://doi.org/10.18653/v1/2020.coling-main.313>
- Bird, S. (2022). Local languages, third spaces, and other high-resource scenarios. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 7817–7829). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.acl-long.539>
- Boutros, F., Struc, V., Fierrez, J., & Damer, N. (2023). Synthetic data for face recognition: Current state and future prospects. *Image and Vision Computing*, 135, 104688. <https://doi.org/https://doi.org/10.1016/j.imavis.2023.104688>
- Brinklow, N. T. (2021). Indigenous language technologies: Anti-colonial oases in a colonizing (digital) world. *WINHEC: International Journal of Indigenous Education Scholarship*, (1), 239–266.
- Brinklow, N. T., Littell, P., Lothian, D., Pine, A., & Souter, H. (2019). Indigenous Language Technologies & Language Reclamation in Canada. *Proceedings of the 1st International Conference on Language Technologies for All*, 402–406.
- Chen, G., Wu, Y., Liu, S., Liu, T., Du, X., & Wei, F. (2023). Wavmark: Watermarking for audio generation. <https://doi.org/10.48550/arXiv.2308.12770>
- Chen, Y.-J., Tu, T., Yeh, C., & Lee, H.-Y. (2019). End-to-end text-to-speech for low-resource languages by cross-lingual transfer learning. *Proc. Interspeech*, 2075–2079. <https://doi.org/10.21437/Interspeech.2019-2730>
- Chu, W., & Alwan, A. (2009). Reducing F0 frame error of F0 tracking algorithms under noisy conditions with an unvoiced/voiced classification frontend. *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 3969–3972. <https://api.semanticscholar.org/CorpusID:5637941>

- Clark, R., & Bakos, G. (2015) (1.0). Centre for Speech Technology Research. <https://www.cstr.ed.ac.uk/research/projects/speechrecorder/>
- Common Voice. (2022, February 3) (release-v1.66.0). Mozilla. <https://github.com/common-voice/common-voice>
- Conrad, M. (2020). Tacotron2 and Cherokee TTS. <https://www.cherokeeleasons.com/content/tacotron2-and-cherokee-tts/>
- Cooper, E., Huang, W.-C., Toda, T., & Yamagishi, J. (2022). Generalization ability of MOS prediction networks. *Proc. ICASSP*, 8442–8446. <https://doi.org/10.1109/ICASSP43922.2022.9746395>
- Coqui. (2021a). Audio preprocessing [Online; accessed 17-November-2023]. <https://github.com/coqui-ai/TTS/discussions/267>
- Coqui. (2021b). Ways to prevent misuse of TTS technology? [Online; accessed 17-November-2023]. <https://github.com/coqui-ai/TTS/discussions/1036>
- Coqui. (2022). coqpit [Online; accessed 17-November-2023]. <https://github.com/coqui-ai/coqpit>
- Coqui. (2023a). Coqui Studio: realistic, emotive text-to-speech through generative AI. [Online; accessed 17-November-2023]. <https://coqui.ai/>
- Coqui. (2023b). Dataset Analysis [Online; accessed 17-November-2023]. [https://github.com/coqui-ai/TTS/tree/dev/notebooks/dataset\\_analysis](https://github.com/coqui-ai/TTS/tree/dev/notebooks/dataset_analysis)
- Coqui GmbH. (2021). Humble FAQ [Online; accessed 17-November-2023]. <https://tts.readthedocs.io/en/latest/faq.html>
- Czaykowska-Higgins, E. (2009). Research Models, Community Engagement, and Linguistic Fieldwork: Reflections on Working within Canadian Indigenous Communities. *Language Documentation & Conservation*, 3(1), 15–50.
- Défossez, A., Synnaeve, G., & Adi, Y. (2020). Real Time Speech Enhancement in the Waveform Domain. *Proc. Interspeech 2020*, 3291–3295. <https://doi.org/10.21437/Interspeech.2020-2409>
- Demirsahin, I., Jansche, M., & Gutkin, A. (2018). A unified phonological representation of South Asian languages for multilingual text-to-speech. *The 6th Intl. Workshop Spoken Language Technologies for Under-Resourced Languages*, 80–84. <https://doi.org/10.21437/SLTU.2018-17>
- Do, P., Coler, M., Dijkstra, J., & Klabbers, E. (2022). Text-to-speech for under-resourced languages: Phoneme mapping and source language selection in transfer learning. *Proc. SIGUL 2022 @ LREC 2022*, 16–22.

- Draxler, C., & Jänsch, K. (2004). SpeechRecorder - a universal platform independent multi-channel audio recording software. *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. <https://www.bas.uni-muenchen.de/Bas/software/speechrecorder/>
- Faith Comes By Hearing. (2021). Site Terms [Online; accessed 17-November-2023]. <https://www.faithcomesbyhearing.com/terms>
- Farley, P., & Microsoft. (2023). Code of conduct for Azure AI Speech text to speech [Online; accessed 17-November-2023]. <https://learn.microsoft.com/en-us/legal/cognitive-services/speech-service/custom-neural-voice/code-of-conduct>
- Gallegos, P. O., Williams, J., Rownicka, J., & King, S. (2020). An Unsupervised Method to Select a Speaker Subset from Large Multi-Speaker Speech Synthesis Datasets. *Proc. Interspeech 2020*, 1758–1762. <https://doi.org/10.21437/Interspeech.2020-2567>
- Gölge, E., & The Coqui TTS Team. (2021). *Coqui tts* (0.13.0). Coqui AI. <https://github.com/coqui-ai/TTS>
- Green, T. J., & Maracle, O. B. (2018). The root-word method for building proficient second-language speakers of polysynthetic languages: Onkwawén:na Kentyókhwa adult Mohawk language immersion program. In *The Routledge Handbook of Language Revitalization* (pp. 146–155). Routledge.
- Gutkin, A., Jansche, M., & Merkulova, T. (2018). FonBund: A library for combining cross-lingual phonological segment data. *Proc. 11th Int. Conf. Language Resources and Evaluation*, 2236–2240.
- Hammerly, C., Fougère, S., Sierra, G., Parkhill, S., Porteous, H., & Quinn, C. (2023). A text-to-speech synthesis system for Border Lakes Ojibwe. *Proceedings of the Sixth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, 60–65.
- Harper, E., Majumdar, S., Kuchaiev, O., Jason, L., Zhang, Y., Bakhturina, E., Noroozi, V., Subramanian, S., Nithin, K., Jocelyn, H., Jia, F., Balam, J., Yang, X., Livne, M., Dong, Y., Naren, S., & Ginsburg, B. (2023a). NeMo TTS Primer [Online; accessed 30-November-2023]. [https://github.com/NVIDIA/NeMo/blob/stable/tutorials/tts/Evaluation\\_MelCepstralDistortion.ipynb](https://github.com/NVIDIA/NeMo/blob/stable/tutorials/tts/Evaluation_MelCepstralDistortion.ipynb)
- Harper, E., Majumdar, S., Kuchaiev, O., Jason, L., Zhang, Y., Bakhturina, E., Noroozi, V., Subramanian, S., Nithin, K., Jocelyn, H., Jia, F., Balam, J., Yang, X., Livne, M., Dong, Y., Naren, S., & Ginsburg,

- B. (2023b). NeMo TTS Primer [Online; accessed 17-November-2023]. [https://github.com/NVIDIA/NeMo/blob/stable/tutorials/tts/NeMo\\_TTS\\_Primer.ipynb](https://github.com/NVIDIA/NeMo/blob/stable/tutorials/tts/NeMo_TTS_Primer.ipynb)
- Harrigan, A., Arppe, A., & Mills, T. (2019). A Preliminary Plains Cree Speech Synthesizer. *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1 (Papers)*, 64–73. Retrieved July 19, 2021, from <https://aclanthology.org/W19-6009>
- Hayashi, T., Yamamoto, R., Yoshimura, T., Wu, P., Shi, J., Saeki, T., Ju, Y., Yasuda, Y., Takamichi, S., & Watanabe, S. (2021). ESPnet2-TTS: Extending the edge of TTS research. *arXiv preprint arXiv:2110.07840*.
- Hermes, M., & Engman, M. M. (2017). Resounding the clarion call: Indigenous language learners and documentation. In *Language documentation and description, vol 14* (pp. 59–87). EL Publishing.
- Hinton, L. (2014). Orthography wars. *Developing orthographies for unwritten languages*, 139–168.
- Hsieh, C.-P., Ghosh, S., & Ginsburg, B. (2023). Adapter-Based Extension of Multi-Speaker Text-To-Speech Model for New Speakers. *Proc. INTERSPEECH 2023*, 3028–3032. <https://doi.org/10.21437/Interspeech.2023-2313>
- Hu, Q., Bleisch, T., Petkov, P., Raitio, T., Marchi, E., & Lakshminarasimhan, V. (2021). Whispered and Lombard neural speech synthesis. *Proc. IEEE Spoken Language Technology Workshop*, 454–461. <https://doi.org/10.1109/SLT48900.2021.9383454>
- Huang, W. C., Cooper, E., Tsao, Y., Wang, H.-M., Toda, T., & Yamagishi, J. (2022). The VoiceMOS challenge 2022. *Proc. Interspeech 2022*, 4536–4540. <https://doi.org/10.21437/Interspeech.2022-970>
- Ito, K., & Johnson, L. (2017). The LJ speech dataset.
- James, J., Shields, I., Berriman, R., Keegan, P. J., & Watson, C. I. (2020). Developing Resources for Te Reo Māori Text To Speech Synthesis System. In P. Sojka, I. Kopeček, K. Pala, & A. Horák (Eds.), *Text, Speech, and Dialogue* (pp. 294–302). Springer International Publishing. [https://doi.org/10.1007/978-3-030-58323-1\\_32](https://doi.org/10.1007/978-3-030-58323-1_32)
- Jia, F., Majumdar, S., & Ginsburg, B. (2021). Marblenet: Deep 1D time-channel separable convolutional neural network for voice activity detection. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6818–6822. <https://doi.org/10.1109/ICASSP39728.2021.9414470>



