The First Text-to-Speech System for Yiddish

די ערשטע טעקסט־רייד־סיסטעם פֿאַר ייִדיש

Samuel Kwan-lok Lo

Master of Science
Speech and Language Processing
School of Philosophy, Psychology and Language Sciences
University of Edinburgh

2021

Abstract

This project introduces the first ever neural text-to-speech (TTS) engine for Yiddish, a West Germanic language primarily spoken by Ashkenazi Jewish communities around the world. Audiobooks from the Yiddish Book Centre and those compiled by the University of Haifa are first cleaned, then used as training data for the TTS system using FastSpeech 2. It is hypothesised that intelligibility and naturalness of the synthetic voices are dependent on the 1) quality of the dataset, 2) presence of diacritics, and 3) spelling of Hebrew-origin words. MUSHRA listening tests are done by Yiddish speakers and one expert evaluator, who rated the intelligibility and naturalness of the resulting voices. It is found that higher transcription quality of the training data corresponds to better-sounding voices. However, contrary to the hypotheses, removing diacritics did not increase quality of the voices, and instead reduced it. Respelling Hebrew-origin words into more phonemic spelling also reduced the overall accuracy because the method it determines whether a word is Hebrew-origin is too naïve. Despite these findings, the overall quality of the voices are still very good according to the evaluators, which is an encouraging first step in the development of Yiddish TTS.

Keywords: *Yiddish*, *Jewish languages*, *low-resource languages*, *neural networks*, *speech synthesis*

Acknowledgements

First and foremost, I would like to thank my supervisor Jacob Webber, who has given me a lot of guidance throughout the whole project, both general and technical. This project will also not be possible without the help of linguist Isaac Bleaman, who has given his time and effort helping this project using his expertise of Yiddish, including processing the dictionary of phonetic respellings of Hebrew-origin words, and helping me find participants for the listening test. I also want to thank Yiddish instructor Eliezer Niborski for compiling the aforementioned dictionary, and acting as our expert evaluator for the project. I would like to extend my gratitude to Mindl Cohen and Amber Kanner Clooney from the Yiddish Book Center for their preservation of the audiobooks and making them publicly accessible, as well as the two narrators for narrating the books used as training data in this project: Peretz Zilberberg, and Sara Blacher-Retter, who was an Israeli nurse. Finally, I want to thank my fellow student Aidan Pine for sharing his code and helping me with technical issues with model implementation.

Contents

1 Introduction

Digital text-to-speech (TTS) technology was first implemented in 1961 for English (Klatt, 1987), and has quickly developed since. It is the basis for various technologies, from audiobooks to giving a voice to those with speech disorders. Modern neural TTS models require human speech data to train, in the form of segmented short transcribed utterances. These models have since been used to expand TTS support to other lower-resource languages (Dagba & Boco, 2014; Gutkin et al., 2016; Rama et al., 2002). This project is a preliminary effort to expand that coverage to Yiddish, a West Germanic language spoken by Ashkenazi Jews. Giving these communities the technology is one way to empower the language and provide speakers with an opportunity to use technologies enabled by TTS in their own language. Such systems may slow the abandonment of the language due to limited technical support.

This system is valuable for Yiddish speakers who wish to access speech technology in Yiddish, which is often the sole language spoken in their communities. Possible uses range from listening to audiobooks, reading out emails, and screen readers for visually impaired people. It also forms the basis for more advanced applications like question answering systems and virtual personal assistants.

This project may also serve as a reference for others who want to build a TTS system for a low-resource language with pre-existing textual and audio material using neural TTS models. It also serves as an example for other Jewish languages spoken around the world such as Ladino and various Judeo-Arabic varieties (Kahn & Rubin, 2017), which would encounter similar problems to Yiddish TTS, such as the pronunciation of Hebrew-origin words. Yiddish is a good first attempt at these problems because it is the most widely spoken non-Hebrew Jewish language, and has the most resources to use as training data.

1.1 Structure

This dissertation outlines the process of creating neural TTS voices for Yiddish, the associated challenges specific to the language, and the evaluation of the voices by native or fluent speakers.

The thesis starts with background information on Yiddish, low-resource TTS in general and challenges specific to Yiddish. Then, the dataset used in the proposed systems is described, including its source and any post-processing. The subsequent section outlines the neural model

used in this project, hypotheses for how the quality of the TTS system correlates with various factors, and the experimental design of the listening test deployed. The results of the test are aggregated and outlined, including qualitative feedback from an expert evaluator, as well as quantitative feedback in the form of ratings by Yiddish speakers who do not have prior experience in speech technology. The results are finally discussed and analysed, and possible future work is outlined.

2 Background

2.1 Status of Yiddish

Yiddish is a West Germanic language spoken mainly by Ashkenazi Jewish communities all over the world. Yiddish is estimated to be at its height before the Second World War, at 11 million speakers (Center for Applied Linguistics, 2012). The current number of speakers are estimated to be 371,000 (Eberhard et al., 2021), most notably in the New York metropolitan area (U.S. Census Bureau, 2015). Significant Yiddish-speaking populations also exist in the rest of the US, Canada, Israel, and Europe (YIVO Institute for Jewish Research, 2014). Although most Yiddish speakers are able to access language technology in more popular languages such as English, there are still some who prefer to access technology in their native language, be it Yiddish software user interface, text-to-speech, or ASR technology.

Yiddish is the most vital within Hasidic communities, such as those in New York City. A significant amount of underage residents of Hasidic neighbourhoods in the city still speak Yiddish according to census data, indicating that Yiddish is still actively passed onto younger generations (Comenetz, 2006), meaning that there is an increasing number of technology-literate people who will be the target users of Yiddish TTS technology. Therefore, it has much potential in developing into more advanced applications. In non-orthodox Jewish communities, where Yiddish speakers are more likely to have acquired it naturally without formal instruction, TTS technology can also help Yiddish speakers illiterate in Hebrew to understand written Yiddish.

2.2 Low-resource TTS

Low-resource languages are defined here as languages with little available data (e.g. written materials, video or audio recordings), usually associated with a small number of speakers and minority status in their respective regions. In the context of TTS, it also means the language has little to no TTS support. Such a language might have no existing TTS dataset available as training data for neural models, and might have few researchers developing technologies for it. If the language is not written in the Latin alphabet, the script might not be supported in existing TTS systems, posing significant challenges to building a TTS system from scratch for a language like Yiddish.

Tan et al. (2021) surveyed recent developments in neural TTS, and mentioned various techniques

for doing low-resource TTS, such as self-supervised training and cross-speaker transfer. The former can mitigate the lack of transcribed speech by using texts and audios from different sources, as Billa (2021) has also done for unsupervised ASR. These more advanced techniques are out of the scope of this project, since Yiddish has a significant amount of usable training data. Nevertheless, these extensions are worthy of investigation for even lower-resource languages (more details in Section 7.1).

2.3 Current state of Yiddish technological presence

There is often limited technology support for low-resource languages, and Yiddish is not an exception. Although Yiddish language technology is underdeveloped, some significant works touch on similar issues that can provide insight into developing speech technologies for Yiddish.

2.3.1 Text input

Yiddish typing was first supported by typewriters. Since Yiddish is written in the Hebrew script, Hebrew typewriters can be repurposed as Yiddish typewriters. It is still possible to type Yiddish without its special diacritics, which are not usually used in Hebrew (also known as *pointing*, as described in Section 2.4.1). Figure 1 shows an example of such a typewriter. They are commonly used to type Yiddish despite being a Hebrew typewriter (Hetko, 2021).

Yiddish typing on computers is also quite well-supported. They are abundant because they are fairly straightforward to create technically, so they are freely available for computers and mobile devices. The standard Yiddish script, based on the Hebrew script, is an alphabet with 46 letters, digraphs and trigraphs, many of which are composed of multiple letters, or letters with diacritics. Yiddish letters can also be reproduced with a Hebrew keyboard, but letters with diacritics (pointed letters) are less convenient to access, which means that Yiddish keyboards are still optimal for typing Yiddish. One example is Yiddish Klal (Bleaman, 2021), a QWERTY-based layout for typing Yiddish. It is easier to type letters such as *pasekh alef* אַ and *komets alef* אָ, which are absent in the standard Hebrew alphabet.

