Zurich University
of Applied Sciences

School of
Engineering

Master Thesis

ZHAW School of Engineering

CAI, Centre for Artificial Intelligence

No Shots Required: Zero-Shot Voice-Adaptation TTS for Swiss German Dialects

Author:
Samuel Stucki

Supervisor:
Prof. Dr. Mark Cieliebak

Secondary Supervisor:
Dr. Jan Deriu

January 31, 2025

Abstract

Zero-Shot (ZS) voice adaptation in text-to-speech (TTS), which enables speech synthesis for unseen speakers without training on their voices, has seen significant progress in high-resource languages like German and English. However, low-resource languages and dialects remain underdeveloped due to data scarcity, limiting robust speech synthesis. This work addresses this gap for Swiss German, a group of dialects spoken in Switzerland, by compiling a large-scale dataset of 5000 hours from podcasts and online broadcasts. We integrate this dataset with existing corpora and apply weak labelling for Swiss German speech to Standard German text, along with dialect identification and others, to fine-tune a multilingual pre-trained Zero-Shot-TTS model. The best-performing setups, evaluated on 43 unseen speakers, achieved a WER of 0.259 and a CER of 0.141, with a weighted F1 score of 0.78 across seven dialect regions for longer sentences. Human evaluation across 30 unseen speakers yielded a 3.60 speaker similarity MOS, -0.42 comparative MOS, and 3.98 intelligibility score, demonstrating reasonable voice adaptation. The findings highlight the potential of using weakly labelled audio data to enhance Zero-Shot voice adaptation performance for Swiss German dialect TTS.

Preface

This thesis was supported by Prof. Dr. Mark Cieliebak and Dr. Jan Deriu of the Centre for Artificial Intelligence (CAI) at the Zurich University for Applied Sciences (ZHAW) in Switzerland. I want to thank them for their invaluable input, without which this thesis would not have been possible. Additionally, I want to thank the CAI Infrastructure Team, specifically Marc Stadelmann and Murteda Al Kaysi, for their proactive support on various issues during training. Lastly, I want to thank my family and friends for supporting me throughout the last few years. You gave me the strength and support (and healthy distraction) to complete my studies successfully.

行到水窮處 坐看雲起時。

Contents

Chapter 1

Introduction

Over the past decade, advances in Artificial Intelligence (AI), and more specifically Natural Language Processing (NLP), have revolutionized the field of text-to-speech (TTS) systems, driven by dramatic increases in computational power and the availability of large-scale datasets. This progress has been marked by a shift from traditional statistical methods to the dominance of Deep Neural Networks (DNN). These powerful models have propelled TTS from basic rule-based systems to State-Of-The-Art (SOTA) solutions capable of producing highly natural and human-like speech. By enabling machines to deeply understand and process human language in written and spoken forms, these advancements have allowed for more accurate text-to-speech conversion. Modern TTS models now excel in generating clear and natural audio and capturing the subtle nuances of tone, intonation, and context, paving the way for applications across diverse languages and dialects.

The task of TTS aims to synthesise human-like, natural-sounding voices from written text. Since the release of the transformer architecture by Vaswani et al. [1], the task has seen great improvements. In general, TTS follows a pipelined approach. A fundamental component of TTS systems is text preprocessing, where input text is converted into a phonetic or linguistic representation. This stage includes text normalization, which handles abbreviations, numbers, and special characters to ensure accurate pronunciation. The synthesized parameters are passed through a vocoder, which generates the final waveform output. Deep-learning vocoders, such as WaveGlow and HiFi-GAN, have significantly improved the quality of synthesized speech, making it nearly indistinguishable from human speech. [2]

Single speaker TTS refers to a synthesis system trained on speech data from a single speaker. These models are optimized to generate high-quality, natural-sounding speech that closely resembles the speaker's voice used for training. Traditional single-speaker TTS systems used pre-recorded speech units to form words and sentences [3]. However, modern deep-learning-based approaches, such as Tacotron [4] and WaveNet [5], have significantly improved the quality and naturalness of synthesized speech. Since the model is trained on data from only one speaker, it achieves high quality in replicating that specific voice. However, this approach lacks flexibility for changing to new speakers without retraining on additional data. Additionally,

the reliance on a large amount of high-quality audio from a voice actor is often not feasible due to time and cost constraints.

Zero-shot voice adaptation is an advanced technique that allows a TTS model to generate speech in a new speaker's voice without requiring retraining on that speaker's data. This is particularly useful for applications where obtaining high-quality speech data for each new speaker is impractical. Unlike traditional speaker adaptation methods, which require fine-tuning with speaker-specific data, zero-shot adaptation relies on speaker embeddings extracted from a short audio sample of the target speaker.

A common approach to zero-shot voice adaptation involves using speaker encoders trained on a sizeable multi-speaker dataset. These encoders learn to generate fixed-length speaker representations (embeddings) that capture voice characteristics such as timbre, pitch, and speaking style. The embeddings are then integrated into a TTS model, enabling it to synthesize speech in the target voice after just a few seconds of reference audio [6]. An example is YourTTS, which leverages self-supervised learning for improved speaker adaptation [7].

1.1 Motivation

Developing a Zero-shot voice adaptation TTS system for Swiss German dialects represents a unique challenge in the NLP field. Swiss German is not a singular, standardized language but a collection of diverse dialects that vary significantly across regions. Unlike standard German, Swiss German lacks a widely accepted written standard, which makes it particularly challenging to develop computational models and resources. Additionally, Swiss German is a low-resource language with limited datasets and annotated corpora available for research and development.

This undertaking presents two primary challenges: limited resources and evaluation complexities. The low-resource challenge stems from the scarcity of available data for training speech models. Due to the relatively small number of speakers for many dialects, collecting sufficient training data is particularly challenging. This limitation contrasts with high-resource languages like English, where systems are often trained on datasets containing up to 50,000 hours of speech, or even Standard German. Moreover, the absence of standardized orthography and linguistic resources adds another layer of difficulty, requiring innovative data collection and processing approaches.

The evaluation challenge is even more intricate than conventional TTS systems. In addition to generating natural and high-quality audio from text, the system must accurately capture and reproduce the distinct features of each dialect. Dialect-specific nuances—such as unique phonetic patterns, vocabulary, and intonation demand fine-grained modelling beyond standard TTS capabilities. Furthermore, the lack of universal evaluation metrics for dialectal TTS systems complicates assessing their performance. These issues highlight the technical and linguistic hurdles that must

be overcome to create a truly effective system.

Despite these challenges, this thesis is motivated by the immense opportunity to fill a critical gap in both Swiss German research and TTS technology. Modern advancements in DNN have demonstrated the potential of TTS systems to produce natural, context-aware, and expressive speech. Applying these techniques to Swiss German offers the potential to preserve its rich linguistic diversity while addressing the practical needs of native speakers. A TTS system capable of representing Swiss German dialects would bridge technological gaps, support linguistic heritage preservation, and serve as a blueprint for developing TTS systems for other low-resource languages.

1.2 Objectives

The objectives of this thesis are threefold. First, a data crawling and transcription pipeline will be created to download and weakly label audio data for Swiss German dialects. Secondly, a new dataset based on Swiss German audio will be created using the data pipeline for further use in experiments. Lastly, make an exploratory Zero-Shot voice-adaptation TTS model that ingests a Standard German transcript and can generate a Swiss German audio in one of the seven given dialect regions. These regions include Basel, Bern, Graubünden, Innerschweiz, Ostschweiz, Wallis, and Zurich. The evaluation of the TTS output will be used to gauge the applicability of this new dataset and, more generally, the approach for Swiss German voice-adaptation TTS as a whole.

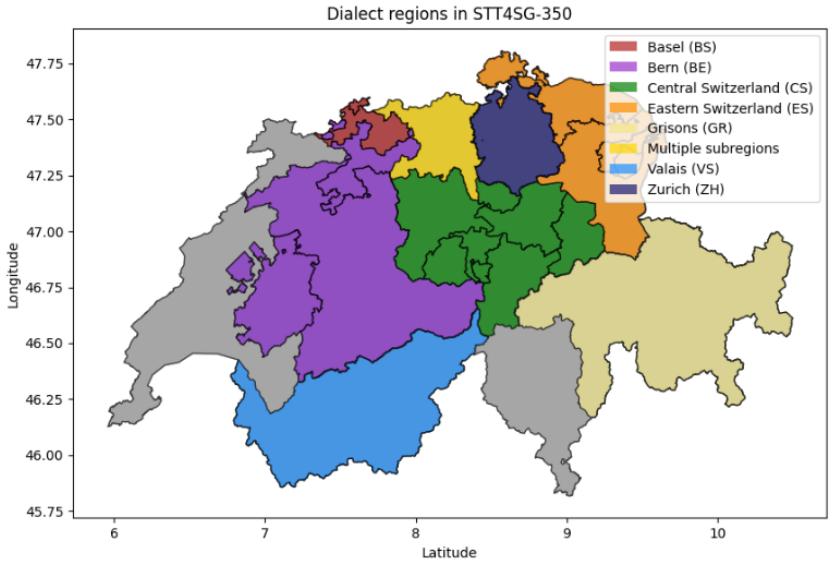

Figure 1.1: Map of Switzerland with approximate dialect regions based on canton level. Figure taken from [8].

1.3 Contributions

This thesis explicitly contributes three things: first, a data pipeline to download, diarize, cut, and weakly label standard German and Swiss German audio data. Secondly, a corpus of nearly 5000 hours of Swiss German audio from two public sources. It has to be noted that the corpus can not be released to the broader public as we do not possess the consent of either the SRF or the individual people involved in producing the utilized Swiss German audio. Lastly, a Swiss German audio fine-tuned zero-shot voice-adapting TTS model, based on [9], that can produce speech from the various Swiss German dialect regions in various performance degrees.

1.4 Outline

This thesis begins with an introduction into the field of text-to-speech in NLP that addresses the motivation behind the research. Chapter 2 reviews the existing literature to establish the context for the study. Following this, Chapter 3 explores the two datasets used in this thesis, involving STT4SG-350 [10], SwissDial [11], and the newly crawled SRF-corpus. The Chapter explains the labelling process for the SRF-corpus, an evaluation of the various performances in each pipeline step, and performs data analysis on the two main corpora. In Chapter 4, the training setup and the utilized pre-trained model by [9] is explained. Chapter 5 then evaluates the output by the trained models outlined in the previous Chapter. It also discusses the Human Evaluation that was carried out. Lastly, Chapter 6 discusses all the findings of this thesis and provides pointers where future work could be done.

Chapter 2

Background and Related Work

Public audio datasets for Swiss German dialects are sparse. However, growing interest in the research community has led to a steady increase in the number and sizes of such corpora. In 2020, Plüss et al. [12] presented the Swiss Parliament Corpus consisting of 293 hours of data collected from the Grosser Rat Kanton Bern. It consists of automatically aligned Swiss German speech to Standard German text. The SwissDial dataset by Schönenberger et al. [11] is a parallel multidialectal corpus across eight major dialects, including Standard German and Swiss German text references of around 28 hours total. The audio recordings for each dialect are performed by a single speaker with the motivation to provide a basic dataset for Swiss German research.

SDS-200 [13] was one of the first larger-scale multi-dialect datasets, including annotations of gender and age. The corpus comprises 4000 speakers and 200 hours covering much of Switzerland's Swiss German-speaking area. The dataset was unbalanced, with specific speakers and dialects being overrepresented. Lastly, the STT4SG-350 dataset [10] consists of approximately 343 hours of speech from all dialect regions. The samples are annotated with the 316 speakers' age group, gender, and dialect. The authors cluster the dialects into seven distinct regions based on their linguistic similarity. To the author's best knowledge, this is the most extensive publicly available Swiss German corpus to date.

Applying these datasets to Swiss German topics in NLP has thus become a large topic in the community. Sicard et al. [14] proposed a novel loss that took the semantic distance between predicted and ground-truth labels into account. Their on SDS-200 [13] fine-tuned whisper [15] achieved a WER of 20.6 and BLEU of 66.6. In [16] Bollinger et al. researched the quality of the variational inference with VITS [17]. Additionally, they developed a textual translation system from Standard German to Swiss German, which is used to feed the TTS system. The model trained on the [13] reached a MOS score of 3.1. Timmel et al. [18] investigated the potential of fine-tuning whisper-model [15] on data from the Swiss Parliament Corpus [12], the SDS-200 [13], and the STT4SG-350 [10] Schweizer Radio und Fernsehen (SRF). Additionally, they added Swiss German data from various broadcasts by the Schweizer Radio und Fernsehen (SRF). They achieved SOTA on all test sets compared to the

baseline large-v2 model.

Generally, for ZS-TTS, significant developments were made with the Vall-E model by Wang et al. [19]. It utilized 60K hours of English speech for pre-training to synthesise high-quality speech by an unseen speaker with only a 3-second recording. The model also seemed to preserve the speaker's emotion and the acoustic environment of the recording. Vall-E-2 by Chen et al. [20] improved upon the previous design by introducing Repetition Aware Sampling and Grouped Code Modelling, achieving Human parity in naturalness for the first time. The most significant breakthrough in the model series for low-resource languages was the VALL-E-X cross-lingual model by Zhang et al. [21]. More generally, Casanova et al. [7] was one of the first to work on multilingual ZS voice adaptation, culminating in the XTTS [9] architecture which achieved SOTA in most of its 16 supported languages. In [22] Lux et al. applied a Language agnostic meta learning procedure and modifications to a TTS encoder, showing in the process that it is possible to teach the model to speak a new language using just 5 minutes of training data without losing the ability for ZS voice adaptation.

Applying ZS-TTS to low-resource languages and dialects has thus become potentially feasible. Doan et al. [23] utilized the XTTS [9] architecture to perform Zero-Shot voice adaptation on Arabic dialects. They fine-tuned the model on QASR [24] containing 2000 hours of Arabic speech from the Al-Jazeera news channel, on which DID had to be performed first before fine-tuning the model. The resulting unbalanced dataset was then used to train two XTTS models, one without dialect tags and one with. They evaluated the models with 31 speakers from the test set. The trained models did not beat the baseline pre-trained XTTS model on WER or MOS but exhibited better speaker similarity performance, showing very good promise in the model's capabilities.

Chapter 3

Data Collection

Public datasets containing Swiss German audio are sparse. The most extensive available corpus, the STT4SG-350 by Plüss et al. [10], containing 350 hours of high-quality Swiss German audio, was taken as an entry point for this thesis. The seven dialect regions defined therein, "Bern", "Basel, "Innerschweiz", "Graubünden", "Ostschweiz", "Wallis", and "Zurich" were utilized in this thesis as well. Additionally, the SwissDial dataset by Dogan-Schönberger et al. [11] of around 28 hours of audio was appended to the STT4SG-350 corpus due to the high-quality audio contained within. The two datasets have slightly differing dialect regions, so the SwissDial split was integrated into the STT4SG-350 definition. The canton "AG" was merged with the Zurich dialect¨, the canton "LU" was merged with the Innerschweiz dialect, and the canton "SG" was merged with the Ostschweiz dialect. The resulting merger of these two corpora is termed the STT4SG-350-corpus. The corpus and its dialect splits were utilized to train and validate the TTS-models. However, the 350 hours were insufficient to train a model to synthesize Swiss German dialects sufficiently.

3.1 Data Crawling

To increase the available hours of speech for training, it was necessary to collect them online. Large sources of Swiss German speech are found in podcasts and radio programs, either done professionally or as a hobby by private individuals. Podcasts are generally uploaded to specific platforms such as Spotify, Apple Podcast, or YouTube (YT). Audio from YT can be indirectly downloaded using third-party packages. The biggest source of directly available Swiss German audio is the state-owned Schweizer Radio und Fernsehen (SRF) TV station. It is possible to download metadata through their API [25] of the various podcasts and radio programs the TV station self-produces or funds. In the metadata included were download links to the episodes of a given podcast, which allowed us to utilize these podcasts in training as well. Upon this discovery, it was decided to use the two sources, SRF and YT, to download more audio data. As described above, we used the official SRF-API [25] to get SRF related podcasts. For YT, we used the Python pytube library [26], specifically the pytubefix library by Jan Bindez [27], as the official repository has been stale for over two years and can no longer be used with the YouTube website

due to natural changes on the platform over time.

As no catalogues were accessible for both SRF and YT containing existing podcasts and their contents programmatically, a manual selection process had to be utilized to collect them. In the case of YouTube, two websites were used as the initial reference point. The first was Podcastclub [28], which defines itself as "...bringing Swiss audio producers to exchange ideas and to help each other..." by advertising and pushing Swiss German audio mediums online. The second platform used was Podcastschmiede [29], a podcast incubator in the Swiss-German language space. Advertised podcasts from both sources were then searched on YouTube to verify if they uploaded their content to the platform. Verification was performed if a podcast was available, checking if Swiss People hosted the podcast in either Swiss German or Standard German and that the audio quality was agreeable by skipping through up to six podcast episodes.

For SRF, the audio was verified by parsing through the publicly accessible audio library [30] and noting down the different podcasts that matched the goal of this thesis. As such, radio programs containing a lot of music or singing were automatically removed. Comedy-oriented podcasts were also removed due to frequent laughter, clapping, and general background noise. The generated list is not exhaustive, and during the writing of this thesis, more podcasts were identified that can be utilized for training. Due to time constraints, they were subsequently left out of the data collection. Future work may follow up on these identified podcasts and use them in their data collection. They are listed in the Appendix in Table A.1 and have 9767 hours of uncut audio in total.

After filtering the SRF podcasts and searching the episodes through the metadata API, it was discovered that not all were available for download. This further limited the list of actual usable podcasts. Consequently, the podcasts were categorized into three different categories, "ch", "de", and "mixed". These terms were given manually after skipping through the episodes and verifying if the hosts spoke Swiss German only - the "ch" tag -, German only - the "de" tag, or if there was a mix of both present - the "mixed" tag. The tagging did not replace the actual Dialect Identification (DID) but was done to prioritize which podcasts were processed first, ordering them from top to bottom with "ch", "mixed", and then "de". The German-only podcasts were also kept under the assumption that it helped with the fine-tuning process of multilingual models. Additionally, it is important to note that Standard German, spoken by Swiss German, has a very distinct accent and its own name, "Swiss Standard German", which can be detected very quickly by native speakers. While significant training in normal Standard German may reduce this accent, journalists and news broadcasters from outside Switzerland who work for SRF are often encouraged to learn this accent to sound like a person from Switzerland, as shown in the case of Anna-Lisa Achtermann [31]. This is due to considerable scrutiny by the public for Swiss news and radio to sound distinctly Swiss and not German.

The resulting audio dataset, termed "SRF-corpus" from here on out, comprises 12 podcasts from YT with 688 hours and 25 podcasts from SRF with a total of 4714 hours of uncut audio, creating a dataset with a total of 5403 hours of audio. Detailed dataset analysis, including the STT4SG-350-corpus, is given in Section 3.3. Table 3.9 lists the different podcasts and their respective size.

3.2 Data Pipeline

The SRF-corpus was not yet usable for training as the audio was still on an episode basis and not on a speaker basis. As such, a data pipeline had to be created, automatically performing Speaker Diarization (SD), segmentation, and weakly labelling the data for further use. The complete pipeline is visualized in Figure 3.1, and each pipeline component will now be discussed. All components were evaluated for their effectiveness against samples taken from the popular SRF podcast "Zivadiliring"[1]. The podcast comprises three hosts: Yvonne Eisenring, Gülsha Adilji, and Maja Zivadinovic. This podcast was chosen because of the general adherence to speaking in Swiss German dialects with few guest appearances, which streamlined evaluation. Gülsha Adilji was born in Uzwil, in the canton of St. Gallen, designated in the dialect region of "Ostschweiz". The two other hosts were both born in Zurich and speak the "Zurich" dialect.

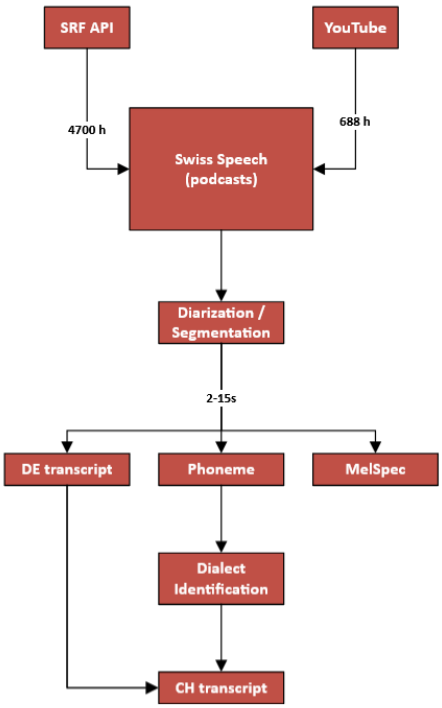

Figure 3.1: Data pipeline for labelling SRF and YT audio.

[1]https://www.srf.ch/audio/zivadiliring

3.2.1 Speech Diarization

The first step after downloading the SRF-corpus was Speaker Diarization (SD). Speaker Diarization refers to partitioning audio into segments where specific speakers are talking and automatically classifying them, as visualized in Figure 3.2. By using Voice Activity Detection (VAD), a method of detecting human speech in audio, it is possible to accurately filter out silence or noise and keep the relevant audio segments for further processing. The powerful pyannote speaker diarization 3.0 pipeline [32][33] on Huggingface[2] was used for this. Compared to the speech segmentation pipeline[3] by the same company, the diarization pipeline can handle longer audio rather than being limited to 10 seconds like the segmentation pipeline and due to this limitation, choosing the latter as the sole pipeline was necessary. The pipeline ingests mono audio sampled at 16kHz and generates a file in the .rttm format containing the different speech segments and their assigned speaker. An example of such a .rttm file can be found in Figure 3.3. The pipeline was trained on a combination of different training sets originating from AISHELL [34], AliMeeting [35], and AMI [36], among others.

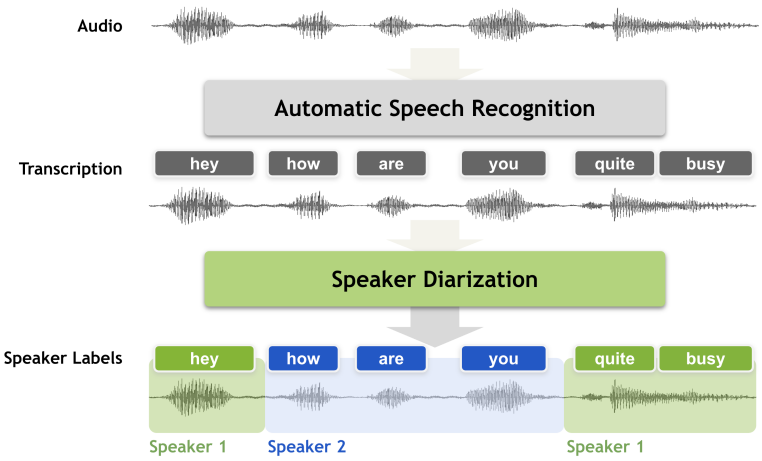

Figure 3.2: Speaker diarization visualization, Figure taken from [37]

Upon listening to the various podcasts, it was found that giving an accurate number of speakers for each podcast was impossible due to shifting hosts, guest appearances or lack thereof, missing hosts due to sickness, and so on. Some podcasts had accurate information in the episode descriptions, while others had none or did not list the present hosts. As such, it was decided that the pipeline would be configured to have at least two speakers ($min\_speakers = 2$) and, at most, six speakers per episode ($max\_speakers = 6$). This allowed for a general setup that could handle the various podcasts and gave the pipeline enough flexibility in its detection.

