Low-Resource Multilingual and Zero-Shot Multispeaker TTS

Florian Lux and **Julia Koch** and **Ngoc Thang Vu**
University of Stuttgart
florian.lux@ims.uni-stuttgart.de

Abstract

While neural methods for text-to-speech (TTS) have shown great advances in modeling multiple speakers, even in zero-shot settings, the amount of data needed for those approaches is generally not feasible for the vast majority of the world's over 6,000 spoken languages. In this work, we bring together the tasks of zero-shot voice cloning and multilingual low-resource TTS. Using the language agnostic meta learning (LAML) procedure and modifications to a TTS encoder, we show that it is possible for a system to learn speaking a new language using just 5 minutes of training data while retaining the ability to infer the voice of even unseen speakers in the newly learned language. We show the success of our proposed approach in terms of intelligibility, naturalness and similarity to target speaker using objective metrics as well as human studies and provide our code and trained models open source.

1 Introduction

The applications of modern TTS systems are omnipresent and bring major benefits in a very diverse range of tasks. For example, low-resource TTS can be used to revitalize and conserve languages with diminishing numbers of speakers (Pine et al., 2022). Other recent applications go into the direction of protecting the privacy of a speaker, by exchanging their voice for a different voice, while not affecting the content of what is said (Meyer et al., 2022). Even in literary studies, TTS systems can be applied to investigate perceptive aspects of poetry reading (Koch et al., 2022). However, while the first of those examples can be done with just a single speaker, the latter two require the TTS system to be able to exchange the voice of the utterance that is produced, which usually requires large amounts of clean multispeaker data. The same requirement exists for many other such applications, which can also be seen in the rise of interest in the research community on voice-cloning technologies (Wu et al., 2022; Casanova et al., 2022; Neekhara et al., 2021; Hemati and Borth, 2021; Cooper et al., 2020). The communities of speakers of low-resourced languages are thus mostly locked out of plenty of the applications that modern TTS enables. For many instances of such languages, like the Taa language, which is famous for its 83 click sounds or the Yoruba language, in which the tones bear so much meaning, that the language can be mostly whistled, it would be extremely difficult to collect the required amounts of data, and transfer learning to such unique languages is very challenging. Still, we believe that a single model that speaks many languages with any voice can exhibit strong generalizing properties and is a promising first step towards fixing these inequalities.

In this work we ask the following question: Can a multilingual TTS system be used to achieve zero-shot multispeaker TTS in a low-resource scenario? Our approach is to use crosslingual knowledge-sharing to enable 1) finetuning a TTS on just 5 minutes of data in an unseen language in an unseen branch in the phylogenetic tree of languages and 2) transferring zero-shot multispeaker capabilities from the pretraining languages to the unseen language. To achieve this, we propose changes to a TTS encoder to better handle multilingual data and disentangle languages from speakers. Further, we show that the LAML pretraining procedure (Finn et al., 2017; Lux and Vu, 2022) can also be used to train general speaker-conditioned models. To verify the effectiveness of our contributions, we train models on just 5 minutes of German and Russian while excluding all Germanic and Slavic languages from the pretraining respectively. We choose a simulated low-resource scenario over an actual low-resource scenario in order to get more reliable evaluations using both objective measures as well as human studies. Furthermore, we show that models trained with this approach do not only serve

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 741–751
November 20–23, 2022.

as a basis for low-resource finetuning with greatly reduced data-need, they can also be used without finetuning as strong multispeaker and multilingual models. We train a model on 12 languages simultaneously and show that it can transfer speaker identities across all languages, even the ones where it has only seen a single speaker during training.

All of our code, as well as the trained multilingual model are available open source[1]. An interactive demo[2] and a demo with pre-generated audios[3] are available.

2 Related Work

2.1 Zero-Shot Multispeaker TTS

Zero-shot multispeaker TTS has first been attempted in (Arik et al., 2018). The idea of using an external speaker encoder as conditioning signal was further explored by (Jia et al., 2018). (Cooper et al., 2020) attempted to close the quality gap between seen and unseen speakers in zero-shot multispeaker TTS using more informative embeddings. With the use of attentive speaker embeddings for more general speaking style encoding (Wang et al., 2018; Choi et al., 2020) as well as different decoding approaches in the acoustic space such as generative flows (Casanova et al., 2021), further attempts have been made at closing the quality gap between seen and unseen speakers. This is however still not a fully solved task. Furthermore, zero-shot multispeaker TTS requires a large amount of high quality data featuring many different speakers to cover a variety of voice properties.

2.2 Low-Resource TTS

In some languages, even a single speaker TTS is not feasible due to the severe lack of high-quality training data available. Attempts at enabling TTS on seen speakers in low-resource scenarios have been made by (Azizah et al., 2020; Xu et al., 2020; Chen et al., 2019) through the use of transfer learning from multilingual data, which comes with a set of problems due to the mismatch in the input space (i.e. different sets of phonemes) when using multiple languages. Training a model jointly on multiple languages to share knowledge across languages has been attempted by (He et al., 2021; de Korte et al., 2020; Yang and He, 2020). One solution to the problem of sharing knowledge across different phonemesets is the use of articulatory features, which has been proposed in (Staib et al., 2020; Wells et al., 2021; Lux and Vu, 2022).