2.3.2 Language learning

Yiddish learning online is only supported to a small extent, compared to the amount of resources for learning major languages of the world. The California Institute for Yiddish Culture and Language (2017) collected various online resources for learning Yiddish, though most of them

Figure 1: An Erika Hebrew typewriter from the 1970s, part of the typewriter collection of the Yiddish Book Center. Image from Hetko (2021).

are short courses without much structure. Most structured classes are only available in-person instead of online and asynchronously. One relatively new resource for learning the language online is the Yiddish course on Duolingo, launched in April 2021 (Blanco & Moline, 2021). Since Duolingo is a developed language-learning platform, with over 500 million users as of 2020 (Blanco, 2020), having a course for Yiddish on Duolingo means that it is now able to receive funding and technical support from a big company recently publicly listed on the NASDAQ. Before the project, I have learnt some basic Yiddish on Duolingo, which taught me to read the script and construct basic sentences. Rudimentary knowledge in Yiddish made data processing significantly easier.

2.3.3 Machine translation

Genzel et al. (2009) describes the first machine translation (MT) system for Yiddish. While it is only tangentially related to TTS, it faces similar challenges as the text processing steps in a TTS pipeline, including the orthography, word origins and the computer encoding of Yiddish, which

are explained in more detail in Section 2.4. They also mention the lack of a high-quality OCR system for Yiddish, which makes digitising existing Yiddish text extra difficult, as explained in Section 3.1.

2.3.4 Speech technology

The work of Ćavar et al. (2016) is the most relevant to Yiddish TTS. They created a Yiddish speech corpus and an ASR system for the AHEYM Project (Kerler et al., 2021), which is an oral history project with an archive of videos with speech in Yiddish (and other languages). To aid the project, they developed an unpublished TTS module for Yiddish with eSpeak (Duddington, 2021), using the formant synthesis method. This method can generate speech by setting different parameters of the speech (including formants, fundamental frequency, and volume) are automatically generated using rule-based methods, without any human speech input. This makes the resulting voice sound extremely robotic, and inappropriate for applications where high naturalness is important. However, the paper only used the eSpeak TTS module for creating an alignment between the transcription (which can be converted into phonemes) and the audio with forced alignment, so the TTS voice is not intended to be heard by consumers. The voices created in this project are neural, therefore they are an improvement compared to the current "state of the art". However, since the eSpeak Yiddish module is not published, it is unable to serve as the baseline voice for the experiments in this project. The also unpublished ASR system would be a useful complement to the TTS system, although they went into very little detail about how the system was created.

2.4 Challenges

Making a TTS system for Yiddish presents unique challenges, because of the idiosyncrasies associated with the language, which are not shared by most other phylogenetically similar languages such as German.

2.4.1 Pointed letters

Yiddish is normally written with a modified version of the Hebrew alphabet. The standard Yiddish orthography makes use of Hebrew letters with diacritics, also known as *pointed letters*, which include אַ, אָ, בּ, בֿ, וּ, וֹ, יִ, יַ, כּ, פּ, פֿ, שׁ, and תּ. The diacritics can be used to disambiguate the sound it represents. For example, the diacritics in the letters *pey* פּ and *fey* פֿ are often omitted

(written as פּ), although the variant without the diacritic (unpointed) letter is prescribed to only represent *fey*. The vowels officially transcribed as *a* and *o* are written אַ and אָ respectively. The unpointed *alef* א is silent, indicating that the following letter should be pronounced with its vocalic form, which is useful when it is in front of the letter *yud* י, since it can be pronounced either as a vowel /i/ or a semivowel /j/. This can create problems when using the unpointed form would create ambiguity. For example, the unpointed word פאר can have four interpretations: *far* פַֿאַר "for", *for* פֿאָר "I go", *por* פָּאָר "pair", and *per* פּאר "magnificence", the last of which is a Hebrew-origin word, preserving the original spelling without phonemic respelling.

This practice of using the unpointed forms is quite common, including official publications such as books and newspapers, including the sources used for this project, described in Section 3.1. It is also common within Hasidic communities(Bleaman, 2020). This presents a challenge for a character-based TTS system, which would need to disambiguate the various pronunciations of an unpointed letter based on the context, i.e. the other words surrounding the word in question. A phone-based TTS system with a respelling dictionary would also need to disambiguate between these words and choose the correct reading, since if the lexicon is originally pointed, it needs to take into account the fact that the input text might be unpointed. One possible method of training a neural TTS system to be able to disambiguate these letters is to remove diacritics from (depoint) all pointed letters in the training data, and then train a character-based model. This is analogous to training a neural TTS engine for English where each character can correspond to multiple pronunciations, such as the pronunciation of the letter *c* in *car*, *cell*, and *cello*.

Besides the training data, it also presents problems during synthesis time, where users might input entirely unpointed text. This would not be a problem if the underlying TTS system was trained using unpointed data, but the input will suffer from a lack of pointing information if the training data were pointed. One possible solution is to look up the corresponding pointed forms, although it is error-prone as one unpointed word could possibly correspond to multiple pointed words. The input text into an unpointed TTS system must first be normalised and depointed before synthesis.

2.4.2 Hebrew-origin words

Being a language used by many Jewish communities, Yiddish has its fair share of words originating from Hebrew, which is the primary language used to write the Hebrew Bible. The

Hebrew script is an abjad, meaning that only consonants are normally indicated, and vowels are not considered full letters. When words from Hebrew are borrowed into Standard Yiddish, the spelling of these words are kept, without any respelling. Some examples include *sakh* סך "many/much" and *khaver* חבֿר "friend". Only the consonants in these words are written (*skh* and *khvr*), which means it is impossible to predict the correct vowels without prior knowledge of the pronunciation. A character-based TTS system would then have trouble mapping vowels to the glyphs of Hebrew-origin words, even more so than in English, where vowels are indicated no matter how irregular the orthography.

It is therefore necessary to either use a dictionary that maps from spelling to pronunciation, or convert the Hebrew-origin words into phonemic spelling, i.e. how they would be written if they were native Germanic words. This respelling has been accomplished before in Soviet Yiddish orthography, used between 1920 and the dissolution of the USSR (Estraikh, 2004). This orthography neutralised the difference between Hebrew- and Germanic-origin words, and spelt everything phonemically (Estraikh, 2004). For example, *shbt* שבת "Sabbath" is respelt *shabes* שאַבעס. In addition, it also converts all word-final allographs (ד, ם, ן, ף, ץ) to word-initial and -medial ones (כ, מ, נ, פּ, צ) (Estraikh, 2004). For example, *grenets* גרענעץ "border" was respelt גרענעצ. The Soviet orthography therefore serves as the inspiration for the phonemic spelling for modern Yiddish, without having to remove word-final allographs.

2.4.3 Unicode encoding

The computer encoding of Yiddish characters is standardised under the Hebrew section of Unicode (`U+0590`–`U+05FF`, and the Alphabetic Presentation Forms section (`U+FB00`–`U+FB4F`) (The Unicode Consortium, 2020). Some letters can be represented in two ways in Unicode. Pointed letters can be either displayed as one precomposed character, or two characters: the unpointed letter and a combining diacritic. For example, *tof* תּ can be represented as just `U+FB4A HEBREW LETTER TAV WITH DAGESH`, or a combination of *sof* ת (`U+05EA HEBREW LETTER TAV`) and the combining diacritic ּ `U+05BC HEBREW POINT DAGESH OR MAPIQ`. Certain digraphs can be represented as one combined digraph or two separate characters. *Tsvey vovn* וו "two vovs" can either be encoded as `U+05F0 HEBREW LIGATURE YIDDISH DOUBLE VAV`, or two consecutive ו `U+05D5 HEBREW LETTER VAV`s.