To evaluate the performance of the diarization pipeline, a manual SD was performed by the author using ELAN [38] and the podcast episode "Wie wir wirklich

[2]https://huggingface.co/pyannote/speaker-diarization-3.0
[3]https://huggingface.co/pyannote/segmentation-3.0

sind"[4]. The episode was chosen randomly and has a duration of 42 minutes and 38 seconds with the three previously mentioned hosts as the speakers. ELAN does not allow direct conversion of the diarized audio to the .rttm format, so it had to be converted using the pympi [39] Python package. The generated .rttm file by pyannote, termed "hypothesis", is illustrated in Figure 3.3 while the manual diarization, termed "reference", is visualized in Figure 3.4. The speakers' names were not translated to their real names but contained the same host in the background. For example, Speaker_05 is the same speaker in both reference and hypothesis, i.e. Gülsha Adilji.

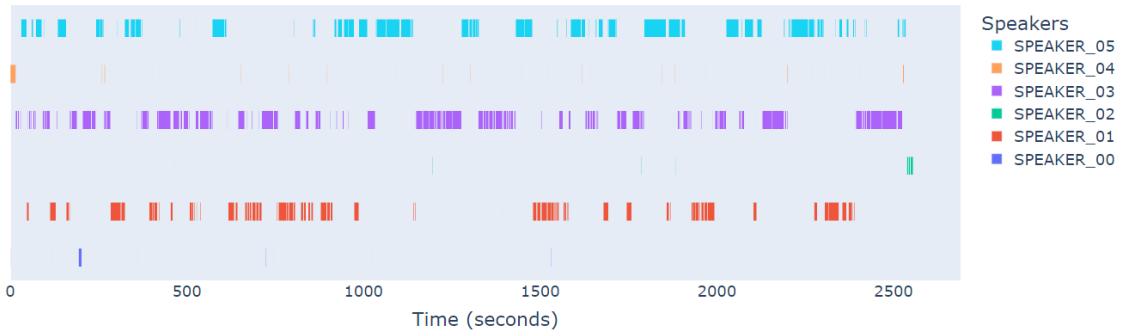

Figure 3.3: Hypothesis .rttm file generated by pyannote pipeline of an episode from the Zivadiliring podcast.

Figure 3.4: Reference .rttm file created by manually performing the diarization using ELAN[38] of an episode from the Zivadiliring podcast.

Before evaluating the pair with metrics, a short inspection was made to find out why, in the hypothesis, there were six different detected speakers while the reference only contained four. Firstly, it was found that the intro and outro of the podcast episode were detected as individual speakers due to voice changes performed on the actual speaker. As such, the intro was assigned to its own speaker, Speaker_00, in blue for both reference and hypothesis. Next, a segment at the beginning of the episode was assigned to Speaker_04 in the hypothesis, while in actuality, it was

[4]https://www.srf.ch/audio/zivadiliring/wie-wir-wirklich-sind?id=384359e2-ace7-4a92-8c6d-ba07ccf41316

Speaker_05. It was misidentified because the person sang a happy birthday song in English in a different tone than the rest of their segments. Lastly, the outro was assigned to Speaker_02 as it again had significant voice changes with music in the background, which may have confused the pipeline. The clue was in the outro, where the speaker identified themselves as Yvonne Eisenring, who was Speaker_03.

The pyannote metrics [40] library was used for metrics-based evaluation as it contained predefined metrics to quantify various aspects of the diarization pipeline. Three different metrics were looked at, which will be explained here.

Detection Error Rate

The first metric is the Detection Error Rate (DEER), which evaluates the VAD module. It is formalized in Equation 3.1 where the $false\ alarm$ is the duration of non-speech incorrectly classified as speech, $missed\ detection$ is the duration of speech incorrectly classified as non-speech, and the $total$ is the total duration of speech in the reference. [40]

$$DetectionErrorRate = \frac{false\ alarm\ +\ missed\ detection}{total} \tag{3.1}$$

The Detection Error Rate between hypothesis and reference .rttm was 0.0146 or 1.46% with 33.4550 missed detections, 0 false alarms, and a total speech duration of 2288.698 seconds or 38.13 minutes. Compared to the original episode duration of 42.63 minutes, VAD reduced the usable audio by 10.55% by removing unnecessary parts.

Segmentation Purity Coverage

The Segmentation Purity Coverage score is used to evaluate the change detection in a system, i.e. when the speakers change. Instead of Precision and Recall, the library's authors believe that a segment-wise purity and coverage score is better suited for quantifying segmentation. Segment-wise coverage is calculated for each reference segment by determining the ratio between the duration of its intersection with the most overlapping hypothesis segment and the total duration of the reference segment. For example, the coverage for reference segment 1 is 100% because hypothesis segment A completely covers it. [40]

Purity, on the other hand, is a complementary metric that measures that measures how pure hypothesis segments are. For instance, hypothesis segment A has a purity of 65%, as it overlaps 65% with reference segment 1 and 35% with reference segment 2. The final metrics are computed as duration-weighted averages across all segments, with the weighted term $Beta$ being equal to 1 per default. [40]

$$F - Measure = (1 + Beta * Beta) * \frac{Purity * Coverage}{Beta * Beta * Purity + Coverage} \tag{3.2}$$

During the evaluation, the total duration of calculated segments was 2340.7420 seconds, of which purity had an intersection duration of 2207.92 seconds and coverage had an intersection duration of 2205.392 seconds. This resulted in a weighted F-measure of 0.9427 for the Segmentation Purity Coverage.

Diarization Error Rate

Lastly, the Diarization Error Rate is the standard metric for evaluating and comparing speaker diarization systems, as formalized in Equation 3.3. $False\ alarm$ refers to the duration of non-speech mistakenly classified as speech, *missed detection* is the duration of speech incorrectly identified as non-speech, $confusion$ represents the duration of speaker misclassification, and the total corresponds to the sum of the reference speech durations across all speakers. It is important to note that this metric accounts for overlapping speech, which may result in higher missed detection rates if the speaker diarization system lacks an overlapping speech detection capability. [40]

$$DER = \frac{false\ alarm + missed\ detection + confusion}{total} \tag{3.3}$$

In the comparison of the hypothesis and the reference .rttm, the Diarization Error Rate was found to be 14.15% with $false\ alarm$ accounting for 49.548 seconds, $confusion$ for 213.911 seconds, $missed detection$ for 80.832 seconds, and a total duration of 2433.022 seconds. This was found to align with the pipeline's performance on the AISHELL-4 corpus [34] where it had a DER of 14.1% [33] and is, as such, acceptable.

3.2.2 Speech Segmentation

The next step entailed performing the actual segmentation of the diarized speech into smaller samples. The chosen approach was to load the .rttm files into the pyannote library and automatically cut the audio files into their smaller constituents based on certain rules. These rules will now be explained.

First, a segment of a single speaker must be longer than two seconds but less or equal to 15 seconds. Next, overlapping audio segments were cut so that only one speaker at any time was speaking based on the diarized speech. The untangled segments were included as a sample if they were longer than two seconds. Neighbouring segments of the same speaker were combined to create longer segments, as many of the diarized segments were found to be extremely small (<0.5 seconds). However, this last rule had an exception: if the length between segment n and segment $n+1$ was larger than 2 seconds, the merging of segments was halted. This was done to reduce the Introduction of unnecessary long silences in the speech samples. Such a case can be seen in Figure 3.5. Manually checking segmented samples afterwards confirmed that the segmentation was not perfect, as it was noted that, at times, multiple speakers were in a single sample or that certain background noises were also present, such as clapping, laughter, sneezes, or background music that was

playing during the speech. Considering the DER of 14.15%, these imperfections were accepted. The number of samples per podcast is given in Table 3.9.

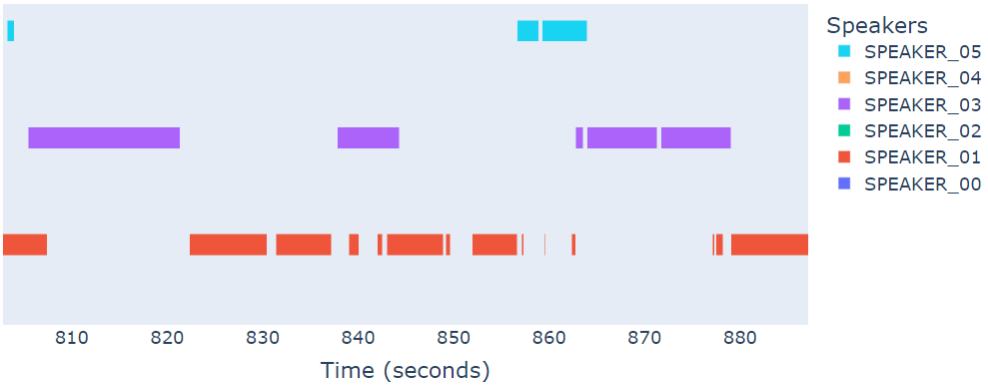

Figure 3.5: Silence of longer than 2 seconds in Speaker 01 around the 850 seconds mark, which would not have been combined with the following segment due to the segmentation process.

The cut samples were saved directly to hdf5 files on an on-podcast basis using the h5py library [41]. This was done to streamline the storage of the different podcast files into a central location and allowed for any generated metadata, such as transcripts, to be saved as attributes in the hdf5. The audio samples were converted to .wav files and sampled to 16kHz as multiple models required this in the downstream data pipeline and reduced the file size of the corpus. The sampling decision did provide a challenge for the training of our model later on and will be elaborated upon in Chapter 4. The segmentation process produced 1'810'479 unique samples for the SRF-corpus.

A significant error was propagated from this step to the rest of the pipeline, which had consequences for the training of the models later on. The segmented audio samples of 15.0 seconds were left as is instead of running them through a translation model with forced alignment, such as whisper-x [42], to reduce their length by cutting the segments on a sentence-basis. This was only detected after realizing that the time distribution was primarily concentrated on 15-second samples, as shown in 3.6, and that the resulting token distribution of the SRF-corpus was bimodal, i.e. had two significant peaks as seen in Figure 3.7. At that point, it was too late to change the pipeline and execute the transcription again. Analysis of the impact this decision had is done in Chapter 5. Future work may improve upon this and rerun the experiments.

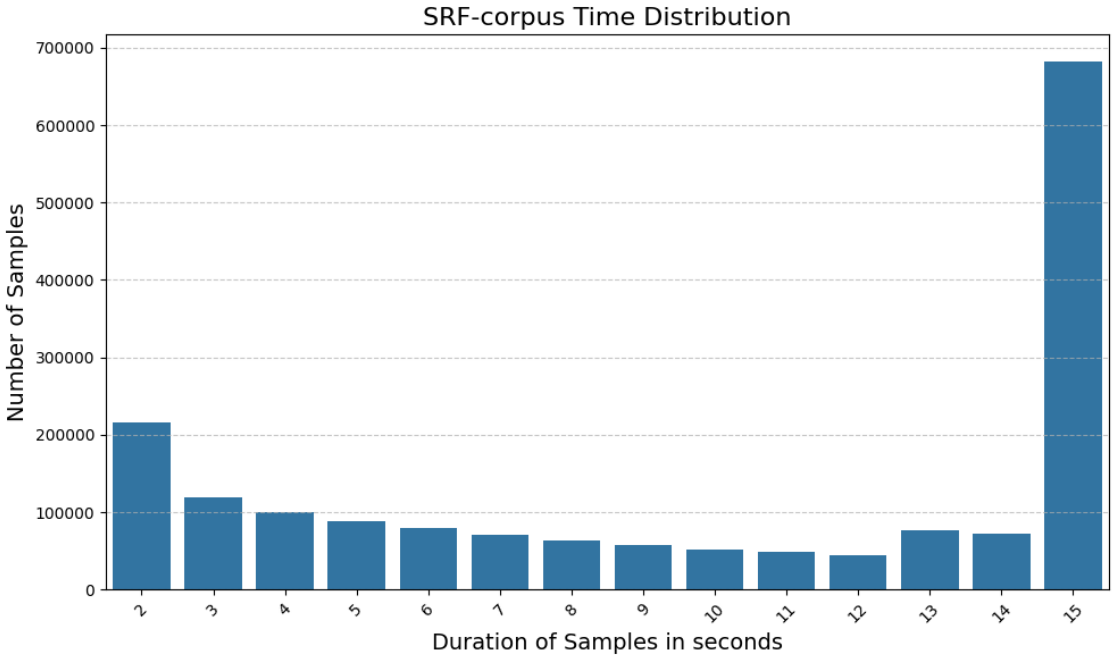

Figure 3.6: Time distribution of SRF-corpus. The duration of samples has been rounded up towards the next bigger integer.

3.2.3 Speech to Standard German Text

The first enrichment process included the transcription of the samples to Standard German. As outlined in the Introduction, Swiss German does not possess a standardized grammar or vocabulary and is, as such, extremely difficult to transcribe. It is thus the norm to use Standard German to train and evaluate any Swiss German model. The model chosen for this task was Whisper-large-v3 [15] using the Huggingface repository[5]. During transcription, an edge case was found with the whisper model, which tended to generate repeated word segments for a whole sample. Some research into the problem yielded that this was occurring in other languages as well, i.e. Japanese[6], and was not limited to the task of transcribing Swiss German to Standard German. A parameter was then found that could control the generation of tokens, called *no_repeat_ngram_size*. We set this parameter to $no\_repeat\_ngram\_size = 2$, which reduced the occurrences of these repeated word segments. If it was detected during transcription, the sample was transcribed again. If the issue persisted again, the sample was labelled with "NO_TEXT" and not used during training. In total, 2811 samples were filtered due to this issue, which was around 0.155% of the total dataset.

The transcription of the different podcasts was run in parallel on three different A100 GPU instances with a batch size of 32 on each. The resulting token distribution, calculated using spaCy [43], of the SRF-corpus is illustrated in Figure

[5]https://huggingface.co/openai/whisper-large-v3
[6]https://github.com/openai/whisper/discussions/1059

3.7. The image clearly depicts a bimodal distribution of tokens, which confirmed the finding in the segmentation part of the pipeline, in which the time distribution was focused on 15-second samples. The significant drop in sentences with token counts of 14 to 40 was found to be the speaking characteristics in podcasts. While controlled environments, like in [10] focus on clarity and controlled speed in their speech, podcasts tend to be faster, more chaotic and characterized by interruptions by the different hosts, speaking over each other in turns or simultaneously, or long segments of monologues by hosts telling a story, reading letters, news broadcasts, or similar. The former characteristic of interruption results in the first peak seen in the graphic between 7 and 14 tokens. In contrast, the latter characteristic of monologue produces the second peak in the distribution for token counts of 40 to 53. The STT4SG-350-corpus had a single peak and much smaller token counts, as visualized in Figure 3.13.

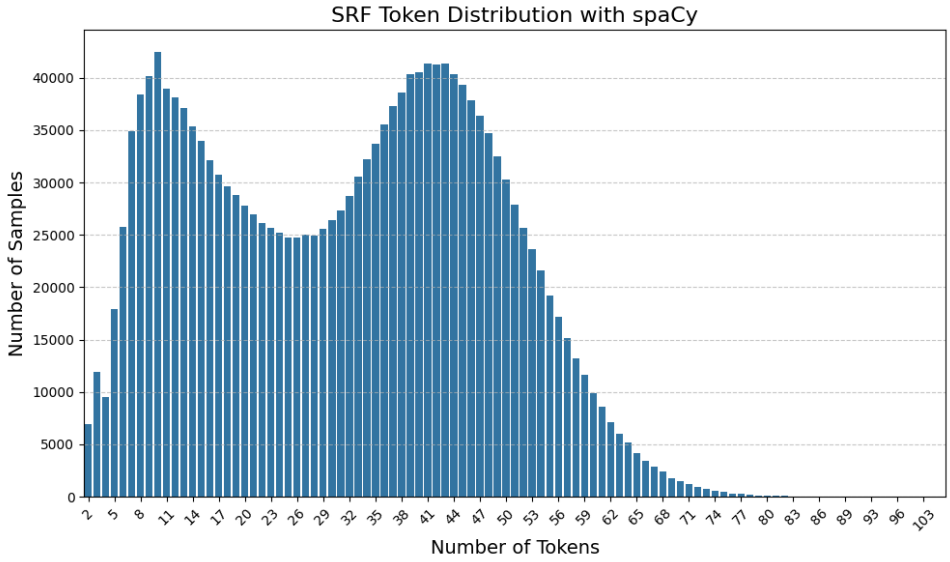

Figure 3.7: SRF-corpus token distribution produced with spaCy.

The output of Whisper was evaluated by sampling 100 random audio segments of the completely transcribed Zivadiliring podcast and manually transcribing them. The reference and hypothesis samples were then evaluated using different metrics, which will be explained here. Several such manually transcribed texts with their associated hypothesis can be viewed in Table 3.1.

Word Error Rate

Word Error Rate (WER) by is derived from the Levenshtein Distance [44]. It quantifies how many errors exist in a generated text by measuring the number of insertions, deletions, and substations needed to transform it to the given reference text. WER is widely used in domains like Automatic Speech Recognition (ASR) and machine translation with the formula given in Equation 3.4 where S are the number of substitutions, D the number of deletions, I the number of insertions, and N the total number of words in the reference text.

$$WER = \frac{S + D + I}{N} \tag{3.4}$$

Character Error Rate

Character Error Rate (CER) is also based on the Levenshtein Distance [44], but unlike WER, quantifies the single-character transformations needed instead of the word-level transformations. The formula is the same as the WER and given in Equation 3.5.

$$CER = \frac{S + D + I}{N} \tag{3.5}$$

BLEU-Score

BiLingual Evaluation Understudy (BLEU) is a widely adopted evaluation metric in machine translation, originally proposed by Papineni et al. [45]. The core idea is to assess how well a machine-generated translation matches human reference translations by counting overlapping n-grams (i.e., sequences of 1, 2, 3, or more words) between the hypothesis and the reference. A significant drawback of this metric is that it does not capture nuances, such as context or fluency, due to its reliance on n-grams. [46][47] The formula is given below in Equation3.6, where BP is the Brevity Penalty, p_n the modified n-gram precision, and w_n the weights for different n-grams.

$$BLEU = BP * exp\left(\sum_{n=1}^{N} w_n * log(p_n)\right) = BP * \prod_{n=1}^{N} p_n^{w_n} \tag{3.6}$$

BERTScore

BERTScore was introduced by Zhang et al. [48] and evaluates the performance of the text generation model by using contextual embeddings of the reference and hypothesis sentences from a pre-trained BERT [49] model. These embeddings capture semantic meaning, allowing BERTScore to evaluate text quality based on meaning rather than exact matches. A cosine similarity between the two sentences is then used to calculate the score. BERTscore is an F1-score based on recall, measuring how the reference tokens match the hypothesis tokens, and precision, measuring how the hypothesis covers the reference tokens and is formalized in Equation 3.7. The metric can counterbalance more strict metrics such as BLEU [45] or ROUGE [50] and is thus provided in this evaluation.

$$F1_{BERT} = 2 * \frac{P_{BERT} * R_{BERT}}{P_{BERT} + R_{BERT}} \tag{3.7}$$

Evaluation Result

Evaluation for WER and CER was performed using jiwer[51], BLEU using nltk [52], and BERTScore the bert_score [53] library. To provide more insight, except for BERTScore, the scores were provided by comparing the sentences normally and with lowered characters only. However, the focus of this thesis is not on case-sensitive spelling but on the content itself. Owing to this, only the lowered scores will be discussed from here on out, with the normal scores being provided as general insight. Additionally, the differences between normal and lowered were minimal, often within 1-2 scores, as shown in Table 3.2.

Length (s)	Hypothesis Sentence	Reference Sentence
6.835	Und dann ist quasi die Idee, wenn du als Burning Man gehst, dass du etwas wie einen Provider machst.	Dann ist quasi die Idee auch, dass du, wenn du an das Burning Man gehst, etwas providest
12.015	Aber es ist nicht gescheitert. Nein, ich bin ja so hyperemotional. Dann verplatzt es mich und dann bin ich aber wieder ruhig nach vier Sekunden. Aber ich habe dann schon wahrscheinlich ein bisschen umgewettert.	Aber es ist nicht gescheitert an der Wäsche. Nein, ich bin ja schon, ich bin ja so Hyperemotional, dann verplatzt es mich, dann bin ich aber auch wieder ruhig nach vier Sekunden. Aber habe dann wahrscheinlich schon herumgeflucht.
4.826	Oder was ist er? Weisst du, mit dem Rettchen wüsstest du, über was wir reden. Was ist er gestern gewesen? Was ist er heute?	Oder was ist er? Weisst du damit wir wissen über was wir reden. Was war er gestern? Was ist er heute?

Table 3.1: Comparison of Generated and Manual Sentences with durations

Statistics	WER	WER Lower	CER	CER Lower	BERTScore	BLEU	BLEU Lower
Mean	0.3000	0.2868	0.1951	0.1907	0.9130	0.5462	0.5571
Median	0.2124	0.2053	0.1179	0.1112	0.9405	0.6408	0.6412
Std Dev	0.2642	0.2538	0.1949	0.1920	0.0747	0.3433	0.3375

Table 3.2: Whisper evaluation metric scores normal and lowered comparisons

The evaluation yielded an average WER of 0.2867, a CER of 0.1906, BERTScore of 0.91, and a BLEU of 0.5571, which are very similar to the in [54] observed results. As a reference, the WER performance of an XLS-R Wav2Vec 1B model [55] trained on the STT4SG-350 [10] corpus reached 14.0±0.1 on the test split. Considering the difficult task of transcribing Swiss German speech in an uncontrolled environment like ours, the scores were considered acceptable.

3.2.4 Speech to Phoneme

The second enrichment process concerned the creation of phoneme text based on the audio. Phonemes are the smallest speech units in a language and can differ based on the given language. [56] This process utilized a Wav2Vec2 model [55] by Xu et al. [57] hosted on Huggingface[7]. The phonemes were needed for the Dialect Identification (DID) and subsequent clustering of samples. As with the Standard German transcription, the phonemes were generated on three A100 instances with a batch size of 32. An issue was encountered in generating the phonemes; the model did not generate any phonemes for certain samples. The reason as to why this occurred could not be found. If the model did not generate a phoneme text for a sample after a second attempt, the sample was tagged with "NO_PHONEME" and ignored in the subsequent pipeline steps, similar to the Standard German transcription process issue. 231 samples were filtered out due to this issue, corresponding to 0.012% of the SRF-corpus. The phonemes were extrinsically evaluated in the DID step of the pipeline, as it was impossible to verify the accuracy of the texts by hand due to time and cost constraints.

3.2.5 Phoneme to Dialect Identification

As this thesis aims to generate Swiss German in different dialects, the samples of the SRF-corpus had to be classified by their spoken dialect. A Naive Bayes classifier developed by Bolliger et al. [58] was utilized for this step. The training consists of feeding the model phonemes based on one of the seven dialects outlined in [10]. Training and Evaluation data was taken from a subset of the STT4SG-350 [10] corpus, comprising around 30 hours for each dialect. The phonemes were constructed using the same Wav2Vec2 model by Xu et al. [57] used in the phoneme transcription step. The training was performed twenty times, and the models were evaluated using a macro average F1-score, with the best-performing model, termed "naive-ch-only", reaching a score of 0.87. Generally, it was found that the model performed best when providing 30 seconds or more of phonemes generated by the same speaker. The confusion matrix for that model can be seen in Figure 3.8. The evaluation shows that most dialects are detected very accurately, with only the Bern and Basel regions showcasing minor difficulties with being misclassified by each other.