- Junker, M.-O. (2024). Data-mining and extraction: The gold rush of AI on indigenous languages. In S. Moeller, G. Agyapong, A. Arppe, A. Chaudhary, S. Rijhwani, C. Cox, R. Henke, A. Palmer, D. Rosenblum, & L. Schwartz (Eds.), *Proceedings of the seventh workshop on the use of computational methods in the study of endangered languages* (pp. 52–57). Association for Computational Linguistics. <https://aclanthology.org/2024.computel-1.8>
- Kaneko, T., Tanaka, K., Kameoka, H., & Seki, S. (2022). iSTFTNet: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time Fourier transform. *Proc. ICASSP*, 6207–6211.
- Kazantseva, A., Maracle, O. B., Maracle, R. J., & Pine, A. (2018). Kawenñón:nis: The wordmaker for Kanyen’kéha. *Proc. Workshop Computational Modeling Polysynthetic Languages*, 53–64.
- Kharitonov, E., Vincent, D., Borsos, Z., Marinier, R., Girgin, S., Pietquin, O., Sharifi, M., Tagliasacchi, M., & Zeghidour, N. (2023). Speak, Read and Prompt: High-Fidelity Text-to-Speech with Minimal Supervision. *Transactions of the Association for Computational Linguistics*, 11, 1703–1718. [https://doi.org/10.1162/tacl\\_a\\_00618](https://doi.org/10.1162/tacl_a_00618)
- Kim, J., Kong, J., & Son, J. (2021). Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In M. Meila & T. Zhang (Eds.), *Proc. 38th int. conf. machine learning*. PMLR.
- Korte, M. d., Kim, J., & Klabbbers, E. (2020). Efficient neural speech synthesis for low-resource languages through multilingual modeling. *Proc. Interspeech*, 2967–2971. <https://doi.org/10.21437/Interspeech.2020-2664>
- Kosgi, S., Sivaprasad, S., Pedanekar, N., Nelakanti, A., & Gandhi, V. (2022). Empathic machines: Using intermediate features as levers to emulate emotions in text-to-speech systems. *Proc. 2022 Conf. North American Chapter Association Computational Linguistics: Human Language Technologies*, 336–347. <https://doi.org/10.18653/v1/2022.naacl-main.26>
- Kuchaiev, O., Ginsburg, B., Gitman, I., Lavrukhin, V., Case, C., & Micikevicius, P. (2018). OpenSeq2Seq: Extensible toolkit for distributed and mixed precision training of sequence-to-sequence models. *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, 41–46. <https://doi.org/10.18653/v1/W18-2507>
- Kuchaiev, O., Li, J., Nguyen, H., Hrinchuk, O., Leary, R., Ginsburg, B., Kri-man, S., Beliaev, S., Lavrukhin, V., Cook, J., Castonguay, P., Popova,

- M., Huang, J., & Cohen, J. M. (2019). NeMo: A toolkit for building AI applications using neural modules. <https://doi.org/10.48550/arXiv.1909.09577>
- Kuhn, R., Davis, F., Désilets, A., Joanis, E., Kazantseva, A., Knowles, R., Littell, P., Lothian, D., Pine, A., Running Wolf, C., Santos, E., Stewart, D., Boulianne, G., Gupta, V., Maracle Owennatékha, B., Martin, A., Cox, C., Junker, M.-O., Sammons, O., . . . Souter, H. (2020). The Indigenous languages technology project at NRC Canada: An empowerment-oriented approach to developing language software. *Proceedings of the 28th International Conference on Computational Linguistics*, 5866–5878. <https://doi.org/10.18653/v1/2020.coling-main.516>
- Kürzinger, L., Winkelbauer, D., Li, L., Watzel, T., & Rigoll, G. (2020). CTC-segmentation of large corpora for German end-to-end speech recognition. In A. Karpov & R. Potapova (Eds.), *Speech and computer* (pp. 267–278). Springer International Publishing.
- Łańcucki, A. (2021). Fastpitch: Parallel text-to-speech with pitch prediction. *Proc. ICASSP*, 6588–6592. <https://doi.org/10.1109/ICASSP39728.2021.9413889>
- Latorre, J., Bailleul, C., Morrill, T., Conkie, A., & Stylianou, Y. (2021). Combining speakers of multiple languages to improve quality of neural voices. *Proc. SSW*, 37–42. <https://doi.org/10.21437/SSW.2021-7>
- Le Ferrand, E., Bird, S., & Besacier, L. (2022). Learning from failure: Data capture in an Australian Aboriginal community. *Proc. 29th Int. Conf. Computational Linguistics*, 4988–4998.
- Lee, S.-g., Ping, W., Ginsburg, B., Catanzaro, B., & Yoon, S. (2023). BigV-GAN: A universal neural vocoder with large-scale training. *The Eleventh International Conference on Learning Representations*. [https://openreview.net/forum?id=iTtGCMDEzS\\_](https://openreview.net/forum?id=iTtGCMDEzS_)
- Liakin, D., Cardoso, W., & Liakina, N. (2017). The pedagogical use of mobile speech synthesis (TTS): Focus on French liaison. *Computer Assisted Language Learning*, 30(3-4), 325–342. <https://doi.org/10.1080/09588221.2017.1312463>
- Lim, Y. (2022). Using text-to-speech technology for integrative listening tasks. *The Institute for Education and Research Gyeongin National University of Education*, 42(s), 39–52. <https://doi.org/10.25020/je.2022.42.s.39>

- Littell, P., Joanis, E., Pine, A., Tessier, M., Huggins Daines, D., & Torkornoo, D. (2022). ReadAlong studio: Practical zero-shot text-speech alignment for indigenous language audiobooks. In M. Melero, S. Sakti, & C. Soria (Eds.), *Proceedings of the 1st annual meeting of the elra/isca special interest group on under-resourced languages* (pp. 23–32). European Language Resources Association. <https://aclanthology.org/2022.sigul-1.4>
- Littell, P., Stewart, D., Davis, F., Pine, A., & Kuhn, R. (2024). Gramble: A tabular programming language for collaborative linguistic modeling. *Proceedings of Interspeech*.
- Lo, C.-C., Fu, S.-W., Huang, W.-C., Wang, X., Yamagishi, J., Tsao, Y., & Wang, H.-M. (2019). MOSNet: Deep learning-based objective assessment for voice conversion. *Proc. Interspeech 2019*, 1541–1545. <https://doi.org/10.21437/Interspeech.2019-2003>
- Lu, Y., Littell, P., & Rice, K. (2024). Empowering Oneida Language Revitalization: Development of An Oneida Verb Conjugator. *Proceedings of Interspeech*.
- Lu, Y.-J., Chang, X., Li, C., Zhang, W., Cornell, S., Ni, Z., Masuyama, Y., Yan, B., Scheibler, R., Wang, Z.-Q., Tsao, Y., Qian, Y., & Watanabe, S. (2022). ESPnet-SE++: Speech Enhancement for Robust Speech Recognition, Translation, and Understanding. *Proceedings of Interspeech*, 5458–5462.
- Lux, F., Koch, J., Schweitzer, A., & Vu, N. T. (2021). The IMS Toucan system for the Blizzard Challenge 2021. *Proc. Blizzard Challenge Workshop*.
- Lux, F., & Vu, T. (2022). Language-Agnostic Meta-Learning for Low-Resource Text-to-Speech with Articulatory Features. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 6858–6868. <https://doi.org/10.18653/v1/2022.acl-long.472>
- Maniati, G., Ellinas, N., Markopoulos, K., Vamvoukakis, G., Sung, J. S., Park, H., Chalamandaris, A., & Tsiakoulis, P. (2021). Cross-lingual low resource speaker adaptation using phonological features. *Interspeech*, 1594–1598. <https://doi.org/10.21437/Interspeech.2021-327>
- Medin, D. L., & Bang, M. (2014). Who’s Asking? Native Science, Western Science and Science Education. *Science Education*.
- Montler, T. (2018). *SENĆOFEN: A dictionary of the Saanich language*. University of Washington Press.

