Proceedings of SIGUL2022 @LREC2022, pages 169–175
Marseille, 24-25 June 2022

Building Open-Source Speech Technology for Low-Resource Minority Languages with Sámi as an Example – Tools, Methods and Experiments

Katri Hiovain-Asikainen, Sjur Nørstebø Moshagen
Department of Language and Culture
UiT the Arctic University of Norway
firstname.lastname@uit.no

Abstract

This paper presents a work-in-progress report of an open-source speech technology project for indigenous Sámi languages. A less detailed description of this work has been presented in a more general paper about the whole *GiellaLT* language infrastructure, submitted to the LREC 2022 main conference. At this stage, we have designed and collected a text corpus specifically for developing speech technology applications, namely Text-to-speech (TTS) and Automatic speech recognition (ASR) for the Lule and North Sámi languages. We have also piloted and experimented with different speech synthesis technologies using a miniature speech corpus as well as developed tools for effective processing of large spoken corpora. Additionally, we discuss effective and mindful use of the speech corpus and also possibilities to use found/archive materials for training an ASR model for these languages.

Keywords: speech corpus, speech processing, minority languages, indigenous languages, TTS, ASR, speech technology

1. Introduction

The current paper will describe ongoing work for developing open-source speech technology applications for two Sámi languages, Lule and North Sámi. The Sámi languages, belonging to the Uralic language family, are related to, e.g. Finnish and Estonian and thus share some structural and lexical features. Lule and North Sámi are neighboring languages, spoken in the northernmost parts of Scandinavia. While Lule Sámi is spoken in Norway and Sweden, North Sámi is spoken in three countries: Norway, Sweden and Finland. For both languages, generally all speakers are bilingual in Sámi and at least one of the majority languages: Norwegian, Swedish and Finnish. The two languages are structurally similar, and after some training, they are mutually intelligible to some extent. However, as part of language revitalization and preservation as well as accelerating digitalization, separate languages need separate language and speech technology tools to meet the needs of modern language users.

Lule and North Sámi differ remarkably in terms of the amount of speakers or language users. According to Ethnologue (Lewis, 2009), North Sámi has by far the largest number of language users among the Sámi languages: 25 000 in all three countries. Lule Sámi, on the other hand, has considerably fewer speakers: total of 2000 in both countries it is spoken in. All Sámi languages are classified as endangered by UNESCO (Moseley, 2010) and Lule Sámi as severely endangered. Perhaps consequently, as North Sámi has most language users among the Sámi languages, it has also most language resources and tools available. An infrastructure of dictionaries, morphological analyzers, spell checkers and language learning tools etc. have been maintained and developed since 2001 by the Divvun and Giellatekno groups[1].

A Text-to-speech tool is made to be able to synthesize intelligible speech output from any unseen text input in a particular language. A key objective for developing speech technology tools for indigenous languages generally is to meet the needs of modern language users in all language communities equally. For the Sámi languages, this would mean equal possibilities to use Sámi in the same contexts as the majority languages are being used. In this way, developing speech and language technology tools for the Sámi languages also contribute to the revitalisation of these languages. Additionally, speech technology tools are important for many language users, also those with special needs. These include language learners (see, e.g., (Yaneva, 2021)), people with dyslexia, vision impaired individuals, (native) users of the language that are not used to read Sámi etc. Additionally, speech technology is bringing more accessibility to many kinds of contents and utilities: a user can for example choose to listen to the news instead of reading the text, or a speech synthesis tool could be integrated into an online dictionary to allow listening to the correct pronunciations of the words.