2.3 Multilingual Multispeaker TTS

The task of multilingual (not even considering low-resource languages) zero-shot multispeaker TTS is mostly unexplored. YourTTS (Casanova et al., 2022) claims to be the first work on zero-shot speaker transfer across multiple languages and was developed concurrently to this work. At the time of writing, there is only a preprint available, so our comparison to their model and methods may differ to a later version. YourTTS reports similar results to ours on high-resource languages using the VITS architecture (Kim et al., 2021) with a set of modifications to handle multilingual data. The authors find that their model doesn't perform as well with unseen voices in languages that have only seen single speaker training data. Through the low-resource focused design, our approach does not exhibit this problem, while being conceptually simpler. It is shown that just one minute of data suffices to achieve very good results in adapting to a new speaker in a known language with YourTTS. This is consistent with our results, however we go one step further and show that 5 minutes of data is enough to not only adapt to a new speaker, but also to a new language. Also consistent with their results we see that the speaker embedding learns to attribute noisy training data to certain speakers, so not all speakers perform equally well. Ideally we would want to also disentangle the noise modeling from the speakers and languages. The GST approach (Wang et al., 2018) has shown that disentangling noise from speakers is possible, it is however not trivial to also disentangle languages, since language properties are also relevant to the encoder, not only the decoder.

Finally, combining the task of zero-shot multispeaker TTS with the task of low-resource TTS has to the best of our knowledge only been attempted once in a very recent approach that was developed concurrently to ours (Azizah and Jatmiko, 2022). Their system uses a multi-stage transfer learning process, that starts from a single speaker system which is expanded with a pretrained speaker encoder. They add the required components for speaker and language conditioning

[1] https://github.com/DigitalPhonetics/IMS-Toucan

[2] https://huggingface.co/spaces/Flux9665/IMS-Toucan

[3] https://multilingualtoucan.github.io/

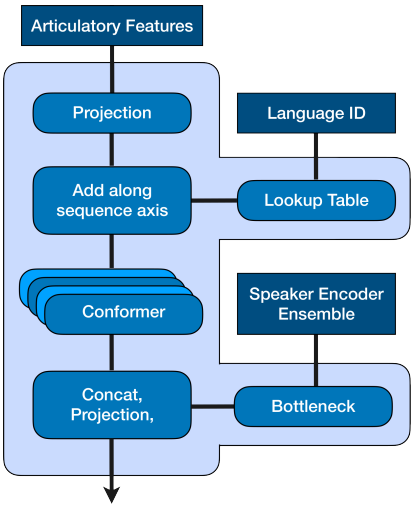

Figure 1: Overview of the encoder design. All of the projections project to the same dimensionality, which we chose to be 384. Round corners mean trainable. Conformer blocks include relative positional encoding.

and apply finetuning to only those parts of the architecture. The main difference of our system to theirs is that we train the full architecture jointly on the high-resource source domain using the LAML pretraining procedure.

3 Proposed Method

3.1 System Architecture

Due to its elegant solution to the one-to-many problem of speech synthesis, we choose FastSpeech 2 (Ren et al., 2020) as the basis for our method. There is however no reason why this procedure should not work in conjunction with any comparable architecture, making the approach mostly model agnostic.

We use the Conformer architecture (Gulati et al., 2020) in both encoder and decoder. This is the same as the basic implementation in the IMS Toucan toolkit (Lux et al., 2021) which is in turn based on the ESPnet toolkit (Hayashi et al., 2020, 2021).

To handle the zero-shot multispeaker task, we condition the TTS on an ensemble of pretrained speaker embedding functions that consist of ECAPA-TDNN (Desplanques et al., 2020) and X-Vector (Snyder et al., 2018) trained on Voxceleb 1 and 2 (Nagrani et al., 2019, 2017; Chung et al., 2018) using the SpeechBrain toolkit (Ravanelli et al., 2021) as suggested in (Meyer et al., 2022). Consistent with (Jia et al., 2018) we find that the best ability to produce speech from voices unseen during training is achieved when injecting the speaker embeddings into the output of the encoder. First we bottleneck the speaker embeddings and apply the SoftSign function, as suggested in (Gibiansky et al., 2017). Then we concatenate them to the encoder's hidden state and project them back to the size of the encoder's hidden state. At inference time, a speaker embedding of a reference audio can be used to make the synthesis speak in the voice of the reference speaker. An important trick we found is to add layer normalization right after the embedding is injected into the hidden state. This does not affect the synthesis of speakers seen during training, however it helps with unseen speakers.

In order to disentangle the languages from the speakers, we add an embedding for the language of the current sample along the sequence axis to the phoneme embedding sequence at the start of the encoder. This fits well to the intuition of a TTS encoder dealing with the text and the decoder dealing with the speech, since the text processing should not rely on speaker information, as a text does not have an inherent speaker. So we infuse the language information at the text stage and the speaker information at the speech stage of the model's information flow. Since, unlike the amount of possible voices, the amount of languages in the world is finite, we simply use an embedding lookup table to get embeddings of languages which receive their meaning purely through backpropagation during training. A text based language embedding could allow for zero-shot language adaptation, which we plan to investigate in the future. An overview of the multilingual multispeaker encoder is shown in Figure 1.

To transform the spectrograms that the FastSpeech 2 based synthesis produces into a waveform, we make use of the HiFi-GAN architecture (Kong et al., 2020) as implemented in the IMS Toucan toolkit (Lux et al., 2021). As is shown in (Liu et al., 2021), neural vocoders can do super-resolution as well as spectrogram inversion. We apply the same trick to transform the 16kHz spectrograms the synthesis produces into 48kHz waveforms.