[7] https://huggingface.co/facebook/wav2vec2-xlsr-53-espeak-cv-ft

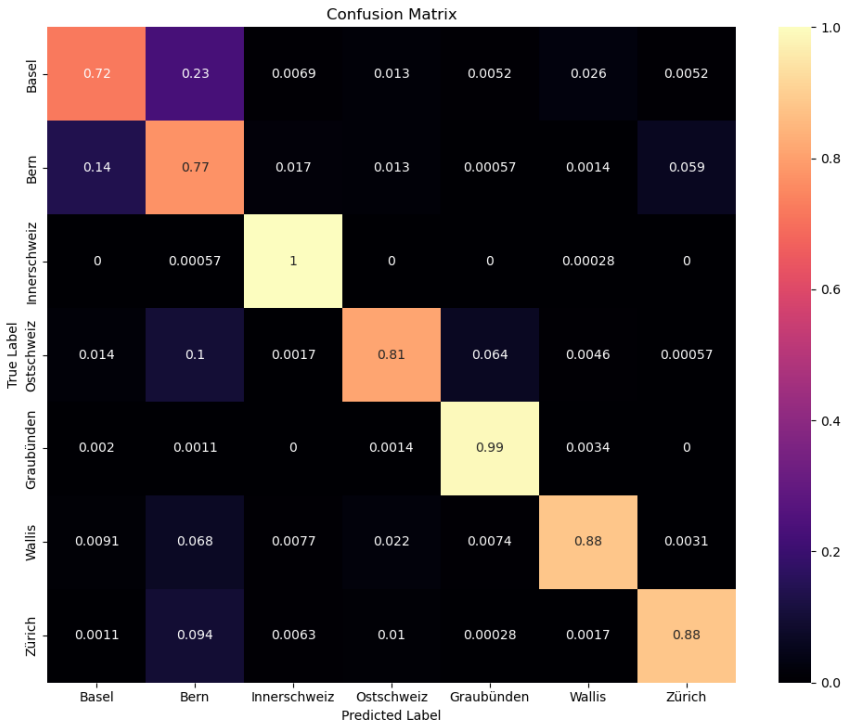

Figure 3.8: Confusion matrix of Naive Bayes CH-only model classifying only Swiss German dialects.

This naive-ch-only model was only used to evaluate the TTS-model, as the generated audio samples only required a classifier for Swiss German. However, an additional class for speech in Standard German was needed in the data pipeline owing to the inclusion of German in the SRF-corpus. As such, additional data had to be added to the training. This data was taken from the v19.0 CommonVoice corpus [59]. As outlined above, each Swiss German dialect consisted of 30 hours of audio, so filtering the German CommonVoice corpus down was necessary. Apart from duration, the samples were randomly chosen based on the approximate distribution of gender and age in the subset of the STT4SG-350 dataset splits and then appended to the training. The STT4SG-350-corpus splits are shown in Table 3.3 for gender and Table 3.4 for the age groups.

Dataset Split	**Male**		**Female**	
	Count	**Freq (%)**	**Count**	**Freq (%)**
Train	93186	46.66	106519	53.334
Validation	12161	52.38	11056	47.62
Test	10394	42.24	14211	57.76

Table 3.3: Gender distribution across SNF-dataset splits for Dialect Identification model training.

The German samples were automatically transcribed to phonemes using the Wav2Vec2 model by Xu et al. [57]. Training, mirroring the naive-ch-only model, also consisted of running 20 different iterations and choosing the best-performing model. Said model, termed "naive-with-de", reached an F1-score of 0.88 over the

Age Group	Train		Validation		Test	
	Count	Freq (%)	Count	Freq (%)	Count	Freq (%)
Teens	8678	4.34	2204	9.49	661	2.69
Twenties	68217	34.16	8843	38.09	8918	36.24
Thirties	37363	18.71	1110	4.78	4217	17.14
Forties	37238	18.65	6619	28.51	5234	21.27
Fifties	22638	11.34	1109	4.78	1382	5.62
Sixties	22845	11.44	3332	14.35	4193	17.04
Seventies	2726	1.37	0	0.0	0	0.0

Table 3.4: Age Distribution across dataset splits of STT4SG-350-corpus

eight different classes. Figure 3.9 illustrates the resulting confusion matrix. What is interesting to note in the Figure is that German was identified correctly in nearly 100% of cases, showcasing the significant difference in pronunciation between Swiss German and Standard German. Additionally, the exhibited misclassification by the regions of Bern and Basel in the ch-only model largely disappeared in the naive-with-de model. Instead, the neighbouring regions of Zurich and Innerschweiz now exhibited misclassification issues.

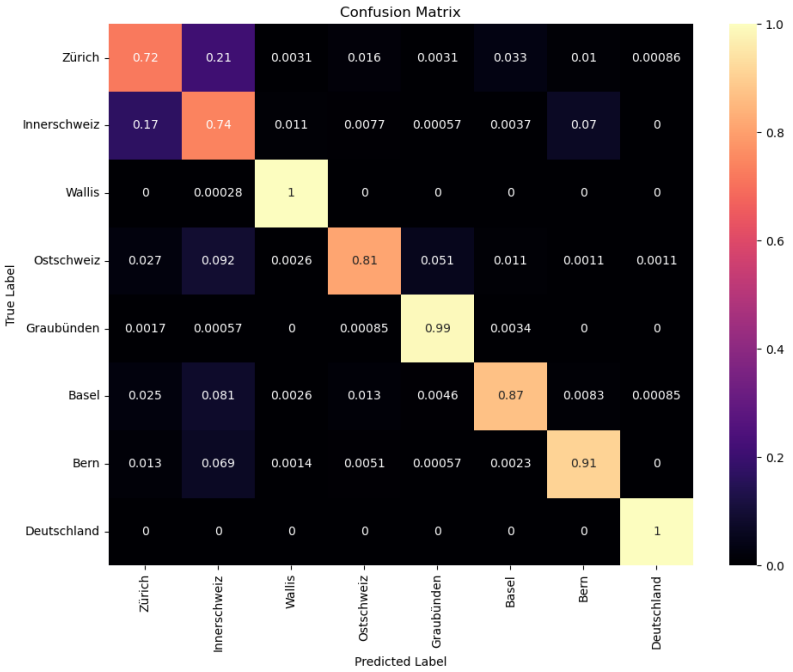

Figure 3.9: Confusion matrix of Naive Bayes model classifying Swiss German dialects and Standard German.

As with previous pipeline steps, evaluating the model's performance on the Zivadiliring podcast was necessary before applying it to the complete SRF-corpus. We compared three approaches: First, we combined all phonemes generated by a speaker in a given episode and fed them as a single string to the naive-ch-only and the naive-with-de models. The ch-only model was included to see if adding German

to the model introduced any significant changes in the classification of data from the SRF-corpus. The second approach consisted of combining segments of around 100 seconds by a given speaker in a given episode and applying a simple majority voting of the model. This approach was only applied to the naive-with-de model. Lastly, an internally at CAI whisper-small [15] model was also compared with the Naive Bayes models to provide insight into the performance of a different architecture. The model was trained on classifying 30-second samples using a Train-k-samples approach, with $k = 20$ and reached an F1-score of 0.77.

Due to two hosts' upbringing in the Zurich region and one host's upbringing in the Ostschweiz region, most Zivadiliring samples were expected to be classified in those regions. This was confirmed by all four models as illustrated in Figure 3.10, in which approximately 2/3 of the samples were classified as Zurich, 1/3 as Ostschweiz, and a small proportion in other regions. The samples classified as Basel were confirmed to be accurate, as the episode "Dirty talk auf serbisch"[8], to which most of these samples belong, hosted a guest named Milan Milanski, who grew up in Basel. The Innerschweiz classified samples were found to be misclassified mainly by the model originating from the overall difficulty the model exhibited with Zurich and Innerschweiz as illustrated in Figure 3.9. The approach of using majority voting largely offsets this issue due to the 3/4 correct classification of both Innerschweiz and Zurich. Lastly, while the whisper model did show similar performance to the Naive Bayes model, it was significantly slower than it. Owing to these findings, it was decided to use the naive-with-de model utilizing the majority voting approach for all classifications in the SRF-corpus.

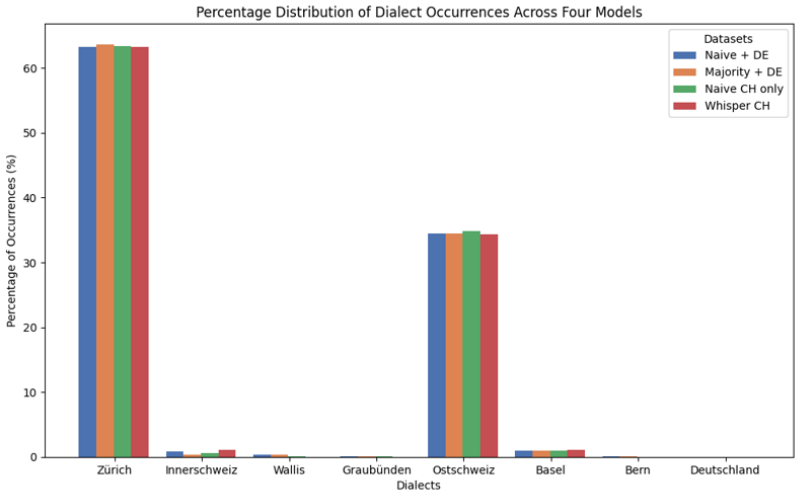

Figure 3.10: Distribution of detected dialects by four evaluated approaches.

[8]https://www.srf.ch/audio/zivadiliring/dirty-talk-auf-serbisch?id=24fb5e87-231a-48ad-a14c-9191adc82d37

3.2.6 Standard German Text to Swiss German Text

Using the generated German text and the DID, a T5 model by Bollinger et al. [16] was applied to generate Swiss German text. The model was trained using the textual data of the SwissDial corpus [11]. To gain insight into the model's performance, a similar evaluation to that of the Standard German text generation was performed using the same 100 sentences. The model thus received as input a dialect tag of the form "[ch_XY]:", with XY denoting the specific dialect region and a Standard German sentence. The Standard German sentence was the manually transcribed sentence, not the sentence generated by the whisper model. As the Introduction outlines, Swiss German has no unified grammar or spelling, so it changes from one person to another, even if they belong to the same dialect region. The sentences were transcribed into Swiss German to the best of the author's ability. However, the result may be skewed due to a bias from the author's upbringing in the Ostschweiz region. There was insufficient time to source a person from Zurich to transcribe the sentences. The author thus attempted to copy the Zurich style as best as possible. The result of the transcription can be seen in Table 3.5.

Statistics	**WER**	**WER Lower**	**CER**	**CER Lower**	**BERTScore**	**BLEU**	**BLEU Lower**
Mean	0.639	0.63	0.396	0.392	0.805	0.109	0.115
Median	0.636	0.632	0.344	0.341	0.802	0.0	0.0
Std Dev	0.253	0.251	0.285	0.284	0.064	0.167	0.169

Table 3.5: T5 evaluation metric scores normal and lowered comparisons.

The scores show that the generated texts in their current form are very inaccurate. This most probably occurred due to the non-standardized way Swiss German is written. Table provides a good, average, and bad sample from this evaluation to allow a more insightful understanding. The Swiss German text was not used for any task downstream.

3.2.7 Speech to Mel Spectrogram

The last enrichment process was the creation of a Mel Spectrogram for each sample using the librosa library [60]. As with the DID process, it was also decided to use the joblib library [61] for parallel computation of the Mel Spectrogram with up to 16 simultaneous jobs. The data was not used for any further process and can be viewed as a general addition to the SRF-corpus.

Dialect	Hypothesis Sentence	Reference Sentence	CER	BLEU
Zürich	Si känt scho mal din Name, fast. Er isch Content Creator, er isch berüehmt im Internet und er isch super.	Sie kennt scho mal din Name, fast. Er isch Content-Creator, er isch berüehmt im Internet und er isch	0.100	0.6383
Ostschweiz	I mein, wa de Onur alles seit. Nur will me zemme wohned isch jetzt nöd de Informationsfluss.	Ich meine, was dä Onur alles seit. Nur will mir zemme wohnet isch ezt do nöd de Informationsfluss.	0.1326	0.2855
Zürich	Drum händs so gfunde, ja du bisch irgendwie d'Muetter und denn au irgendwie nöd. Ich glaub, es git nöd die definiert Rolle. Aber ich han so gfunde d'Klaschtante isch no härzig.	Drum hät si d Mueter gfunde, dass si das au nöd gseh hät, dass Klatschstunde no härzig isch.	0.698	0.0

Table 3.6: Comparison of Generated and Manual Annotated Swiss German Sentences

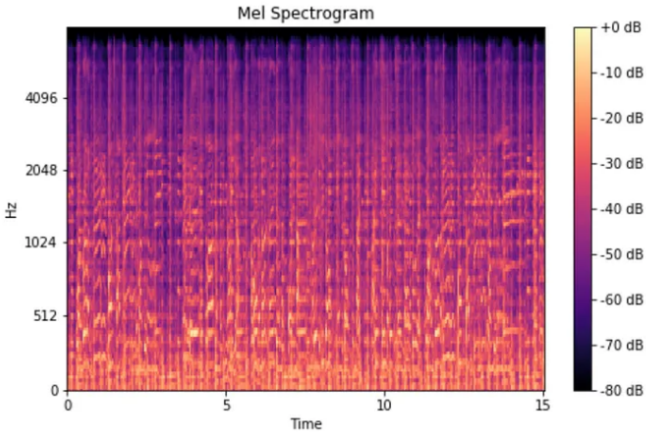

Figure 3.11: Image of a Mel Spectrogram, Figure taken from [62]

3.3 Data Analysis

This section will go into a data analysis of the STT4SG-350-corpus and the SRF-corpus on characteristics not discussed during the general description of the data pipeline. The aim is to provide a comprehensive insight into both corpora before delving into the TTS setup.

3.3.1 STT4SG-350-corpus

First, the smaller STT4SG-350-corpus will be examined, specifically the training and test split. As the corpus consists of pre-existing datasets, this analysis will be kept to a minimum. Further information is found in the original papers about the STT4SG-350 [10] and SwissDial [11] dataset.

There are 246'694 unique samples in the training set shared across 240 speakers in the STT4SG-350 and eight speakers from the SwissDial dataset. The duration distribution of the samples is found in Figure 3.12, which shows that the distribution is centred around three to five seconds. When checking the token distribution in Figure 3.13, it seems largely centred around 7 to 12 tokens, correlating well with the length of a single sentence. Sentences were largely unique in the training split for the samples, so only a small subset of speakers shared the same sentence. Contrary to the training set, in the test split of the STT4SG-350 dataset, which was also used as the source of the test split in this thesis, the same 3515 sentences were recorded across the seven dialect regions, with each region consisting of at least eight unique speakers. This enables a fairer comparison, much required for downstream evaluation of the TTS model. [10]

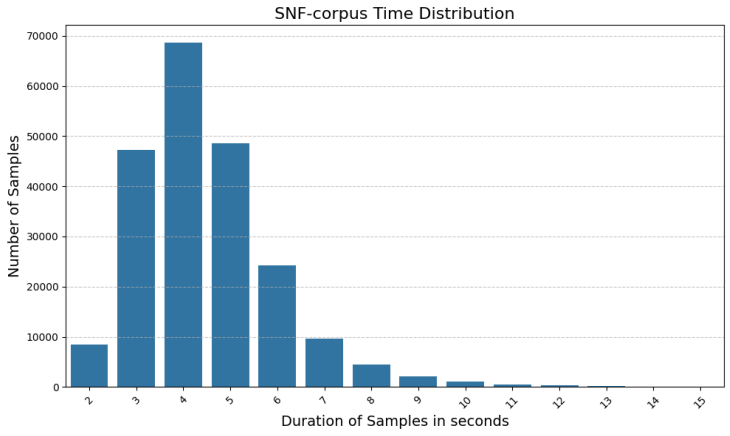

Figure 3.12: Time distribution of SNF-corpus. The duration of samples has been rounded up towards the next bigger integer.

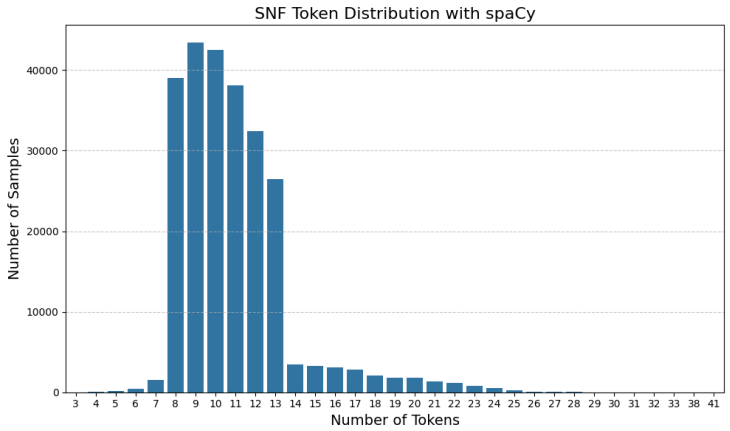

Figure 3.13: SNF-corpus token distribution, generated with spaCy.

The dialect regions in the training set are represented by a similar audio distribution, each region having between 13.24% and 15.51%, as listed in Table 3.7. The distributions based on sample duration, visualized in Figure 3.14, and for the German token, illustrated in Figure 3.15 also remain similar to the overall distribution of the dataset.

Region	**Samples**	**Length (h)**	**% of Dataset**	**Tokens**
Basel	31903	43.02	13.24%	343423
Bern	32860	44.25	13.62%	353495
Graubünden	37691	42.29	13.01%	442139
Innerschweiz	35187	46.15	14.20%	377044
Ostschweiz	36868	50.41	15.51%	395326
Wallis	35148	49.83	15.33%	377183
Zürich	37037	49.00	15.08%	408029

Table 3.7: SNF-corpus statistics by region concerning number of samples, duration, percentage of total duration, and number of Standard German tokens calculated using spaCy.

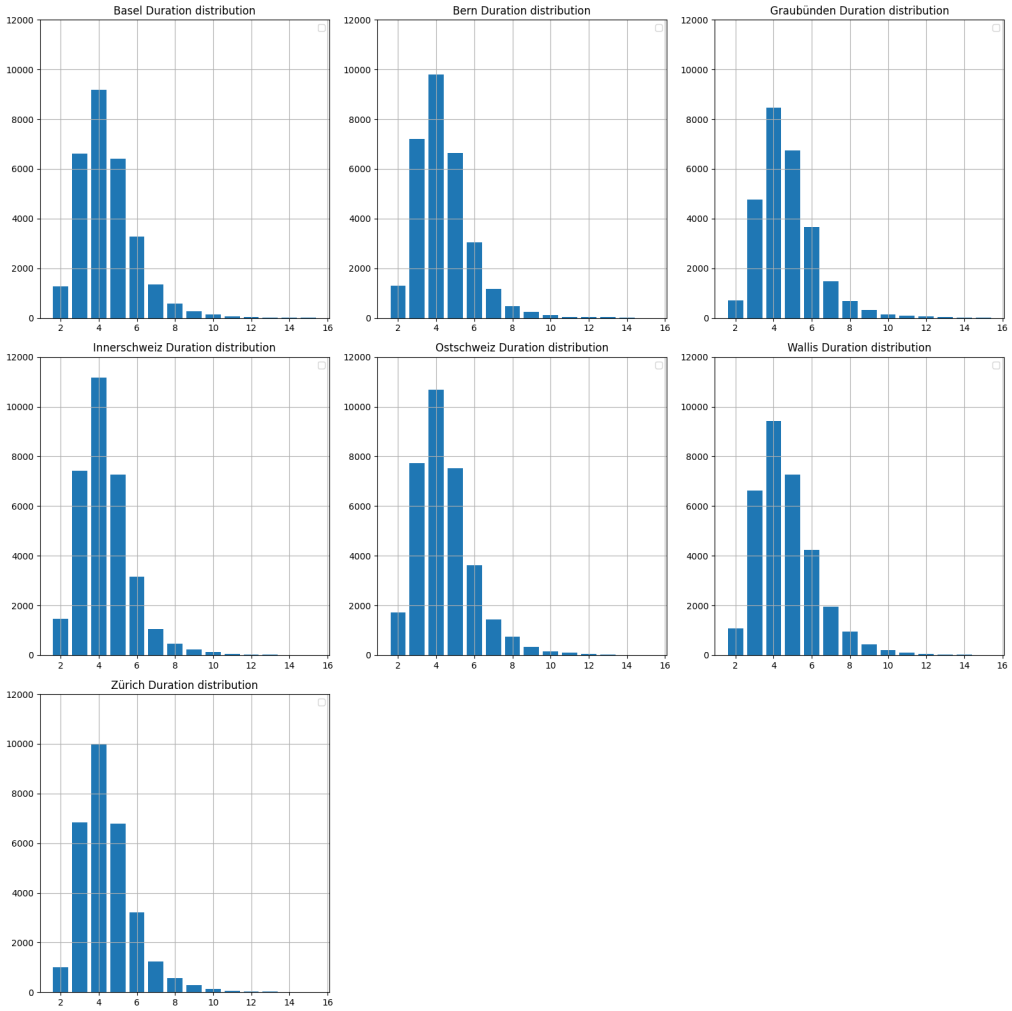

Figure 3.14: Time distribution of the SNF corpus in seconds. The duration of samples has been rounded up towards the next bigger second.

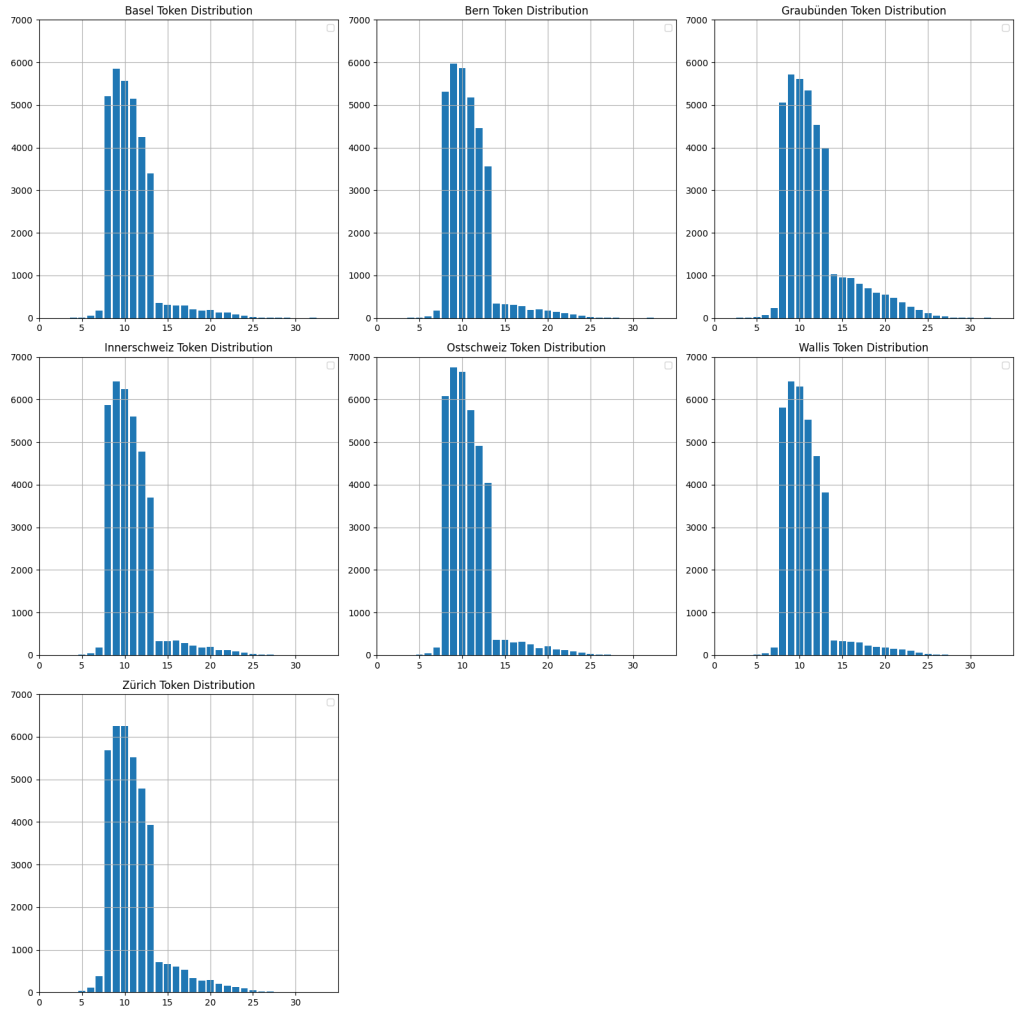

Figure 3.15: SNF-corpus token distribution produced with spaCy.