The first Text-to-speech (TTS) tool for the Sámi languages was developed in 2015 for North Sámi by Divvun and Acapela[2]. This tool was produced as closed-source and thus neither the framework used to develop the tool nor the speech corpus used for it are publicly available[3]. Also, the company has ended support for certain operating systems, blocking access to

[1] https://giellatekno.uit.no/, https://divvun.no/fi/

[2] `https://divvun.no/fi/tale/tale.html`

[3] We hope to be able to make the speech corpus publicly available in the future.

the voices for new users on these operating systems. For this reason, we are now working on a modern, open-source TTS system that could be openly available for anyone who wants to develop speech technology for minority languages. The system will make all language-independent parts integrated into the larger GiellaLT infrastructure[4], ensuring that maintenance and updates are done regularly. When finished, it will also ensure that all voices will be available on all supported platforms, and that new platforms will be available to all existing voices. The research and development groups behind the GiellaLT infrastructure have existed for about twenty years, and given the governmental support for the Sami languages, the sustainability prospects are good.

2. Requirements and Related Works

Developing TTS for an indigenous language with few resources (such as grammars, language learning books or phonetic descriptions) available can be challenging. Such resources are important in designing the project, building and checking the corpora and evaluating the TTS output phonetically. If a phonetic description of the language is scarce or it is made within a different framework, one might need to make a description from scratch. Any linguistic description is useful for this, but for speech technology purposes, what is important is to have at least some amount of speech material and corresponding text, provided by a native speaker of the language. In this way, it is possible to study the relationship between text and speech in a particular language and to produce a phonetic description in a form of a grapheme-to-phoneme mapping. This mapping (or *text-to-IPA* rule set) can already be used to build a very simple and ”old-fashioned” but still usable TTS application, such as the Espeak formant synthesis (Kastrati et al., 2014; Pronk et al., 2013). As this framework does not require a speech corpus but only a set of phonetic and phonological rules, any language can be added to the list of the languages covered by Espeak, only utilising the knowledge of native speakers. The downside of this is that while it might be a working synthesizer, the users' expectations for the quality of a TTS system are very high due to the examples from well-resourced languages such as English.

The development of a TTS system as a whole requires multidisciplinary input from fields like natural language processing (NLP), phonetics and phonology, machine learning (ML) and digital signal processing (DSP). Tasks connected to NLP are important in developing the text front-end for the TTS – these are, for example, automatically converting numbers and abbreviations to full words in a correct way. Phonetics and phonology are essential in corpus design, making text-to-IPA rules and evaluating the TTS output. Also, by using phonetic annotations of the texts, it is possible to address phenomena that are not visible in the orthographic texts. The importance of ML is growing in the field of speech technology, as neural networks are used to model the acoustics of human speech, allowing for realistic and natural-sounding TTS. Procedures related to DSP are important in (pre)processing the audio data: these include filtering, resampling and normalizing the corpus for suitable audio quality. Furthermore, the resulting TTS system can be used in developing more advanced speech technology frameworks, such as dialogue systems (see, e.g. Jokinen et al. (2017; Wilcock et al. (2017; Trong et al. (2019)) and various kinds of mobile applications.

Some of the typological and phonetic features of for example North and Lule Sámi are setting challenges in building a high quality TTS. One of these is the ternary quantity system in both of these languages. In both North and Lule Sámi, there are triplets of word forms that differ only by the quantity, the length of the intervocalic consonant in a disyllabic foot. The orthography does not differentiate between the Quantity 3 (Q3) and Quantity 2 (Q2) forms in all contexts, and the long (Q2) and overlong (Q3) geminates are written identically in those cases (see Tables 1 and 2 for examples). Our first experiments on building an open-sourced TTS have shown that a simple rule-based formant synthesis (such as Espeak) is not able to fully cover for this phonetic phenomenon without a separate syntactically disambiguated text-processing pipeline.

At present, several minority language communities with a weak literary tradition try to strengthen the position of the language in society. In doing so, they find themselves in a situation lacking the infrastructure needed to do so, infrastructure that majority language speakers take for granted. Minority language communities do not equally benefit from the technological advances, compared to languages like English or Mandarin. By adapting existing state-of-the-art speech technology to a form suitable for low-resource languages, we contribute to the strengthening the language infrastructure for the Sámi languages and widening the modalities where the languages can be used.

In what follows, we present our plans for our Sámi TTS project and discuss some directions for our future work.