The Unicode Consortium has developed a system to formalise this variation in the representation

of digraphs and letters with diacritics, called the Unicode normalisation forms (The Unicode Consortium, 2020). The multiple ways of representing the same abstract character is defined as *canonical equivalence*, and so the two ways of encoding תּ are considered canonically equivalent. A piece of text with both ways can be normalised into Normalization Form C (NFC), where any precomposed characters are canonically decomposed, then recombined. For the purposes of our TTS system, all the text in the training data is normalised according to this formalisation, to ensure best performance for the character-based TTS model.

3 Dataset

3.1 Sources

A neural TTS system requires transcribed audio as training data. The two main sources of data used in this project are audiobooks from the Yiddish Book Center (2021) and those transcribed and compiled by researchers at the University of Haifa (Daniel et al., 2008).

The two books from the Yiddish Book Center are *A Shtetl* (Asch, 1904), and *Khayim Lederers Tsurikkumen* (Asch, 1927), both written by Sholem Asch in the early 20$^{\text{th}}$ century. Both books are narrated by Peretz Zilberberg at the Jewish Public Library in Montréal, Québec, Canada in the 1990s, and recorded on cassette tapes, later digitised by the Book Center. This speaker was chosen on the recommendation by Mindl Cohen and Amber Kanner Clooney from the Book Center, since Zilberberg's recordings are clear and reasonably high-quality. The audiobooks seem to be recorded in a studio, although there is the occasional audible page flipping and a moderate amount of print-through. This means that the cassettes were stored for a long time before they were digitised, so that other parts of the magnetic tape could be heard in some silent sections. From the average length of each raw audio file of about 80 minutes and the presence of print-through, the audiobooks were likely originally recorded on C180 tapes, characterised by its long total playing time (180 minutes on both sides) and print-through caused by the the tapes' thinness. However, it does not seem to affect the quality of the final output voice. The processed audio is 4 hours and 11 minutes long. The Book Center has more books narrated by Zilberberg, but they are not selected due to time constraints of the project. The Book Centre first scanned the books, then digitised them using the open-source Jochre optical character recognition (OCR) tool (Urieli, 2021). The quality of the transcription is still high, although it makes frequent mistakes in some cases. It is unclear how significantly this affects the performance of the resulting synthesised voices. The transcription is also often unpointed, such as using *alef* א to write the vowels אַ and אָ. A solution for this problem is proposed in Section 4.4.

The books compiled by Daniel et al. (2008) are short stories, novel excerpts, satires and biographies written by various Yiddish authors, including Mendele Moykher-Sforim and Sholem Aleichem (see Daniel et al. (2008) for the full list of works). The audiobooks are narrated by Sara Blacher-Retter, who was an Israeli nurse. The processed audio is 6 hours and 28 minutes long.

The audiobooks she narrated are much higher-quality than those by Zilberberg, presumably because Blacher-Retter's were recorded later, and there was no significant background noise. The transcriptions for these books are hand-typed, proofread, and has correct pointing. This contributed to the high quality of this dataset compared to the Book Center's.

3.2 Manual editing

Since the quality of both the audio and text for the books from Daniel et al. (2008) are all quite high, this section will mainly be about the preliminary data cleaning procedure for the two Book Center audiobooks.

The audio files for each book were divided into multiple files, about 75 to 80 minutes long each, corresponding to one side of the cassette. The beginning and end of each audiobook contain information about the Book Center and the recording session, with background music. These are first manually removed using Audacity, so that the audio files only contain the main content.

The OCR texts were downloaded directly from the Book Center's website. They are unedited outputs of the OCR software, therefore cleaning is needed. The beginning and end of the books contained materials not read aloud, and were thus removed. All page numbers, and page headers containing the book and author name were removed as well. Quotation marks, ellipses and pipes were removed, and the punctuation and spacing were formatted more consistently, e.g. putting no space before and one space after a comma.

3.3 Sentence segmentation

All the audiobooks are long audio files between 6 minutes and 80 minutes, therefore they need to be segmented into shorter clips when used for training because of the computer's memory limit. Aeneas (ReadBeyond, 2017) is first used to segment the transcript by labelling the audio with the starting and ending times of each sentence.

Aeneas works by first generating audio through the eSpeak module of a specific language, then aligning it with the original audio via dynamic time warping. The resulting output specifies the timestamps for each text segment, separated by line breaks. The line breaks were placed heuristically at the presence of a fullstop, which may possibly break up abbreviations, although none was found from a cursory manual inspection.

After adding line breaks, a Python script was written to romanise the text to resemble German orthography using a rule-based system, and then passed a German eSpeak module. The output audio is finally fed into Aeneas. Table 1 details the conversion, with an additional rule that converts clusters *schp* and *scht* into *sp* and *st* (the latter only when followed by vowels or *r*, as well as a heuristic conversion from *eup-* and *pun* to *euf-* and *fun*, disambiguating the pointed forms of פ. Table 2 shows an example sentence from the Universal Declaration of Human Rights in ordinary Yiddish orthography, the standard romanisation scheme devised by the YIVO Institute for Jewish Research (2021), and this heuristic German-like orthography.

Yiddish	Roman	Yiddish	Roman	Yiddish	Roman
א	∅	וי	eu	נ ן	n
אַ	a	ז	s	ס ת	ss
אָ	o	ח כ ך	ch	ע	e
ב בּ	b	ט ת	t	פ	p
בֿ וו	w	י יִ	i	פֿ פ ף	f
ג	g	יי ײַ	ei	ז	z
ד	d	כּ ק	k	ר	r
ה	h	ל	l	ש	sch
ו וּ	u	מ ם	m		

Table 1: Conversion table from Yiddish written in Hebrew characters to a German-like orthography.

Script/language	Sentence
Original Yiddish	יעדער מענטש ווערט געבוירן פֿרײַ און גלײַך אין כּבֿוד און רעכט
YIVO romanisation	yeder mentsh vert geboyrn fray un glaykh in koved un rekht
German-like spelling	ieder mentsch wert gebeurn frei un gleich in kwud un recht
German translation	Alle Menschen sind frei und gleich an Würde und Rechten geboren
English translation	All human beings are born free and equal in dignity and rights

Table 2: Example of Yiddish in the various orthographies and the corresponding German and English translations. *Koved* כּבֿוד is a Hebrew-origin word, and here the YIVO transcription indicates its actual pronunciation, whereas our rule-based script directly transliterates the Hebrew characters.

After alignment, the timestamps were verified and manually adjusted if necessary, although Aeneas performs very well with Yiddish audio and the romanised text despite only using a German eSpeak module. The proofreading process also helps to spot any mistakes present in the original transcription.

Two other alignment configurations were attempted. The Yiddish text is first romanised with the YIVO system, and both an English and German eSpeak module were used for alignment. Resulting alignment quality still seems to be high between the two configurations, although not as high as using the German-like romanisation as input, so the latter is chosen for the project.

Aeneas seems to give better alignments when the overall length of the audio is shorter, probably because one misalignment would often affect subsequent timestamps, so dividing them into smaller chunks "resets" the alignments and makes them more accurate. However, manually splitting a long audio file is time-consuming and error-prone, although the tradeoff might be worthwhile for more accurate alignments.

The OCR errors do not seem to negatively affect the quality of the output alignments, meaning that what the eSpeak system generates is enough to align correctly with ground-truth audio, even though transcription errors exist.

Once we have the aligned timestamps for every sentence, the audio and text files are segmented according to those timestamps. Recordings that are empty, too short (below 1 second), or two long (over 10 seconds) are removed. There are 3,079 resulting utterances for recordings by Zilberberg, and 7,879 utterances for Blacher-Retter.

3.4 Phone alignment

Alignment on a phone-level is needed to prepare the audio files for training a neural TTS system. However, since there is no pre-existing phone dictionary for Yiddish, a character-based system is used as a temporary solution, i.e. the "phones" of each word is simply each character of the word. For example, instead of having IPA on the right (ייִדיש j ɪ d ɪ ʃ), the dictionary file would have the entry with each separate letter as "phones" (ש י ד ִי י ייִדיש).