3.3.2 SRF-corpus

Next, the newly created SRF-corpus is analyzed. In total, the dataset is made up of 1'767'156 unique samples with a total audio length of 4979 hours containing around 55.85 Million tokens. These statistics removed the samples mentioned in the Standard German transcription step in section 3.2.3 and the phoneme generation in section 3.2.4. It was impossible to calculate the accurate number of unique speakers in time due to the intensive clustering process required to assign each speaker found in the diarization step. This is due to, for example, two episodes of the same podcast with the same speakers having different numbering assigned to them in the diarization process, which would need to be realigned in a clustering process using speaker embeddings. Future work may investigate a way to calculate this in an efficient way. Table 3.8 lists the results for each dialect region. What is visible is the unequal distribution of the dataset, with half of the dataset consisting of two regions, Standard German comprising a third and Zurich 23.67%. Wallis has the least data by a large margin, which was expected due to the small population that speaks the dialect, which is further reduced due to infrequent appearances in broad-

casts or podcasts. Future work may consider investing time in finding more Wallis data to better balance the dialect regions.

All regions shared a similar sample duration distribution, visualized in Figure 3.16, as the overall dataset, shown in Figure 3.6 in section 3.2.2. It reinforces that long segments of the same person speaking were in the majority. The German distribution especially had a large concentration of samples with a length of 15 seconds. It is suspected that this was due to the prevalence of Standard German being spoken in news broadcasts or knowledge-based podcasts. This is due to the fragmented nature of Swiss German, in which Standard German serves as a lingua franca between the regions. These segments often have a single person speaking in a monologue for extensive durations at a time, which in turn results in the concentration of samples in the maximum sample length of 15 seconds.

However, when inspecting the distribution of the tokens, visualized in Figure 3.17, this did not translate directly to longer segments of generated tokens. The bimodal distribution is strong in most regions except Wallis. Interestingly, most token counts were in the range of 5 to 20. When investigating the reason for this deviation, it was found that intros to podcasts were often classified as the Wallis dialect region containing between 2 and 7 tokens, often comprising 15-second segments. This explained the similar distribution in sample duration and the large gathering of shorter token samples.

Region	**Samples**	**Length (h)**	**% of Dataset**	**Tokens**
Basel	179269	460.81	9.25%	5359839
Bern	293114	771.38	15.49%	8984805
Deutschland	538687	1685.72	33.86%	17237642
Graubünden	57226	151.33	3.04%	1744917
Innerschweiz	121994	341.22	6.85%	3950169
Ostschweiz	121248	350.60	7.04%	4009260
Wallis	15520	39.46	0.79%	431576
Zürich	440098	1178.58	23.67%	14132021

Table 3.8: SRF-corpus statistics by region concerning number of samples, duration, percentage of total duration, and number of Standard German tokens.

A different interesting find was in the Standard German token distribution. There, the prevalence of long token samples was significantly larger than in other regions. This could also be attributed to the previously mentioned usage of Standard German in more information-dense podcasts, such as news broadcasts and knowledge-based podcasts.

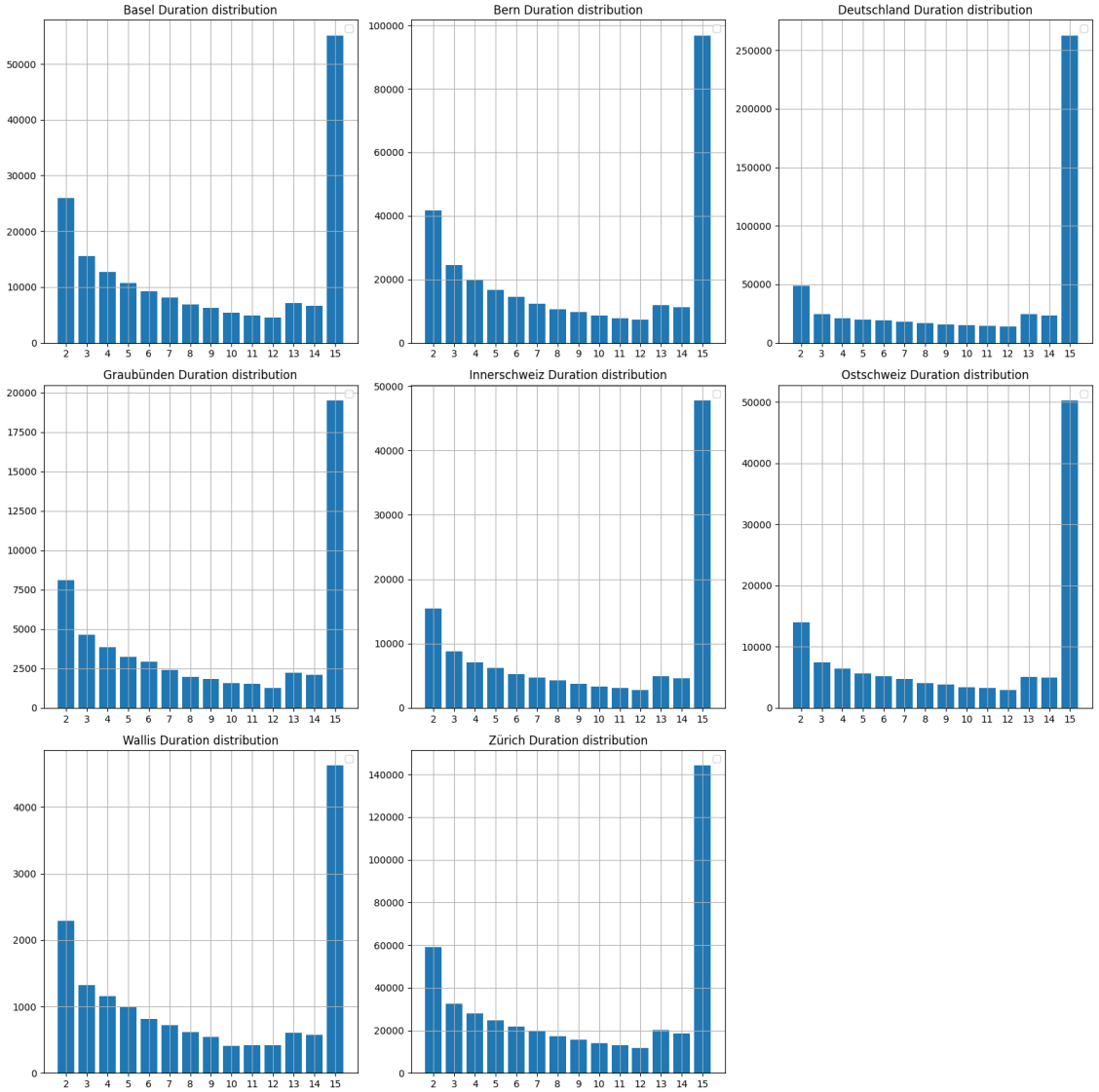

Figure 3.16: Time distribution of the SRF corpus in seconds. The duration of samples has been rounded up towards the next bigger second.

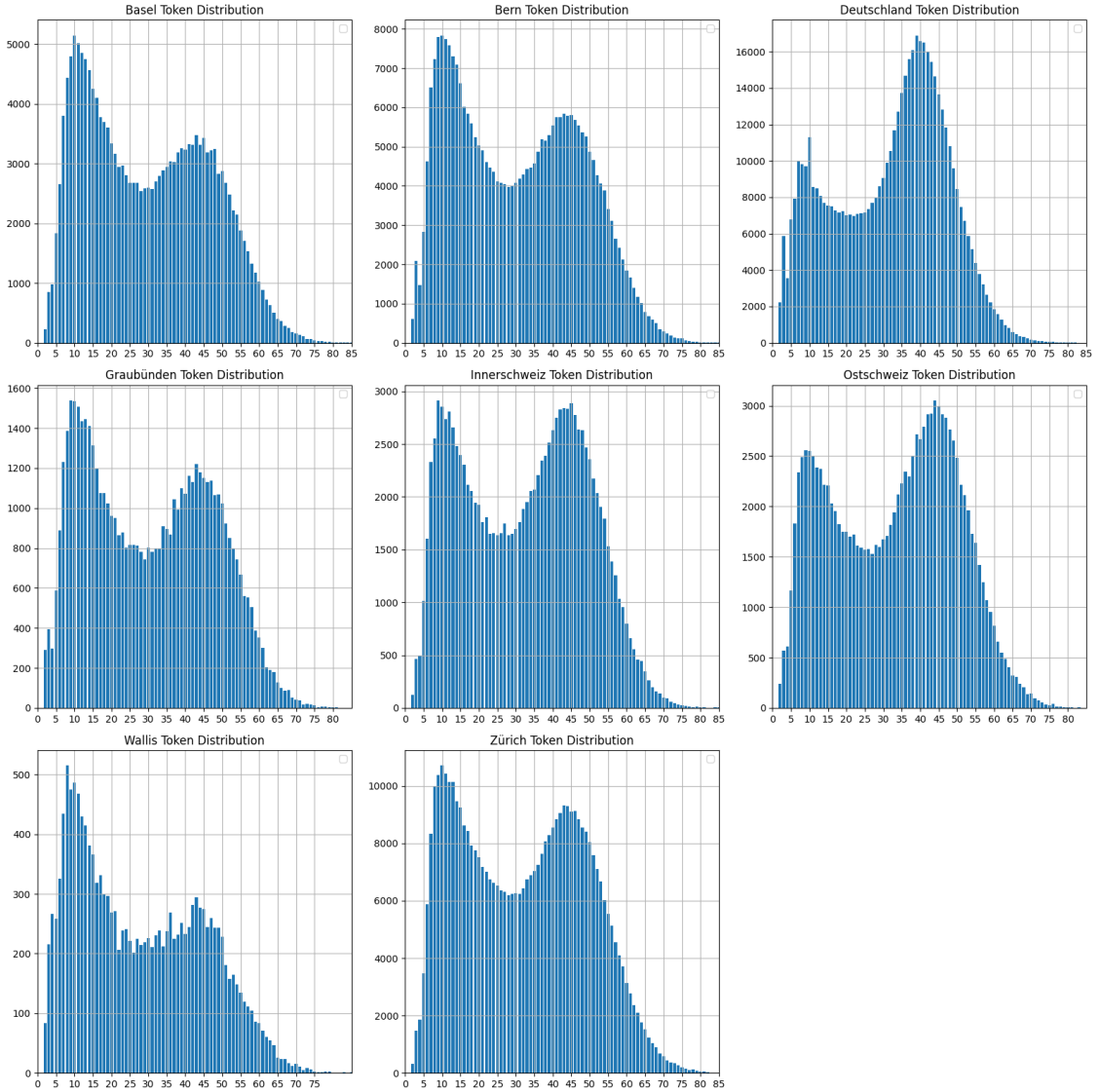

Figure 3.17: SRF-corpus token distribution produced with spaCy.

Podcast	**Source**	**Language**	**Spoken uncut (h)**	**Spoken cut (h)**	**No. of Samples**	**Audio Reduction (%)**
#SRFglobal	SRF	ch	36.9731	35.7933	10716	3.1910
100 Sekunden Wissen	SRF	de	186.7464	125.4665	36051	32.8145
Debriefing 404	SRF	ch	245.1364	187.6421	96335	23.4540
Digital Podcast	SRF	mixed	428.0475	412.5512	136358	3.6202
Dini Mundart	SRF	ch	39.3878	35.7651	14184	9.1975
Gast am Mittag	SRF	ch	33.1414	32.8135	10505	0.9894
Geek-Sofa	SRF	ch	317.2767	275.3522	117235	13.2139
Kopf voran	SRF	mixed	45.0494	41.2057	14911	8.5322
Kultur-Talk	SRF	de	55.8419	53.5841	16206	4.0432
Literaturclub - Zwei mit Buch	SRF	mixed	31.7908	29.9500	9547	5.7904
Medientalk	SRF	de	66.4631	66.4164	20140	0.0703
Pipifax	SRF	ch	9.0800	8.5439	2769	5.9042
Podcast am Pistenrand	SRF	ch	18.2869	16.0808	6910	12.0638
Samstagsrundschau	SRF	ch	404.1439	398.0435	128992	1.5095
Sternstunde Philosophie	SRF	de	159.3853	146.4778	53852	8.0983
Sternstunde Religion	SRF	de	60.8175	55.9756	20904	7.9614
Sykora Gisler	SRF	mixed	152.2172	131.4531	56231	13.6411
Tagesgespräch	SRF	mixed	1661.3344	1624.5331	507534	2.2152
Ufwärmrundi	SRF	ch	60.9786	56.3705	19882	7.5569
Vetters Töne	SRF	ch	25.4194	23.2970	9068	8.3495
Wetterfrage	SRF	ch	67.6753	61.8935	20140	8.5434
Wirtschaftswoche	SRF	de	122.2953	122.1185	37862	0.1446
Wissenschaftsmagazin	SRF	mixed	393.6119	384.8805	127025	2.2183
Zivadiliring	SRF	ch	50.0306	41.1712	19577	17.7080
Zytlupe	SRF	ch	44.7367	39.8512	10195	10.9206
Auf Bewährung - Leben mit Gefängnis	YT	mixed	3.0030	2.5871	767	13.8495
Berner Jugendtreff	YT	ch	127.8043	90.9402	56051	28.8442
Ein Buch Ein Tee	YT	ch	3.7301	3.4089	1346	8.6110
expectations - geplant und ungeplant kinderfrei	YT	ch	16.8355	15.3204	5870	8.9994
fadegrad	YT	ch	49.9476	45.9155	15808	8.0727
Feel Good Podcast	YT	ch	319.5960	271.8658	128979	14.9345
Finanz Fabio	YT	ch	58.4377	52.3411	21510	10.4326
Scho ghört	YT	ch	23.4483	21.7405	7918	7.2833
Sexologie - Wissen macht Lust	YT	mixed	15.4066	14.6454	4742	4.9407
Über den Bücherrand	YT	mixed	14.5319	13.3127	5081	8.3898
Ungerwegs Daheim	YT	ch	38.6651	31.8721	13616	17.5688
Wir müssen reden - Public Eye spricht Klartext	YT	de	17.5236	16.7155	5167	4.6115

Table 3.9: Podcast names used in the SRF-copus with their origin and length of audio in hours

Chapter 4

TTS System Design

This section will provide details about the TTS architecture utilized, training doctrine, changes made to the architecture's training setup, and any interesting challenges faced during training.

4.1 TTS Architecture

It was decided to build upon the existing XTTSv2 model architecture by Casanova et al. [9], as it is already pre-trained on 27k hours of multilingual data and the existence of a library. The v2 model was released by Coqui-ai in 2024, but the company has since gone defunct. The source code is, however, available on GitHub[1] and can be utilized. XTTS is a massive multilingual Zero-Shot (ZS) TTS model supporting 16 languages out-of-the-box. The model can perform cross-language ZS-TTS and supports low/medium resource languages. It achieves a streaming latency of less than 150ms and is built upon the Tortoise model by James Betker [63] but includes several modifications.

The model's original architecture comprises three main parts, which will be explained now. The whole architecture is visualized in Figure 4.1. First, a Vector Quantised-Variational AutoEncoder (VQ-VAE) is applied that ingests a mel-spectrogram and encodes each frame with a codebook of 8192 codes sampled at a 21.53Hz frame rate. After training, only the first 1024 most frequent codes were kept. Next, a GPT-2 encoder with 443M parameters, which is a decoder-only transformer similar to the one in the Tortoise architecture [63], receives text tokens as an input, which were obtained with a 6681-token-custom Byte-Pair Encoding (BPE) tokenizer [64]. The output of the encoder is the prediction of the VQ-VAE audio codes. The GPT-2 encoder is conditioned by a Conditioning Encoder, which processes mel-spectrograms to produce 32 embeddings, each with 1,024 dimensions, for every audio sample. This Conditioning Encoder consists of six 16-head Scaled Dot-Product Attention layers, followed by a Perceiver Resampler [65], which generates a fixed number of embeddings regardless of the audio input's length. Unlike the approach in [63], where only a single 1,024-dimensional embedding was used to con-

[1] https://github.com/coqui-ai/TTS

dition the GPT-2 encoder, the Perceiver Resampler enhances the model's ability to process diverse audio inputs. Lastly, the decoder is built on the HiFi-GAN vocoder by Kong et al. [2] with 26M parameters and takes the latent vectors generated by the GPT-2 encoder as input. Due to their high compression rate, directly reconstructing audio from the VQ-VAE codes can result in pronunciation errors and artefacts. To address this, as in [63], they utilize the GPT-2 encoder's latent space as input to the decoder instead of the VQ-VAE codes. The decoder is further conditioned with speaker embeddings derived from the H/ASP model [66], which are incorporated into each upsampling layer via linear projection. Inspired by [7], they introduced Speaker Consistency Loss (SCL) to enhance speaker similarity. The VQ-VAE and GPT-2 encoder were trained using 22.5 kHz audio signals for faster inference. However, the decoder was trained with input vectors linearly upsampled to the correct length, enabling the production of 24 kHz audio. The resulting model was released with Fine-tuning support, which this thesis will utilize using the STT4SG-350- and SRF-corpora. [9]

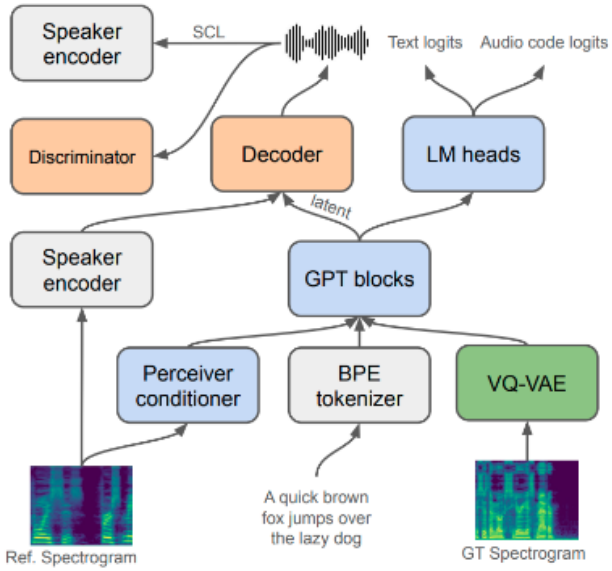

Figure 4.1: Training architecture of XTTS model, Figure taken from [9]

Apart from supporting Fine-tuning for low/medium resource languages, the framework also allows for voice adaptation of the speaker. This is important for evaluating the model with which speakers from the test split can be cloned and used in a Human Evaluation to gauge the accuracy of the generated speech, including the speaker's voice.

4.2 Adapting to Dialects

The model is structured to use tokens as language tags to differentiate the various languages. The dialects were added to the tokenizer using custom tokens, listed in

Table 4.1. No changes had to be made for German as the language was supported out of the box with the "de" tag.

Dialect	**Tag**
Deutschland	de
Bern	ch_be
Basel	ch_bs
Graubünden	ch_gr
Innerschweiz	ch_in
Ostschweiz	ch_os
Wallis	ch_vs
Zürich	ch_zh

Table 4.1: Mapping of Swiss German Dialects, including Standard German, to tags.

4.3 Goal of Training

This training aims to have a zero-shot voice adaptation TTS system for Swiss German dialects, which, instead of ingesting Swiss German text or phonemes, can directly utilize Standard German text. This would reduce the development of future TTS systems, as no costly conversion and verification of Swiss German phonemes would be required. On the other hand, the Swiss-German text approach is extremely difficult to solve due to the large variety of writing styles in the dialects and the population at large. Utilizing a standardized language like German largely removes these barriers and is an important milestone in developing Swiss German voice adaptation TTS.

4.4 Training Pipeline

As outlined in Chapter 3, two distinct datasets were chosen for training, the STT4SG-350 [10] corpus and the SRF-corpus. The datasets were stored in hdf5 files based on their podcast. This was unified into a single hdf5 file for each region, including the metadata generated in the data pipeline, resulting in 8 different files. This included creating a custom dataset formatter that read the speaker ID and the Standard German transcribed text. As outlined in section 3.3.2, different speaker IDs may refer to the same person in the SRF-corpus due to no existing clustering approach to identifying the speakers over the episodes in a podcast. Future work may optimize this step to reduce the possible mixing of training and validation data. Speaking of validation data, the library automatically samples the evaluation data based on both a given percentage and a maximum of absolute samples. For training evaluation, 2% of the complete dataset was chosen with no maximum number of samples.

To observe the training, a custom logger was implemented to stream the results directly to wandb [67]. This also enabled us to stream test sentences directly to the platform and follow the progress by checking the loss functions and listening to

samples generated by the model for all eight regions, including Standard German. These sentences could be generated using reference samples of a speaker, so-called conditioning samples. These were sampled at 22050 Hz as the VQ-VAE, and GPT-2 encoder were trained with this sample rate. The impact of these conditioning samples is very high, so they should be chosen carefully and of high quality. It was decided to generally use up to five unique sentences that were made up of around 25 seconds. The test samples are automatically sampled at 24000 Hz by the decoder.

The previously mentioned 22.05k Hz also provided a small challenge for the training, as the STT4SG-350- and SRF-corpus were sampled to 16kHz and had to be upsampled. However, upsampling the data in the hdf5 files would have taken too long due to issues on the CAI infrastructure, drastically limiting the I/O operation on the cluster. As such, a more pragmatic solution was found by upsampling the samples during training. This was done using the PyTorch [68] Audio library [69], specifically the $torchaudio.functional.resample$ function. Tests exhibited a limited increase in loading time after implementing this change and were, as such, an acceptable solution.