3. Methodology

3.1. Building the Corpora

3.1.1. Text Corpus

Building a corpus with good quality requires selecting native language texts from different domains to build a special-purpose corpus (i.e. for speech technology) from scratch.

Texts in Sámi languages are published daily in both media and by public bodies required to communicate in writing in Sámi. Since most of the publishers (typically online) have to provide their site in both Sámi and the majority languages. Having gathered text since 2005,

[4] giellalt.github.io, github.com/divvun

the largest Sámi corpus is the one for North Sámi, with 38.94 million tokens. The North Sámi corpus is a quite big corpus for an indigenous language, but on the other hand small compared to majority languages.

Our aim is to have a balanced corpus for the other Sámi languages as well, with regard to regional dialects of the same language. As the majority of North Sámi speakers are in Norway, and the legal protection for the Sámi languages is stronger in Norway than in Sweden, both our North Sámi and our Lule Sámi corpus therefore mostly consist of text written in Norway. This has consequences for some of the tools we are developing, including TTS: the synthesis will reflect the characteristics of the Norwegian variety better.

3.1.2. Speech Corpus

The modern approaches to TTS involve machine learning and complex modelling of speech, which brings in the requirement for relatively big amounts of speech data to build the models from. This is because in a data-driven or *corpus-based* speech synthesis, that utilize deep neural networks, the association between textual features and acoustic parameters is learned directly from paired data – the sentence-long sound files and the corresponding texts. The sum of the learned knowledge from the paired data construct the acoustic model (see, e.g., Watts et al. (2016)). This is especially the case in the modern end-to-end or sequence-to-sequence approaches that merge the front-end to the neural model, such as in the Tacotron 2 framework (Shen et al., 2018).

The building of the speech corpus starts from collecting a suitable multi-domain text corpus which corresponds to at least 10 hours of recorded read speech, that has been shown to be enough to achieve an end-user suitable TTS system for North Sámi (Makashova, 2021). This amount is also going to be recorded to build our Lule Sámi voice. Our plan is to build both male and female voices and thus altogether 20 hours of speech is going to be recorded.

A question of *data efficiency* has been discussed in a new study by Săracu and Stan (2021). This study evaluated the amount of data required by the Tacotron 2 speech synthesis model to produce good quality output, and showed that if the training data is carefully constructed to present all common graphemes in a language, the data requirement can be significantly lowered. In the present project, we have checked that our corpus covers all important phonological contrasts and sound combinations by calculating frequencies of all trigrams in our corpus. Additionally, we calculated frequencies of all consonant gradation patterns from the Lule Sámi TTS corpus, using a grammatical description of the language (Spiik, 1989). In the case of missing gradation patterns, we added additional sentences to cover for these as well.

In the present project, we focus on open-source methodologies, in which case it is important to build a collection of open source texts as well, with a CC-BY (Creative Commons) licence.

To build our new TTS text corpus, we reused a part of the Lule Sámi gold corpus[5] developed in 2013 within the GiellaLT community, and collected additional texts of various text styles we knew to be well written. The resulting Lule Sámi text corpus for TTS consists of text styles such as news, educational, parliament etc. with altogether over 74,000 words (see Figure 4 for word counts per domain).

3.2. Corpus Processing and Modeling

When using machine-learning methods to build up a speech model for TTS, the quality of the recordings has to be excellent, i.e., room reverberation or background noise has to be avoided in the recordings, because the noise would be modelled as well. Thus, the recordings have to be done in a sound-treated room with professional microphones and recording set-up. The minimal requirement for the audio recording is so-called *CD quality* (44.1 kHz sample, 16-bit).