The Montreal Forced Aligner (MFA) (McAuliffe et al., 2017a) is used to segment the audio clips on both the word level and the phone (in this case character) level using forced alignment, based on the Kaldi toolkit (see McAuliffe et al. (2017b) for the specific implementation). The MFA outputs a TextGrid file which labels the appropriate starting and ending points for each word and character, which is used in FastSpeech 2 for training the TTS system.

3.4.1 Phonetic respelling

In addition to a purely character-based system, the experiments in this project also includes normalised text where Hebrew-origin words are respelt phonetically, similar to the Soviet orthography. E. Niborski (2021) compiled a dictionary of such a respelling, based on words listed in Y. Niborski (1999). The dictionary contains 7,648 entries, including root words, derived words and phrases. This dictionary is not built into a certain synthetic voice, but is instead

used for text pre-processing during synthesis time. For example, the Hebrew-origin word *kvud* כּבֿוד is respelt as *koved* קאָוועד "dignity". All words present in the dictionary are respelt prior to synthesis, so that they will be pronounced correctly using the character-based TTS model.

4 Model & Experiments

4.1 Model

The neural TTS system used in this project is FastSpeech 2 (Ren et al., 2021), which takes in raw text as training data and produces a Mel spectrogram as output, training a character-based model that learns a mapping between characters and a Mel spectrogram, which will finally be vocoded into a waveform. Compared to the original FastSpeech model, FastSpeech 2 trains on ground-truth audio, instead of a lossy version of it, and adds more features parameterising the audio, including pitch, energy and duration. The model also requires the input to be transcribed into phonemes, which the Montreal Forced Aligner provides, as described in Section 3.4.

The implementation used in this project is mainly based on the work of Chien (2021), with additional code from Pine (2021), which added support for customising the system to new languages. The voices used in this project are all trained with this implementation, using the segmented data from Section 3. A total of 4 voices are trained from the FastSpeech 2 model, which is expanded into 6 experimental conditions for evaluation (detailed in Section 4.4). All voices are trained on a computer using two NVIDIA GTX1060 6GB GPUs. Training the model for less than one day already produces very good results.

The vocoder used in this project is HiFi-GAN (Kong et al., 2020), which takes Mel spectrograms as input and then outputs waveforms using generative adversarial networks. The generator tries to upsample the Mel spectrograms to raw waveforms, while there are two discriminators, where one tries to discriminate narrower context and only allows regular periodic signals (such as in a vowel which can last up to a second), and the other deals with wider windows of duration.

4.2 Hypotheses

These experiments conducted for this project investigates the relationship between intelligibility and naturalness of the synthetic voices, and the 1) quality of the dataset (especially accuracy of the transcription), 2) pointedness of the Yiddish letters, and 3) the spelling of Hebrew-origin words.

First, higher dataset quality, including more accurate transcriptions and better audio quality, should translate to higher intelligibility and naturalness ratings, meaning that the content of the

output audio will be more understandable, and the resulting speech will be more human-sounding. Inaccurate transcriptions resulting from the imperfect OCR system will require more training data to cancel out, which may make the model learn wrong mappings between character and sound, resulting in mispronunciations in the final output. Low-quality audio will also affect the overall quality of the voice. Although it may not directly correspond to reduced naturalness, it will certainly translate to lower intelligibility if the training audio is noisy.

Second, if the transcription is depointed prior to trianing, it should be able to neutralise potential inconsistent pointing present in the training data, thus increasing the robustness of the character-to-Mel model, making the resulting voices higher-quality. It also has the advantage of allowing unpointed input during synthesis time, because a TTS system trained on entirely pointed text will fail to pronounce unpointed letters correctly. For example, if *alef* א exclusively corresponded to silence in the training data (since the vowels אַ and אָ would always pointed), it will not pronounce words with unpointed *alef* correctly.

Finally, pre-processing the text by respelling Hebrew-origin words normalises the orthography to be more phonemic, which should also at least increase intelligibility. The training data is limited in size, therefore it might not cover many Hebrew-origin words. Those that are present in the training data might be pronounced correctly, but the TTS system may fall short for unseen Hebrew-origin words. Having a separate respelling dictionary during synthesis time increases the robustness of the system and allows new words to be easily added, which increases the accuracy of pronunciation.

4.3 Voices

This project uses a total of 10 experimental conditions, from 4 uniquely trained neural voices (covering 6 experimental conditions), 2 eSpeak voices, and 2 ground-truth voices. Out of these 10 conditions, 5 belong to Zilberberg and the other 5 belong to Blacher-Retter. Each half contains either 1) actual human recordings, 2) a neural synthetic voice derived from, or 3) an eSpeak voice resembling each of the two speakers.

Table 3 summarises all 10 conditions. ZB and BB are the **b**aseline systems, where the pointing of the original text was unchanged (i.e. inconsistent pointing for Zilberberg and full pointing for Blacher-Retter). ZU and BU are the same as the baselines except the transcriptions are all **u**npointed, and digraphs were all separated as different characters (e.g. the digraph וו is

Code	Speaker	Unpointed	Respelt	eSpeak	Gold
ZB	Zilberberg				
ZU	Zilberberg	✓			
ZR	Zilberberg		✓		
ZE	Zilberberg			✓	
ZG	Zilberberg				✓
BB	Blacher-Retter				
BU	Blacher-Retter	✓			
BR	Blacher-Retter		✓		
BE	Blacher-Retter			✓	
BG	Blacher-Retter				✓

Table 3: All experimental conditions used in this project.

split into two characters וו). ZR and BR are the exact same voices as the baselines ZB and BB respectively, but they qualify as different experimental conditions since all the input text fed into ZR and BR are **r**espelt according to the dictionary from Section 3.4.1. ZE and BE are made using the German **e**Speak module, where sentences are generated by using the romanised version of the original Yiddish, as outlined in Section 3.3. ZG and BG are **g**old-standard audio, consisting of the original human recordings by the two speakers.

4.4 Experimental design

The TTS voices are evaluated with a listening test, which is split into two parts. The first part is taken by 8 adult Yiddish native or fluent speakers with normal hearing, and one expert evaluator who has linguistic knowledge in Yiddish. The second part is qualitative feedback from the expert, who can provide more holistic comments on the synthetic voices.

4.4.1 MUSHRA

The MUSHRA method is used in this listening test, a standard specified by the International Telecommunication Union (2015). Within the field of TTS, it is usually used to evaluate the naturalness of the same sentence read in variants of synthetic voices. In a MUSHRA test, listeners are first presented with a reference recording, which is usually a real human recording. Then the listener will be able to hear the same utterance read aloud in different voices, one of which being the same as the reference. They can rate each audio from 0% to 100%, and they are required to rate the reference as 100%, so that scores across listeners will be comparable.

The MUSHRA listening test used in this project is hosted on the Qualtrics XM Platform, a platform for online surveys. The main questions of the survey was generated with the Qualtreats

script (Webber et al., 2021), which automatically generates a MUSHRA test in the Qualtrics survey format which can then be imported onto the Qualtrics website.

4.4.2 Structure

Since much of the text used to train the Zilberberg voices are from OCRed versions of books, they are often inaccurate and/or unpointed. The text is manually proofread, and the diacritics are added back to the transcriptions with the help of a native speaker. This verification step is essential because it ensures the text being synthesised is written the most accurately, which improves the quality of the resulting pronunciations.

The listening test first asks two questions about the evaluator's speaker status, including whether they speak Yiddish at home or learnt it formally at an institution (such as a training programme), as well as the exact dialect(s) that they speak.

Evaluators are then asked 20 MUSHRA questions, 10 for each of the two speakers. For each question, participants are first presented with a human recording as a reference, alongside the corresponding YIVO-compliant (with correct pointing) transcription. Then they are presented with the same sentence read by all five voices of the same speaker including the reference (i.e. from ZB to ZG, and BB to BG as shown in Table 3). They are then asked to rate the intelligibility of the five recordings from 0% to 100%, where the reference must be rated 100%, as specified by the MUSHRA standard.