3.2 Input Representation

To make the use of multilingual data with only partially overlapping phonemesets easier, we represent the inputs to our system as articulatory feature vectors rather than identity based vectors, the same as is introduced in (Lux and Vu, 2022). On top of this, we add an additional mechanism to deal with the

multilinguality of the data.

Word boundaries are something that in most languages is very clearly visible in text. In spoken form however, word boundaries do not cover their own segment, but are instead only noticeable through cues in pitch and energy. This is why in TTS, word boundaries are usually removed. However we believe that in a multilingual setting, it is important to make the TTS model aware of word boundaries. We assume that this helps the model learn to distinguish how morpheme boundaries work in each language individually, as this is something that rarely holds across languages.

In our design, word boundaries are considered in the encoder of the TTS model, which intuitively corresponds to the encoding of the text, in which word boundaries do exist on the surface level, but not in the decoder, which intuitively corresponds to the decoding of the speech, where word boundaries are deeply embedded in the prosody as boundary tones. We achieve this by simply keeping track of the indexes of the word boundaries throughout the encoder and overwriting their predicted durations to be always zero. The upsampling mechanism in the length regulator will then remove their encoded vectors from the sequence as the information is passed to the decoder, while it was still available as contextual information in the encoder. This is illustrated in Figure 2. It is to be noted that as polar opposite to word boundaries, pauses do exist in speech, but not necessarily in text. For that reason, we treat pauses as separate units from word boundaries. Pauses receive a non-zero duration in the encoder and have their own spectrogram frames associated to them, unlike the word-boundaries. To detect pauses in the text, we use occurrences of commas and dashes in the text as a simple heuristic. This heuristic works in surprisingly many languages. Sentence marks like the question mark, the exclamation mark and the full stop are also treated as separate units, because they hold prosodic significance, even though they are mostly realized as a pause on the time axis.

3.3 Data Preparation

Furthermore we average the energy and pitch values extracted from the gold-audio over the spectrogram frames that belong to a single phoneme according to the alignment. This is introduced in FastPitch (Łańcucki, 2021) and allows for great controllability, but also makes model training more robust against low-quality data, which is an important feature for dealing with multilingual data since its quality greatly varies over the languages.

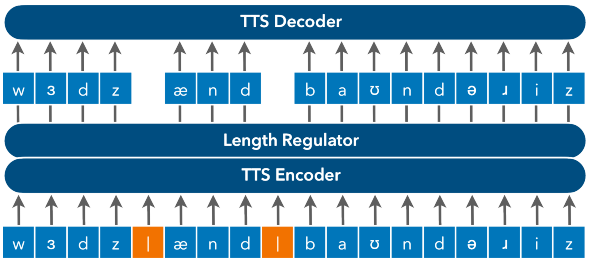

Figure 2: Example of the information flow of phonemes through the text encoder and speech decoder. The word boundaries (orange) are used in the encoder to contextualize the phoneme encoding, due to the length regulator however they do not reach the decoder.

Due to our reliance on spectrogram frames with their energy and pitch values being attributed to the correct phoneme, we make use of a lightweight self-contained aligner. We train this aligner as an automatic speech recognition system (ASR) using CTC (Graves et al., 2006) and an L_1 reconstruction loss of its inputs and the outputs of an auxiliary TTS that backtranslates the frame-wise ASR predictions to a spectrogram inspired by (Pérez-González-de Martos et al., 2021). Alignment is then found by ordering the posteriograms of the ASR by the phonemes which we expect and then performing monotonic alignment search from start to end (Kim et al., 2020) using the efficient implementation from (Badlani et al., 2022). This aligner was introduced and is further described in (Lux et al., 2022).

3.4 Training Procedure

To train the TTS we make use of the LAML procedure (Lux and Vu, 2022), which means that we treat different languages as tasks from a meta learning perspective. In order to solve all of these tasks simultaneously, an initialization point is iteratively refined to take fewer steps to get close to a good solution for each task. Such an initialization point that is well suited for all tasks seen in training is usually also suitable for unseen tasks (i.e. unseen languages in our context). To achieve this with TTS, we calculate the loss for one batch per language and sum them up. The samples from each language that go into each batch are chosen randomly, so the speakers are mixed throughout, resulting in also the ability to finetune to specific speakers on tiny amounts of data.

Since phonemes should in theory be language agnostic, we also train the aligner on a massive amount of multilingual and multispeaker data described in section 4.1 following the same procedure resulting in low-resource finetuning capabilities.

With regards to the vocoder we find that it can not only perform spectrogram inversion and super-resolution, but also slight speech enhancement. We inject random noise with a signal-to-noise ratio of 5db into the spectrogram for every tenth sample to increase the robustness of the vocoder against some noise in the synthesis induced by mixed quality data in some languages.

4 Experiments

4.1 Data Used

In our experiments we use a variety of speech datasets with accompanying text labels in a total of 12 languages. The total amount of hours per language used is shown in parentheses in the following. For English (85h), we use the Blizzard Challenge 2011 dataset (King and Karaiskos, 2011), LJSpeech (Ito and Johnson, 2017), LibriTTS (Zen et al., 2019), HiFi-TTS (Bakhturina et al., 2021) and VCTK (Veaux et al., 2017). For German (80h) we use the HUI-Audio-Corpus-German (Puchtler et al., 2021) and the Thorsten corpus (Müller and Kreutz, 2021). Spanish (30h) includes the Blizzard Challenge 2021 dataset (Ling et al., 2021) and the CSS10 dataset (Park and Mulc, 2019), from which we also use the Greek (4h), Finnish (11h), French (39h), Russian (21h), Hungarian (10h) and Dutch (34h) subsets. The Dutch and French subsets of the Multilingual LibriSpeech (Pratap et al., 2020) are also included, as well as its Polish (20h), Portuguese (25h) and Italian (30h) subsets. Greek, Finnish, Russian and Hungarian each only have a single speaker. To have a high variety of data, but keep the computational cost manageable, we only use a maximum of 20,000 randomly chosen samples per corpus.