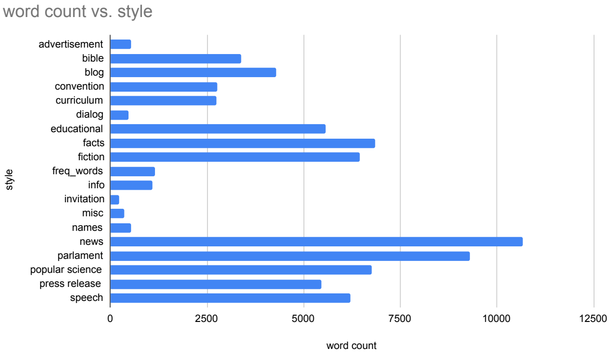

Figure 1: The word counts per style of the Lule Sámi text corpus for TTS. Altogether, 74,737 words that correspond roughly to 12.46 hrs of speech recordings.

3.2.1. Text Processing

Most orthographies are underspecified with respect to the pronunciation of the text. This creates interesting questions when converting a standard orthographic text to audio waves. In the cases of Lule and North Sámi there is a class of nouns where consonant gradation (i.e. length alternation) is not expressed in the orthography, while still being grammatically crucial, as it is the sole marker of the difference between different syntactic functions, especially *singular nominative* vs *singular genitive*, and for North Sámi also *singular accusative*. That is, for this class of nouns the only difference between the subject and the possessor or (for North Sámi) between the subject and the object, is expressed through a length distinction that is *not* present in the standard orthography, as seen in Tables 1 and 2. This distinction is phonetically significant, as shown in a number of acoustic phonetic studies, such as in Magga

[5] gtsvn.uit.no/freecorpus/goldstandard/converted/smj/

(1984) and Hiovain et al. (2020) for North Sámi and Fangel-Gustavson et al. (2014) for Lule Sámi.

The distinction has to be recreated when converting the orthographic text to a phonemic representation. There are also other underspecifications in the orthography, but these are the most crucial.

	Orth.	IPA	Transl.
Q3	*oarre*	[ʔo͡ɑrːrɪɛ]	’a squirrel’ Nom.Sg
Q2	*oarre*	[ʔoɑrːɪɛ]	’a squirrel’s’ Gen.Sg
			’a reason’ Nom.Sg
Q1	*oare*	[ʔoɑrɪɛ]	’a reason’s’ Gen.Sg

Table 1: Ternary length contrast of consonants in Lule Sámi, underspecified in the orthography. Abbreviations: Q3 – overlong, Q2 – long, Q1 – short. Examples originally presented in Fangel-Gustavson et al. (2014).

	Orth.	IPA	Transl.
Q3	*beassi*	[pe͡æsːsɪ]	’birchbark’ Nom.Sg
Q2	*beassi*	[peæsːɪ]	’birchbark’ Acc.Sg
			’(bird’s) nest’ Nom.Sg
Q1	*beasi*	[peæsɪ]	’(bird’s) nest’ Acc.Sg

Table 2: Ternary length contrast of consonants in North Sámi, underspecified in the orthography. Abbreviations as in Table 1.

The foundation for all linguistic processing and thus also for the text processing for speech technology in the *GiellaLT* infrastructure is the morphological analyser, built using formalisms from Xerox. From these source files, the *GiellaLT* infrastructure creates *finite state transducers* (FST’s) using one of three supported FST compilers: Xerox tools (Beesley and Karttunen, 2003), *HFST* (Lindén et al., 2013), or Foma (Hulden, 2009). All language models are written as rule-based, full form lexicons with explicit morphological descriptions and morphophonological alternations. This makes it possible to create language models for any language, including minority and indigenous languages with few or non-existing digital resources.

FST’s are useful in speech technology especially in the task of converting orthographic texts to IPA characters, by using an FST model of the language to analyze the corpus texts. The length contrast is encoded in the FST model at an intermediate level, but during compilation, this information is lost. We have enhanced the code for the *HFST* utility `hfst-pmatch` to allow the analyser/-tokeniser FST to be an on-the-fly composition of two separate FST’s, and outputting that intermediate string representation, in effect creating a fake three-tape FST. With the morphological analysis of all tokens available, we can proceed by disambiguating the sentence, and leaving only the analyses that fit the morphosyntactic context. The end result is that we will be left with the proper analysis (subject or object) *and* with information of the proper length of the word form, to be fed to the module for conversion to IPA. As always, this is done using rule-based components, to have full control of every step and be able to correct errors in the IPA transcription. There is still a fallback module for cases of unknown words and names.