Afterwards, there are 40 further MUSHRA questions, 20 for each speaker, in the same format as the above 20 intelligibility questions. However, there is no transcription provided, as naturalness is instead the target of rating in this section.

Finally, participants can optionally give further comments on the voices, and describe how TTS technology can help their respective communities.

5 Results

5.1 Quantitative evaluation

8 respondents in total answered the survey. One evaluator's answers are discarded because they did not consistently rate the reference recordings as 100% as per the instructions. The expert evaluator, Eliezer Niborski, also took the survey besides giving qualitative comments.

Analysis of variance (ANOVA) tests are done to determine the statistical significance of the results, as recommended by Mendonça and Delikaris-Manias (2018). Four ANOVA tests are carried out for each combination of the two speakers (Zilberberg/Blacher-Retter) and rating criteria (intelligibility/naturalness), among all 7 valid non-expert evaluators. It is found that all four groups achieve a p-value of less than 2×10^{-16}, significantly smaller than the traditional 0.05 threshold. This means all the results are extremely significant despite the small number of evaluators.

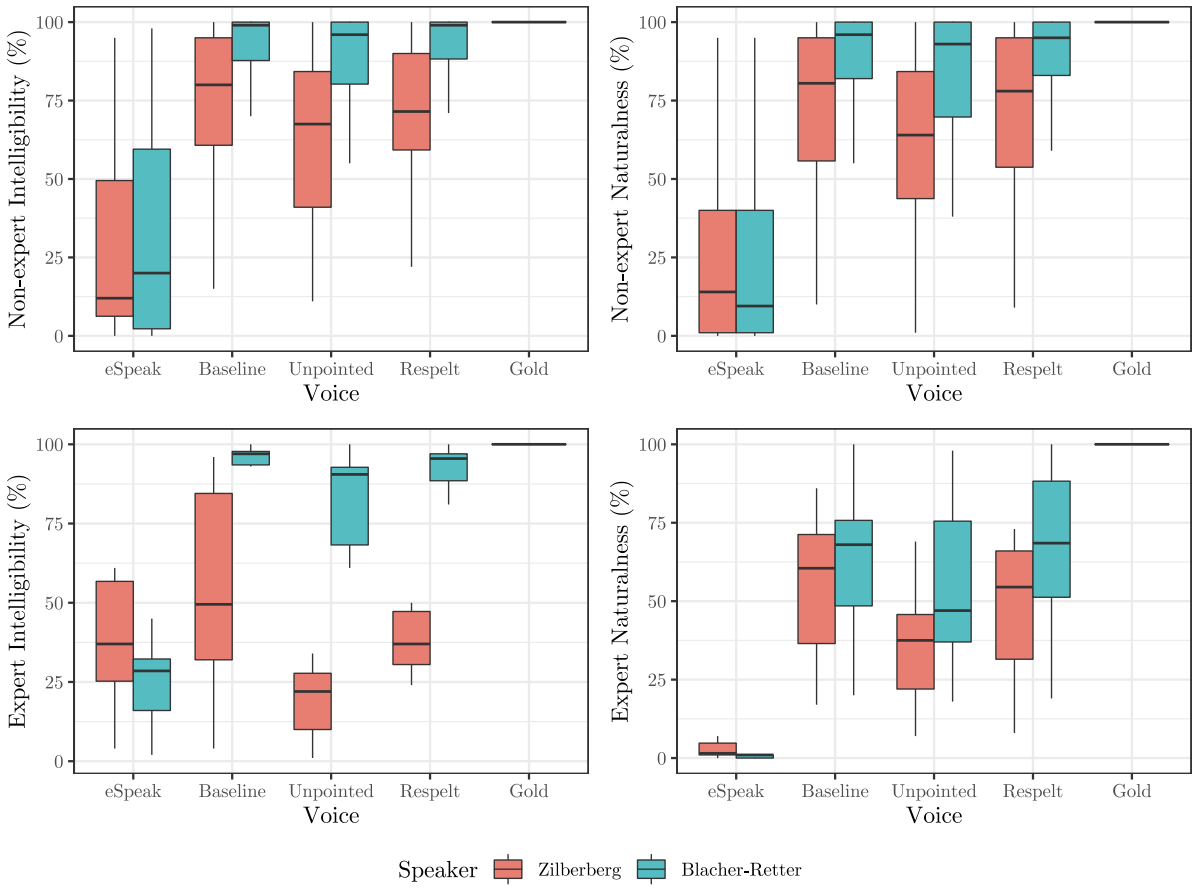

Figure 2: Box plot of the intelligibility and naturalness ratings from both non-experts (top) and the expert evaluator (bottom).

Figure 2 shows the intelligibility and naturalness ratings of all 10 voices of the listening test. The top two graphs show results from all valid non-experts. The bottom two graphs show the results for the one expert evaluator.

Both sets of responses follow a common pattern, where the eSpeak voices are rated the lowest by all evaluators, with no intelligibility or naturalness score higher than 40%, which is expected given that it is a parametric TTS system with no human recordings used as training data. All Gold ground-truth voices by definition get a 100% rating, although one evaluator excluded from the final results rated the Baseline voices higher than the human reference. Evaluators also show a larger variance in the ratings of the Zilberberg voices, indicating that either evaluators disagreed more with one another, or different test sentences received highly divergent ratings. On the other hand, ratings for the Blacher-Retter voices have lower variance, especially in terms of intelligibility.

The neural Blacher-Retter voices received better ratings than the neural Zilberberg voices across the board, which corroborates the first hypothesis stating that quality of the training set positively correlates with the resulting intelligibility and naturalness ratings.

For the neural Zilberberg voices, all evaluators agree that BU sounds the most intelligible and natural, the ZR is second and ZU is worst. The performance of the neural Blacher-Retter voices are similar to Zilberberg, but BR performs virtually the same as the BB. This result does not seem to support the second hypothesis, which states that unpointing the Zilberberg training data would neutralise the inconsistency in pointing and would thus boost the resulting system's quality. The fact that the Respelt voices perform worse on the baseline also does not support the third hypothesis, which says that respelling Hebrew-origin words will increase output speech quality.

The expert evaluator generally gave lower ratings than non-experts, where the expert gave an overall average rating of 58.87% for all 10 voices, compared to the 72.39% of non-experts. There is also a discrepancy between how non-experts and the expert rate intelligibility and naturalness. The difference between intelligibility and naturalness ratings for non-experts is 3.13% on average and 9.17% at most, whereas the difference for the expert is 15.05% on average and 30.55% at the maximum. It is evident that the expert tends to give harsher ratings, possibly because he is able to analyse the output quality more closely using his linguistic expertise, rather than

rating them solely by impression as most non-experts might. This also reveals that the voices are considered acceptable to the general population (at least a 70% for either metric for all of the neural voices). It also shows that non-experts might not fully understand the difference between intelligibility and naturalness, and instead only rated the general quality of the voices.

5.2 Qualitative evaluation

5.2.1 Non-experts

The end of the survey contains a text box that allows free-form responses from survey-takers. Evaluators who left comments generally states that the all the voices are generally clear to understand, with the Blacher-Retter voices being more comprehensible. Some evaluators noticed that the Zilberberg voices struggled with letters with pointed and unpointed variants (e.g. *f* ﬠ vs *p* פ), making certain sentences incomprehensible.

One person commented that they felt the intelligibility section should feature multiple options of transcripts that one can choose from, instead of being provided a transcript and asked to rate the intelligibility directly. They expressed that listening to different neural voices has increased their ability to comprehend the eSpeak voices, and they would not have understood them without listening to the other voices first.

Evaluators were also asked what applications they think Yiddish TTS technology can be used for. One person suggested using it in a language learning application (such as Duolingo), so that developers would not have to make human recordings for all new sentences, making the course more scalable. Another suggested application in education is helping students prepare for class, but only if the quality of the synthesised speech is high. Several people also suggested that it can be used for reading emails out loud, both for proofreading purposes as well as for visually impaired users. There are also more ambitious suggestions such as using the TTS system as part of a virtual home assistant system. Besides practical applications, some evaluators also pointed out that there are also symbolic benefits of having TTS technology available for Yiddish, putting it on par with higher-resource languages of the world.