4.2 Experimental Setup

To verify our first contribution, we exclude German, Dutch and English data (Germanic languages) from the pretraining and then finetune a model on randomly chosen samples from a single speaker which add up to a total duration of just 5 minutes of German speech. We do the same with excluding Russian and Polish (Slavic languages) from the pretraining and then finetune on 5 minutes of Russian speech. In the evaluations we will refer to these models as the low-resource (LR) models. The two languages were chosen to simulate a low-resource scenario, rather than using an actual low-resource language, to still be able to get reliable and accurate measures on intelligibility and naturalness. We compare the two LR models to human speech as well as a single speaker model trained on 29 hours of German and 21 hours of Russian respectively. These models will be referred to as the high-resource (HR) models in the evaluation. Since the aligner and the vocoder are speaker and language agnostic, we exclude the Germanic and Slavic languages from their training and do not finetune them at all.

Intelligibility To assess intelligibility, we calculate the phone-error-rate (PER) of the German and Russian IMS-Speech (Denisov and Vu, 2019) ASR systems on 3000 unseen sentences. This includes the case of an unseen speaker in the LR models.

Naturalness To verify the naturalness, we conduct a mean opinion score (MOS) study in which human raters give scores on a scale from 1-5 to 10 samples of human, LR and HR speech. For the case of German, we consider the HR model the upper bound, since the data is very high quality. Also, in this case the two largest and cleanest subsets of data were removed from the pretraining. So for German, we are investigating how close we can get to the performance of a very strong system. For Russian however, we can benefit from the high-quality pretraining that is met with less high-quality in-domain data and aim to even outperform the HR system.

Speaker Transfer To verify our second contribution, we will measure the cosine similarity of speaker embeddings derived from synthetic speech to the embeddings derived from the human references used across all languages, including those which have seen only one speaker during training and the LR models from the previous experiment. A low standard deviation across all languages for each speaker (including the LR models) would indicate that the zero-shot multispeaker TTS properties are shared across all languages.

Word Boundaries The impact of the word boundaries can be mostly found in the intonation, but this includes cases where the intonation leads to incorrect phrasing and thus also incorrect word

boundaries in the output. To verify their importance, we run the intelligibility experiment with a different configuration: We evaluate word-error-rate (WER) instead of PER and we only evaluate the German models, since the data quality is higher in that one, which gives us more reliable results. We compare each model to a version that is trained completely analogous, but without word boundaries in the input. Since the HR models are monolingual, we hypothesize to see no change in WER, but an increase for the LR models, when the word boundaries are removed.

Accent Transfer To investigate the impact of the language embedding on its own, we focus on the languages which have only seen a single speaker during training, which are Greek, Russian and the two LR models. In these cases, it might be possible that the model has learned to associate the language with the voice of the speaker, since they always co-occur. We measure whether the cosine similarity to a target speaker in each of the other languages changes if we change the language embedding to one of the single-speaker languages. A small deviation would mean, that the language embedding does not affect the voice of the speaker, which is what we desire.

5 Results

5.1 Intelligibility

The PERs of the different TTS systems are reported in Table 1. The single speaker model for German almost matches the intelligibility of the human voice, indicating a very strong baseline. While the PER of the model trained on 5 minutes of a male German voice is worse relative to the single speaker model, the low absolute PER still indicates good intelligibility. When exchanging the speaker embedding for that of a female speaker, the PER increases further. This might be caused by the exclusion of the most varied and clean parts of the training data from the pretraining for this experiment, which reduces the overall quality for certain voices. It might however also simply be caused by the voice itself. Unfortunately, we do not have the same 3000 samples spoken by another speaker to investigate the impact of the voice on its own.

The Russian LR model also has a worse PER compared to human speech and the HR baseline. Looking into the cases where the LR model performed worse than the HR model, we mostly find near-misses, like producing the unvoiced variant of a consonant rather than the voiced variant. So while the small amount of data used paired with the lower quality of the finetuning data certainly negatively impacts the intelligibility, it is not as bad perceptively as the scores seem at first. Interestingly the impact of using a very different speaker embedding does not affect the PER significantly in this case. We assume this is because of the more diverse pretraining data that this model has seen.

Language	Speech Type	Voice	PER
German	Human	Male	3.58%
	TTS - HR	Male	3.59%
	TTS - LR	Male	4.34%
		Female	5.91%
Russian	Human	Male	7.65%
	TTS - HR	Male	9.22%
	TTS - LR	Male	12.32%
		Female	12.64%

Table 1: PER of an ASR trained for the corresponding language. Reference speaker for LR speech is varied. The same 3000 samples are used to calculate each PER.