Additionally, the model had to be configured to the sample durations of the SRF-corpus, with samples of up to 15 seconds. The max_wav_length of the GPT-2 model arguments had to be modified to $15s * 22050 = 330750$. The training was then performed using 2 NVIDIA H200 GPUs with a batch size of 36 and grad accumulation steps of 14, resulting in an effective batch size of 36 * 14 * 2 = 1008. The authors [9] recommend at least 252 or a multiple thereof. We mostly applied the same setup as [9] for the training setup, applying an AdamW optimizer with betas 0.9 and 0.96 and weight decay 0.01. The learning rate was changed to 6e-5 from the original 5e-5 due to internal tests and listening to the generated audio files. Weight decay was applied only to the weights, and the learning rate was decayed using MultiStepLR with a gamma of 0.5 using milestones 5000, 150000, and 300000. Table 4.2 lists all essential parameters and their values. In total, we aimed for 9 to 11 epochs for one training of a model, with each epoch consisting of around 27000 steps and the complete training generally taking around 6-7 days.

Multiple smaller trainings were performed to fine-tune various aspects of the model and the output. The most significant change in the training regiment was excluding all samples with less than seven tokens. This was done after noticing a considerable speech impediment of the model that resulted in mumbled speech or outright nonsense. It was assumed that smaller samples contained chaotic speech and prevented the model from forming actual sentences. The sample size was thus reduced by 71995 with the change. In this stage, the impact of the bimodal distribution of the SRF-corpus on test samples became apparent. Shorter segments, like a single sentence, were becoming more difficult for the model to generate the longer the training went on. On the other hand, longer or multiple sentences together brought better quality to the sentences. The data analysis we did on the corpus confirmed this apparent issue, in which the model was trained to generally generate samples of 15 seconds, which also heightened the ueueue-frequency in samples if sentences were

Parameter	Parameter Type	Value
max_wav_length	GPTArgs	330750
batch_size	GPTTrainerConfig	36
eval_batch_size	GPTTrainerConfig	36
eval_split_size	GPTTrainerConfig	0.02
log_step	GPTTrainerConfig	1000
save_step	GPTTrainerConfig	2000
save_n_checkpoints	GPTTrainerConfig	1
run_eval_steps	GPTTrainerConfig	2000
optimizer	GPTTrainerConfig	AdamW
optimizer_wd_only_on_weights	GPTTrainerConfig	False
optimizer_params_betas	GPTTrainerConfig	[0.9, 0.96]
optimizer_params_eps	GPTTrainerConfig	1e-8
optimizer_params_weight_decay	GPTTrainerConfig	1e-2
lr	GPTTrainerConfig	6e-05
lr_scheduler	GPTTrainerConfig	MultiStepLR
lr_scheduler_milestones	GPTTrainerConfig	[50k * 18, 150k * 18, 300k * 18]
lr_scheduler_gamma	GPTTrainerConfig	0.5
lr_scheduler_last_epoch	GPTTrainerConfig	-1
shuffle	GPTTrainerConfig	True
grad_accum_steps	Trainer	14

Table 4.2: Important XTTSv2 model args and their set value.

short. The evaluation for these issues will be explained in more detail in Chapter 5.

The final setup thus consists of 3 models, which will now be explained. First, the mixed dataset trained "SRF+STT4SG-Mixed" model, which trained for 9 epochs or 238k steps on the complete corpora. Next, the "Mixed+STT4SG-FT" model took the last checkpoint of the SRF+STT4SG-Mixed as a starting point and Fine-tuned only on the STT4SG-350 [10]corpus. This was an attempt to remedy the model's issue with short sentences, as [10] did not have the bimodal distribution and was centred around sentences with a length of 7 to 14 tokens. The training was run for 26 epochs. Lastly, the "STT4SG-Only" baseline model was trained using only [10] and ran for 25 epochs.

Model Name	Description	Epochs
SRF+STT4SG-Mixed	Fine-tuned on both SNF- and SRF-corpus	9
Mixed + STT-SG FT	Used last SRF+STT4SG-Mixed checkpoint and fine-tuned on SNF-corpus	9 + 26
STT4SG-Only	Baseline model	25

Table 4.3: Trained models for experiments with description.

Chapter 5

Evaluation

This section will give insight into the evaluation approach of the trained models. The evaluation is diverse as the TTS models are targeted to solve multiple problems. For this, numerous evaluation types using different lengths of sentences and differing numbers of voice adaptations were used to understand the actual performance. Next, a human evaluation will be performed to evaluate the voice adaptation, the audio quality, and the interpretability of the generated audio samples. More details are in Section 5.4. This thesis did not evaluate Standard German due to the focus on Swiss German dialects, but specific findings will be presented in the Dialect Identification Section 5.3.4. Mentions of token counts in this Chapter are always calculated using spaCy [43]. Additionally, any ±X.yz values given in the tables refer to the observed standard deviation.

5.1 Voice Adaptation Speaker Selection

A subset of the STT4SG-350 [10] test-split was chosen for the voice adaptation, as the models have not seen the speakers in the training before. The same sentences had to be spoken in all six dialect regions for the evaluations to be comparable. However, the speakers in the test set did not all produce audio for all sentences; only a few had overlapping sentences for multiple samples. The filtering thus prioritised these combinations, and speakers were chosen based on the filtered sentences of other speakers. This filtering process results in each dialect having the same 15 spoken sentences, listed in Table 5.1, with the split having 43 unique speakers, presented below. The 15 sentences per dialect are similar in size to the 10 sentences per dialect that the authors in SwissDial [11] let annotators annotate. Figure 5.1 visualises the number of unique speakers per dialect. Generally, most dialects have six or more unique speakers, except the Wallis region. This was due to the low number of speakers in the complete test split and the previously mentioned prioritisation of overlapping sentences by speakers, which Wallis happened to be most represented by.

Figure 5.2 illustrates the gender distribution for the number of samples spoken by the given gender per dialect region, which is unbalanced. Three areas are particularly skewed towards a specific gender. Wallis can be easily explained, as the test split only contains female speakers. Bern and Zurich had 2/3 of their samples belonging

to males, with the sentence selection process filtering those in unintentionally. Next, in Figure 5.3, the number of sentences each speaker has in the test split is visualized. A significant portion of the speakers only produce one sentence, which may make evaluation harder due to differing sample quality by each speaker. However, this was deemed acceptable due to the necessity of comparing the same sentences with each other. Lastly, Figure 5.4 provides insight into the age distribution of the various speakers, with people in their twenties and forties making up the majority. Table C.1 also gives an overview of the chosen speakers.

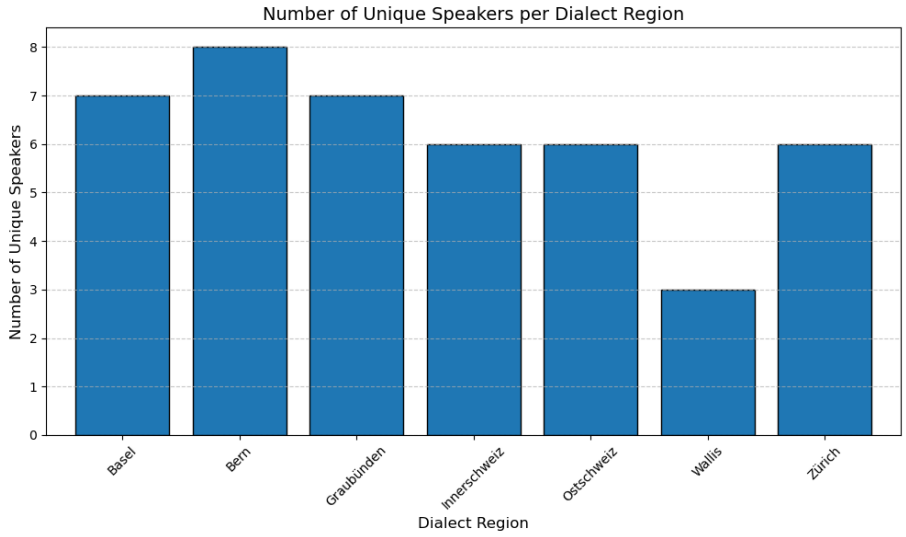

Figure 5.1: Number of unique speakers per dialect region for the evaluation test split.

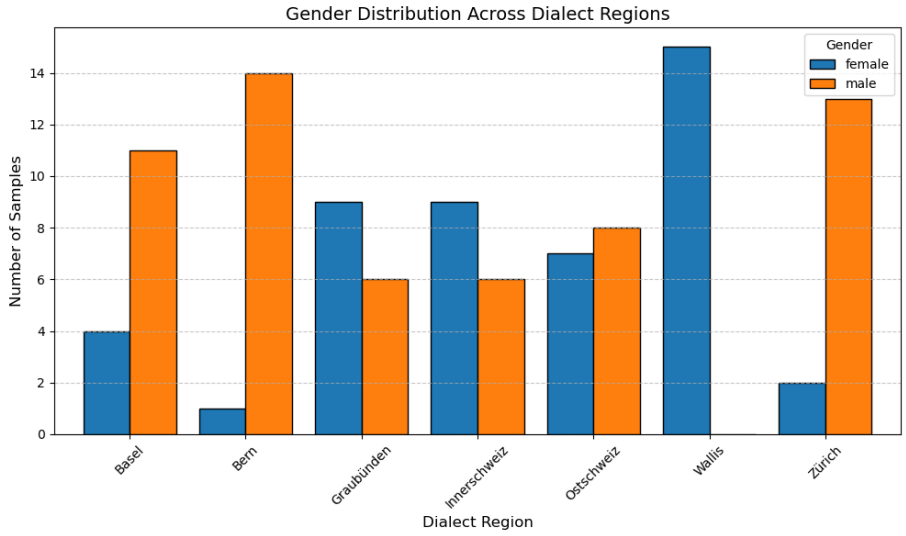

Figure 5.2: Gender distribution of samples per dialect region for the evaluation test split.

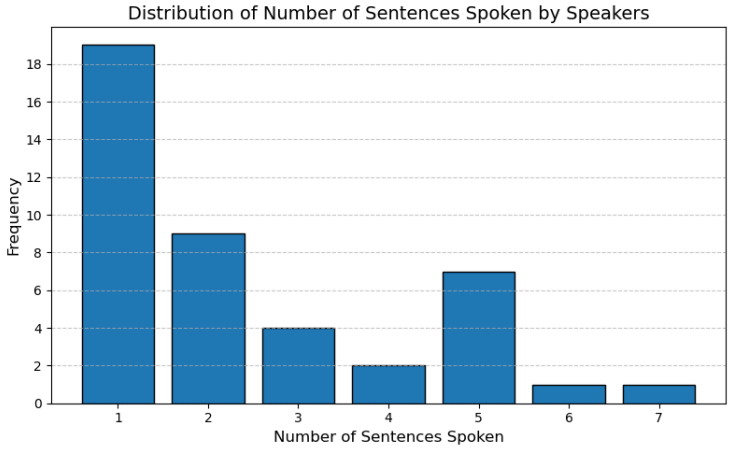

Figure 5.3: Distribution of sentences spoken by speakers for the evaluation test split.

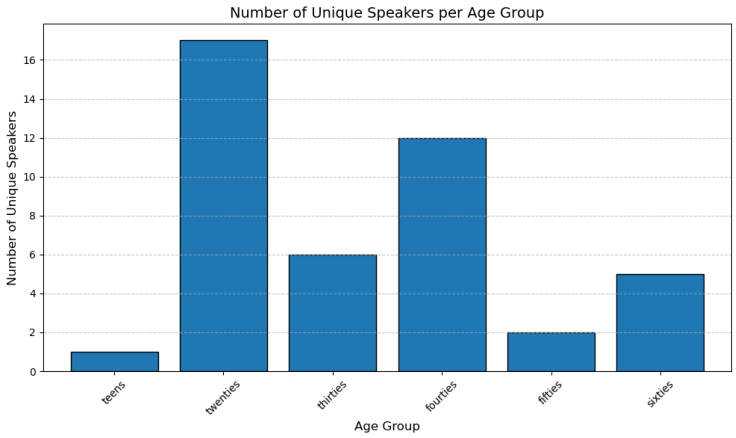

Figure 5.4: Number of unique speakers per age group for the evaluation test split.

Quality was a considerable concern as the audio of each speaker was recorded in their own home with their audio setup. Manual inspection of audio samples yielded a difference in quality, which was significant depending on the speaker. Some had incomprehensible speech with echos, and others had microphones that made them sound robotic. For the evaluation to be as equal as possible, some way of discerning the quality of each sample needed be found for each speaker's conditioning sample selection. Dataspeech [70][71] is a library used by HuggingFace in the training of their Parler-TTS [72] model family to discern which audio samples can be used for training and which should be filtered out. The library provides insight into pitch, speaking rate, Signal to Noise Ratio (SNR), reverberation, speech monotony, Scale-Invariant Signal-to-Distortion Ratio (SI-SDR) [73], and Perceptual Evaluation of Speech Quality (PESQ). All samples of the 43 speakers were run through the library, out of which the condition samples were then chosen randomly based primarily on the PESQ, but also the SI-SDR, SNR, and reverberation. Pitch and speaking rates were largely ignored in this selection process as they were deemed speaker-dependent, not audio-dependent, and the model should be able to handle different types of voice

adaptions. Additionally, it was verified that the conditioning samples did not match any of the samples selected in one of the evaluations in Section 5.2. Each speaker thus had five condition samples assigned to them, which were used in all subsequent experiments and evaluations.

Sentence	**Token Count**
Da kann man nicht in der Jugendherberge absteigen.	9
Dabei braucht einem der Winter keine Angst zu machen.	10
Die Hinweise seien von Anwohnern und über soziale Medien eingegangen.	11
Diese müssen sie an grössere abgeben.	7
Er hinterlässt eine Frau und einen Sohn.	8
Geplant sind laut Mitteilung Werkstätten mit Arbeitsplätzen für qualifizierte Handwerker.	11
Im Herbst sind im Schutz der Dunkelheit wieder vermehrt Einbrecher unterwegs.	12
Mehr Aufmerksamkeit auf den Strassenverkehr würde auch ein Head-up-Display liefern.	11
Obwohl bereits dannzumal das "Nichterreichen" der Vorgaben klar erwartet werden musste.	14
Sie zeigen die beiden Entführten vor einem grossen schwarzen Banner sitzend.	12
Trotzdem möchte die Finanzlage anhand der Kennzahlen gesamthaft würdigen.	10
Und auch die hohen Lebenshaltungskosten trüben das Bild von der Schweiz.	12
Vielen Dank für das Votum des Finanzkommissionspräsidenten, François Chapuis.	11
Wir werden dein unglaubliches Talent und deine endlose Inspiration nie vergessen.	12
Zumindest in den ersten drei Episoden überwiegt letzterer Einfluss.	10

Table 5.1: Selected sentences for Voice Adaptation, taken from the STT4SG-350 [10] dataset.

5.2 Evaluation Types

As outlined in Section 4.4, internal tests on models trained with the SRF-corpus exhibited a degradation in performance on short sentences. Four different evaluation types were thus defined, of which three were evaluated in both automated and human evaluation setups. GPT-Random was only evaluated in automated evaluation to measure the mix of short and long sentence lengths on the three models.

5.2.1 SNF-Short

The first evaluation type is termed SNF-Short and is directly based on the test sentences selected for the voice adaptation. Table 5.1 lists the 15 sentences for each of the seven dialect regions for the 43 speakers. This set of sentences was used to test the performance of the models on short sentences and voice adaptation, as there are reference samples in the test split. This means that each sentence had a corresponding speaker in a given dialect and could thus be used to make direct comparisons between references and hypotheses. In total, there are 105 samples split across the 43 speakers.

5.2.2 SNF-Long

SNF-Long is an evaluation type based on the same sentences used in the SNF-Short evaluation type but combined two sentences, forming longer text segments. This was done after realizing the previously discussed tendency of the model to prefer longer text segments due to the prevalence of such segments in the SRF-corpus. The speaker for each of these longer sentences was kept the same as in SNF-Short by using the speaker, which had a corresponding audio file for the first sentence in the text segments. The list of sentences is given in Table 5.2. As with SNF-Short, there are 105 samples in total.

5.2.3 GPT-Random

GPT-Random is based on sentences generated by ChatGPT [74]. The model was asked to create 900 random German sentences of varying lengths to have more and longer sentences available to evaluate the models automatically. GPT-Random was primarily used in the internal tests, as the sentences varied greatly in length and gave important insight into the model's performances. In these internal tests, up to 100 were randomly chosen and used with a single speaker. The speaker, originating from the Innerschweiz region with ID acf67674-c912-42c0-ba3c-f85e2db965ac in the STT4SG-350 [10] dataset, was chosen due to the excellent audio quality and clear speech that was observed in the dataspeech [70] output and generally good performance on sentences in the SNF-Short evaluation. To evaluate the three models described in Section 4.4, 100 sentences were randomly sampled once and then utilized for all three models and dialect regions, with their token counts illustrated in Figure 5.5. Both the complete 900-sentence set and the smaller subset of 100 sentences are available on GitHub[1]. A single run of GPT-Random generated 100 * 7 regions = 700 sentences.

[1]https://github.com/stucksam/swiss-zero-shot-va-tts/

Sentences	Token Count
Da kann man nicht in der Jugendherberge absteigen. Zumindest in den ersten drei Episoden überwiegt letzterer Einfluss.	19
Dabei braucht einem der Winter keine Angst zu machen. Da kann man nicht in der Jugendherberge absteigen.	19
Die Hinweise seien von Anwohnern und über soziale Medien eingegangen. Dabei braucht einem der Winter keine Angst zu machen.	21
Diese müssen sie an grössere abgeben. Die Hinweise seien von Anwohnern und über soziale Medien eingegangen.	18
Er hinterlässt eine Frau und einen Sohn. Diese müssen sie an grössere abgeben.	15
Geplant sind laut Mitteilung Werkstätten mit Arbeitsplätzen für qualifizierte Handwerker. Er hinterlässt eine Frau und einen Sohn.	19
Im Herbst sind im Schutz der Dunkelheit wieder vermehrt Einbrecher unterwegs. Geplant sind laut Mitteilung Werkstätten mit Arbeitsplätzen für qualifizierte Handwerker.	23
Mehr Aufmerksamkeit auf den Strassenverkehr würde auch ein Head-up-Display liefern. Im Herbst sind im Schutz der Dunkelheit wieder vermehrt Einbrecher unterwegs.	23
Obwohl bereits dannzumal das "Nichterreichen" der Vorgaben klar erwartet werden musste. Und auch die hohen Lebenshaltungskosten trüben das Bild von der Schweiz.	26
Sie zeigen die beiden Entführten vor einem grossen schwarzen Banner sitzend. Obwohl bereits dannzumal das "Nichterreichen" der Vorgaben klar erwartet werden musste.	26
Trotzdem möchte die Finanzlage anhand der Kennzahlen gesamthaft würdigen. Sie zeigen die beiden Entführten vor einem grossen schwarzen Banner sitzend.	22

Table 5.2: Sentences and their Token Counts

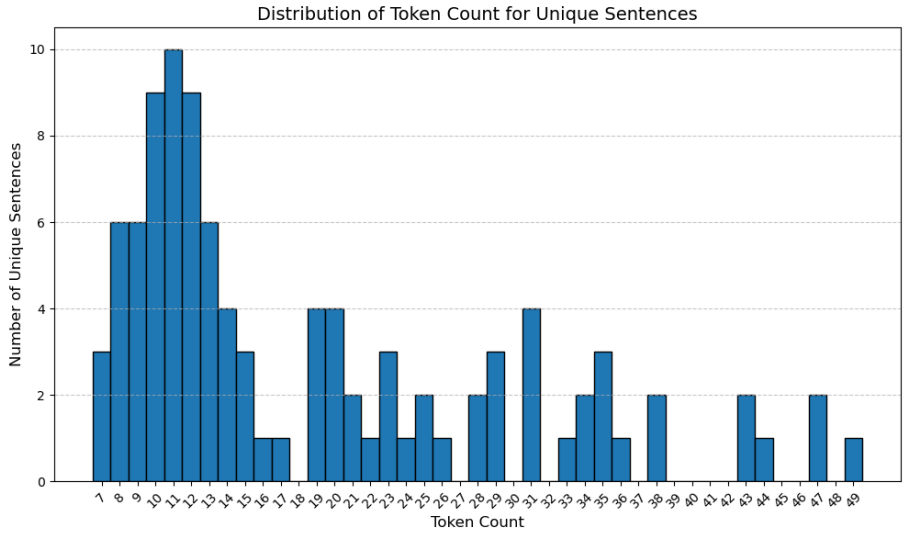

Figure 5.5: Distribution of token counts in the GPT-Random evaluation type of the 100 individual sentences.

5.2.4 GPT-Long

GPT-Long is based on the same 900 sentences generated in the GPT-Random step. However, two significant changes were made to the evaluation. First, the sentences had to have a token count of more than 20. Secondly, only thirty sentences were chosen, which were run for every speaker selected in the voice adaptation. The execution of GPT-Long resulted in 30 * 43 speakers = 1290 individual generated samples. This approach enabled us to evaluate long text segments and the DID, as the dialect classifier tended to require 30 seconds or more of audio to classify the speakers accurately. Figure 5.6 shows the token count distribution of GPT-Long. As with GPT-Random, the sentences are available on GitHub[2].

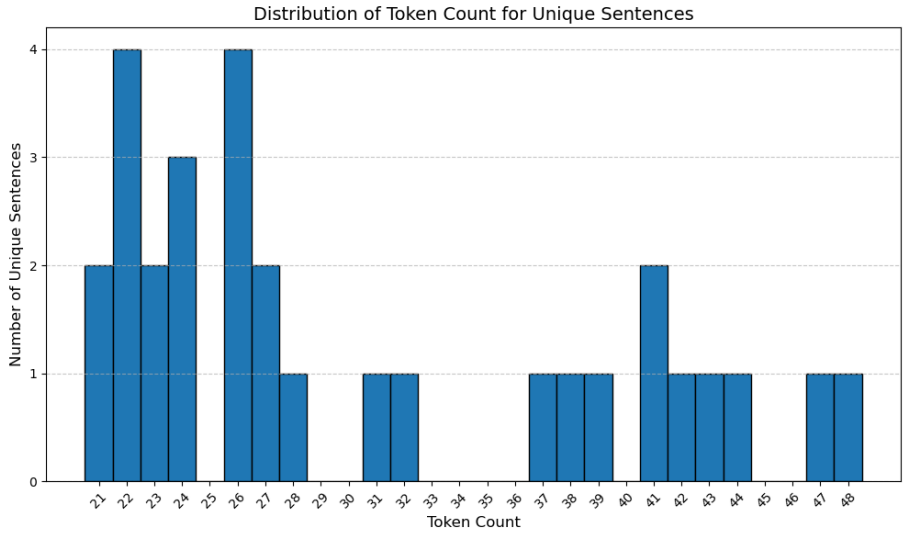

Figure 5.6: Distribution of token counts in the GPT-Long evaluation type of the 30 individual sentences.

5.3 Automated Evaluation

The automated evaluation pipeline uses three enrichment processes: Standard German transcription, phoneme generation, and dialect classification. All processes, including the investigated metrics, have already been described in Chapter 3, with the only difference being the in Section 3.2.5 described usage of a classification model that does not contain the German class to reduce classification errors.

5.3.1 Sentence Verification

This section will evaluate the performance of the TTS models in replicating the given reference sentence. In these text-based evaluations, as with the data pipeline, only the lower text metrics will generally be discussed. For metrics, CER is particularly interesting, as the authors of the XTTS architecture [9] based their evaluation on them. Each Evaluation type will be discussed separately first, with an overall

[2]https://github.com/stucksam/swiss-zero-shot-va-tts/

average score provided at the end. The dialects will also be investigated from a general on-model basis. The in Section 3.2.3 defined setup for Standard German transcription was replicated for evaluation, including the suppression of repeated n-grams.