5.2.2 Expert feedback

Main points from the expert's evaluation are summarised below. Please see the Appendix for the full feedback.

The quality of the voices are generally good, and they can capture the various pronunciations of allophones, such as those of /r/ and /l/. However, performance is better when the speaker dialect is closer to the written standard. The expert elaborated that Blacher-Retter's dialect (Lithuanian) is closer to the written standard than Zilberberg's (Polish), especially in the realisation of the *u* ו vowel. Minor mistakes are also present in the pronunciation of specific consonants and vowels in both speakers' voices.

Regarding the loshn koydesh component (another word for Hebrew-origin words used by the expert, meaning "holy language"), their pronunciations are mostly correct, especially for common words. However, it usually breaks down at compound words, since these words tend to be absent in the respelling dictionary. There are also many false positives when determining whether a word in the input text is from Hebrew. For example, *tsum* צום (normally meaning "to the") is an entry in the respelling dictionary, with the respelt form *tsom* צאָם "to fast". An instance of צום is more likely to be the grammar particle than the verb, but the respelling step of the TTS systems converts all instances to צאָם, resulting in the wrong pronunciation. Therefore, a disambiguation step is required to determine the correct reading.

The quality of the voices depend heavily on the quality of the orthography. For example, words with apostrophes and hyphens are seen as separate words instead of one, resulting in incorrect word stress. Inconsistent pointing also produces relatively low-quality pronunciations.

Finally, there are some issues with sentential words stress, where certain less idiomatic phrases (i.e. not as common or not present at all in the training data) are pronounced with stress on the wrong word. Additionally, the intonation of the sentences are not varied, because the training data only consists of narration and not more spontaneous conversation.

6 Discussion

6.1 Dataset quality

It is evident from the results that a higher-quality dataset corresponds to higher intelligibility and naturalness ratings. This is probably caused by the same reason as stated in the hypothesis in Section 4.2, that the overall textual quality of the Blacher-Retter dataset is higher due to an extra manual proofreading step. The expert also pointed out more specific problems in pronunciation in the Zilberberg voices than the Blacher-Retter voices. For example, the letter *ayin* ע before a word-final *lange nun* ן represents an actual vowel /e/ in the modern Yiddish orthographic standard. However, the books that Zilberberg narrated might be in an older non-standardised orthography, where the *ayin* ע is often silent. This inconsistency in the orthographic convention has trained the model wrongly, and it somehow skipped all *ayin* ע before a *lange nun* ן, even in monosyllabic words where the letter is always pronounced, such as *men* מען "one" and *ken* קען "I know". On the other hand, the texts read by Blacher-Retter all conform to the YIVO standard orthography, so the same problems are not present in those voices.

6.2 Pointedness

The results shown in Section 5 are contrary to the hypothesis. Removing the diacritics from the training data and the input text (as well as dividing digraphs into two separate characters) universally reduces both the intelligibility and naturalness of the resulting voices, which is unexpected. A non-expert evaluator pointed out that the Zilberberg voices have difficulty pronouncing the letter *pey/fey* פ correctly, although it is unsure whether they are referring to the Baseline voice or the Unpointed voice.

A possible explanation that it performs worse for the Zilberberg voices is that the quality of the transcription is already low, and depointing the input data does not change that fact that the transcriptions are often inaccurate. Without manually correcting the transcription, processed noisy data will still yield noisy data. Depointing also removes information from the transcriptions that are pointed, and the effect of this neutralisation of the inconsistency in pointing is not enough to offset the low quality of the original transcripts.

A similar issue possibly exists in the BR voice. The transcripts are completely human-typed

and verified, therefore there is no need to depoint the training data, since the pointing is already consistently present. Depointing only removes data that would otherwise be used to disambiguate pronunciation, therefore results in worse intelligibility scores.

The importance of correct pointing might have been previously overestimated, since the difference between the Baseline and Unpointed voices does not seem to be as significant as the inter-speaker difference.

6.3 Respelling

The results for this section also go against the hypothesis originally set out. Respelling is supposed to increase intelligibility for Hebrew-origin words, especially those not present in the training data, because the TTS system would not otherwise have a good idea of how those words should be pronounced. Without a simple respelling dictionary to convert the Hebrew spelling to more phonemic spelling usually reserved for Germanic words, the model would require knowledge in Hebrew to have any hope of having enough data to sufficiently guess the pronunciation of an unseen Hebrew-origin word.

However, the main problem with the current respelling algorithm is that it always replaces a word as long as it is in the dictionary, creating many false positives, as explained in Section 5.2.2. To mitigate this problem, a more sophisticated algorithm can be created to decide when a word should be respelt. One more accurate but also more sophisticated method is to disambiguate pronunciation by its part-of-speech (POS) tag. A POS tagger trained from Yiddish text hand-annotated with POS tags can effectively label each word with the correct POS tag. This method is time-consuming, but it will significantly improve pronunciation accuracy for Hebrew-origin words, since many ambiguous words can be disambiguated if the POS is known. For example, the word *mid* מיד "tired" is an adjective, but it is a homograph with the Hebrew-origin word *miyad* מיד "immediately" (respelt as מיִיאַד phonemically), which is an adverb. A POS tagger that can distinguish these two POS will be able to disambiguate these two readings. However, it still would not work in some cases, such as for קין, which could be the native word *kin* "chin" or the Hebrew name *Kayn* "Cain", which are both nouns. It would require a tagger that is able to identify named entities as well.

6.4 Other limitations

Word stress and prosody are two features that are harder to train correctly with the current dataset, which is a limitation of the current TTS voices. The dataset exclusively contains narration of prose and novels, which is dominated by one kind of intonation, i.e. that of carefully read speech. A dataset with more spontaneous recordings and texts from wider domains will be essential for developing a TTS system with more variation in prosody.

Another limitation of this project comes from the fact that it has a developer who is not a Yiddish speaker. Having a non-speaker developer makes it substantially harder to process the data efficiently and spot mistakes. Being proficient in Yiddish enables one to be aware of mistakes otherwise not noticeable. It is generally not easy to always involve fluent speakers in speech and language processing studies in minority languages, since the pool of people who both speak the language and have knowledge in language technology is drastically smaller.

7 Conclusion

The quality of the neural voices built for this project are generally quite high, with the Blacher-Retter voices sounding better because of its high quality human-typed transcriptions, as opposed to the scanned books Zilberberg narrated. It is demonstrated that 6 hours of high-quality audio can already produce a good-sounding voice with FastSpeech 2, which gives hope for other low-resource languages if they wish to develop their own TTS systems.

It is also found that contrary to previous beliefs, removing diacritics and naïvely respelling Hebrew-origin words do not necessarily yield better-sounding voices. The quality of the resulting voices mainly depends on the transcription quality. Hopefully, technologies like this TTS system will be able to inspire more work in Yiddish TTS and other low-resource languages.

7.1 Future work

Although the quality of the resulting TTS systems is satisfactory according to the listening tests, future advancements could be added to further improve the intelligibility and naturalness of the systems. The one with the most immediate gain is a normaliser for numbers and abbreviations, which the current system cannot cope with. Only a simple transducer is needed for Yiddish numbers to normalise them from digits to words (e.g. *12* to *tsvelf* צוועלף) since Yiddish numbers do not inflect by gender or case. Normalising abbreviations is also fairly straightforward, although possible disambiguation is needed if one abbreviation corresponds to multiple meanings. The dataset used in this project contains a number of abbreviations, which could be used to construct a normalisation dictionary for future use.

Model-wise, transfer learning from another bigger language such as German is possible. The phonology of Yiddish and German have many similarities, therefore finetuning a pre-trained German TTS model with Yiddish data is a possible extension. Multi-speaker systems would also be able to increase the robustness of the system regarding synthesising different accents. A voice with little data could be synthesised if its model parameters are shared with other speakers'. One use case would be building a Hasidic voice by fine-tuning a pre-trained Yiddish TTS model with a small amount of data from a Hasidic speaker.