The IPA transcription provided by the FST technology described above can further improve the accuracy of the TTS, especially for the alignment between sounds and characters. When training a speech model with the IPA transcriptions as text input instead of standard orthography, in a deep neural network structure, the letter-to-sound correspondence will likely be more transparent, especially with ternary quantity cases described above. This rule-based approach, reusing many components from other parts of the *GiellaLT* infrastructure, also means that high quality speech synthesis is within reach for not only Sámi languages, but for other low-resource languages as well.

3.2.2. Experiments with Different TTS Frameworks

We have experimented with two different open source ML based TTS methodologies: Ossian (Suni et al., 2014) and a *Tacotron implementation* (largely based on Shen et al. (2018)), specially adapted for low-resource languages, like the Sámi languages (Makashova, 2021). Both of these methodologies require standard preprocessing procedures such as splitting the training data into sentence-long files as well as some sound filtering and normalisation techniques to ensure good quality and accuracy of the speech modeling.

The texts have to accurately match the corresponding audio files for the modelling to be successful, thus, a text normalisation procedure (part of the front-end) has to be conducted for the whole data. This covers, e.g., converting numbers, acronyms and abbreviations to orthographic text. Also, as explained in Section 3.2.1., it is useful to make a letter to sound (or text-to-IPA) rule mapping of a given language as this makes the relationship between speech and the corresponding text (when used as training data for speech modeling) more transparent.

In our first experiment, we used a data set consisting of approximately one hour of speech from a native speaker of Lule Sámi, using the Ossian TTS. Ossian consists of a rule-based, statistical front-end and a deep neural network-based acoustic modelling. We used Ossian with the HTS (HMM/DNN-based Speech Synthesis System, see also Zen et al. (2007)) recipe to train an experimental Lule Sámi voice, generating relatively intelligible speech (see Figure 2 for a spectrogram image of a sample sentence).

With one hour of training data and an HP ZBook 15 G6 (Intel i7 CPU), it took approximately 3 hours to train an experimental Lule Sámi voice. Although Ossian TTS or similar would technically be more suitable for a low-resource setup, its machine-like voice quality does not meet the requirements of a modern speech technology user. However, from this experiment, it was clear that

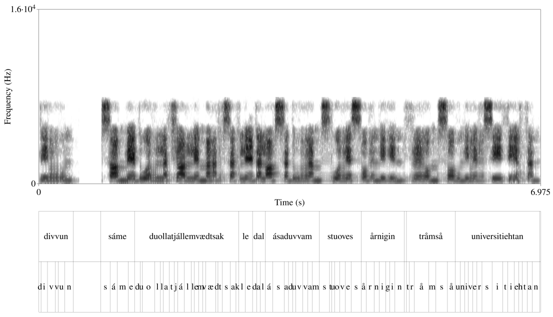

Figure 2: A spectrogram of a sample sentence from the Ossian TTS model trained on 1 hour of Lule Sámi speech. Sentence text: "*Divvun, sáme duollatjállemvædtsak, le dal ásaduvvam stuoves årnigin Tråmså universitiehtan*."

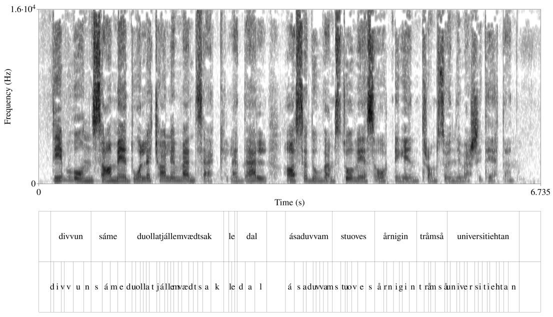

Figure 3: A spectrogram of a sample sentence from a human native speaker of Lule Sámi, reading the exact same sentence as in the Ossian sample.

for getting better results, more training data would be needed, but piloting the methods using small experimental data gives us better insight on the requirements for the speech corpus, i.e. the size and audio quality of the data.