Due to the small sample size of SNF-Short and SNF-Long, for each metric in each model comparison, a homogeneity test using Levene [75] and a normality test using Shapiro-Wilk [76] is performed. If both variance and homogeneity do not fail ($p \leq 0.05$), a statistical significance test is run with One-Way ANOVA [77] and an additional pairwise t-test is executed using Tukey's HSD [78] if any of the values have p value of ≤ 0.05. In cases where tests for homogeneity or normality fail, the Kruskal-Wallis test [79] is applied, and followed Post-Hoc Dunn's Test for pairwise comparisons where appropriate ($p \leq 0.05$). If no specific pairwise comparisons are mentioned in the discussion, it should be assumed that the differences between the models were not statistically significant.

SNF-Short

SNF-Short was internally deemed the most difficult evaluation type due to the general issues observed during fine-tuning. Unsurprisingly, the average scores of each model, listed in Table 5.3, confirmed this, even the baseline. The Mixed+STT4SG-FT model achieved the best performance in terms of WER (0.523) and CER (0.240), demonstrating higher accuracy in both word- and character-level recognition compared to the SRF+STT4SG-Mixed model, which had the highest WER (0.676) and CER (0.380). Interestingly, BLEU scores presented a different trend: the SRF+STT4SG-Mixed model achieved the highest BLEU score (0.288), indicating better alignment with reference sentences, while the Baseline STT4SG-Only model scored the lowest (0.212). All models achieved similar BERTScores, with minor variations ranging between 0.830 and 0.844, suggesting similar semantic similarity across models. Important to keep in mind is that whisper [15] requires 30-second segments of audio as input, which are automatically padded with silence if the sample is smaller. This can have an impact on scores due to the erroneous transcription of whisper.

The Tukey HSD test results for the WER metric showed that there was a significant difference between the Mixed+STT4SG-FT model and the SRF+STT4SG-Mixed model, with a mean difference of 0.1523 ($p = 0.0283$). The CER metric analysis showed that the Mixed+STT4SG-FT model significantly outperformed the SRF+STT4SG-Mixed model, with a mean difference of 0.1403 ($p = 0.0032$). Additionally, the SRF+STT4SG-Mixed model significantly improved over the STT4SG-Only model (mean difference = -0.1014, $p = 0.0471$). However, no significant difference was found between the Mixed+STT4SG-FT and STT4SG-Only models (mean difference = 0.0389, $p = 0.6332$).

Table 5.4 lists the performance of the models on a dialect region level. Generally,

Model	WER	CER	BERTScore	BLEU
SRF+STT4SG-Mixed	0.676±0.60	0.380±0.47	0.834±0.1	**0.288±0.32**
Mixed+STT4SG-FT	**0.523±0.31**	**0.240±0.17**	**0.844±0.1**	0.249±0.3
STT4SG-Only	0.575±0.31	0.278±0.20	0.830±0.11	0.212±0.27

Table 5.3: Average SNF short transcription results on model basis.

except for the Basel region, the lowest WER and CER are produced by the same model for a given dialect region. The Mixed+STT4SG-FT performs best in these metrics in four regions. Surprisingly, its performance is much stronger compared to the SRF+STT4SG-Mixed model in the Wallis region. Contrary to this is the already discussed performance of the SRF+STT4SG-Mixed model on the BLEU metric in four regions, surprisingly including Wallis in which it had the worst WER and CER. The Baseline model STT4SG-Only generally was the second best in each region, except for Basel and Ostschweiz. The SRF+STT4SG-Mixed and Mixed+STT4SG-FT scores did not directly reflect the dialect distribution of the SRF-corpus, which was an interesting find as both heavily utilized the SRF-corpus in their training. An example is the Zurich area, which was second to last in performance overall but contained 23.67% of all samples.

All comparisons yielded p-values greater than 0.05, indicating that the observed differences between models on a dialect region basis were not statistically significant. Further studies with larger sample sizes might be required to detect subtler effects. Future work should consider increasing the SNF short evaluation type size and rerun the experiment.

Dialect	Model	WER	CER	BERTScore	BLEU
Basel	SRF+STT4SG-Mixed	**0.476±0.32**	0.280±0.28	**0.887±0.07**	**0.376±0.35**
	Mixed+STT4SG-FT	0.604±0.31	**0.261±0.18**	0.827±0.11	0.184±0.26
	STT4SG-Only	0.688±0.31	0.304±0.17	0.812±0.11	0.152±0.28
Bern	SRF+STT4SG-Mixed	0.789±0.55	0.473±0.43	0.793±0.12	0.267±0.30
	Mixed+STT4SG-FT	**0.475±0.31**	**0.224±0.17**	**0.851±0.11**	**0.313±0.35**
	STT4SG-Only	0.628±0.31	0.361±0.26	0.789±0.13	0.190±0.28
Graubünden	SRF+STT4SG-Mixed	0.503±0.42	0.271±0.29	0.852±0.11	**0.413±0.38**
	Mixed+STT4SG-FT	**0.460±0.22**	**0.216±0.15**	0.856±0.09	0.273±0.23
	STT4SG-Only	0.463±0.31	0.234±0.25	**0.861±0.12**	0.334±0.29
Innerschweiz	SRF+STT4SG-Mixed	0.558±0.37	0.273±0.21	0.862±0.09	0.331±0.38
	Mixed+STT4SG-FT	**0.364±0.31**	0.166±0.18	**0.905±0.08**	**0.401±0.38**
	STT4SG-Only	0.426±0.32	**0.174±0.16**	0.882±0.10	0.277±0.34
Ostschweiz	SRF+STT4SG-Mixed	0.605±0.42	0.305±0.26	0.830±0.09	**0.270±0.25**
	Mixed+STT4SG-FT	0.521±0.34	0.228±0.20	0.857±0.12	0.267±0.29
	STT4SG-Only	**0.485±0.26**	**0.221±0.16**	**0.858±0.11**	0.262±0.27
Wallis	SRF+STT4SG-Mixed	0.936±1.06	0.594±0.94	**0.803±0.11**	**0.204±0.27**
	Mixed+STT4SG-FT	**0.629±0.23**	**0.300±0.14**	0.796±0.08	0.133±0.23
	STT4SG-Only	0.706±0.32	0.353±0.18	0.789±0.11	0.129±0.20
Zurich	SRF+STT4SG-Mixed	0.862±0.67	0.462±0.40	0.814±0.11	0.157±0.31
	Mixed+STT4SG-FT	**0.610±0.36**	**0.281±0.17**	0.818±0.11	**0.174±0.29**
	STT4SG-Only	0.629±0.28	0.301±0.15	**0.819±0.10**	0.140±0.22

Table 5.4: Average SNF short transcription result on dialect basis.

SNF-Long

SNF-Long evaluation was expected to yield better scores than SNF-Short, confirmed in Table 5.5. The two evaluation types are directly comparable because the same sentences are used for the voice adaptation. SNF-Long consistently performed better across all metrics; specifically, SRF+STT4SG-Mixed gained the most. For WER, the SRF+STT4SG-Mixed model improved by 0.263, the Mixed+STT4SG-FT model improved by 0.08 and the Baseline STT4SG-Only model improved by 0.084. The CER improved by 0.174, 0.045, and 0.057 for the three models, respectively. BERTScore saw a moderate increase between 0.012 and 0.036, and BLEU improved by 0.161, 0.11, and 0.1, respectively.

Model	WER	CER	BERTScore	BLEU
SRF+STT4SG-Mixed	**0.413±0.26**	0.206±0.18	**0.870±0.08**	**0.449±0.26**
Mixed+STT4SG-FT	0.441±0.25	**0.195±0.13**	0.858±0.09	0.359±0.28
STT4SG-Only	0.491±0.25	0.221±0.15	0.842±0.08	0.312±0.23

Table 5.5: Average SNF long transcription result on model basis.

Statistical analysis found differences in the BERTScore and BLEU. The BERTscore Tukey result yielded that the comparison between SRF+STT4SG-Mixed and STT4SG-Only was statistically significant (p = 0.0353), with a mean difference of -0.0285. In the BLEU Tukey result, both the comparison between Mixed+STT4SG-FT and SRF+STT4SG-Mixed (p = 0.0301) and the comparison between SRF+STT4SG-Mixed and STT4SG-Only (p = 0.0004) were significant, with mean differences of 0.0903 and -0.1366, respectively, indicating meaningful differences between these pairs.

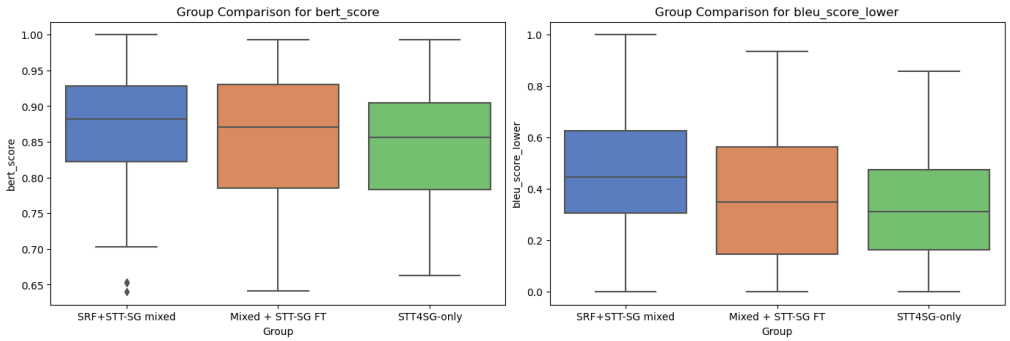

Figure 5.7: Boxplot of BERTScore and BLEU for the overall result of the SNF long evaluation.

Breaking the values down on a dialect basis, shown in Table 5.6, the in SNF-Short seen fragmentation of metric performances largely disappeared. Apart from Ostschweiz, a single model always had the best scores in all metrics for a given region in SNF-Long. The SRF+STT4SG-Mixed performed best in four out of the seven dialects, while Mixed+STT4SG-FT performed best in the Zurich and Innerschweiz

regions. Wallis remained the most challenging region for all models, which was surprising to see in the Baseline STT4SG-Only model. As the STT4SG-350 [10] corpus used in training was approximately balanced, Wallis should also score similarly to the dialect regions. Currently, it is assumed that this is due to difficulties with the pronunciation of dialects.

Statistical analysis showed four interesting differences. First, Tukey's results showed a significant difference between BLEU of Mixed+STT4SG-FT and SRF+STT4SG-Mixed ($p = 0.0391$) models for the Graubünden dialect region, with a mean difference of 0.2239. However, no significant differences were found between Mixed+STT4SG-FT and STT4SG-Only (p = 0.7138) or between SRF+STT4SG-Mixed and STT4SG-Only ($p = 0.199$). More interestingly were the findings of the Basel dialect region, in which WER, BERTScore, and BLEU exhibited interesting patterns. For the WER metric, only the comparison between SRF+STT4SG-Mixed and STT4SG-Only was significant ($p = 0.0223$), with a mean difference of 0.2279, while the other comparisons were insignificant. For the BERTScore metric, the comparison between SRF+STT4SG-Mixed and STT4SG-Only was significant ($p = 0.0173$), with a mean difference of -0.0847. The other comparisons between Mixed+STT4SG-FT and the other groups were insignificant. Lastly, in the BLEU metric, Kruskal-Wallis test results were significant ($p = 0.0027$). The only significant pair-wise difference was between SRF+STT4SG-Mixed and STT4SG-Only ($p = 0.001881$), while the other pairwise comparisons were insignificant. Figure 5.8 gives box plots for the three Basel metrics.

Dialect	Model	WER	CER	BERTScore	BLEU
Basel	SRF+STT4SG-Mixed	**0.430±0.26**	**0.190±0.14**	**0.876±0.07**	**0.453±0.24**
	Mixed+STT4SG-FT	0.503±0.23	0.239±0.14	0.831±0.10	0.279±0.22
	STT4SG-Only	0.658±0.18	0.281±0.11	0.791±0.07	0.145±0.16
Bern	SRF+STT4SG-Mixed	0.452±0.27	0.231±0.17	0.844±0.09	0.384±0.27
	Mixed+STT4SG-FT	0.421±0.26	0.193±0.14	0.859±0.10	0.392±0.28
	STT4SG-Only	**0.342±0.22**	**0.132±0.09**	**0.893±0.08**	**0.465±0.26**
Graubünden	SRF+STT4SG-Mixed	**0.251±0.14**	**0.112±0.06**	**0.914±0.05**	**0.620±0.17**
	Mixed+STT4SG-FT	0.385±0.20	0.138±0.10	0.882±0.08	0.396±0.27
	STT4SG-Only	0.463±0.31	0.234±0.25	0.861±0.12	0.334±0.29
Innerschweiz	SRF+STT4SG-Mixed	0.364±0.13	0.160±0.06	0.889±0.04	0.481±0.19
	Mixed+STT4SG-FT	**0.301±0.21**	**0.119±0.11**	**0.901±0.07**	**0.501±0.24**
	STT4SG-Only	0.336±0.16	0.135±0.09	0.880±0.06	0.463±0.19
Ostschweiz	SRF+STT4SG-Mixed	**0.349±0.23**	0.168±0.13	**0.890±0.07**	**0.494±0.29**
	Mixed+STT4SG-FT	0.379±0.25	0.156±0.12	0.885±0.08	0.442±0.33
	STT4SG-Only	0.388±0.15	**0.154±0.07**	0.874±0.06	0.361±0.21
Wallis	SRF+STT4SG-Mixed	**0.537±0.34**	**0.304±0.25**	**0.833±0.10**	**0.346±0.29**
	Mixed+STT4SG-FT	0.694±0.19	0.342±0.10	0.778±0.07	0.131±0.15
	STT4SG-Only	0.713±0.24	0.359±0.16	0.776±0.07	0.150±0.14
Zurich	SRF+STT4SG-Mixed	0.507±0.32	0.282±0.26	0.846±0.07	0.364±0.25
	Mixed+STT4SG-FT	**0.408±0.23**	**0.178±0.12**	**0.868±0.08**	**0.370±0.30**
	STT4SG-Only	0.486±0.24	0.234±0.18	0.841±0.08	0.321±0.19

Table 5.6: Average SNF long transcription result on dialect basis.

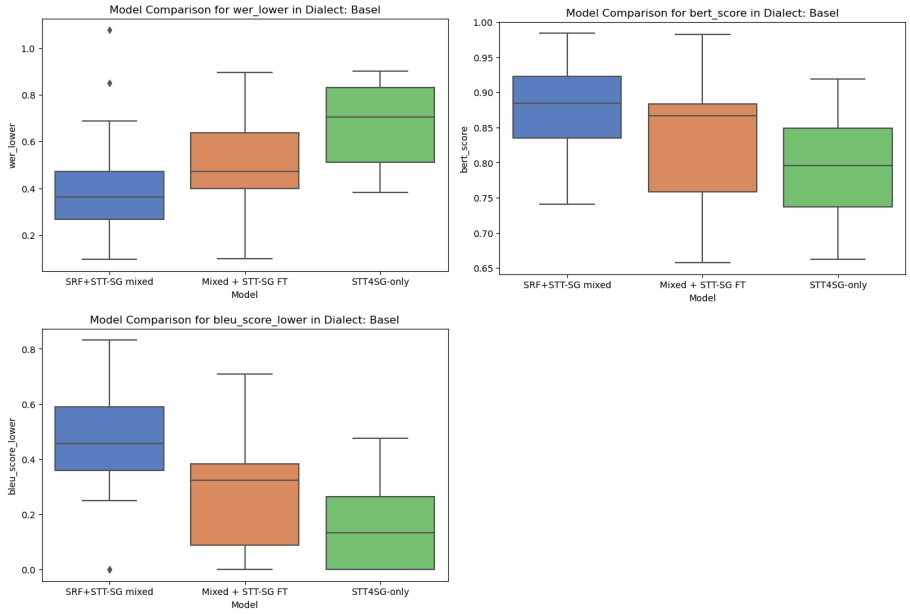

Figure 5.8: Boxplot of metrics values for models which exhibited statistical significance in the Basel dialect.

GPT-Random

As the token count distribution of GPT-Random was more diverse than the two previous evaluation types, this evaluation gauged the general performance of the models on various sentence lengths. Additionally, as only a single speaker with very good audio quality was used to generate the samples, quality-specific voice adaptation issues due to the speaker's audio setup disappeared.

The evaluation exhibited improvements in all metrics compared to the SNF-Long evaluation. The best-performing model is the Mixed+STT4SG-FT, reaching a WER of 0.238, CER of 0.123, BERTScore of 0.937 and a BLEU of 0.610, which was unexpectedly high. more in-depth analysis of the CER provided insight into the distribution of the scores, as visualized in Figure 5.9. The fine-tuned Mixed+STT4SG-FT showed improvements over the non-fine-tuned SRF+STT4SG-Mixed model, confirming the suspected improvement on shorter text segments. This evaluation run also confirmed the internally noted performance degradation during inference on smaller samples by models trained exclusively on mixed corpora, such as the SRF+STT4SG-Mixed. Shorter samples showed significant deviations, while longer samples generally performed well for this model.

Model	WER	CER	BERTScore	BLEU
SRF+STT4SG-Mixed	0.548±0.53	0.354±0.36	0.865±0.12	0.434±0.34
Mixed+STT4SG-FT	**0.238±0.26**	**0.123±0.14**	**0.937±0.08**	**0.610±0.34**
STT4SG-Only	0.326±0.29	0.173±0.17	0.911±0.09	0.498±0.35

Table 5.7: Average GPT-Random transcription results on model basis.

Generally, as the distribution of GPT-Random is skewed towards shorter segments,

the scores naturally tend to centre around 10 to 20 tokens for all three models. However, it is interesting to note that performance stays relatively stable the longer the segments become.

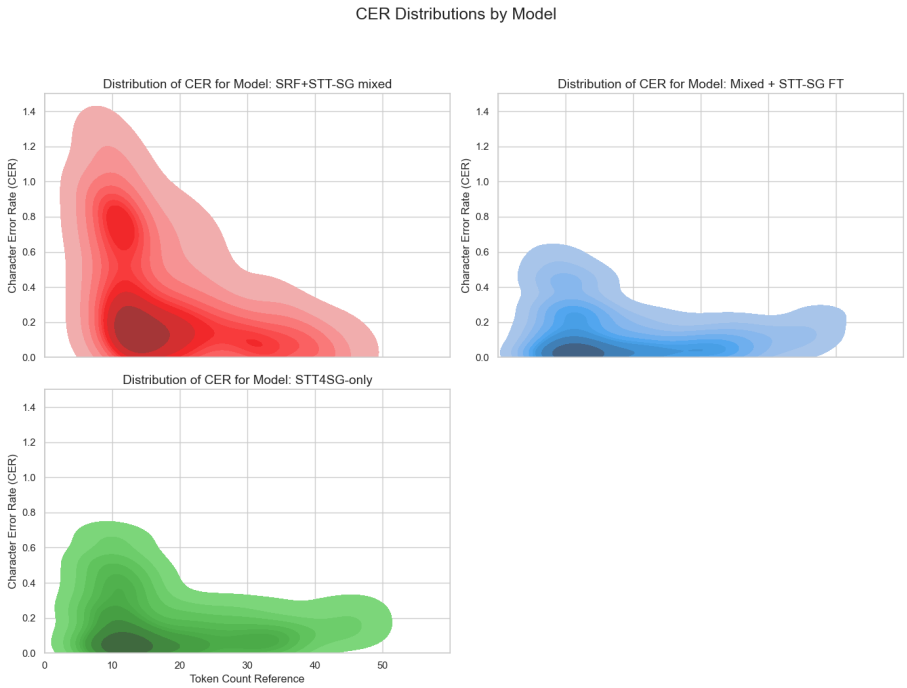

Figure 5.9: GPT-Random CER score distribution across token counts.

Dialect	Model	WER	CER	BERTScore	BLEU
Basel	SRF+STT4SG-Mixed	0.422±0.42	0.280±0.32	0.892±0.10	0.499±0.32
	Mixed+STT4SG-FT	**0.242±0.24**	**0.124±0.13**	**0.935±0.07**	**0.591±0.33**
	STT4SG-Only	0.390±0.30	0.217±0.18	0.891±0.09	0.428±0.34
Bern	SRF+STT4SG-Mixed	0.540±0.55	0.353±0.38	0.870±0.12	0.459±0.34
	Mixed+STT4SG-FT	**0.226±0.28**	**0.118±0.15**	**0.942±0.08**	**0.651±0.35**
	STT4SG-Only	0.312±0.28	0.170±0.18	0.917±0.09	0.516±0.34
Graubünden	SRF+STT4SG-Mixed	0.519±0.50	0.313±0.31	0.876±0.12	0.439±0.36
	Mixed+STT4SG-FT	**0.179±0.20**	**0.092±0.12**	**0.955±0.06**	**0.661±0.33**
	STT4SG-Only	0.242±0.23	0.127±0.15	0.940±0.06	0.569±0.33
Innerschweiz	SRF+STT4SG-Mixed	0.430±0.49	0.285±0.35	0.895±0.10	0.539±0.31
	Mixed+STT4SG-FT	**0.150±0.21**	**0.075±0.12**	**0.962±0.06**	**0.738±0.31**
	STT4SG-Only	0.228±0.24	0.106±0.12	0.942±0.06	0.607±0.33
Ostschweiz	SRF+STT4SG-Mixed	0.551±0.55	0.364±0.40	0.872±0.11	0.430±0.32
	Mixed+STT4SG-FT	**0.284±0.28**	**0.139±0.15**	**0.925±0.08**	**0.538±0.36**
	STT4SG-Only	0.334±0.30	0.177±0.17	0.905±0.09	0.476±0.34
Wallis	SRF+STT4SG-Mixed	0.789±0.59	0.515±0.39	0.798±0.13	0.272±0.33
	Mixed+STT4SG-FT	**0.386±0.30**	**0.212±0.16**	**0.891±0.09**	**0.439±0.32**
	STT4SG-Only	0.451±0.32	0.252±0.20	0.869±0.10	0.374±0.35
Zurich	SRF+STT4SG-Mixed	0.582±0.54	0.365±0.35	0.856±0.13	0.404±0.34
	Mixed+STT4SG-FT	**0.203±0.23**	**0.105±0.13**	**0.950±0.06**	**0.655±0.32**
	STT4SG-Only	0.328±0.31	0.164±0.17	0.913±0.09	0.513±0.36

Table 5.8: Average GPT-Random transcription result on dialect basis.

Splitting the scores based on dialects, as listed in Table 5.8, confirms the high overall performance of the STT4SG-350 [10] fine-tuned Mixed+STT4SG-FT model. It performed the best in all seven dialect regions. What is interesting to note is that the best performance was achieved in the Innerschweiz region, from which the speaker stems. This may hint towards an easier voice adaptation for the models when the speaker has the same source dialect as its target dialect. It performed the worst in all dialect regions by a large margin.

GPT-Long

The last evaluation type, GPT-Long, was expected to favour the SRF+STT4SG-Mixed model more than previous setups, as it only contained sentences of token counts larger than 20. Additionally, the 43 unique speakers produced 30 samples each, with which better insights can be gathered on dialect regions than in the restricted SNF short and SNF long setups.