5.2 Naturalness

For the studies on the naturalness, we received a total of 330 ratings per speech type from 33 raters in German and 140 ratings per speech type from 14 raters in Russian. The results are shown in Table 2. Considering that the setup for the German LR TTS is the most difficult, the model achieves a MOS that is surprisingly close to that of the baseline trained on 350 times more data, especially when considering the standard deviations, which indicate a large overlap in ratings. There is a rather large gap between the absolute values for human speech and synthetic speech, which is likely due to the very high quality of the human samples causing the raters to compare samples rather than rate them independent of each other. This causes even small imperfections to trigger a strong aversity. For Russian, the LR system even significantly outperformed the baseline trained on 250 times more data. We suspect that the mixed quality of samples in the Russian corpus (i.e. multiple different microphones and recording environments used) caused the single speaker model to not learn a consistent voice. It is however not a weak model, as the good performance on the intelligibility experiment confirms. In our interpretation, this shows that the

Language	Speech Type	MOS	σ
German	Human	4.57	± 0.69
	TTS - LR	3.06	$\pm 1,35$
	TTS - HR	3.35	± 1.02
Russian	Human	4.37	± 0.86
	TTS - LR	3.57	± 1.25
	TTS - HR	2.07	± 1.02

Table 2: Mean opinion scores by human raters. All synthetic samples within a language are generated in the same voice.

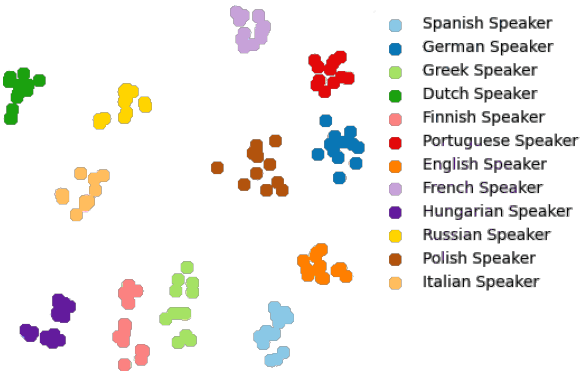

Figure 3: Visualization of speaker embeddings for 12 unseen speakers (1 speaker per language) each speaking 2 sentences in 12 different languages + the respective human speech reference. Each color corresponds to one speaker. Each point in a certain color is spoken in a different language.

pretraining can effectively leverage vast amounts of high-quality data in high-resource languages to perform well in underresourced languages.

5.3 Speaker Transfer

In preliminary experimentation we found that finetuning on the 5 minutes of data alone leads to rapid overfitting and the model loses its zero-shot multispeaker TTS capabilities. To prevent this, we finetune by including the small dataset into the LAML training procedure and train jointly for 5,000 batches. Further we found that when training with just one language per batch, the model does not converge to a usable state, whereas combining all languages to equal amounts in each batch (i.e. the LAML procedure) converges in just 60,000 steps, which shows the necessity of using LAML for this setup.

	$\varnothing$	σ		$\varnothing$	σ
English	0.81	0.02	Dutch	0.79	0.03
German	0.86	0.02	Finnish	0.79	0.02
French	0.85	0.01	Greek	0.82	0.03
Hungarian	0.77	0.04	Italian	0.71	0.03
Portuguese	0.75	0.03	Polish	0.71	0.03
Russian LR	0.80	0.03	Spanish	0.81	0.03
German LR	0.81	0.03	Russian	0.79	0.03

Table 3: Cosine similarities of speaker embeddings of synthetic samples spoken in all 12 languages compared to the speaker embedding of the human reference speaker. Two utterances of the same human speaker leads to a similarity of 0.87 on average, defining an upper bound. $\varnothing$ is the average within-speaker similarity, σ is standard deviation of the within-speaker similarity.

Table 3 shows the average similarity that samples spoken in all 12 languages we investigated achieve compared to their human reference. The language column refers to the language of the speaker that the reference was taken from. A low standard deviation means, that the voice sounds similar regardless of the language it is currently speaking, indicating a good disentanglement of speakers and languages. While table 3 shows that the cloning of the speaker identity worked in some cases nearly perfect (German, French), there were also some cases where they didn't work as well (Italian, Polish). Investigating whether the language had an impact on this however showed, that the low scores are only due to the specific speakers which we randomly chose as the reference for those languages. Other speakers speaking either of those languages produced much higher similarities with their synthetic counterparts. So how well a voice can be cloned depends on the voice, but not on the language. The overall low standard deviations furthermore indicate that the speaker identity is consistent across all languages, regardless of which voice in which language is used as the reference. For the LR variants included in this table, a different speaker than was seen in the training is used. The high similarity and low standard deviation indicates that the level of fulfillment of the zero-shot multispeaker TTS task exhibited by the full model is still present in the LR models. The results are supported by the visualization in Figure 3. The clusters shown are linearly separable, indicating distinct speaker identities despite the switches in languages and high similarity to the human reference across all languages, even the ones where only a single speaker was seen during training.

5.4 Word Boundaries

As can be seen from Table 4, the models that are aware of where word boundaries should go perform

significantly better at placing the correct prosodic cues to indicate word boundaries in the output in the multilingual scenario. The impact of the boundaries on the monolingual model are insignificant.

Model	WER
LR multilingual with boundaries	13.71%
LR multilingual without boundaries	19.83%
HR monolingual with boundaries	11.32%
HR monolingual without boundaries	11.91%

Table 4: Impact of monolingual and multilingual German models being aware of word boundaries as measured by an ASR system in terms of WER.

5.5 Accent Transfer

Table 5 shows whether the language embedding impacts the voice that is produced. While the change of the language embedding did not significantly impact the similarity to the target speaker, we discovered that the information about the language encoded in the language embedding can actually be used to control the accent of the produced speech completely independent of language and speaker.