Additionally, the dataset used in this project can be further standardised for future use in more accessible pre-defined formats, such as following the formats of the LJ Speech Dataset (Ito

& Johnson, 2017) or the VCTK Corpus made by the Centre for Speech Technology Research (Yamagishi et al., 2019).

Bibliography

Asch, S. (1904). *A shtetl* [Audiobook]. Narrated by Peretz Zilberberg, Jewish Public Library 1993. Retrieved July 13, 2021, from https://www.yiddishbookcenter.org/collections/audio-books/smr-257SholemAschAShtetl.CD1Of4.ReadByPeretzZilberbergYID

Asch, S. (1927). *Khayim lederers tsurikkumen* [Audiobook]. Narrated by Peretz Zilberberg, Jewish Public Library 1997. Retrieved July 13, 2021, from https://www.yiddishbookcenter.org/collections/audio-books/smr-sholem-asch-khayim-lederers-tsurikkumen-000

Billa, J. (2021). Improving low-resource asr performance with untranscribed out-of-domain data.

Blanco, C. (2020). *2020 Duolingo Language Report: Global Overview*. Retrieved August 18, 2021, from https://blog.duolingo.com/global-language-report-2020/

Blanco, C., & Moline, E. (2021). *Yiddish is now on Duolingo!* Retrieved August 18, 2021, from https://blog.duolingo.com/yiddish-is-now-on-duolingo/

Bleaman, I. (2020). Implicit standardization in a minority language community: Real-time syntactic change among Hasidic Yiddish writers. *Frontiers in Artificial Intelligence*, *3*, 35. https://doi.org/10.3389/frai.2020.00035

Bleaman, I. (2021). *Typing in Yiddish on a Mac*. Retrieved July 21, 2021, from https://www.isaacbleaman.com/resources/yiddish_typing

California Institute for Yiddish Culture and Language. (2017). *Yiddish Resources*. Retrieved August 18, 2021, from https://yiddishinstitute.org/yiddish-resources/

Ćavar, M., Ćavar, D., Kerler, D.-B., & Quilitzsch, A. (2016). Generating a Yiddish speech corpus, forced aligner and basic ASR system for the AHEYM project, In *Proceedings of the tenth international conference on language resources and evaluation (LREC'16)*, Portorož, Slovenia, European Language Resources Association (ELRA). https://aclanthology.org/L16-1744

Center for Applied Linguistics. (2012). *Heritage language spotlights: Yiddish*. Retrieved August 14, 2021, from https://www.cal.org/heritage/yiddish.html

Chien, C.-M. (2021). FastSpeech 2 - PyTorch Implementation. GitHub. Retrieved August 8, 2021, from https://github.com/ming024/FastSpeech2

Comenetz, J. (2006). Census-based estimation of the Hasidic Jewish population. *Contemporary Jewry*, *26*, 35–74. https://doi.org/10.1007/BF02965507

Dagba, T. K., & Boco, C. (2014). A text to speech system for Fon language using multisyn algorithm [Knowledge-Based and Intelligent Information & Engineering Systems 18th Annual Conference, KES-2014 Gdynia, Poland, September 2014 Proceedings]. *Procedia Computer Science*, *35*, 447–455. https://doi.org/https://doi.org/10.1016/j.procs.2014.08.125

Daniel, T., Sheinwald, D., Freer, J. P., Goldenberg, R., & Prager, L. (2008). Di velt fun Yidish: Audio stories. Narrated by Sara Blacher-Retter. Retrieved July 13, 2021, from http://yiddish.haifa.ac.il/Stories.html

Duddington, J. (2021). *eSpeak text to speech*. Retrieved July 23, 2021, from http://espeak.sourceforge.net

Eberhard, D. M., Simons, G. F., & Fennig, C. D. (Eds.). (2021). *Ethnologue: Languages of the World* (24th ed.). Dallas, TX, USA, SIL International. http://www.ethnologue.com

Estraikh, G. (2004). *Soviet Yiddish: Language planning and linguistic development*. Clarendon Press. Retrieved July 23, 2021, from https://oxford.universitypressscholarship.com/view/10.1093/acprof:oso/9780198184799.001.0001/acprof-9780198184799

Genzel, D., Macherey, K., & Uszkoreit, J. (2009). Creating a high-quality machine translation system for a low-resource language: Yiddish.

Gutkin, A., Ha, L., Jansche, M., Pipatsrisawat, K., & Sproat, R. (2016). TTS for low resource languages: A Bangla synthesizer, In *Proceedings of the tenth international conference on language resources and evaluation (LREC'16)*, Portorož, Slovenia, European Language Resources Association (ELRA). https://aclanthology.org/L16-1317

Hetko, A. (2021). *The Story Behind the Yiddish Book Center's Yiddish Typewriter Collection*. Retrieved August 17, 2021, from https://www.yiddishbookcenter.org/language-literature-culture/vault/story-behind-yiddish-book-centers-yiddish-typewriter-collection

International Telecommunication Union. (2015). *Recommendation ITU-R BS.1534-3: Method for the subjective assessment of intermediate quality level of audio systems*. Retrieved August 12, 2021, from https://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1534-3-201510-I!!PDF-E.pdf

Ito, K., & Johnson, L. (2017). The lj speech dataset.

Kahn, L., & Rubin, A. D. (2017). *Handbook of Jewish languages: Revised and updated edition.* Brill. https://books.google.co.uk/books?id=3IJ1DwAAQBAJ

Kerler, D.-B., Veidlinger, J., Lemster, M., Quilitzsch, A., Vaisman, A., Valles, M., & Schulman, S. (2021). *The Archives of Historical and Ethnographic Yiddish Memories.* Retrieved July 23, 2021, from https://aheym.com

Klatt, D. H. (1987). Review of text-to-speech conversion for english. *Acoustical Society of America Journal*, *82*(3), 737–793. https://doi.org/10.1121/1.395275

Kong, J., Kim, J., & Bae, J. (2020). HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis.

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., & Sonderegger, M. (2017a, January 17). *Montreal Forced Aligner* (Version 0.9.0). http://montrealcorpustools.github.io/Montreal-Forced-Aligner

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., & Sonderegger, M. (2017b). Montreal Forced Aligner: trainable text-speech alignment using Kaldi, In *Proceedings of the 18th Conference of the International Speech Communication Association.*

Mendonça, C., & Delikaris-Manias, S. (2018). Statistical tests with MUSHRA data.

Niborski, E. (2021). List of Hebrew- or Aramaic-origin words in Yiddish spelt phonetically. Retrieved July 21, 2021, from https://editions.yiddish.paris/wp-content/uploads/2021/06/%D7%A4%D6%BF%D7%90%D6%B8%D7%A0%D7%A2%D7%98%D7%99%D7%A9%D7%A2%D7%A8-%D7%90%D7%99%D7%A0%D7%93%D7%A2%D7%A7%D7%A1.pdf

Niborski, Y. (1999). *Verterbukh fun loshn-koydesh-shtamike verter in yidish [Dictionary of Hebrew- and Aramaic-origin words in Yiddish].* Paris, France, Bibliothèque Medem.

Pine, A. (2021). FastSpeech 2 - PyTorch Implementation. GitHub. Retrieved August 8, 2021, from https://github.com/roedoejet/FastSpeech2

Rama, G. L. J., Ramakrishnan, A. G., Muralishankar, R., & Prathibha, R. (2002). A complete text-to-speech synthesis system in tamil, In *Proceedings of 2002 ieee workshop on speech synthesis, 2002.* https://doi.org/10.1109/WSS.2002.1224406

ReadBeyond. (2017, March 15). *Aeneas* (Version 1.7.3). https://www.readbeyond.it/aeneas

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.

Tan, X., Qin, T., Soong, F., & Liu, T.-Y. (2021). A survey on neural speech synthesis.