The performance increase of the mixed-only model was confirmed as listed in Table 5.9, as it ranked second overall. Mixed+STT4SG-FT again had the best performance, with a WER of 0.232, CER of 0.138, BERTScore of 0.937 and a BLEU of 0.633. The scores of this model were very similar to those in the GPT-Random setup. The Baseline model STT4SG-Only performed the worst on all metrics in this setup, which was expected due to longer samples not being present in the STT4SG corpus. Interestingly, the model still improved marginally when compared to the GPT-Random setup.

Model	**WER**	**CER**	**BERTScore**	**BLEU**
SRF+STT4SG-Mixed	0.273±0.25	0.168±0.18	0.930±0.06	0.615±0.26
Mixed+STT4SG-FT	**0.233±0.2**	**0.138±0.13**	**0.937±0.06**	**0.633±0.26**
STT4SG-Only	0.293±0.22	0.172±0.15	0.919±0.07	0.555±0.27

Table 5.9: Average GPT-Long transcription result on model basis.

Looking at the distribution of CER scores over token counts, two distinct clusters can be seen for all three models in varying degrees. The first is found in token counts between 20 and 30, and the second for token counts between 35 and 45. This directly correlates to the general distribution of the evaluation type, seen in Figure 5.6. However, while the SRF+STT4SG-Mixed model seems to have more variability for segments in the first cluster, it performs better in the second. This is directly correlated with the bimodal distribution seen in Figure 3.7 in the SRF-corpus, where the token count of 20-30 was in the valley while token counts in the range of 35-45 around the second peak. Mixed+STT4SG-FT, on the contrary, has fewer difficulties with the first cluster, but due to the fine-tuning of the STT4SG corpus, it lost some performance on the second cluster of long sentences.

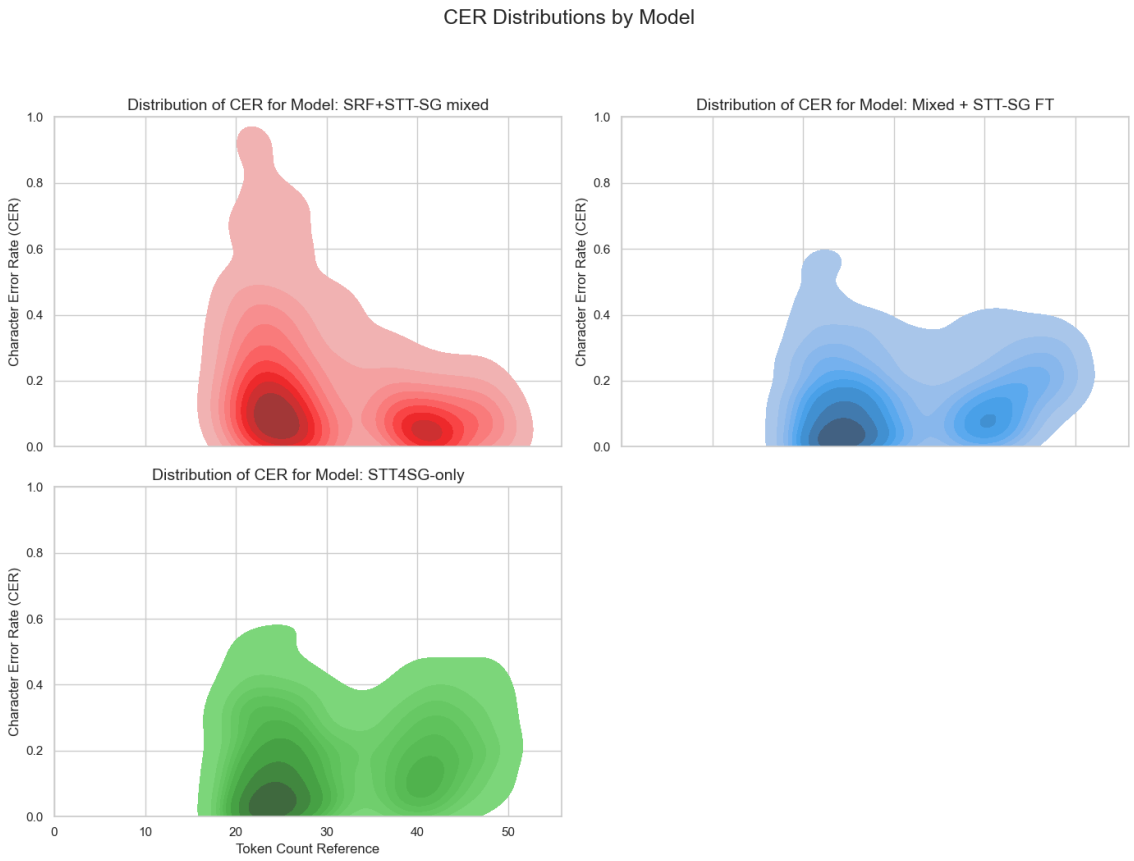

Figure 5.10: GPT-Long CER score distribution across token counts.

Looking into the dialect regions, the two models trained on the SRF-corpus outperformed the Baseline model in six out of seven dialect regions. The best-performing region was again the Innerschweiz with a WER of 0.222, CER of 0.095, BERTScore of 0.941, and a BLEU of 0.745 by the Mixed+STT4SG-FT model. The worst performing region remained Wallis with the SRF+STT4SG-Mixed achieving the best results of a WER of 0.416, CER of 0.263, BERTScore of 0.897, and a BLEU of 0.489. The other five regions had a similar performance overall by the Mixed+STT4SG-FT model with a WER between 0.221 and 0.248, CER between 0.125 and 0.149, BERTScore between 0.929 and 0.944, and a BLEU between 0.611 and 0.655.

It is important to note that the dialect regions are not balanced in this setup due to the different number of speakers in each dialect, as outlined in Figure 5.1. Specifically, the Wallis region may have suffered due to only containing 3 unique speakers, resulting in 90 samples, compared to, for example, Zurich, with six unique speakers, resulting in 180 samples. Future work may use a more balanced setup for the evaluation of dialect-specific setups. However, as seen in the GPT-Random setup, in which dialects were balanced with 100 samples in each, Wallis was still the worst-performing dialect region.

Dialect	Model	WER	CER	BERTScore	BLEU
Basel	SRF+STT4SG-Mixed	**0.229±0.20**	0.134±0.14	**0.944±0.05**	**0.644±0.24**
	Mixed+STT4SG-FT	0.231±0.19	**0.132±0.12**	0.938±0.06	0.629±0.26
	STT4SG-Only	0.336±0.20	0.189±0.12	0.908±0.06	0.475±0.25
Bern	SRF+STT4SG-Mixed	0.273±0.26	0.169±0.20	**0.929±0.07**	0.617±0.27
	Mixed+STT4SG-FT	**0.221±0.20**	**0.133±0.13**	0.941±0.06	**0.647±0.25**
	STT4SG-Only	0.331±0.25	0.196±0.16	0.907±0.08	0.528±0.28
Graubünden	SRF+STT4SG-Mixed	0.236±0.22	0.143±0.16	**0.942±0.05**	**0.655±0.23**
	Mixed+STT4SG-FT	**0.225±0.17**	**0.131±0.11**	0.942±0.05	0.639±0.23
	STT4SG-Only	0.257±0.24	0.155±0.17	0.930±0.08	0.614±0.26
Innerschweiz	SRF+STT4SG-Mixed	0.233±0.18	0.140±0.13	0.937±0.05	0.647±0.23
	Mixed+STT4SG-FT	**0.151±0.15**	**0.095±0.12**	**0.960±0.05**	**0.745±0.20**
	STT4SG-Only	0.196±0.17	0.117±0.14	0.948±0.06	0.678±0.22
Ostschweiz	SRF+STT4SG-Mixed	0.279±0.24	0.172±0.17	0.928±0.07	0.611±0.26
	Mixed+STT4SG-FT	**0.222±0.16**	**0.125±0.10**	**0.941±0.05**	**0.644±0.23**
	STT4SG-Only	0.232±0.18	0.132±0.11	0.941±0.05	0.620±0.25
Wallis	SRF+STT4SG-Mixed	**0.416±0.36**	**0.263±0.26**	**0.897±0.09**	**0.489±0.29**
	Mixed+STT4SG-FT	0.444±0.25	0.272±0.17	0.872±0.08	0.386±0.26
	STT4SG-Only	0.451±0.23	0.273±0.15	0.864±0.08	0.366±0.27
Zurich	SRF+STT4SG-Mixed	0.326±0.30	0.213±0.24	0.911±0.09	0.568±0.29
	Mixed+STT4SG-FT	**0.248±0.20**	**0.149±0.14**	**0.932±0.06**	**0.611±0.26**
	STT4SG-Only	0.316±0.21	0.185±0.15	0.914±0.07	0.523±0.27

Table 5.10: Average GPT-Long transcription result on dialect basis.

Overall

Combining the four different evaluation types for an overarching performance comparison (support = 105 + 105 + 700 + 1290 = 2200), the Mixed+STT4SG-FT model seems to have performed the best. It consistently outperformed the others, achieving the lowest error rates (WER: 0.259, CER: 0.141) and the highest semantic similarity (BERTScore: 0.929) as well as BLEU (0.594). This indicates that fine-tuning the mixed model with the STT4SG dataset improved transcription accuracy, semantic alignment, and n-gram overlap. The SRF+STT4SG-Mixed model, while ranking third in most metrics, showed the second best BLEU performance (0.534), suggesting its outputs align well with reference n-grams despite higher error rates (WER: 0.386, CER: 0.239). Finally, the Baseline STT4SG-Only model demonstrated intermediate performance, outperforming the SRF+STT4SG-Mixed model in WER (0.327), CER (0.179), and BERTScore (0.909), but achieving the lowest BLEU score (0.508). This suggests that while this model captures semantic similarity better, it may lack consistency in generating exact matches with reference text.

Model	WER	CER	BERTScore	BLEU
SRF+STT4SG-Mixed	0.386±0.41	0.239±0.29	0.902±0.10	0.534±0.31
Mixed+STT4SG-FT	**0.259±0.24**	**0.141±0.14**	**0.929±0.07**	**0.594±0.30**
STT4SG-Only	0.327±0.26	0.179±0.16	0.909±0.08	0.508±0.31

Table 5.11: Overall transcription average result on model basis.

The distribution of CER visualized a combination of the four evaluation types dis-

cussed earlier. Three distinct clusters can be detected in all three models, which directly correspond to short (up to 14 tokens), medium (between 20 and 30 tokens) and long text segments (between 35 and 45 tokens). While variation in the SRF+STT4SG-Mixed is substantial for shorter sentences, it reduces considerably in longer segments. The Mixed+STT4SG-FT model has the lowest variations in all three token count clusters and thus confirms its overall strength. The Baseline STT4SG-Only model has comparable variance with the Mixed+STT4SG-FT model except for shorter segments that performed worse.

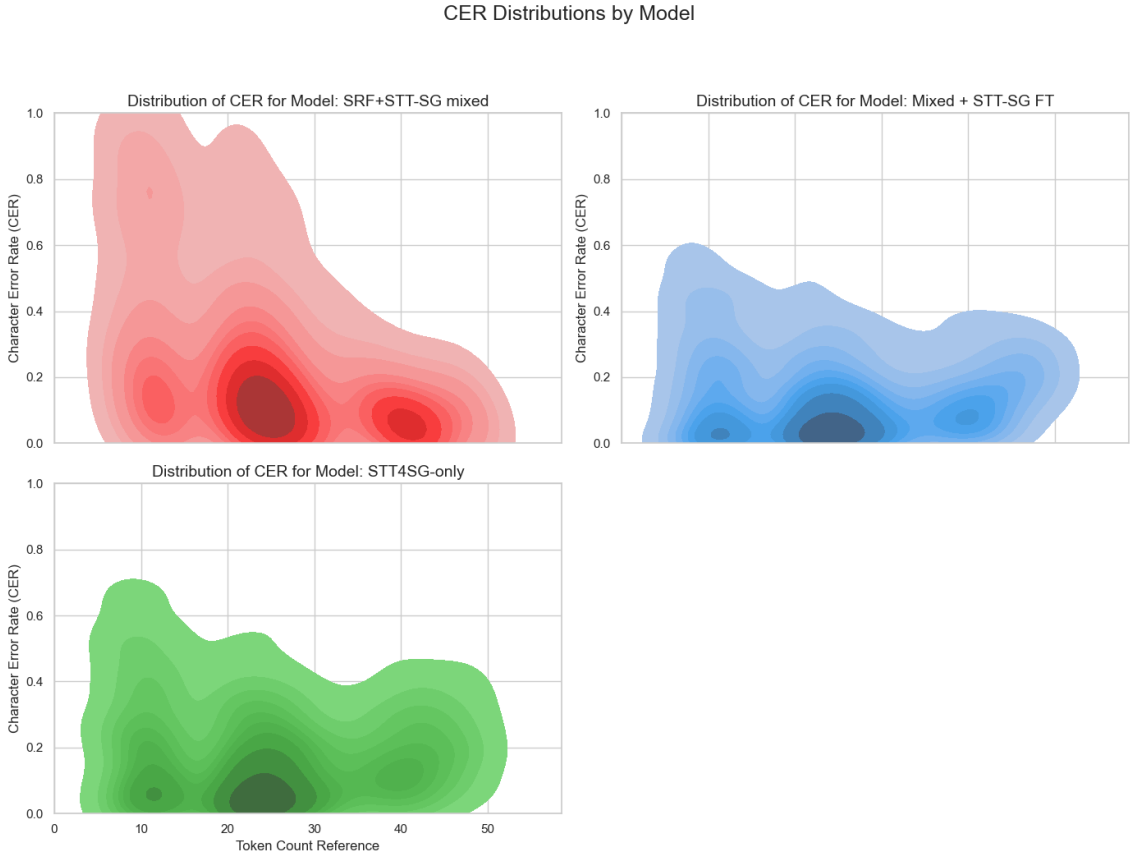

Figure 5.11: Overall CER score distribution across token counts.

The Mixed+STT4SG-FT model also performed best in all dialect regions, emerging as the most robust model across dialects. Innerschweiz was the best-performing region overall, while Wallis was the worst-performing region, as already seen in the earlier evaluations. The SRF+STT4SG-Mixed model performed inconsistently, often ranking third across WER, CER, and BERTScore. However, it occasionally showed increased BLEU performance (e.g., BLEU: 0.581 for Basel and 0.579 for Graubünden), suggesting it aligns well with reference text structures in certain regions despite higher error rates. The STT4SG-Only model generally falls in the middle, achieving reasonable scores across most metrics but lagging in BLEU (e.g., Zurich: 0.492, Ostschweiz: 0.544). This suggests it captures semantic information better than SRF+STT4SG-Mixed but lacks the refinement of the STT4SG-fine-tuned model. Table 5.12 lists all scores in detail.

Dialect	Model	WER	CER	BERTScore	BLEU
Basel	SRF+STT4SG-Mixed	0.306±0.30	0.186±0.22	0.923±0.08	0.581±0.28
	Mixed+STT4SG-FT	**0.262±0.24**	**0.140±0.13**	**0.927±0.07**	**0.582±0.30**
	STT4SG-Only	0.382±0.25	0.207±0.15	0.894±0.08	0.432±0.29
Bern	SRF+STT4SG-Mixed	0.373±0.40	0.234±0.28	0.904±0.10	0.551±0.31
	Mixed+STT4SG-FT	**0.241±0.23**	**0.135±0.14**	**0.934±0.07**	**0.624±0.30**
	STT4SG-Only	0.345±0.27	0.198±0.17	0.902±0.09	0.501±0.31
Graubünden	SRF+STT4SG-Mixed	0.332±0.36	0.197±0.23	0.918±0.09	0.579±0.30
	Mixed+STT4SG-FT	**0.229±0.19**	**0.124±0.12**	**0.939±0.06**	**0.619±0.28**
	STT4SG-Only	0.266±0.24	0.149±0.17	0.928±0.08	0.582±0.29
Innerschweiz	SRF+STT4SG-Mixed	0.319±0.34	0.194±0.24	0.918±0.08	0.589±0.28
	Mixed+STT4SG-FT	**0.168±0.19**	**0.093±0.12**	**0.955±0.06**	**0.714±0.27**
	STT4SG-Only	0.224±0.21	0.117±0.13	0.939±0.07	0.625±0.28
Ostschweiz	SRF+STT4SG-Mixed	0.386±0.40	0.240±0.28	0.903±0.09	0.530±0.30
	Mixed+STT4SG-FT	0**.264±0.23**	**0.136±0.13**	**0.929±0.07**	**0.582±0.30**
	STT4SG-Only	0.285±0.24	0.152±0.14	0.922±0.07	0.544±0.30
Wallis	SRF+STT4SG-Mixed	0.629±0.57	0.403±0.42	0.841±0.12	0.361±0.33
	Mixed+STT4SG-FT	**0.447±0.28**	**0.251±0.16**	**0.869±0.09**	0**.375±0.30**
	STT4SG-Only	0.486±0.29	0.275±0.18	0.855±0.10	0.339±0.31
Zurich	SRF+STT4SG-Mixed	0.444±0.44	0.277±0.30	0.885±0.11	0.485±0.32
	Mixed+STT4SG-FT	**0.259±0.24**	**0.142±0.14**	**0.929±0.07**	**0.592±0.31**
	STT4SG-Only	0.343±0.26	0.186±0.16	0.905±0.08	0.492±0.31

Table 5.12: Average Overall Transcription Result on Dialect Basis.

5.3.2 Regression

To further explore the data setup, a regression was performed on variables we deemed to impact the model output quality. As such, a run of the GPT-Long was utilized, in which 30 audio samples were generated for each of the 43 speakers and automatically evaluated for WER. Afterwards, a random forest with $n = 100$ estimators, and a Gradient Boosting Regressor with $n = 100$ estimators and $lr = 0.01$ were trained with four distinct features listed in Table 5.13 for the target variable WER.

Variable Name	Description
dialect_sample_ratio	The ratio of a speakers dialect in comparison to the complete mixed training corpora of STT4SG-350 [10] and SRF-corpus
token_count_ref	The number of tokens in the reference sentence, calcualted using spaCy [43]
average_cond	The number of conditioning samples classified as average by dataspeech. In total, each speaker only had 5 conditioning samples.
good_cond	The number of conditioning samples classified as good by dataspeech. In total, each speaker only had 5 conditioning samples.

Table 5.13: Description of Regression Variables

The Random Forest was completed with a Mean Squared Error (MSE) of 0.06305 and an R-squared of 0.12709, and Gradient Boosting Regression had a MSE of

0.053197 and an R-squared of 0.26359. The regression result is listed in Table 5.14. Both models agree that the most important features influencing the WER are the number of tokens in a sentence (token_count_ref) and the ratio of the speaker's dialect compared to the complete dataset (dialect_sample_ratio). While the quality of the conditioning samples is still important, their influence on WER is less significant than the other two features.

Variable Name	Random Forest	Gradient Boosting
token_count_ref	0.485184	0.422241
dialect_sample_ratio	0.325478	0.340617
average_cond	0.099546	0.130195
good_cond	0.089793	0.106946

Table 5.14: Regression Models and Their Scores

5.3.3 Conditioning Samples

During experiments, the question of whether the ordering of the five conditioning samples had an impact on the performance of the XTTS model arose. As such, an experiment was started by rerunning the GPT-Random sentences seven times with the same ordering of the conditioning samples and seven times with a random order different from the others. The utilized speaker was the same as the one selected in the GPT-Random evaluation and had great audio quality. The model used for this was the Mixed+STT4SG-FT model.

Figure 5.12 confirmed the assumptions that the ordering of the conditioning samples had a significant impact on the output quality of the model. All four metrics exhibited a larger variation of scores when evaluating the shuffled-order generated audio samples (blue), including better and worse scores compared to the same-order runs (orange). Interestingly, even with the same ordering, the generated samples may have a noticeable difference when compared to each other.

Important performance improvements could be achieved if the samples are optimized before inference. Future work may thus investigate why these variations occur and develop an automatic conditioning sample optimizer for future voice adaption experiments. It is also important to investigate if the dialect and speaker adaptation still work as intended or if there are significant changes to the synthesised voice.

The evaluation results of this shuffle experiment can be found on GitHub[3]

[3]https://github.com/stucksam/swiss-zero-shot-va-tts/

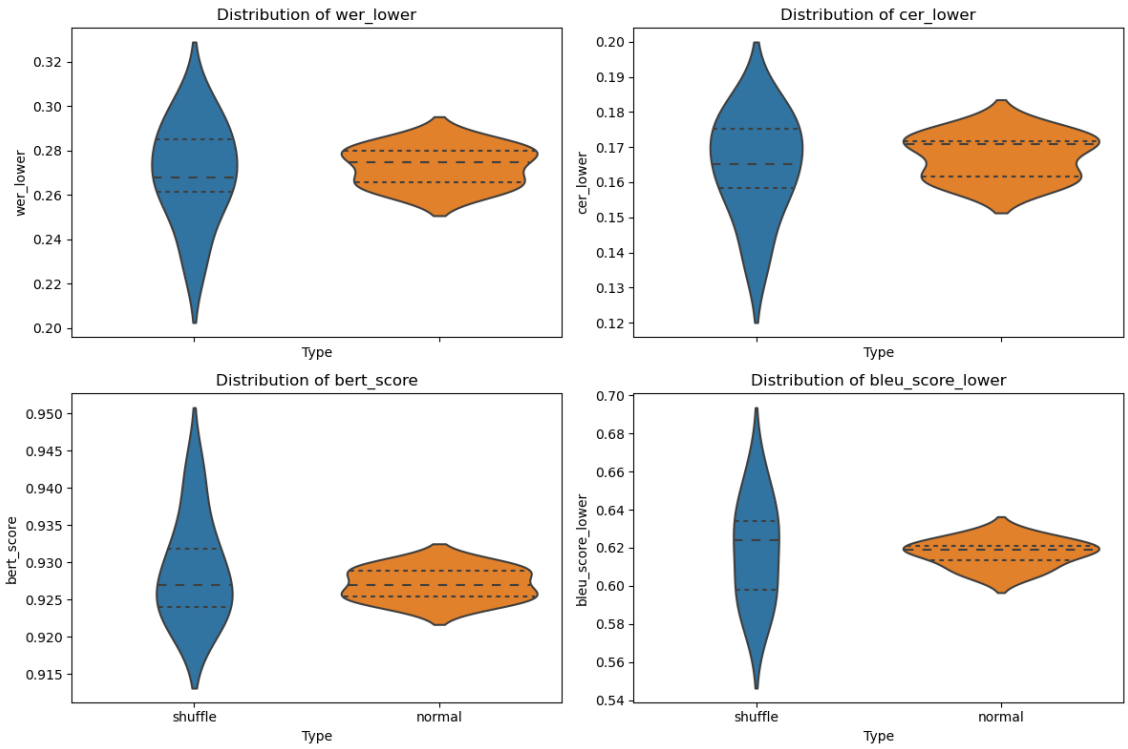

Figure 5.12: Violin plots for all four metrics with shuffle referring to conditioning samples being uniquely reordered and normal as conditioning samples remaining in the same order.

5.3.4 Dialect Identification

The strength of the models in reproducing the dialects they targeted was done using the output of SNF-Short and GPT-Long. As the Naive Bayes classifier is only accurate when providing phonemes of samples > 30 seconds, it was impossible to accurately classify the samples on a stand-alone basis. For SNF-Short, additional samples were created for each speaker using the other 15 sentences in the evaluation set. Thus, each speaker had 15 generated samples, transcribed to phonemes and concatenated with each other during DID for processing by the Naive Bayes model. SNF-Short evaluation is balanced on a dialect region basis (n=15) but unbalanced on a speaker basis.