Embed.	ΔSim	Embed.	ΔSim
Greek	0.001	German LR	0.002
Russian	0.008	Russian LR	0.004

Table 5: Average deviation in cosine similarity from target speaker in each language when the language embedding is switched to a language with only a single speaker.

6 Discussion

Language Embedding Investigation The accent transfer has interesting implications on how the distribution of realizations of a phoneme shifts with each language, independent of the context, which can be investigated by synthesizing individual phonemes with only the language embedding changed. We find language typical patterns, even in the languages that have only been trained on 5 minutes of data. So it seems that very little data is enough to capture a lot about how a language is usually spoken.

Implicit Morpheme Vocabularies Although word boundaries are not explicitly denoted as segmental units in speech, they still have considerable influence on the phonetic realization. Consider for example the phenomenon of velar softening, i.e. a velar plosive is realized as alveolar fricative when followed by a long or short i ([ɪ] or [ay]) in some contexts, such as in *electri[k]* → *electri[s]ity*. This does however not hold across word boundaries as in *electri[k] igniter*. Another example where word boundaries cause changes in the phone sequence is the phenomenon of final devoicing: voiced obstruents become voiceless if they occur in word-final position e.g. the German word *Hunde (dogs)* is pronounced [hʊndə] in its plural form but in singular *Hund* becomes [hʊnt]. Such rules are however highly dependent on the language. Final devoicing is for example observed in German, Dutch and Polish, but not in English or French.

While many of these language specific lexical rules are already captured by the phonemizer, the situation is different in cases where word boundaries are not reflected by the phone sequence itself but only in the intonation, such as in ['acid] → [ac'id+ic]. While in the latter, there is still a morpheme boundary after *acid*, this is not a word boundary. This highlights the importance of differentiating between actual word boundaries and word-internal morpheme boundaries in order to produce correct intonation which is crucial for generating intelligible speech.

Monolingual TTS models actually seem to learn an implicit vocabulary of morphemes as well as an intuition in which contexts morpheme boundaries can denote a word boundary in the language they are trained in. But in the case of multilinguality, this vocabulary of morphemes is difficult to construct, because every language has different morphemes. Thus, since multilingual models face a more difficult task to identify morphemes, they struggle even more distinguishing morpheme from word boundaries. Even with the language embedding, it seems like this is a property that the TTS can no longer implicitly capture, at least not given small amounts of data.

We especially observe this in compound-nouns in our model trained on German in a low-resource setting. A model without explicit word boundaries adds boundary tones in the middle of the word causing an unnatural intonation that reduces the intelligibility of the word. If the model is trained with word boundaries, even though there are no word boundaries within the composite-noun, the pronunciation becomes much more fluent with the intonation being consistent throughout the word.

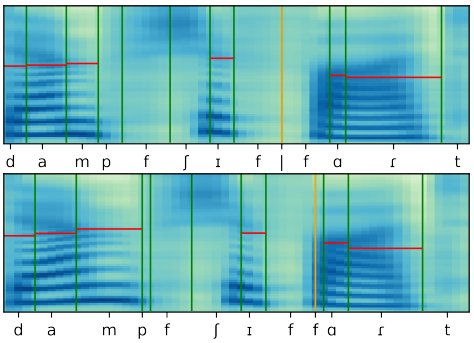

Figure 4: Spectrogram of the German noun-composite "Dampfschifffahrt" (steamboat ride) as produced by the word-boundary aware multilingual TTS (upper) and the multilingual TTS without word-boundaries (lower). Pitch predictions per phoneme are displayed in red, phoneme boundaries are displayed in green and the boundary between "steamboat" and "ride" in orange, which is however invisible to the models.

Figure 4 illustrates this with an example. It depicts spectrograms of a German word that consists of three parts: [dampf], [ʃɪf] and [fɑɾt]. The components translate to English as steam, boat and ride. The proper phrasing within this word would be to combine the [dampfʃɪf] into one unit with the pitch being the highest on [ɪ] and a falling pitch towards the end of the word throughout [fɑɾt]. This is the case in the model that is aware of the word-boundaries. For illustration purposes, we include the boundary between steamboat and ride in the plot, the model however does not see this boundary as it happens in the middle of one word. The model which is unaware of the word-boundaries lowers its pitch already at the [ɪ] and lengthens the [dampf] part of the word. This makes the second instance sound as if the model was saying "steam boatride" rather than "steamboat ride".

We conclude that by simply making word boundaries explicit, the model no longer overestimates intonation phrase boundaries and boundary tones at every possible morpheme boundary.

Low-Resource Capabilities Our experiments on low-resource scenarios show three major things: 1) it is possible to generalize into unseen branches in the phylogenetic tree of languages and reduce data-need even for languages with significant differences from the languages that have been trained on, which makes us hopeful that the direction of zero-shot learning to speak in a language is possible. 2) even from extremely little data in a target language, a lot of knowledge about the language can be abstracted. Language embeddings seem to encode language specific realizations of phones even when trained only on a few minutes of data. 3) the quality of data can be transferred across languages. Pretraining on high-quality data and then finetuning on low-quality data leads to a better model than when trained on much more of the low-quality data. This suggests that found data can be sufficient for TTS in a new language, because its quality can be improved by studio data in the pretraining.