- Mortensen, D. R., Littell, P., Bharadwaj, A., Goyal, K., Dyer, C., & Levin, L. (2016). PanPhon: A resource for mapping IPA segments to articulatory feature vectors. *Proc. 26th Int. Conf. Computational Linguistics: Tech. Paper*, 3475–3484.
- Mosby, I. (2013). Administering colonial science: Nutrition research and human biomedical experimentation in aboriginal communities and residential schools, 1942–1952. *Histoire sociale/Social history*, 46(1), 145–172.
- Nishihara, M., Wells, D., Richmond, K., & Pine, A. (In Submission). Low-dimensional Style Token Control for Hyperarticulated Speech Synthesis. *Proc. INTERSPEECH 2024*.
- NVIDIA. (2023). TTS Vocoder Hifigan Version 1.0.0rc1 [Online; accessed 24-November-2023]. [https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/tts\\_hifigan](https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/tts_hifigan)
- OpenAI. (2023). openai/whisper-base pretrained model [Online; accessed 28-November-2023]. <https://huggingface.co/openai/whisper-base>
- Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). Fairseq: A fast, extensible toolkit for sequence modeling. *Proceedings of NAACL-HLT 2019: Demonstrations*.
- Patton, B., Agiomyrgiannakis, Y., Terry, M., Wilson, K., Saurous, R. A., & Sculley, D. (2016). AutoMOS: Learning a non-intrusive assessor of naturalness-of-speech. *NIPS 2016 End-to-end Learning for Speech and Audio Processing Workshop*.
- Perkins, M. (2020). 3 inferring prescriptivism: Considerations inspired by Hobongan and minority language documentation. In D. Chapman & J. D. Rawlins (Eds.), *Values, ideologies and identity* (pp. 32–45). Multilingual Matters. <https://doi.org/doi:10.21832/9781788928380-004>
- Pine, A., Huggins-Daines, D., Joanis, E., Littell, P., Tessier, M., Torkornoo, D., Knowles, R., Kuhn, R., & Lothian, D. (2023). ReadAlong studio web interface for digital interactive storytelling. In E. Kochmar, J. Burstein, A. Horbach, R. Laarmann-Quante, N. Madnani, A. Tack, V. Yaneva, Z. Yuan, & T. Zesch (Eds.), *Proceedings of the 18th workshop on innovative use of nlp for building educational applications (bea 2023)* (pp. 163–172). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2023.bea-1.14>
- Pine, A., Littell, P., Joanis, E., Huggins-Daines, D., Cox, C., Davis, F., Antonio Santos, E., Srikanth, S., Torkornoo, D., & Yu, S. (2022).  $G_i2P_i$  rule-based, index-preserving grapheme-to-phoneme transformations.

- Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, 52–60.
- Pine, A., & Turin, M. (2017). Language revitalization. <https://doi.org/10.1093/acrefore/9780199384655.013.8>
- Pine, A., Wells, D., Brinklow, N., Littell, P., & Richmond, K. (2022). Requirements and motivations of low-resource speech synthesis for language revitalization. *Proc. 60th Annu. Meeting Association Computational Linguistics (Volume 1: Long Papers)*, 7346–7359. <https://doi.org/10.18653/v1/2022.acl-long.507>
- Polyak, A., Adi, Y., Copet, J., Kharitonov, E., Lakhotia, K., Hsu, W.-N., Mohamed, A., & Dupoux, E. (2021). Speech Resynthesis from Discrete Disentangled Self-Supervised Representations. *Proc. Interspeech 2021*, 3615–3619. <https://doi.org/10.21437/Interspeech.2021-475>
- Pratap, V., Tjandra, A., Shi, B., Tomasello, P., Babu, A., Kundu, S., Elkahky, A., Ni, Z., Vyas, A., Fazel-Zarandi, M., Baevski, A., Adi, Y., Zhang, X., Hsu, W.-N., Conneau, A., & Auli, M. (2024). Scaling Speech Technology to 1,000+ Languages. *Journal of Machine Learning Research*, 25(97), 1–52. <http://jmlr.org/papers/v25/23-1318.html>
- Qamhan, M. A., Alotaibi, Y. A., Seddiq, Y. M., Meftah, A. H., & Selouani, S. A. (2021). Sequence-to-sequence acoustic-to-phonetic conversion using spectrograms and deep learning. *IEEE Access*, 9, 80209–80220. <https://doi.org/10.1109/ACCESS.2021.3083972>
- Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. *Proceedings of the 40th International Conference on Machine Learning*.
- Ravanelli, M., Parcollet, T., Plantinga, P., Rouhe, A., Cornell, S., Lugosch, L., Subakan, C., Dawalatabad, N., Heba, A., Zhong, J., Chou, J.-C., Yeh, S.-L., Fu, S.-W., Liao, C.-F., Rastorgueva, E., Grondin, F., Aris, W., Na, H., Gao, Y., . . . Bengio, Y. (2021). SpeechBrain: A general-purpose speech toolkit [arXiv:2106.04624].
- Ren, Y., Hu, C., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T.-Y. (2021). FastSpeech 2: Fast and high-quality end-to-end text to speech. *Int. Conf. Learning Representations*.
- Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T.-Y. (2019). FastSpeech: Fast, robust and controllable text to speech. *Advances in Neural Information Processing Systems*, 32.

- Rix, A., Beerends, J., Hollier, M., & Hekstra, A. (2001). Perceptual evaluation of speech quality (PESQ) - a new method for speech quality assessment of telephone networks and codecs. *Proc. ICASSP*, 2, 749–752.
- Saunders, M. (2008). Instructions for implementing a new language "voice" for Speak on the XO [Online; accessed 17-November-2023]. [https://wiki.laptop.org/go/Instructions\\_for\\_implementing\\_a\\_new\\_language\\_%22voice%22\\_for\\_Speak\\_on\\_the\\_XO](https://wiki.laptop.org/go/Instructions_for_implementing_a_new_language_%22voice%22_for_Speak_on_the_XO)
- SchemaStore. (2023). JSON Schema Store [Online; accessed 17-November-2023]. <https://www.schemastore.org/json/>
- Schnarch, B. (2004). Ownership, control, access, and possession (OCAP) or self-determination applied to research: A critical analysis of contemporary First Nations research and some options for First Nations communities. *International Journal of Indigenous Health*, 1(1), 80–95.
- Schröder, M., & Trouvain, J. (2003). The German text-to-speech synthesis system MARY: A tool for research, development and teaching. *International Journal of Speech Technology*, 6, 365–377.
- Schwartz, L. (2022). Primum Non Nocere: Before working with Indigenous data, the ACL must confront ongoing colonialism. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 2: Short papers)* (pp. 724–731). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.acl-short.82>
- Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerry-Ryan, R. J., Saurous, R. A., Agiomyriannakis, Y., & Wu, Y. (2018). Natural TTS synthesis by conditioning WaveNet on MEL spectrogram predictions  
Comment: Accepted to ICASSP 2018.
- Smith, L. T. (2023). *Decolonizing methodologies (3rd ed.)* Bloomsbury Academic.
- SpeechBain. (2023). HyperPyYAML [Online; accessed 17-November-2023]. <https://github.com/speechbrain/HyperPyYAML>
- Staib, M., Teh, T. H., Torresquintero, A., Mohan, D. S. R., Foglianti, L., Lenain, R., & Gao, J. (2020a). Phonological features for 0-shot multilingual speech synthesis. *Proc. Interspeech*, 2942–2946. <https://doi.org/10.21437/Interspeech.2020-1821>
- Staib, M., Teh, T. H., Torresquintero, A., Mohan, D. S. R., Foglianti, L., Lenain, R., & Gao, J. (2020b). Phonological Features for 0-Shot Mul-

- tilingual Speech Synthesis. *Proc. Interspeech 2020*, 2942–2946. <https://doi.org/10.21437/Interspeech.2020-1821>
- Taal, C. H., Hendriks, R. C., Heusdens, R., & Jensen, J. (2010). A short-time objective intelligibility measure for time-frequency weighted noisy speech. *Proc. ICASSP*, 4214–4217.
- Taylor, J., & Richmond, K. (2021). Confidence intervals for ASR-based TTS evaluation. *Proc. Interspeech*, 2791–2795.
- Taylor, P., Black, A. W., & Caley, R. (1998). The architecture of the Festival speech synthesis system. *The Third ESCA/COCOSDA Workshop (ETRW) on Speech Synthesis*, 147–152.
- Tesfaye Biru, E., Tofik Mohammed, Y., Tofu, D., Cooper, E., & Hirschberg, J. (2019). Subset Selection, Adaptation, Gemination and Prosody Prediction for Amharic Text-to-Speech Synthesis. *Proc. 10th ISCA Workshop on Speech Synthesis (SSW 10)*, 205–210. <https://doi.org/10.21437/SSW.2019-37>
- UnBQ. (2023). Our Mission [Online; accessed 17-November-2023]. <https://www.bluequills.ca/About/Mission>
- Valentini-Botinhao, C., & King, S. (2021). Detection and analysis of attention errors in sequence-to-sequence text-to-speech. *Proc. Interspeech*, 2746–2750. <https://doi.org/10.21437/Interspeech.2021-286>
- Valin, J.-M. (2018). A hybrid dsp/deep learning approach to real-time full-band speech enhancement. *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, 1–5. <https://doi.org/10.1109/MMSP.2018.8547084>
- Veaux, C., Yamagishi, J., & King, S. (2013). The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. *2013 Proc. Int. Conf. Oriental COCOSDA*, 1–4. <https://doi.org/10.1109/ICSDA.2013.6709856>
- Veysov, A., & Voronin, D. (2022). One voice detector to rule them all [Online; accessed 17-November-2023]. <https://thegradient.pub/one-voice-detector-to-rule-them-all/>
- Wang, C., Hsu, W.-N., Adi, Y., Polyak, A., Lee, A., Chen, P.-J., Gu, J., & Pino, J. (2021). FAIRSEQ  $S^2$ : A scalable and integrable speech synthesis toolkit. *Proc. 2021 Conf. Empirical Methods in Natural Language Processing: System Demonstrations*, 143–152. <https://doi.org/10.18653/v1/2021.emnlp-demo.17>
- Wang, W. S.-Y. (1967). Phonological Features of Tone. *International Journal of American Linguistics*, 33(2), 93–105.

- Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., & Saurous, R. A. (2017). Tacotron: Towards end-to-end speech synthesis. *Proc. Interspeech*, 4006–4010.
- Wang, Y., Stanton, D., Zhang, Y., Ryan, R.-S., Battenberg, E., Shor, J., Xiao, Y., Jia, Y., Ren, F., & Saurous, R. A. (2018). Style tokens: Un-supervised style modeling, control and transfer in end-to-end speech synthesis. *Int. Conf. Machine Learning*, 5180–5189.
- Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Enrique Yalta Soplin, N., Heymann, J., Wiesner, M., Chen, N., Renduchintala, A., & Ochiai, T. (2018). ESPnet: End-to-End Speech Processing Toolkit. *Proc. Interspeech 2018*, 2207–2211. <https://doi.org/10.21437/Interspeech.2018-1456>
- Wells, D., & Richmond, K. (2021). Cross-lingual transfer of phonological features for low-resource speech synthesis. *Proc. SSW*, 160–165.
- Wester, M., Valentini-Botinhao, C., & Henter, G. E. (2015). Are we using enough listeners? no! an empirically-supported critique of interspeech 2014 tts evaluations. *Interspeech 2015*, 3476–3480.
- Wiseman, J. (2021). Py-webrtcvad [Online; accessed 17-November-2023]. <https://github.com/wiseman/py-webrtcvad>
- WSÁNEĆ School Board. (2023). SENĆOTEN Language [Online; accessed 17-November-2023]. <https://wsanecschoollboard.ca/sencoten-language/>
- Xu, Z., Zhang, S., Wang, X., Zhang, J., Wei, W., He, L., & Zhao, S. (2023). MuLanTTS: The Microsoft speech synthesis system for Blizzard challenge 2023. *arXiv preprint arXiv:2309.02743*.
- Yamagishi, J., Veaux, C., & MacDonald, K. (2019). CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit.
- Zen, H., Dang, V., Clark, R., Zhang, Y., Weiss, R. J., Jia, Y., Chen, Z., & Wu, Y. (2019). LibriTTS: A corpus derived from Librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.
- Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A. W., & Tokuda, K. (2007). The HMM-based speech synthesis system (HTS) version 2.0. *SSW*, 6, 294–299.
- Zhang, H., Yuan, T., Chen, J., Li, X., Zheng, R., Huang, Y., Chen, X., Gong, E., Chen, Z., Hu, X., Yu, D., Ma, Y., & Huang, L. (2022). Paddlespeech: An easy-to-use all-in-one speech toolkit. *Proceedings of the 2022 Conference of the North American Chapter of the Associ-*



*ation for Computational Linguistics: Human Language Technologies: Demonstrations.*

Zhang, Y., Weiss, R. J., Zen, H., Wu, Y., Chen, Z., Skerry-Ryan, R., Jia, Y., Rosenberg, A., & Ramabhadran, B. (2019). Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning. *Proc. Interspeech*, 2080–2084. <https://doi.org/10.21437/Interspeech.2019-2668>

## Appendix A. Summary of Brainstorming Questions

The responses to the questions listed above are summarized in the following paragraphs.

**What does it sound like?** The summary of the discussion for this question was that ‘voice quality’ was important, with a particular focus on speech that “captures the song of the language” (i.e. prosody). Participants imagined both clear pronunciation (pedagogical register, see (§4)) as well as a normal conversational register. Variation and relatability of the speakers was cited as a key desire by participants in multiple groups. Specifically, participants described wanting to have control over speaker variation with respect to gender, age, and dialect. Some participants also requested that there be a method to adapt the speech to sound like the user (i.e. speaker adaptation). These are techniques that some participants felt could be useful for giving learners more relatable speech on which to model their pronunciation.

**Where are the people using the tool? What are they doing with it?** This question elicited a wide variety of responses; participants could imagine the tool being used just about anywhere. Although discussions often began with a focus on its use for their students in the classroom, participant responses would often drift towards the use of the tool outside of the classroom to support in-classroom work. Specifically, how it could free teachers up from spending time creating resources for students and allow them to spend more time on instruction and individualized student support. In order to support students outside the classroom, offline use was also described as desirable, as well as support on a wide variety of devices.

**Imagine somebody misusing the technology, what are some things that could go wrong?** This question prompted very animated discussion, as there were many opportunities for unethical applications of the technology that were imagined by participants. Concerns included the production of offensive words, either due to pronunciation errors by the model or offensive inputs, ‘deep fakes’ where a respected Elder’s voice might be used to say something false or insulting, culturally insensitive applications, or the ability to produce speech from a speaker who has since passed away. There were also concerns, as there are with many reference tools, that the standard that happens to have been recorded could set a ‘standard dialect’, which

might result in harmful dialectal bias. There was general consensus among participants that once a speaker had passed away, the voice should no longer sound like them and instead should be anonymized.

Many of these issues can be mitigated through proper restrictions around access and control of the technology. The majority of the applications imagined did not involve general-purpose synthesis where arbitrary and potentially offensive text could be used as the input. There was consensus that for closed systems, such as supplementing the SENĆOTEN dictionary or Kawennón:nis, an identifiable voice is acceptable. However, if open input applications are developed for this project, it was decided that an identifiable voice should never be used.

## Appendix B. Summary of Selected TTS Toolkits

This section summarizes and briefly introduces selected publicly available neural speech synthesis toolkits, which are later compared in Appendix C with respect to features required for our context. We are not including concatenative toolkits such as Festival (P. Taylor et al., 1998), nor are we including statistical parametric speech synthesis (SPSS) systems like HTS (Zen et al., 2007) or MaryTTS (Schröder & Trouvain, 2003) despite their popularity in some low-data scenarios (Harrigan et al., 2019; James et al., 2020). This is because SPSS systems lack the naturalness of neural speech synthesis and concatenative systems require data to be from a single speaker, which is a requirement that we believe is too onerous in a context where many speakers are elderly and their time is extremely limited. Additionally, while there continues to be active development in concatenative and statistical parametric speech synthesis toolkits, the widespread popularity of neural methods means that the tooling around neural-based toolkits is often more up-to-date. Moreover, whereas concatenative and SPSS techniques have historically been the only options for limited-data TTS, some neural methods are now efficient enough to be used in limited-data scenarios (§4.2).

### B.1. Coqui TTS

With over 10 feature prediction (text-to-spectrogram) models, 8 vocoders (spectrogram-to-wav), and 5 end-to-end models implemented, the open-source Coqui TTS toolkit has the widest variety of speech synthesis systems of any publicly available toolkit. The primary focus of Coqui, however, is not research, they are a private business which markets its speech synthesis systems

as a way to replace voice actors with artificially generated voices (Coqui, 2023a). In December 2023 the Coqui company announced it was shutting down, although Coqui TTS continues to exist as an open-source repository.

### *B.2. ESPnet*

The End-to-End Speech Processing toolkit (aka ESPnet) (Hayashi et al., 2021; Watanabe et al., 2018) is an open-source platform for performing a variety of tasks including automatic speech recognition (ASR), speech synthesis, voice conversion, speech translation, and many other tasks. The first version released was primarily focused on ASR, implemented in PyTorch and Chainer, and heavily inspired by Kaldi. The second version, which at the time of writing is still under development, has dropped the requirements on Chainer and Kaldi and includes a host of other improvements including multi-node/multi-GPU training.

### *B.3. FAIRSEQ S<sup>2</sup>*

The FAIRSEQ S<sup>2</sup> toolkit (C. Wang et al., 2021) adds speech synthesis support to the larger Facebook AI engineered FAIRSEQ toolkit (Ott et al., 2019) for sequence modelling.

Their paper is also the only paper of any of the toolkits here that compares itself to existing toolkits. We include all of the toolkits that they include in their survey except for OpenSeq2Seq (Kuchaiev et al., 2018), a TensorFlow-based toolkit developed at NVIDIA that has since been archived in favour of NeMo (B.5).