As the expectations for the quality of TTS are very high due to the examples from well-resourced languages such as English, using a neural vocoder (such as *WaveNet*, Oord et al. (2016) or *WaveGlow*) that produces realistic, human-like speech is necessary for good usability and user experience.

As described in Makashova (2021), the North Sámi TTS voice was trained with a female voice, data set consisting of 3500 training sentences. The TTS model consisted of four components: Tacotron, ForwardTacotron, Tacotron2 and WaveGlow, the two latter ones from the official Nvidia repository. The training of this successful and good quality Tacotron model and the WaveGlow model took one month, and for the ForwardTacotron for three days, on a single GPU. In Divvun, we have access to the Norwegian academic high-performance computing and storage service (Sigma2) and thus the training time could be significantly shorter.

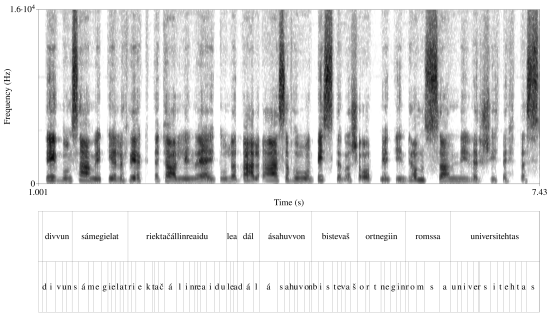

Figure 4: A spectrogram of a sample sentence generated using a Tacotron model of North Sámi. The text is the North Sámi equivalent of the Lule Sámi sentence in the previous figures: "*Divvun, sámegielat riektačállinreaidu, lea dál ásahuvvon bistevaš ortnegiin Romssa Universitehtas.*"

As can be seen from comparing the spectrograms in Figures 2, 3 and 4, the Tacotron sample is also visually similar to the human speech in Figure 3. The Ossian sample has a lot lower frequency range compared to the Tacotron and human samples, and the formant transitions are not smooth. Figure 4 also shows the promising quality of the Tacotron sample: with few hours of training data, realistic and good quality TTS is achievable. Thus, a similar workflow, following the North Sámi one for training the Lule Sámi TTS voice has been planned and started in our project.

It has to be taken into account that the environmental cost for the complex modelling of speech is high in terms of electricity and technical components such as graphical processing units (GPUs). For reducing these costs, there are possibilities to adapt existing speech models by training the models further with additional data and pre-trained models from a "neighbouring" language. This so-called *transfer learning* (Tu et al., 2019; Debnath et al., 2020) allows for utilising smaller data sets for training, making it possible, for example, to use the North Sámi TTS model as the starting point for the Lule Sámi TTS.

At this point, we have made some experiments on a TTS model using transfer learning between North and Lule Sámi. With a miniature data set (approx. one hour of speech data recorded with a cell phone), we were able to train a Lule Sámi voice, but the quality of the output showed that this corpus did not cover all necessary phonemes of the language and thus there were some phonological inaccuracies. Moreover, as the North and Lule Sámi orthographies are somewhat different (for example, the alveolar fricative sound written in English as *sh*, is written as *š* in North Sámi, and as *sj* in Lule Sámi), there were errors in this kind of cases. By converting both North and Lule Sámi texts to IPA characters these differences could be ”eliminated” and thus the transfer learning would presumably be more successful.

A good quality speech corpus of Lule Sámi is going to be produced by autumn 2022. Having experimented with different frameworks and experimental data sets, we have now the required tools and technologies to proceed quickly to producing the end-user suitable TTS for Sámi.

3.3. Future Work: Approaches to Automatic Speech Recognition

In addition to TTS, we are working towards developing a tool for *automatic speech recognition* (ASR) for Sámi. This section describes materials and experiments only for North Sámi, but in the future, we hope to expand our work to Lule Sámi ASR as well.