7 Limitations and Future Work

While the LAML procedure is, as the name suggests, language agnostic, we only include European languages in our training and testing in order to get more reliable results with the resources for testing we have available. The state of the implementation with which the experiments were conducted cannot handle tonal languages, due to the non-segmental nature of tone. This limits the generality of our findings. Our open-sourced code has been updated in the meantime to be able to handle tone and lengthening properly. We plan to extend this work to include a much larger and much more diverse set of languages.

8 Conclusion

We show that through a simple encoder design coupled with a mechanism to encode word boundaries and the LAML training procedure, a low-resource capable multilingual zero-shot multispeaker TTS can be achieved. We are able to train a German and a Russian model on just 5 minutes of data each, which perform comparable or even better to single speaker models trained on 29 and 21 hours of data respectively. We further show that the ability to perform zero-shot multispeaker TTS is shared across languages, even those which have seen only 5 minutes of single speaker data. An additional side-effect is that the language embedding design in the encoder allows us to vary the accent of speech regardless of language of the input text and speaker.

References

Sercan Arik, Jitong Chen, Kainan Peng, Wei Ping, and Yanqi Zhou. 2018. Neural voice cloning with a few samples. *NeurIPS*, 31.

Kurniawati Azizah, Mirna Adriani, and Wisnu Jatmiko. 2020. Hierarchical Transfer Learning for Multilingual, Multi-Speaker, and Style Transfer DNN-Based

TTS on Low-Resource Languages. *IEEE Access*, 8:179798–179812.

Kurniawati Azizah and Wisnu Jatmiko. 2022. Transfer Learning, Style Control, and Speaker Reconstruction Loss for Zero-Shot Multilingual Multi-Speaker Text-to-Speech on Low-Resource Languages. *IEEE Access*, pages 5895–5911.

Rohan Badlani, Adrian Łańcucki, Kevin J Shih, Rafael Valle, Wei Ping, and Bryan Catanzaro. 2022. One TTS alignment to rule them all. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6092–6096. IEEE.

Evelina Bakhturina, Vitaly Lavrukhin, Boris Ginsburg, and Yang Zhang. 2021. Hi-Fi Multi-Speaker English TTS Dataset. In *Interspeech*, pages 2776–2780.

Edresson Casanova, Christopher Shulby, Eren Gölge, Nicolas Michael Müller, Frederico Santos de Oliveira, et al. 2021. SC-GlowTTS: An Efficient Zero-Shot Multi-Speaker Text-To-Speech Model. In *Interspeech*, pages 3645–3649.

Edresson Casanova, Julian Weber, Christopher D Shulby, Arnaldo Candido Junior, Eren Gölge, and Moacir A Ponti. 2022. YourTTS: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone. In *ICML*, pages 2709–2720. PMLR.

Yuan-Jui Chen, Tao Tu, Cheng-chieh Yeh, and Hung-Yi Lee. 2019. End-to-End Text-to-Speech for Low-Resource Languages by Cross-Lingual Transfer Learning. *Interspeech*, pages 2075–2079.

Seungwoo Choi, Seungju Han, Dongyoung Kim, and Sungjoo Ha. 2020. Attentron: Few-Shot Text-to-Speech Utilizing Attention-Based Variable-Length Embedding. *Proc. Interspeech 2020*, pages 2007–2011.

J. S. Chung, A. Nagrani, and A. Zisserman. 2018. VoxCeleb2: Deep Speaker Recognition. In *Interspeech*, pages 1086–1090.

Erica Cooper, Cheng-I Lai, Yusuke Yasuda, Fuming Fang, Xin Wang, et al. 2020. Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings. In *ICASSP*, pages 6184–6188. IEEE.

Marcel de Korte, Jaebok Kim, and Esther Klabbers. 2020. Efficient Neural Speech Synthesis for Low-Resource Languages Through Multilingual Modeling. *Interspeech*, pages 2967–2971.

Pavel Denisov and Ngoc Thang Vu. 2019. IMS-speech: A speech to text tool. *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2019*, pages 170–177.

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. In *Interspeech*, pages 3830–3834.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135. PMLR.

Andrew Gibiansky, Sercan Arik, Gregory Diamos, John Miller, Kainan Peng, et al. 2017. Deep voice 2: Multi-speaker neural text-to-speech. *NeurIPS*, 30.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, pages 369–376.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, et al. 2020. Conformer: Convolution-augmented Transformer for Speech Recognition. *Interspeech*, pages 5036–5040.

Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, et al. 2020. ESPnet-TTS: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In *ICASSP*, pages 7654–7658. IEEE.

Tomoki Hayashi, Ryuichi Yamamoto, Takenori Yoshimura, Peter Wu, Jiatong Shi, et al. 2021. ESPnet2-TTS: Extending the Edge of TTS Research. *arXiv preprint arXiv:2110.07840*.

Mutian He, Jingzhou Yang, and Lei He. 2021. Multilingual Byte2Speech Text-To-Speech Models Are Few-shot Spoken Language Learners. *arXiv preprint arXiv:2103.03541*.

Hamed Hemati and Damian Borth. 2021. Continual Speaker Adaptation for Text-to-Speech Synthesis. *arXiv preprint arXiv:2103.14512*.

Keith Ito and Linda Johnson. 2017. The LJ Speech Dataset. `https://keithito.com/LJ-Speech-Dataset/`.

Ye Jia, Yu Zhang, Ron Weiss, Quan Wang, Jonathan Shen, et al. 2018. Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis. In *NeurIPS*, volume 31.

Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. 2020. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Advances in Neural Information Processing Systems*, 33:8067–8077.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *ICML*, pages 5530–5540. PMLR.

Simon King and Vasilis Karaiskos. 2011. The Blizzard Challenge 2011. In *Proc. Blizzard Challenge Workshop*, volume 2011.

Julia Koch, Florian Lux, Nadja Schauffler, Toni Bernhart, Felix Dieterle, Jonas Kuhn, Sandra Richter, Gabriel Viehhauser, and Ngoc Thang Vu. 2022. PoeticTTS - Controllable Poetry Reading for Literary Studies.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. *NeurIPS*, 33.

Adrian Łańcucki. 2021. FastPitch: Parallel text-to-speech with pitch prediction. In *ICASSP*, pages 6588–6592. IEEE.

Zhen-Hua Ling, Xiao Zhou, and Simon King. 2021. The Blizzard Challenge 2021. In *Proc. Blizzard Challenge Workshop*.

Yanqing Liu, Zhihang Xu, Gang Wang, Kuan Chen, Bohan Li, et al. 2021. DelightfulTTS: The Microsoft Speech Synthesis System for Blizzard Challenge 2021. *Proc. Blizzard Challenge Workshop*, 2021.

Florian Lux, Julia Koch, Antje Schweitzer, and Ngoc Thang Vu. 2021. The IMS Toucan system for the Blizzard Challenge 2021. In *Proc. Blizzard Challenge Workshop*, volume 2021. Speech Synthesis SIG.

Florian Lux, Julia Koch, and Ngoc Thang Vu. 2022. Exact Prosody Cloning in Zero-Shot Multispeaker Text-to-Speech. In *Proc. IEEE SLT*.

Florian Lux and Ngoc Thang Vu. 2022. Language-Agnostic Meta-Learning for Low-Resource Text-to-Speech with Articulatory Features. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 6858–6868.

Sarina Meyer, Florian Lux, Pavel Denisov, Julia Koch, Pascal Tilli, and Ngoc Thang Vu. 2022. Speaker Anonymization with Phonetic Intermediate Representations.

Thorsten Müller and Dominik Kreutz. 2021. Thorsten - Open German Voice (Neutral) Dataset. https://doi.org/10.5281/zenodo.5525342.

A. Nagrani, J. S. Chung, and A. Zisserman. 2017. VoxCeleb: a large-scale speaker identification dataset. In *Interspeech*, pages 2616–2620.

Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. 2019. VoxCeleb: Large-scale speaker verification in the wild. *Computer Science and Language*.

Paarth Neekhara, Jason Li, and Boris Ginsburg. 2021. Adapting TTS models For New Speakers using Transfer Learning. *arXiv preprint arXiv:2110.05798*.

Kyubyong Park and Thomas Mulc. 2019. CSS10: A Collection of Single Speaker Speech Datasets for 10 Languages. *Interspeech*, pages 1566–1570.

Alejandro Pérez-González-de Martos, Albert Sanchis, and Alfons Juan. 2021. VRAIN-UPV MLLP's system for the Blizzard Challenge 2021. *Proc. Blizzard Challenge Workshop*, 2021.

Aidan Pine, Dan Wells, Nathan Brinklow, Patrick Littell, and Korin Richmond. 2022. Requirements and Motivations of Low-Resource Speech Synthesis for Language Revitalization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 7346–7359.

Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, and Ronan Collobert. 2020. MLS: A Large-Scale Multilingual Dataset for Speech Research. In *Interspeech*, pages 2757–2761.

Pascal Puchtler, Johannes Wirth, and René Peinl. 2021. Hui-audio-corpus-german: A high quality tts dataset. In *German Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 204–216. Springer.

Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, et al. 2021. SpeechBrain: A General-Purpose Speech Toolkit.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, et al. 2020. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech. In *ICLR*.

David Snyder, Daniel Garcia-Romero, Alan McCree, Gregory Sell, Daniel Povey, et al. 2018. Spoken Language Recognition using X-vectors. *Odyssey 2018*, pages 105–111.

Marlene Staib, Tian Huey Teh, Alexandra Torresquintero, Devang S. Ram Mohan, Lorenzo Foglianti, et al. 2020. Phonological Features for 0-Shot Multilingual Speech Synthesis. *Interspeech*, pages 2942–2946.

Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, et al. 2017. Superseded-CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit.

Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ-Skerry Ryan, Eric Battenberg, et al. 2018. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. In *ICML*, pages 5180–5189. PMLR.

Dan Wells, Pilar Oplustil-Gallegos, and Simon King. 2021. The CSTR entry to the Blizzard Challenge 2021. In *Proc. Blizzard Challenge Workshop*, volume 2021. Speech Synthesis SIG.

Yihan Wu, Xu Tan, Bohan Li, Lei He, Sheng Zhao, Ruihua Song, Tao Qin, and Tie-Yan Liu. 2022. AdaSpeech 4: Adaptive Text to Speech in Zero-Shot Scenarios. *arXiv preprint arXiv:2204.00436*.

Jin Xu, Xu Tan, Yi Ren, Tao Qin, Jian Li, et al. 2020. *LRSpeech: Extremely Low-Resource Speech Synthesis and Recognition*, page 2802–2812. Association for Computing Machinery, New York, NY, USA.

Jingzhou Yang and Lei He. 2020. Towards Universal Text-to-Speech. In *Interspeech*, pages 3171–3175.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, et al. 2019. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech. In *Interspeech*, pages 1526–1530.