### *B.4. IMS Toucan*

IMS Toucan is an open-source platform developed at the University of Stuttgart. The system was first introduced in the 2021 Blizzard challenge (Lux et al., 2021) and has a variety of specific considerations for a low-resource context. Unlike the other toolkits reviewed, they propose a single architecture; as of writing this is a modified FastSpeech2 (Ren et al., 2021) system paired with either Avocodo (Bak et al., 2023) or BigVGAN (Lee et al., 2023) as the vocoder.

### *B.5. NeMo*

The NeMo, short for ‘**N**eural **M**odules’, toolkit is an open-source toolkit developed by NVIDIA that supports TTS, ASR, and a variety of natural language processing tasks (machine translation, LLMs etc).

### B.6. *SpeechBrain*

SpeechBrain (Ravanelli et al., 2021) is an open-source toolkit for conducting research and development for a variety of speech processing tasks including speech synthesis, ASR, and speech enhancement, among others.

### B.7. *EveryVoice*

In addition to the existing toolkits described above, we include our preliminary development of a new TTS toolkit, named the EveryVoice TTS toolkit, for comparison. EveryVoice TTS, introduced in §5.2, has been designed to provide a unified speech synthesis toolkit specifically tailored to limited data TTS applications. In the following sections, we show that the EveryVoice TTS toolkit combines features related to low-resource TTS in a way that we believe will support our goal to make a repeatable recipe (§2.4), and is currently unsupported in any one existing open source solution. We are releasing the toolkit in a functional, but early, stage of development to help encourage feature requests and additional requirements from the broader community of low-resource speech synthesis practitioners.

	Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS- Toucan	NeMo	Speech Brain
License	MPL- 2.0	Apache- 2.0	MIT	MIT	Apache- 2.0	Apache- 2.0	Apache- 2.0
TTS only	Yes	No	Yes	No	Yes	No	No
Associated Paper	✗	✓ <sup>1</sup>	✓	✓ <sup>2</sup>	✓ <sup>3</sup>	✓ <sup>4</sup>	✓* <sup>5</sup>

Table B.2: Summary of basic information for existing toolkits. Only foundational paper citations are included as there are many citations associated with some of these toolkits. \* indicates the associated paper only appears as a preprint. (Watanabe et al., 2018)<sup>1</sup>, (C. Wang et al., 2021)<sup>2</sup>, (Lux et al., 2021)<sup>3</sup>, (Kuchaiev et al., 2019)<sup>4</sup>, (Ravanelli et al., 2021)<sup>5</sup>

## Appendix C. Overview of Requirements & Implementation Survey

This section explores some of the core requirements of a TTS toolkit required for inclusion in our ‘repeatable recipe’ for limited-data TTS. We have structured the requirements into three distinct sections: modelling requirements (C.1), data preprocessing requirements (C.2), and developer experience requirements (C.3). For each set of requirements, we will compare and

assess the way that each of the aforementioned toolkits addresses these challenges. All of the findings are summarized for ease of reference in Table C.12 on page 69.

### *C.1. Modelling Requirements*

One of the main requirements of a TTS model architecture for our context is data efficiency. However, other considerations are similarly important such as preventing misuse, and the ability to practically fine-tune and evaluate the system.

#### *C.1.1. Data Efficiency*

Perhaps the most obvious technical requirement of a toolkit for use in a limited data context is whether the model is efficient with respect to how much data it requires, as discussed at length in §4.2. If a toolkit requires many hours of audio to produce intelligible speech it will not be able to be used for low-resource TTS.

**Toolkit Implementations.** While a rigorous comparison of the results of listening tests for each model is outside the scope of this paper, we can infer some basic information about data efficiency based on the models that each toolkit implements. For example, it has been shown that the non-autoregressive FastSpeech2/Fastpitch systems are many times more efficient than Tacotron2 (Pine, Wells, et al., 2022). Every toolkit discussed contains an implementation of a FastSpeech2/Fastpitch based system, meaning that data-efficient TTS is possible in any of the toolkits mentioned.

Toolkits differ widely, however, in terms of the documentation and signposting provided to users about which model to use in a low-resource context. FAIRSEQ S<sup>2</sup>, SpeechBrain, and ESPnet do not provide any suggestions about which model to use for low-resource settings. NeMo’s excellent ‘primer on TTS’ (Harper et al., 2023b) includes descriptions of the differences between autoregressive (AR) Tacotron2 and non-AR Fastpitch systems, but does not mention anything related to data efficiency. Similarly, Coqui TTS’s FAQ section (Coqui GmbH, 2021) has an answer to the question ‘How should I choose the right model?’ but it suggests (data inefficient) Tacotron as the first model to try and low-resource considerations are not mentioned.

IMS Toucan is different from the other toolkits in that it explicitly references low-resource TTS as its target use case. It uses an adapted FastSpeech2

model and provides a pre-trained vocoder with instructions for fine-tuning to a new low-resource language.

The EveryVoice TTS toolkit approach to data efficiency is similar to IMS Toucan in that we only include a single model which is specifically curated for a low-resource context. Like IMS Toucan we implement a two-stage system comprised of a feature prediction network based on FastSpeech2 (Ren et al., 2021). The EveryVoice TTS vocoder is based on iSTFT-Net (Kaneko et al., 2022). While so-called end-to-end systems like VITS (Kim et al., 2021) are capable of producing impressively natural speech, two-stage systems are currently more data efficient in low-resource applications since the vocoder is trained without text, lessening the burden of requiring transcribed data.

The feature prediction network we implemented is based on the adapted FastSpeech2 model from (Pine, Wells, et al., 2022), which was demonstrated to be particularly well-suited to low-resource TTS; in listening test evaluations, the adapted FastSpeech2 model trained on only 1 hour of data was shown to have results that were not significantly different from a Tacotron2 baseline trained on 10 hours of data.

Coqui TTS	ESPnet	Every Voice	Fairseq	IMS-Toucan	NeMo	Speech Brain
✓*	✓*	✓	✓*	✓	✓*	✓*

Table C.3: List of toolkits that contain a recommended data-efficient strategy for low-resource TTS. \* indicates that data efficient models are implemented in the toolkit, but there is no documentation present that recommends a particular model for use in low-resource contexts.

### C.1.2. Preventing Misuse

As discussed in Appendix A, misuse could result from the unauthorized use of speech data (i.e. data theft), or production of offensive language that would bring harm to the speaker or to people deceived by the synthesized audio.

There are a variety of strategies for mitigating these issues, from educating practitioners about the importance of data rights and intellectual property, to more technical solutions such as applying a ‘watermark’ to synthesized audio (G. Chen et al., 2023) that allows synthesized audio to be identified with a signature.

**Toolkit Implementations.** None of the strategies available are adequate in preventing all forms of misuse. As of the time of writing, we were unable to find any documentation of strategies to prevent misuse in ESPnet, IMS Toucan, FAIRSEQ S<sup>2</sup>, NeMo, or SpeechBrain. FAIRSEQ S<sup>2</sup> has released pre-trained checkpoints for TTS systems trained on low-resource language data without the permission of the language communities or speakers in question (see §2.3) which has the potential to enable misuse, not prevent it. Coqui TTS has a public discussion (Coqui, 2021b) about the topic, which has garnered quite a bit of interest and promising ideas, but to our knowledge, none of the ideas or strategies have yet been implemented in Coqui TTS. However, Coqui TTS also promotes the use of the problematic ‘Massively Multilingual Speech’ TTS models to be used with Coqui TTS.

Closed-source speech synthesis systems such as Microsoft’s Neural TTS have adopted some stricter regulations for their TTS software, such as limiting access, enforcing a Code of Conduct (Farley & Microsoft, 2023), and using watermarking technology, but closed-source systems are beyond the scope of this paper.

The EveryVoice TTS toolkit addresses potential misuse by requiring a declaration of data permissions before training any model with EveryVoice TTS; that is, users must declare that they have permission to use their data and also include some personal information such as name and contact email address. These answers, along with all other hyperparameters, are stored by default in the model checkpoints produced during training and are required in order to run the model. In this way, we intend to invite users to consider their relationship to the data in question and to confirm that they have appropriate permission. The documentation links the user to resources explaining why this is important, and the command line interface for creating new datasets prompts the user with this information as well.

Unfortunately, none of these methods are adequate in eliminating the possibility of misuse. For example, a user could fill out the data permission questions incorrectly. However, we hope that this measure will indicate to users that they are potentially doing something harmful and prevent cases of inadvertent harm while also providing hurdles for bad actors who might try to misuse EveryVoice TTS.

We encourage future research to investigate methods such as incorporating low-intervention watermark systems that would identify synthesized audio and be embedded in all released pre-trained checkpoints released by publicly available toolkits.



### C.1.3. Practical Finetuning

The ability of a different organization to fine-tune a pre-trained model has practical consequences for the feasibility and accessibility of training a TTS model for a new language. With the current status quo, training of multilingual neural TTS systems is, in practice, limited to larger organizations with access to GPU clusters and large multilingual speech corpora. Beyond hardware limitations, there are modelling constraints that limit the ability to perform cross-lingual fine-tuning such as unifying the input space as discussed at length in §4.3.