The Unicode Consortium. (2020). *The Unicode Standard, Version 13.0*. The Unicode Consortium. Retrieved July 25, 2021, from http://www.unicode.org/versions/Unicode13.0.0/

Urieli, A. (2021). Java Optical CHaracter Recognition. GitHub. Retrieved August 17, 2021, from https://github.com/urieli/jochre

U.S. Census Bureau. (2015). Detailed Languages Spoken at Home and Ability to Speak English for the Population 5 Years and Over for New York-Newark-Jersey City, NY-NJ-PA: 2009-2013. Retrieved July 21, 2021, from https://www.census.gov/data/tables/2013/demo/2009-2013-lang-tables.html

Webber, J., Williams, E., & Wells, D. (2021). Qualtreats. GitHub. Retrieved August 12, 2021, from https://github.com/CSTR-Edinburgh/qualtreats

Yamagishi, J., Veaux, C., & MacDonald, K. (2019). CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (version 0.92). Centre for Speech Technology Research (CSTR), University of Edinburgh. Retrieved August 18, 2021, from https://doi.org/10.7488/ds/264

Yiddish Book Center. (2021). Retrieved July 21, 2021, from https://www.yiddishbookcenter.org

YIVO Institute for Jewish Research. (2014). *Basic Facts about Yiddish*. Retrieved August 16, 2021, from https://www.yivo.org/cimages/basic_facts_about_yiddish_2014.pdf

YIVO Institute for Jewish Research. (2021). *Yididsh Alef-beys (Alphabet)*. Retrieved July 31, 2021, from https://www.yivo.org/yiddish-alphabet

Appendix: Qualitative evaluation from expert

The results of the text-to-speech prototype are quite impressive. Even without particular preparation a paragraph written in correct standard Yiddish is rendered in a way that is mostly intelligible and even sounds here and there quite natural.

The two voices (Perets and Sara) representing two different Yiddish dialect backgrounds are immediately recognizable.

In both cases most individual sounds are rendered quite accurately. A special good job has been done with the distinct [r] sounds of the two model speakers, the nuances of [l], the natural assimilation of neighboring similar sounds, the particular renditions of the final nun (after a consonant, after [g]/[k], after [b]/[p]), some characteristic features of the vowels/diphthongs (for instance Sara's [emitsn]). Since the human models are not speaking freely according to their usual dialects, but instead reading from a literary text, they are themselves influenced in their pronunciation by the written form of that text. It's no surprise that their digital counterparts don't follow either the strict lines of a well-defined Yiddish dialect. This is less obvious for Sara, since her dialect has broadly a better vowels correspondence with normalized written Yiddish, and more noticeable for Perets (confusions [du]/[di], [op]/[up], [gevust]/[gevist]).

Some sounds present specific problems:

1. זש which should be pronounced [zh] sounds like [sh] for Sara, and [z-sh] for Perets.
2. יַי is for some reason pronounced [i] by Perets, or skipped completely. There must be some problem in the coding, because it appears to be quite systematic: zaydene, fayer, vaysn, shtraymlekh, vayb, zayn, taynet, dayge...
3. The ע before a final nun represents in principle a real sound [e], if the given text follows the modern rules of Yiddish spelling. The human model for Perets was probably reading from a text written according to older orthographic standards, with possible silent ע before a final syllabic nun, and the person allowed some variation in the rendition of that letter combination: [funkn]/[tsvangen]. But for some reason, the digital version skips almost every ע before a final nun, which is obviously wrong in monosyllabic words, independently of any orthographic conventions. (זען, קען, דען, וועט, מען, לען, טרען, ברען,

etc...)

This question of -ען concerns also Sara ([farshmurn] instead of [farzhmuren]).

4. The word אויף should receive special treatment since it's one of the most salient deviations – except for the Loshn-Koydesh component – from phonetic spelling in Yiddish.

The treatment of the Loshn-Koydesh component, which is not written phonetically and requires a transcription list for virtually every word is quite interesting and does a great job in identifying a lot of simple words. Globally it seems to be working better with Sara than with Perets.

Let me only point out a few general problems that will require improvement:

1. The program doesn't handle well compound words, recognizing only part of the compound if at all.

2. It assumes by default that "recognized" words are actually LK and should be pronounced according to the LK-table. But for some often occurring homographic short words, this assumption is wrong most of the time: אין, צום, בין, ברי, מיד, קין, שער, מעגן...

3. You can expect some more cases of homographs (not found in the samples) which will probably require context related calculations in order to guess the proper pronunciation (סוד1/סוד2; סמך1/סמך2; זכר1/זכר2; צער1/צער2)

4. It's a pity the phonetic list of LK-words used for the "respelling" doesn't reflect the stress information contained in the original dictionary. This would certainly be of good use to realize a more credible pronunciation of certain words.
   In some cases the "respelt" results are worse than the "baseline" : [khanu'ke] instead of the correct [kha'neke]

5. Since the transcription table contains only the "standard pronunciation" (close to Sara's) of the LK words, it could be useful for Perets to include parts of the phonetic information contained in: Waynraych, M., Shrayb on grayzn (oysgabe B: farn poylishn dialect), 1926

Concerning the rendition of the written characters, a few more details could easily be

improved:

1. An apostrophe is interpreted as a sign separating words. This produces a wrong-sounding rhythm. It would be better to simply erase the apostrophe (and the possible silent letter א or ה following it) and concatenate the two words. Examples: ס'איז, ס'האָט, מ'האָט should sound [siz], [sot], [mot].

2. Something very similar happens with hyphens: they should bring words closer together instead of separating them (khanuke-likhtlekh, erets-yisroel-erd, ergets-vu, beshum-oyfn).

Some special instances suggest there is something to check, but they would require more testing to figure out what is going on:

1. Sara renders the word בוימל [boyml] as [boymele].

2. Perets skips the stressed ע in קליענטקע and has it as [klintke].

From a very small sample of test sentences, it appears that the results are heavily dependent on the orthographic quality of the text. Missing diacritics and spelling variation introduce noticeable and unwanted changes in the rendition of the text, whereas many words should be recognized anyway and pronounced unequivocally: קיינמאָל = קיין מאָל, לאמיר = לאָמיר, וואו = וווּ = וו = וו

After dealing with pronunciation, a few questions arise concerning stress and intonation.

Some words are surprisingly well weighted in their sentences and give a good natural feeling: אויך, אַזוי, גאַנץ, דעמאָלט, בפֿירוש, גרעסערע.

Some others seem to be part of recognized phrases with well-established stress structures: אין יעדן פֿאַל, קיין מאָל נישט...

But some details make a weaker impression:

1. The fundamental feature of strongly stressed coverbs is not always implemented in the fullest manner. It would be nice to hear consistently: [oy'falndik], [o'ystsukemfn] , [tsu'tsushteln] and [aro'ystsubrengen], and not [oyffa'lndik], [o'ystsuke'mfn], [tsu'tsushte'ln], [aro'ystsubre'ngen].
   Even more so when a coverb is separated (but then it is of course much more difficult to recognize as a coverb): [du shraybst mir tsu']

2. In contrast, the verbal prefixes shouldn't get any stress at all. It's the case most of the time but not always ([ba'tamt]).

3. The addition of substantive suffixes also perturbs the stress pattern, when it shouldn't. One would expect to hear [shve'rikayt] and [ekspe'rtshaft] instead of [shve'rika'yt], [ekspe'rtsha'ft].

4. Some words of the "international component" seem to have a wrong or fuzzy stress: [te'rmin], [pre'tsize], [termino'logye], [li'ngvi'st]

The punctuation of a sentence, which gives usually the main clues as to the right intonation, is being taken into account only partially. Actually, only the slight pause at a comma is being implemented and it already helps a lot in structuring the sentences. But it would be very useful to have question marks and other signs influence in some way the rendition of a sentence.

As a conclusion, here are some more expectations for further developments:

1. Take into consideration the punctuation marks (questions, exclamation, ellipsis...)

2. Reduce reliance, where possible, on orthographic accuracy (missing vowel points, superfluous silent letters).

3. Manage to interpret numbers written with digits.

4. Recognize some frequently used abbreviations: the LK ones, as well as ד"ר, פּראָפּ', פּר', מ"ר, מר"ס, א"אַ, אאַז"וו, צב"ש and so on.