In Makashova (2021), TTS and ASR models were trained simultaneously in a dual transformation loop, using the same *read speech* data set, corresponding to only six hours of speech from two speakers, three hours each. The ASR model in this work was based on the Wav2Vec model which is a part of the HuggingFace library. The model was trained for 30 000 steps and it reached a WER (Word-Error-Rate) of 41% and 0.5 loss. The most common error types in the ASR predictions seem to be in word boundaries (**earáláhkai – eará láhkái* and in lengths of some sounds (**rinš**k**it – rinš**kk**it*). However, these kinds of errors would be easy to correct using Divvun's spell checking software.

One of the most important differences between training the TTS and ASR models would be that for TTS, the training material needs to be very clean in terms of sound quality and there needs to be as many recordings from a single speaker as possible. For ASR, on the other hand, the recorded materials can be of poorer sound quality and preferably from multiple speakers and from different areal varieties of a language as long as there are good transcriptions of the speech.

State-of-the-art ASR frameworks normally require up to 10,000 hours of multi-speaker data for training reliable and universal models that are able to generalise to any unseen speaker (Hannun et al., 2014). As collecting these amounts of data from small minority languages is not a realistic goal, alternatives such as utilising existing archive materials can be considered for developing speech technology for Sámi. These are provided by, e.g., *The language bank of Finland* and *The language bank of Norway*. These archive materials contain spontaneous, transcribed spoken materials from various dialects and dozens of North Sámi speakers.

The huge amounts of speech data normally used for ASR thus might require *massive* online data sourcing campaigns, such as the ongoing *Lahjoita puhetta – "Donate your speech"*[6] project for developing Finnish ASR. A similar campaign but in a smaller scale could be considered for the Sámi languages.

The first experiments on using the ASR model from (Makashova, 2021) to predict unseen *spontaneous* North Sámi speech have been promising and there is ongoing work on further development of an ASR tool. We believe such a tool will contribute to the documentation and to better usability of any untranscribed Sámi archive speech corpora. By providing automatic text transcriptions of the materials, they could be easily searchable and thus utilized for, e.g. linguistic research. Additionally, ASR has an important role in modern language-learning applications that have spoken language exercises, such as in Duolingo (Teske, 2017).

4. Conclusion

In summary, the procedures and pipelines described above could be applied to any (minority) language with a low-resource setting, in the task of developing speech technology applications. Most of the applications discussed here can be piloted with or further developed with relatively small data sets (even with < 10 hrs of paired data), compared to the amounts of data used for respective tools for majority languages. This is largely possible thanks to the available open source materials and technologies, especially those relying on, e.g., *transfer learning* methodologies that allow for adapting speech models between related/similar languages.

Additionally, Cooper (2019) suggests that for low-resource languages, certain types of *found data* could be used to build TTS, instead of collecting a synthesis corpus from scratch. In this research, non-traditional sources of data such as (read) ASR data, radio broadcast news and audio books were used to develop usable and natural sounding TTS.

Finally, for tasks like TTS, if a speech corpus must be built from scratch, it has to be designed to prioritise quality over quantity of the corpus. We ensure a good quality and multi-purpose speech corpus by working with professional voice talents and language experts that are native speakers of the language. Additionally, by making the speech corpus used for developing TTS openly available, future needs to collect similar corpora are reduced.

Bibliographical References

Beesley, K. R. and Karttunen, L. (2003). Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford*.

Cooper, E. (2019). *Text-to-speech synthesis using found data for low-resource languages*. Columbia University.

Debnath, A., Patil, S. S., Nadiger, G., and Ganesan, R. A. (2020). Low-resource end-to-end sanskrit tts using tacotron2, waveglow and transfer learning. In *2020 IEEE 17th India Council International Conference (INDICON)*, pages 1–5. IEEE.