**Toolkit Implementations.** Feature-prediction and end-to-end TTS models for Coqui TTS, ESPnet, and SpeechBrain all one-hot encode inputs using either characters or phonemes, which means that the input space varies between languages and causes complexity for fine-tuning (§4.3).

FAIRSEQ S<sup>2</sup>, and NeMo both implement feature prediction networks trained with self-supervised learning representations (SSL) of audio as input (Polyak et al., 2021) instead of text, allowing these models to be fine-tuned without mapping the input space of the target language to the donor language model.

Alternatively, IMS Toucan and EveryVoice TTS both implement feature prediction networks trained with phonological feature vector inputs (Lux & Vu, 2022; Staib et al., 2020b). The implementations differ in that IMS Toucan implements a custom mapping from IPA characters to phonological features, and EveryVoice TTS uses the libraries G<sub>i</sub>2P<sub>i</sub> (Pine, Littell, et al., 2022) and Panphon (Mortensen et al., 2016).

	Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS-Toucan	NeMo	Speech Brain
Phonological Features	✗	✗	✓	✗	✓	✗	✗
SSL Features	✗	✗	✗	✓	✗	✓	✗
Public Checkpoints	✓	✓	✓	✓	✓	✓	✓

Table C.4: List of toolkits that contain features to support fine-tuning. ‘Phonological Features’ specifies whether the toolkit allows inputs to be encoded as multi-hot phonological feature vectors instead of one-hot character or phoneme vectors. ‘SSL Features’ specifies whether the toolkit supports using self-supervised units as inputs to their feature prediction networks.

#### C.1.4. Evaluation

As discussed at length in §4.4, evaluation represents a significant challenge for many low-resource speech synthesis projects, including ours. Listening tests involve time and expertise that are both in short supply, which means that there are practical limitations on the number of models we are capable of evaluating. The inclusion of one or more language-independent methods for performing evaluation is therefore seen as a useful and important feature in our context, where automatic methods could help triage models and decide which ones should be evaluated with listening tests, even if automatic metrics are not reliable replacements for human evaluation.

**Toolkit Implementations.** At the time of writing, Coqui TTS, IMS Toucan, EveryVoice TTS, and SpeechBrain do not provide any methods for evaluating synthesized speech.

NeMo provides a notebook (Harper et al., 2023a) describing how to calculate Mel Cepstral Distortion (MCD) with dynamic time warping. FAIRSEQ S<sup>2</sup> provides automatic evaluation by implementing calculations of MCD as well as Mel Spectral Distortion, character and word error rates (CER/WER) calculated by pre-trained ASR systems, and a variety of F0 error calculations including gross pitch error (GPE) and F0 frame error (FFE) (Chu & Alwan, 2009). It should be noted that while the MCD and FFE metrics are language-independent, calculating word and character error rates using an ASR model requires also having a pre-trained ASR system in the target language which is unlikely in many low-resource contexts.

ESPnet provides five strategies for evaluating synthesized speech; MCD, root mean-squared error (RMSE) of log-F0 estimations, CER/WER, and conditional Fréchet DeepSpeech Distance (cFDS) (Binkowski et al., 2020). They also measure speaker similarity by implementing an automatic calculation of the mean speaker embedding cosine similarity between the reference and synthesized audio using a speaker verification model (Hsieh et al., 2023).

Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS-Toucan	NeMo	Speech Brain
✗	✓	✗	✓	✗	✓	✗

Table C.5: List of toolkits that contain a strategy for performing automatic evaluation.

## *C.2. Data Preparation & Preprocessing Requirements*

Data preparation and preprocessing are umbrella terms to describe the various steps that are needed to gather and transform data into the necessary formats required for training. Many TTS research datasets have often already been preprocessed such that all audio files have a consistent sampling rate and bit depth, some text normalization has been applied, and the dataset is almost always aligned at the utterance level.

In order for us to build a repeatable recipe for other limited-data TTS projects to follow, we need to ensure that the toolkit we use incorporates adequate preprocessing functionality to support data that is potentially less consistent than typical TTS research datasets. Specifically, we require a pipeline for handling audio data that is able to remove noise, detect and remove unnecessary silence, perform utterance-level segmentation of long-form audio, and detect outliers.

### *C.2.1. Noise Removal*

As described in §3.3, making high-quality recordings is a significant logistical challenge. In some cases, recordings have been produced without the ideal equipment or recording conditions for TTS. In these cases, removing noise is an important step in preparing data for training (Xu et al., 2023).

**Toolkit Implementations.** As of writing, we could not find any method for performing denoising within Coqui TTS, IMS Toucan, or NeMo. In a GitHub Discussion (Coqui, 2021a) it appears that Coqui TTS recommends the use of 3rd party software RNNoise (Valin, 2018). IMS Toucan and NeMo do not appear to suggest a noise removal strategy.

SpeechBrain and ESPnet (Y.-J. Lu et al., 2022) both provide speech enhancement pipelines for separating a speech signal from ambient noise and reverberation. These pipelines are not connected with TTS recipes.

FAIRSEQ S<sup>2</sup> provides a dedicated denoising strategy following Défossez et al. (2020) built into their TTS pipeline.

EveryVoice TTS does not yet provide any solution for denoising. Like Coqui TTS, we currently recommend using RNNoise (Valin, 2018) while an integrated solution is being investigated.

### *C.2.2. Silence Removal*

Silence can vary widely within a TTS dataset and this variability can adversely affect TTS models, particularly when jointly modelling text/audio

Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS- Toucan	NeMo	Speech Brain
✗	✓*	✗	✓	✗	✗	✓*

Table C.6: List of toolkits that contain a strategy for removing noise from speech utterances. \* indicates that the method is available but not documented in connection with the TTS pipeline.

alignment. A useful step for data preparation is to trim leading and trailing silences around speech regions to a consistent, relatively short duration. This may be applied directly to audio files before training, or during acoustic feature extraction, in which case any derived Mel spectrogram features might not match original audio durations exactly. This process may also be referred to as voice activity detection (VAD).

**Toolkit Implementations.** ESPnet includes a utility script to trim surrounding silence as detected by a simple power threshold relative to the maximum signal amplitude in each utterance. Additional artificial silence can be added to pad the beginning and end of each utterance to retain a minimum amount of silence (by default 0.01 s). This utility is included as part of many of the recipes in provided in ESPnet for preparing various TTS corpora. Coqui TTS includes a similar utility; both toolkits trim silence while loading original audio files, before extracting acoustic features. FAIRSEQ S<sup>2</sup> uses the open-source WebRTC VAD (Wiseman, 2021), which uses a GMM to model voice probabilities per frame based on energy levels in multiple frequency bands. Leading and trailing silences are removed completely, while utterance-internal silences longer than 300 ms are replaced by 300 ms of artificial silence. EveryVoice TTS allows users to apply the SoX `silence` effect to with defaults for stripping leading, trailing, and utterance-internal silence based on an energy threshold.

NeMo also provides a simple energy threshold-based VAD solution, alongside a pre-trained CNN VAD model (Jia et al., 2021). The option to trim silences during dataset preparation is mentioned briefly in the documentation for NeMo’s TTS configuration files, but only for setting the simple energy threshold. IMS Toucan has the option to run a pre-trained Silero neural VAD model during dataset preparation (Veysov & Voronin, 2022). SpeechBrain also provides a general-purpose interface to a pre-trained CRDNN model, but it does not seem to be applied in any provided TTS training recipes. In-

stead, data preparation in the provided FastSpeech2 recipe trims surrounding silences based on forced alignments.

Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS- Toucan	NeMo	Speech Brain
✓	✓	✓	✓	✓*	✓	✓*

Table C.7: List of toolkits that contain a strategy for trimming surrounding silence from speech utterances. \* indicates that the method is available but not documented in connection with the TTS pipeline.

### C.2.3. Audio Segmentation

As previously mentioned (§3.3.2), we recommend the use of tools like Mozilla’s Common Voice or Speech Recorder (Draxler & Jänsch, 2004) for recording new audio, since the software prompts users for single utterances thus resulting in utterance-aligned data. However, it is often the case that parallel text and audio data already exist for a language but in a longer-form format. Being able to align long-form audio into shorter segments is thus one important step in our repeatable recipe.

**Toolkit Implementations.** As of writing, Coqui TTS, and IMS Toucan do not provide any built-in support for performing long-form audio segmentation.

Implementations vary in terms of which pre-trained models are provided by default, but ESPnet, and SpeechBrain both provide recipes for CTC Segmentation (Kürzinger et al., 2020) which is a powerful method for long-form audio utterance segmentation. However, as of writing, the recipes are not directly linked to speech synthesis recipes, leaving it to the expertise of the user to be aware that this is a method that could be used for segmenting their data. Similarly, FAIRSEQ S<sup>2</sup>’s parent module FAIRSEQ provides steps for performing CTC Segmentation, but it does not appear to be linked in the general TTS recipe.

NeMo’s ‘Dataset Creation Tool’ also provides support for CTC Segmentation which defines a more documented pathway for the user to apply the CTC Segmentation method when creating and preprocessing their dataset.

The EveryVoice TTS Toolkit also provides support for segmenting data using the CTC Segmentation method. The method is built into the main

command line interface with documentation in the help message of the command line and in the user documentation as to why CTC Segmentation might be needed.

Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS- Toucan	NeMo	Speech Brain
✗	✓*	✓	✓*	✗	✓	✓*

Table C.8: List of toolkits that contain a strategy for performing long-form audio segmentation into text/audio aligned utterances. \* indicates that the method is available but not documented in connection with the TTS pipeline.

#### C.2.4. Outlier Detection & Dataset Selection

When building speech synthesis systems, more data is not always better, particularly when the data in question was not purposefully recorded for TTS under controlled conditions (Gallegos et al., 2020; Tesfaye Biru et al., 2019). Choosing a subset of the data to use can be achieved by detecting and removing or filtering outliers when preparing data.

**Toolkit Implementations.** As of the time of writing, SpeechBrain, ESPnet, and NeMo do not appear to have any module specifically dedicated to detecting and removing outliers.

Coqui TTS has a collection of notebooks (Coqui, 2023b) intended to perform ‘dataset analysis’ for the purpose of finding outliers and determining phoneme coverage. The notebooks appear to perform basic sanity checks (i.e. missing files or duplicate data) and create plots for other information, but Coqui TTS does not appear to have a documented method for incorporating these notebooks (or the results of running them) into a TTS recipe.

IMS Toucan provides a ‘scorer’ which passes potentially inconsistent data through a pre-trained model. The script then displays the top 20 samples with the highest loss (as calculated by the pre-trained model) and removes the top 5 samples. FAIRSEQ S<sup>2</sup> also implements a form of outlier filtering using two separate methods; one based on signal-to-noise ratio (SNR) and the other based on Character Error Rate from a pre-trained ASR system (C. Wang et al., 2021). While the former could be applied in a zero-shot manner, the latter would require a pretrained ASR model, ideally trained in the same language as the data in question.

EveryVoice TTS provides a module for detecting and removing outliers by detecting clipping in sample audio as well as statistical outliers with respect to duration, F0 estimation, and energy and speaking rate (calculated as both words/second and characters/second). We have found that extreme outliers with respect to speaking rate are often due to inaccurate alignments.

Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS Toucan	NeMo	Speech Brain
✓	✗	✓	✓	✓	✗	✗

Table C.9: List of toolkits that contain a strategy for performing outlier filtering on potentially noisy datasets.

### C.3. Developer Experience Requirements

One of our goals is to reduce the friction involved in building and training speech synthesis models. We are building our repeatable recipe for the technical user, that is, a user who is familiar with the command line, but not necessarily a TTS expert, or even a machine-learning expert. We separate these considerations into two broad categories; toolkit support for configuring projects and models (C.3.1), and the general availability of documentation and developer support (C.3.2).

#### C.3.1. Assisted Project and Model Configuration

One of the challenges of developing neural speech synthesis systems is related to the number of combinatorial possibilities among hyperparameters. Typically, these hyperparameters are configurable in files that are separate to the model code, but TTS toolkits vary in terms of how much assistance is provided to users in creating or editing these files. For our repeatable recipe, we require a toolkit that assists the user in creating configurations for new projects with new languages or datasets.

These configurations should ideally include validation, to help catch errors and misconfigurations prior to model runtime, so that potential issues can be resolved before training. Configurations should also be hierarchical and composable, allowing certain aspects of model configuration to be shared across different configuration files. Additionally, in many cases it is common to build models from multiple datasets and it can be tedious and error-prone to ask users to manually combine multiple datasets together into a single location, ensure the audio is in the same format, and combine the filelists

themselves. A toolkit for our repeatable recipe would ideally be able to accommodate multiple data sources within a single configuration file.

**Toolkit Implementations.** IMS Toucan does not implement hyperparameter configuration using modular YAML or JSON files. Rather, everything is specified in Python files directly. The toolkit is set up to support specific ‘pipelines’ for training specific models which represent particular combinations of hyperparameters defined together with a model and training routine. This approach is different from the other toolkits surveyed which prioritize separation of hyperparameters and models to allow for easier customization.

SpeechBrain maintains a Python package (SpeechBrain, 2023) that supports custom extensions to the data serialization language YAML. Most importantly, the authors extend the use of YAML tags to simplify the creation of new modules, classes, or functions, for example, `model: !new:collection.Counter` would create a new instance of the Counter class from the collection module in Python as a value to the model key in a standard YAML key/value mapping. Hyperparameters loaded from their extended YAML are used to instantiate models without validation, however, leaving it up to the user to determine the cause of potential errors at runtime.

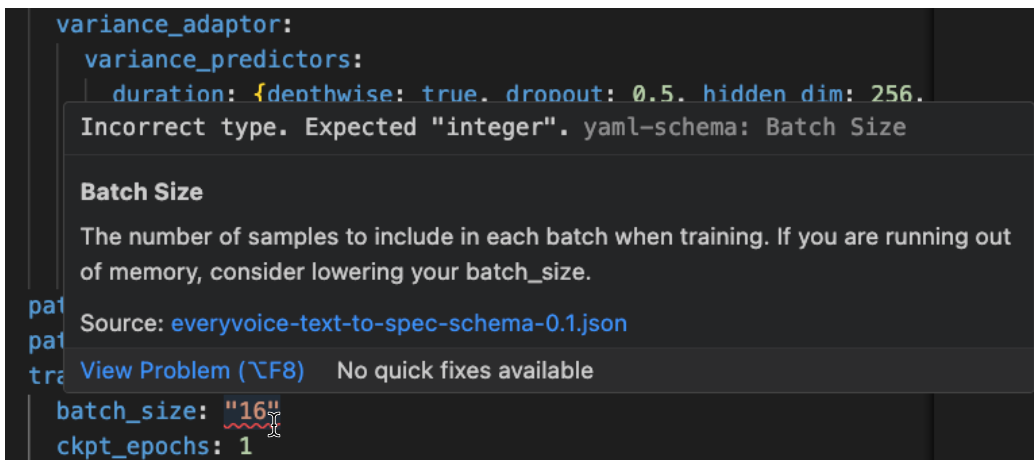
Similar to SpeechBrain, Coqui TTS maintains its own separate Python package for managing hyperparameters (Coqui, 2022). The package, called ‘coqpit’, is a simple but clever implementation for providing meaningful error messages to missing values in a configuration file and allows for defining nested, conditional configurations capable of being overridden easily from command line arguments. Unlike the majority of other toolkits, ‘coqpit’ uses JSON instead of YAML to serialize hyperparameters. A configuration can also be validated by using the provided `check_values` method and `check_argument` function.

NeMo & FAIRSEQ S<sup>2</sup> use Hydra, which is a popular configuration library that combines useful configuration features (such as hierarchical, composable configuration) together with a command-line interface that allows configuration arguments to be overridden in the command line and multiple jobs to be instantiated with a single command. It appears that some modules in ESPnet2 are moving to Hydra, but currently TTS is implemented by loading pure YAML as it is in the original ESPnet. The YAML configurations typically have in-line comments that describe help messages and instructions for



the range and type of possible values, leaving it up to the user to validate their hyperparameter specifications.

Instead of Hydra, EveryVoice TTS uses Pydantic, a widely used data validation library for Python, as well as Typer, which is a separate popular command line interface framework for Python. Beyond basic type-checking, this system allows for many other forms of validation such as checking file or folder paths, or even more complex validations, for example, ensuring that if the vocoder is configured to take 22.05 kHz inputs and produce 44.1 kHz outputs, then the number of upsampling layers (and the kernel sizes of those layers) is configured correctly for 2:1 super-resolution; if not, an early error message is shown describing the problem. Because model configurations are statically-typed, code-completion is also available in code editors that support it, and JSON schemas for each configuration are automatically generated and uploaded to SchemaStore (SchemaStore, 2023), meaning that many popular code editors like VS Code will automatically include syntax highlighting and tab completion when a user is editing a serialized EveryVoice TTS JSON or YAML configuration as seen in Figure C.5.



```
variance_adaptor:
  variance_predictors:
    duration: {depthwise: true, dropout: 0.5, hidden_dim: 256.
Incorrect type. Expected "integer". yaml-schema: Batch Size

Batch Size
The number of samples to include in each batch when training. If you are running out
of memory, consider lowering your batch_size.
Source: everyvoice-text-to-spec-schema-0.1.json
View Problem (⇧F8) No quick fixes available

batch_size: "16"
ckpt_epochs: 1
```

Figure C.5: Screenshot of syntax highlighting in an EveryVoice TTS YAML file showing that when a user passes a string instead of an integer for the ‘batch\_size’ configuration parameter, they are warned in the editor with highlighting. EveryVoice TTS YAML files also include tab completion.

Additionally, EveryVoice TTS allows the user to define multiple sources of data for a single experiment in one configuration file. Each source of data can still be configured independently (e.g. with respect to how much silence

is removed, or how to parse the filelist). The EveryVoice TTS configuration wizard (C.3.2) also supports the user in writing configurations with multiple data sources. Table C.10 summarizes the results across the surveyed toolkits.

	Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS-Toucan	NeMo	Speech Brain
Library	Coqpit	YAML	Pydantic	Hydra	None	Hydra	Hyper-PyYaml
Validation	✓	✗	✓	✗	✗	✗	✗
Hierarchical & Composable	✓	✓	✓	✓	✗	✓	✓
Config Syntax Highlighter	✗	✗	✓	✗	N/A	✗	✗
Multiple Data Sources	✗	✗	✓	✗	✗	✗	✗

Table C.10: Table describing various strategies used by toolkits to assist users in defining configurations for their models. The ‘Library’ field specifies which library is used (if any) to assist with configuration. ‘Hierarchical & Composable’ specifies whether configurations can be shared and nested. ‘Config Syntax highlighter’ specifies whether the serialized format of the configuration (JSON or YAML) highlights errors or provides type hints. ‘Multiple Data Sources’ specifies whether the configuration can handle multiple sources of data or whether datasets have to first be combined by the user.

### C.3.2. Documentation, Guides, & Support

For many applied low-resource TTS projects, in addition to a limited amount of data, there is limited access to speech synthesis expertise and support. Undocumented settings or commands seldom pose issues for teams with direct access to toolkit developers or other experts. For many users of technical toolkits, issues are solved in internal memos, emails, or messaging platforms that are exclusively available to a particular workplace or research team. For applied low-resource TTS projects, an ideal toolkit would have public documentation, guides, and support systems.

**Toolkit Implementations.** ESPnet benefits from a large community (7.3k stars on GitHub and over 200 contributors), publicly available documentation including Kaldi-style ‘recipes’ that are effectively collections of bash scripts to perform a particular task. There are also recordings of in-person tutorials,

lectures, and community-authored instructional videos on toolkit usage for TTS.

Coqui TTS arguably has the largest community of users with 21.1k stars on GitHub, over 130 contributors, an active GitHub Discussions page, and over 2900 members on its Discord channel. However, the member count here is shared between users of its open-source toolkit implementation as well as its business product offerings. The documentation contains a combination of guides, recipes, and notebooks.

Similar to ESPnet and Coqui TTS, NeMo has a large community of users with 8.3k stars and 270 contributors. It also has some excellent guides and Jupyter notebooks for introducing users to speech synthesis and the NeMo-toolkit as well as a GitHub Discussion page.

FAIRSEQ S<sup>2</sup> has a large community of users and contributors. They have a documentation page, but the documentation page does not include any information about speech synthesis. Instead, they have some examples in the form of markdown files within the code base that demonstrate how to prepare data from LJ Speech, VCTK, and Common Voice and commands for running training, inference, and evaluation. They do not have a Discord channel but they do have a Google Group.

IMS Toucan has a relatively smaller community of users than some of the other toolkits discussed here, likely in part because of its focused approach in providing a TTS architecture for low-resource applications. The contributors are active and supportive to the community of users and discussions appear to take place in the GitHub Issues page. There is information in the GitHub repository page readme about how to adapt the training pipeline to a new language, but there is no dedicated documentation page.

SpeechBrain has a large community of users (6.7k stars and over 130 contributors), a dedicated Discord channel with 200+ members, and a variety of forms of documentation including blog-style tutorials, notebooks, and YouTube videos (as of writing there do not appear to be videos for TTS).

As previously mentioned, we are releasing EveryVoice TTS at a functional but early stage of development to help encourage the growth of a community of users; however, that community does not yet exist. EveryVoice TTS has documentation, guides and a GitHub Discussion page, but does not have any associated notebooks, video content, or a Discord channel like other toolkits discussed here. Our team is excited to welcome users, but there is no replacement for user communities like the ones found in Coqui TTS, ESPnet, or SpeechBrain, and they take time to grow. Prospective users of EveryVoice

TTS or other toolkits with limited contributors and communities should take this consideration into account as a possible limitation of the toolkit.

To help guide users, EveryVoice TTS has implemented a textual user interface (TUI) to interactively guide users through the steps of configuring a TTS project with new datasets in a new language.

It is worth mentioning that all of the toolkits evaluated here, including EveryVoice TTS, only provide documentation in English, which introduces a language barrier for prospective users who do not speak English. While this is true for the vast majority of speech processing toolkits, there are exceptions such as PaddleSpeech (H. Zhang et al., 2022), which provides a bilingual English & Mandarin Chinese ‘readme’ and responds to issues and discussion topics in both languages.

	Coqui TTS	ESPnet	Every Voice	FAIRSEQ S <sup>2</sup>	IMS-Toucan	NeMo	Speech Brain
GitHub Stars	21.3k	7.3k	N/A	27.9k	390	8.4k	6.8k
Open Issues	18	185	N/A	1019	28	41	95
Closed Issues	752	2031	N/A	3025	105	1782	861
Contributors	139	231	5	308	3	270	135
Discussions Page	✓	✓	✓	✗	✗	✓	✓
Discord Members	2901	✗	✗	✗†	✗	✗	260
Textual User Interface	✗	✗	✓	✗	✗	✗	✗
Jupyter Notebooks	✓	✓	✗	✗	✗	✓	✓
Documentation Page	✓	✓	✓	✗	✗	✓	✓
Associated Paper	✗	✓ <sup>1</sup>	✓	✓ <sup>2</sup>	✓ <sup>3</sup>	✓ <sup>4</sup>	✓* <sup>5</sup>

Table C.11: Summary of information related to documentation, guides, and support for existing toolkits. GitHub statistics are not available yet for EveryVoice TTS as of writing. Only foundational paper citations are included, there are many citations associated with some of these toolkits. \* indicates the associated paper only appears as a preprint. † FAIRSEQ S<sup>2</sup> has a Google Group with a community of users. <sup>1</sup>(Watanabe et al., 2018), <sup>2</sup>(C. Wang et al., 2021), <sup>3</sup>(Lux et al., 2021), <sup>4</sup>(Kuchaiev et al., 2019), <sup>5</sup>(Ravanelli et al., 2021)

	Coqui TTS	ESPnet	Every Voice	Fairseq	IMS- Toucan	NeMo	Speech Brain	Section
Associated Paper	✗	✓ <sup>1</sup>	✓	✓ <sup>2</sup>	✓ <sup>3</sup>	✓ <sup>4</sup>	✓ <sup>5</sup>	§A
Data Efficiency	✓†	✓†	✓	✓†	✓	✓†	✓†	§C.1.1
Misuse Prevention	✗	✗	✓	✗	✗	✗	✗	§C.1.2
Phonological Features	✗	✗	✓	✗	✓	✗	✗	§C.1.3
SSL Features	✗	✗	✗	✓	✗	✓	✗	§C.1.3
Public Checkpoints	✓	✓	✓	✓	✓	✓	✓	§C.1.3
Evaluation	✗	✓	✗	✓	✗	✓	✗	§C.1.4
Noise Removal	✗	✓*	✗	✓	✗	✗	✓*	§C.2.1
Silence Trimming	✓	✓	✓	✓	✓*	✓	✓*	§C.2.2
Segmentation	✗	✓*	✓	✓*	✗	✓	✓*	§C.2.3
Outlier Filtering	✓	✗	✓	✓	✓	✗	✗	§C.2.4
Config Library	Coqpit	YAML	Pydantic	Hydra	None	Hydra	HyperPy- Yaml	§C.3.1
Config Validation	✓	✗	✓	✗	✗	✗	✗	§C.3.1
Composable Configs	✓	✓	✓	✓	✗	✓	✓	§C.3.1
Config Syntax Highlighter	✗	✗	✓	✗	N/A	✗	✗	§C.3.1

Multiple Data Sources	✗	✗	✓	✗	✗	✗	✗	§C.3.1
GitHub Stars	21.3k	7.3k	N/A	27.9k	390	8.4k	6.8k	§C.3.2
Open Issues	18	185	N/A	1019	28	41	95	§C.3.2
Closed Issues	752	2031	N/A	3025	105	1782	861	§C.3.2
Contributors	139	231	5	308	3	270	135	§C.3.2
Discussions Page	✓	✓	✓	✗	✗	✓	✓	§C.3.2
Discord Members	2901	✗	✗	✗ <sup>‡</sup>	✗	✗	260	§C.3.2
Textual User Interface	✗	✗	✓	✗	✗	✗	✗	§C.3.2
Jupyter Notebooks	✓	✓	✗	✗	✗	✓	✓	§C.3.2
Documentation Page	✓	✓	✓	✗	✗	✓	✓	§C.3.2

Table C.12: A summary table of the results of the survey provided in sections §5, Appendix C, and Appendix B. For additional information please visit the associated sections, referenced in the last column. \* indicates that the feature is available, but not documented in connection with a TTS pipeline or recipe. † indicates that data efficient models are implemented in the toolkit, but there is no documentation present that recommends a particular model for use in low-resource contexts. ‡ FAIRSEQ S<sup>2</sup> does not have a Discord channel, but they have a Google Group where discussions take place. <sup>1</sup>(Watanabe et al., 2018), <sup>2</sup>(C. Wang et al., 2021), <sup>3</sup>(Lux et al., 2021), <sup>4</sup>(Kuchaiev et al., 2019), <sup>5</sup>(Ravanelli et al., 2021)