Fangel-Gustavson, N., Ridouane, R., and Morén-Duolljá, B. (2014). Quantity contrast in Lule Saami: A three-way system. In *Proceedings of the 10th International Seminar on Speech production*, pages 106–109.

[6] yle.fi/aihe/lahjoita-puhetta

Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., et al. (2014). Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

Hiovain, K., Vainio, M. T., and Šimko, J. (2020). Dialectal variation of duration patterns in Finnmark North Sámi quantity. *The Journal of the Acoustical Society of America*, 147(4):2817–2828.

Hulden, M. (2009). Foma: a finite-state compiler and library. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 29–32.

Jokinen, K., Hiovain, K., Laxström, N., Rauhala, I., and Wilcock, G. (2017). Digisami and digital natives: Interaction technology for the North Sami language. In *Dialogues with social robots*, pages 3–19. Springer.

Kastrati, R., Hamiti, M., and Abazi, L. (2014). The opportunity of using espeak as text-to-speech synthesizer for Albanian language. In *Proceedings of the 15th International Conference on Computer Systems and Technologies*, pages 179–186.

M. Paul Lewis, editor. (2009). *Ethnologue: Languages of the World*. SIL International, Dallas, TX, USA, sixteenth edition.

Lindén, K., Axelson, E., Drobac, S., Hardwick, S., Kuokkala, J., Niemi, J., Pirinen, T. A., and Silfverberg, M. (2013). Hfst—a system for creating nlp tools. In *International workshop on systems and frameworks for computational morphology*, pages 53–71. Springer.

Magga, T. (1984). *Duration in the quantity of bisyllabics in the Guovdageaidnu dialect of North Lappish*, volume 11. University of Oulu.

Makashova, L. (2021). Speech synthesis and recognition for a low-resource language: Connecting TTS and ASR for mutual benefit. Master's thesis, University of Gothenburg.

Moseley, C. (2010). *Atlas of the World's Languages in Danger*. Unesco.

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Pronk, R., Intelligentie, B. O. K., and Weenink, D. D. (2013). Adding Japanese language synthesis support to the espeak system. *University of Amsterdam*.

Săracu, G. and Stan, A. (2021). An analysis of the data efficiency in Tacotron2 speech synthesis system. In *2021 International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, pages 172–176. IEEE.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., et al. (2018). Natural tts synthesis by conditioning WaveNet on mel spectrogram predictions. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4779–4783. IEEE.

Spiik, N. E. (1989). *Lulesamisk grammatik*. Sameskolstyrelsen.

Suni, A., Raitio, T., Gowda, D., Karhila, R., Gibson, M., and Watts, O. (2014). The simple4all entry to the Blizzard Challenge 2014. In *Proc. Blizzard Challenge*. Citeseer.

Teske, K. (2017). Duolingo. *calico journal*, 34(3):393–401.

Trong, T. N., Jokinen, K., and Hautamäki, V. (2019). Enabling spoken dialogue systems for low-resourced languages—end-to-end dialect recognition for North Sami. In *9th International Workshop on Spoken Dialogue System Technology*, pages 221–235. Springer.

Tu, T., Chen, Y.-J., Yeh, C.-c., and Lee, H.-Y. (2019). End-to-end text-to-speech for low-resource languages by cross-lingual transfer learning. *arXiv preprint arXiv:1904.06508*.

Watts, O., Henter, G. E., Merritt, T., Wu, Z., and King, S. (2016). From HMMs to DNNs: where do the improvements come from? In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5505–5509. IEEE.

Wilcock, G., Laxström, N., Leinonen, J., Smit, P., Kurimo, M., and Jokinen, K. (2017). Towards samitalk: a Sami-speaking robot linked to Sami wikipedia. In *Dialogues with Social Robots*, pages 343–351. Springer.

Yaneva, A. (2021). Speech technologies applied to second language learning. A use case on Bulgarian. Bachelor's thesis.

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A. W., and Tokuda, K. (2007). The HMM-based speech synthesis system (hts) version 2.0. In *SSW*, pages 294–299. Citeseer.