The additional generation of audio samples was unnecessary for GPT-Long as each speaker already possessed 30 different samples. As with SNF-Short, the phonemes of these samples were appended and then classified by the Naive Bayes model. This tended to assign all samples of a speaker to the same class. Future work may define a more refined approach to evaluate the performance of the dialect adaptation by utilising a classifier that is not dependent on longer audio segments. The GPT-Long evaluation is unbalanced on both the dialect level as well as on the speaker level. Due to this, support for each dialect will also be provided in score listings. Table 5.15 lists the result of the Dialect Identification by the three models.

Eval Type	Model	Weighted F1	Macro F1	Micro F1
SNF-Short	SRF+STT4SG-Mixed	0.4286	0.4286	0.4762
	Mixed+STT4SG-FT	0.4121	0.4121	0.5048
	STT4SG-Only	**0.4823**	**0.4823**	**0.5143**
GPT-Long	**SRF+STT4SG-Mixed**	**0.7876**	**0.7946**	**0.8140**
	Mixed+STT4SG-FT	0.7027	0.7058	0.7209
	STT4SG-Only	0.5868	0.5929	0.6512

Table 5.15: F1-Scores of models on SNF-Short and GPT-Long dialect classification.

First, the performance of the models on SNF-Short is categorized by low scores, with the Baseline STT4SG-Only performing the best with a macro F1 of 0.4823. Both SRF-corpus trained performed considerably worse at 0.4286 and 0.4121, respectively. Some significant differences can be observed when investigating the dialect regions in Table 5.16. Graubünden and Ostschweiz show the highest overall F1-scores – both STT4SG-Only and Mixed+STT4SG-FT achieve an F1-score of 0.8333 in Graubünden, while in Ostschweiz, STT4SG-Only (0.75) and Mixed+STT4SG-FT (0.8108) perform exceptionally well.

Basel and Zürich perform the weakest, with the Mixed+STT4SG-FT model failing to classify them correctly (0.0 F1-score) and Zürich showing extremely low precision and recall across all models. This indicates significant challenges in recognizing these dialects, which is surprising considering the size of both datasets in the SRF-corpus is significant. For Bern and Graubünden, models often achieve perfect recall (1.0) but lower precision, meaning they correctly identify relevant speakers and frequently misclassify others. While the SRF+STT4SG-Mixed achieves high precision in Wallis, Basel, its recall is often low (e.g., 0.20 in Wallis, 0.2667 in Basel and Innerschweiz).

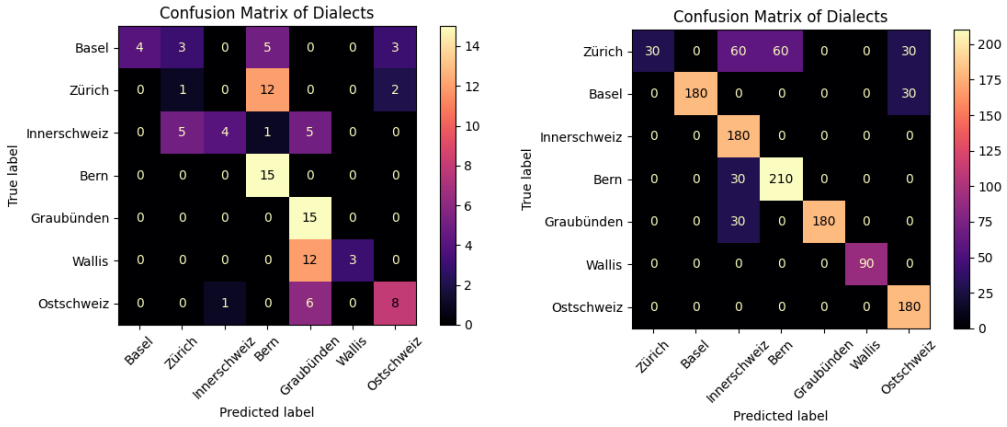

Figure 5.13: Confusion matrices for SNF-Short (left) and GPT-Long (right) classifications for the SRF+STT4SG-Mixed model.

Discussing the GPT-Long results in Table 5.17, the SRF+STT4SG-Mixed outperforms the other two models by a considerable margin, achieving a weighted F1 score

Dialect	Model	Precision	Recall	F1-Score
Basel	SRF+STT4SG-Mixed	1.0000	0.2667	0.4211
	Mixed+STT4SG-FT	0.0000	0.0000	0.0000
	STT4SG-Only	1.0000	0.3333	**0.5000**
Bern	**SRF+STT4SG-Mixed**	0.4545	1.0000	**0.6250**
	Mixed+STT4SG-FT	0.3191	1.0000	0.4839
	STT4SG-Only	0.2308	0.4000	0.2927
Graubünden	SRF+STT4SG-Mixed	0.3947	1.0000	0.5660
	Mixed+STT4SG-FT	0.7143	1.0000	**0.8333**
	STT4SG-Only	0.7143	1.0000	**0.8333**
Innerschweiz	SRF+STT4SG-Mixed	0.8000	0.2667	0.4000
	Mixed+STT4SG-FT	0.5000	0.1333	0.2105
	STT4SG-Only	0.2353	0.2667	**0.2500**
Ostschweiz	SRF+STT4SG-Mixed	0.6154	0.5333	0.5714
	Mixed+STT4SG-FT	0.6818	1.0000	**0.8108**
	STT4SG-Only	1.0000	0.6000	0.7500
Wallis	SRF+STT4SG-Mixed	1.0000	0.2000	0.3333
	Mixed+STT4SG-FT	0.6250	0.3333	0.4348
	STT4SG-Only	0.6000	1.0000	**0.7500**
Zürich	SRF+STT4SG-Mixed	0.1111	0.0667	0.0833
	Mixed+STT4SG-FT	0.3333	0.0667	**0.1111**
	STT4SG-Only	0.0000	0.0000	0.0000

Table 5.16: F1-Scores of models on dialect in SNF-Short result.

of 0.7876. Compared to the Mixed+STT4SG-FT, which achieved an F1 of 0.7027, and the Baseline STT4SG-Only models' significantly lower F1 of 0.5868. This underlines that adding the SRF-corpus significantly improves performance on replicating Swiss German dialects. More interesting is the performance increase of the Baseline model compared to the shorter segments in which it should perform better. This may hint towards XTTS having an easier time replicating speech patterns the longer the segments are.

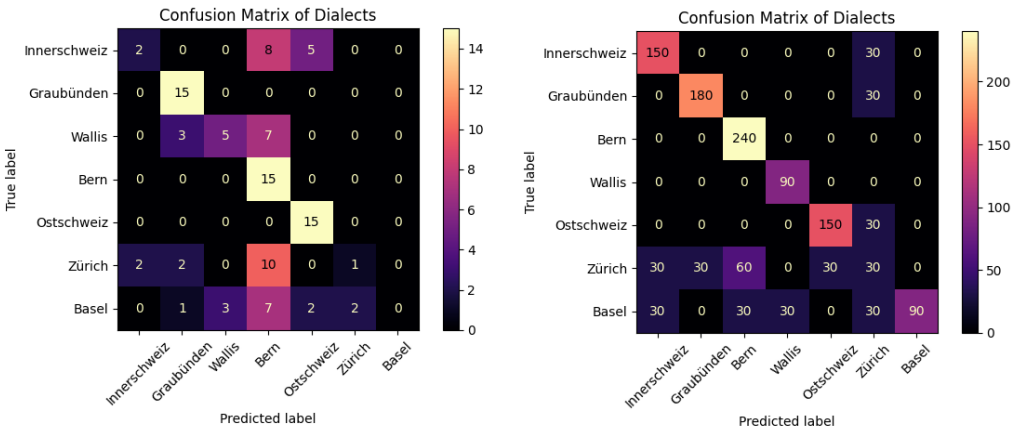

Figure 5.14: Confusion matrices for SNF-Short (left) and GPT-Long (right) classifications for the Mixed+STT4SG-FT model.

Dialect	Model	Precision	Recall	F1-Score	Support
Basel	**SRF+STT4SG-Mixed**	1.0000	0.8571	**0.9231**	210
	Mixed+STT4SG-FT	1.0000	0.4286	0.6000	210
	STT4SG-Only	1.0000	0.1429	0.2500	210
Bern	SRF+STT4SG-Mixed	0.7778	0.8750	0.8235	240
	Mixed+STT4SG-FT	0.7273	1.0000	**0.8421**	240
	STT4SG-Only	0.5385	0.8750	0.6667	240
Graubünden	SRF+STT4SG-Mixed	1.0000	0.8571	0.7500	210
	Mixed+STT4SG-FT	0.8571	0.8571	0.8571	210
	STT4SG-Only	1.0000	1.0000	**1.0000**	210
Innerschweiz	SRF+STT4SG-Mixed	0.6000	1.0000	0.7500	180
	Mixed+STT4SG-FT	0.7143	0.8333	**0.7692**	180
	STT4SG-Only	0.6250	0.8333	0.7143	180
Ostschweiz	**SRF+STT4SG-Mixed**	0.7500	1.0000	**0.8571**	180
	Mixed+STT4SG-FT	0.7143	0.8333	0.7692	180
	STT4SG-Only	0.5868	0.5929	0.6512	180
Wallis	**SRF+STT4SG-Mixed**	1.0000	1.0000	**1.0000**	90
	Mixed+STT4SG-FT	0.7500	1.0000	0.8571	90
	STT4SG-Only	0.6000	1.0000	0.7500	90
Zürich	**SRF+STT4SG-Mixed**	1.0000	0.1667	**0.2857**	180
	Mixed+STT4SG-FT	0.2000	0.1667	0.1818	180
	STT4SG-Only	0.0000	0.0000	0.0000	180

Table 5.17: F1-Scores of models on dialect in GPT-Long result.

Investigating the dialect regions reveals that certain dialects, like Wallis and Graubünden, are replicated with high accuracy across all models, with perfect or near-perfect F1 Scores. This suggests that the models capture their phonetic characteristics well, confirming the very distinct pronunciation patterns of the dialects most native speakers can categorize. On the other hand, Zürich shows extremely low scores across all models, indicating significant difficulty in generating speech that accurately reflects it.

Across most dialects, the SRF+STT4SG-Mixed model consistently performs well, often achieving the highest F1-scores. The Mixed+STT4SG-FT model performs moderately well but does not dominate across dialects, implying that some performance was lost after fine-tuning on the STT4SG-350 [10]. The Baseline STT4SG-Only is inconsistent—while it excels in Graubünden (1.0 F1), it ultimately fails in Zürich (0.0 F1) and has considerable issues with Basel (0.25 F1). This suggests that a model trained exclusively on STT4SG-350 data alone cannot effectively replicate all dialects.

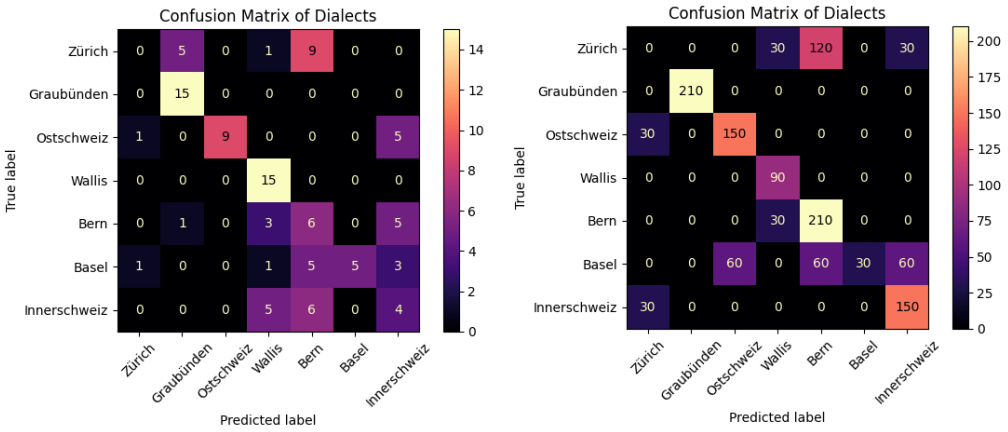

Figure 5.15: Confusion matrices for SNF-Short (left) and GPT-Long (right) classifications for the STT4SG-Only model.

Interesting Findings for Standard German

During the training and evaluation of models, the authors noticed a trend in which the pre-trained German on XTTS slowly changed the pronunciation of Standard German to Swiss Standard German. This was thought to occur because of the in Chapter 3 mentioned accent that is very prevalent in Swiss German speakers and was thus apparent in the German portion of the SRF-corpus as well, due to SRF encouraging the news broadcasters to speak with this accent. Future work may investigate further how accurately the Swiss Standard German could be replicated TTS-models.

5.4 Human Evaluation

Automated Metrics such as WER or BLEU lack contextual understanding as they depend on reference transcriptions and assume standardized pronunciation. Whisper, the transcription model used in this thesis, is trained on high-resource languages like Standard German, introducing biases and inaccuracies when applied to Swiss German dialects as seen in the evaluation of the data pipeline in Section 3.2.3. Evaluating speech naturalness requires assessing the intonation, stress, rhythm, and more. Native speakers of Swiss German can easily judge these characteristics and if they match expected speech patterns, such as those of the speakers used in voice adaptation. This is especially important in Swiss German, where fluency often depends on local phrasing and rhythm. The human evaluation also allows testing intelligibility in practical contexts, such as understanding complex sentences or ambiguous pronunciations, which may not be captured in the automated metrics. As such, a human evaluation was conducted to compare the trained models.

5.4.1 Metrics

Three subjective-based metrics were defined for native speakers to classify the generated audio samples. The classification of the spoken dialect was not one of them. This was due to the task's complexity and the evaluators' potential bias. Swiss German consists of numerous distinct dialects, each with its unique characteristics. While native speakers are often highly proficient in recognizing the dialect of their region, this can create a challenge when evaluating dialects from other areas. For example, a native speaker from Zürich may accurately identify the Zürich dialect but may find it more challenging to assess dialects from regions such as Bern or Innerschweiz, which may have different phonological patterns. This regional bias can lead to inconsistent evaluations, where dialects that are distant from the evaluator's region may be either under or over-rated regarding authenticity or naturalness.

SMOS

Similarity Mean Opinion Score (SMOS) is used to evaluate how similar the synthesized speech is to the reference speaker's voice, irrespective of the content. It measures whether the TTS model captures the speaker's unique characteristics, such as tone, pitch, and speaking style. The SMOS scale ranges from 1 to 5, with increments of 0.5 points.

- 1: Completely Dissimilar.
- 2: Mostly Dissimilar, with minor similarities.
- 3: Somewhat Similar, noticeable differences in tone or style.
- 4: Mostly Similar, minor differences only.
- 5: Same Voice.

CMOS

Comparative Mean Opinion Score (CMOS) is used to evaluate the comparative naturalness of synthesized speech against a given reference speech. "Better" refers to synthesized speech that is smoother, more natural, and closer to human-like qualities than the reference, while "worse" describes speech that sounds robotic, unnatural, or distorted compared to the reference. The spoken text is of no relevance in the evaluation. The CMOS scale ranges from -3 to 3, with intervals of 1:

- -3: Much worse than the reference.
- -2: Worse than the reference.
- -1: Slightly worse than the reference.
- 0: Same as the reference.
- +1: Slightly better than the reference.
- +2: Better than the reference.
- +3: Much better than the reference.

Intelligibility

Intelligibility evaluates how easy it is to understand the synthesized speech and ensures that the synthesized speech is clear and comprehensible when compared to the text prompt. The Intelligibility scale ranges from 1 to 5, with intervals of 1:

- 1 (Bad): Very difficult or impossible to understand.
- 3 (Fair): Moderately clear, with some unclear or mispronounced words.
- 5 (Excellent): Perfectly clear and easy to understand.

5.4.2 Evaluator Sourcing

In total, six evaluators were sourced for this evaluation. The evaluators were volunteers and were, as such, not compensated for their time. Two evaluators were the author's parents, and two others were the author's flatmates. The remaining two were Co-workers at the CAI. Four participants classified themselves as speaking the Ostschweiz dialect, one evaluator the Bern dialect, and the last person the Innerschweiz dialect. Each evaluation took approximately 1 hour, for which we thank the evaluators.

5.4.3 Structure

A mix of sentences utilized in the automated evaluation was deemed necessary for the human evaluation. This enabled cross-evaluation and reused the existing test setup. Considering evaluation fatigue, the evaluation sets were not allowed to be too large, as all three trained models had to be evaluated. A healthy mix between sufficient samples and time investment by an individual evaluator was found to be 84 for each evaluation, split into two sets of 42. Each set consisted of a specific type of sentence structure, with the first set, termed "HumanEval-Short", focusing on short samples from the SNF-Short evaluation type described in Section 5.2.1. The second set, termed "HumanEval-Long", was made up of equal parts (read 21) from the SNF-Long and GPT-Long evaluation types, outlined in Section 5.2.2 and Section 5.2.4 respectively. Three subset evaluation sets were created, with two evaluators voting on a given subset to enable insight into variations in judgment while covering a larger sample pool. As these configurations are not trivial, HumanEval-Short and HumanEval-Long will now be defined more clearly.

HumanEval-Short

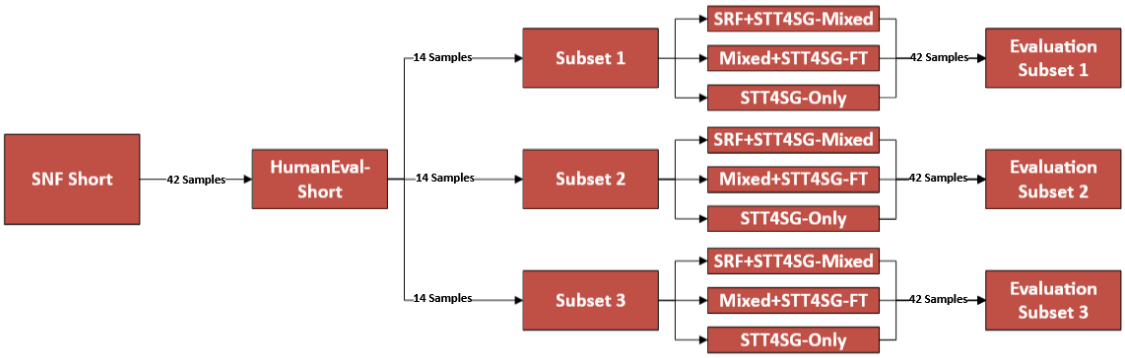

Figure 5.16: HumanEval-Short generation method and subset development.

Figure 5.16 is provided first to support the explanation. The HumanEval-Short subset consisted of 42 unique samples from SNF-Short. These 42 unique sentences had to be evaluated for each model. As such, the effective sample size of this dataset grew to 3*42 = 126. This was too large for an individual evaluator. Following, HumanEval-Short was split into three subsets of 14 unique sentences each, which, due to the three models, had an effective sample size of 42, which the evaluators had to judge.

Next, the sampling of the unique sentences from SNF-Short will be explained. First, the 42 samples were chosen to be equally distributed across the seven dialects. As such, six samples per dialect were chosen randomly based on the speakers' dialect region. The resulting gender distribution per dialect can be seen in Figure 5.17. In total, 30 unique speakers were present in the dataset. The 42 sentences were shuffled randomly before splitting them into three subsets.

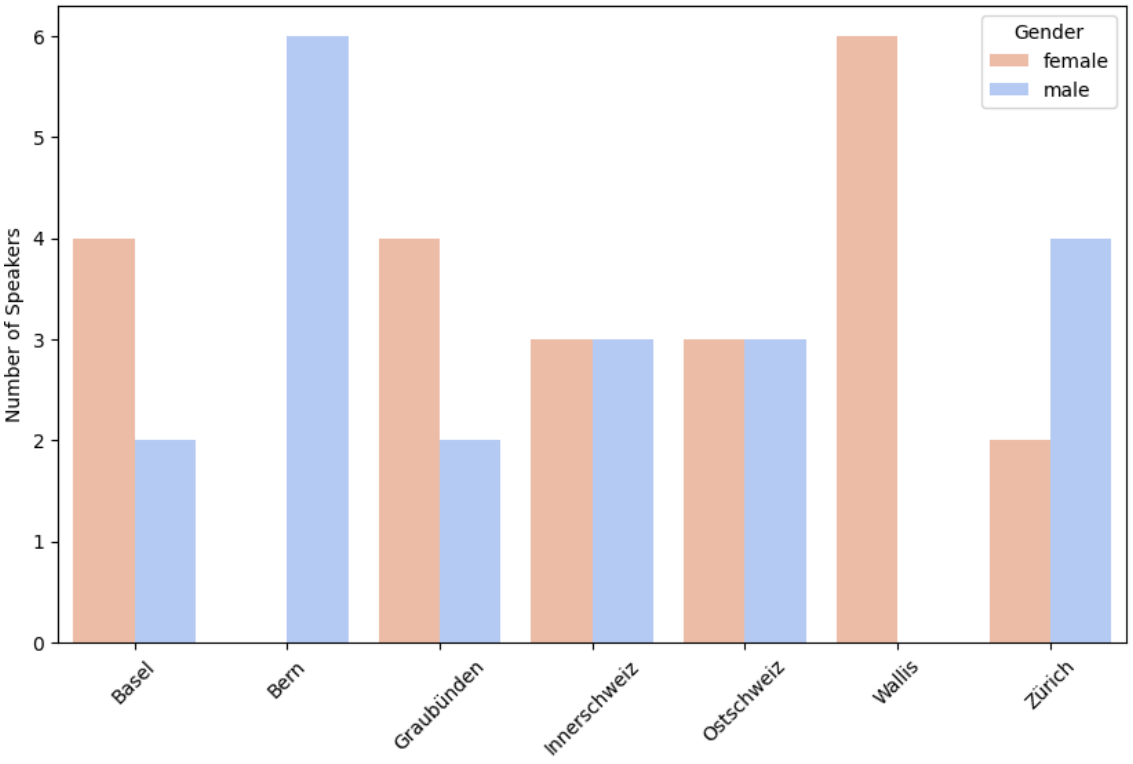

Figure 5.17: HumanEval-Short gender distribution across dialect region.

HumanEval-Long

HumanEval-Long had the same approach as HumanEval-Short but consisted of two sources: SNF- and GPT-Long. As such, 21 unique sentences were sampled from each evaluation type, as shown in Figure 5.18. These were then split again into three subsets for each source, totalling six subsets, each consisting of 7 unique sentences. Each sentence had a corresponding audio sample in each model, resulting in an effective evaluation subset of 21 samples for SNF- and GPT-Long. For the evaluation, the equal numbered subsets in SNF-Long and GPT-Long (e.g. "Eval SNF-Long Subset 1" and "Eval GPT-Long Subset 1") were combined to form the second evaluation set with an effective audio sample size of 42.

The sources were sampled in SNF-Long first by filtering out speakers not present in HumanEval-Short. This allowed cross-comparison of voice adaptation between short and long samples. Next, it was randomly downsampled to contain only three samples per dialect, resulting in 21 total samples. and the gender distribution seen in Figure 5.19. Afterwards, the same speakers were sampled from GPT-Long, and again, 21 samples were filtered so that each dialect region had three unique samples. The 21 samples in SNF-Long and the 21 samples in GPT-Long were randomly shuffled and split into three subsets of 7 sentences each.

The evaluation of the HumanEval-Long group will be evaluated as a whole, not on SNF-Long and GPT-Long basis, due to the goal of this set being the evaluation of longer segments as a whole.

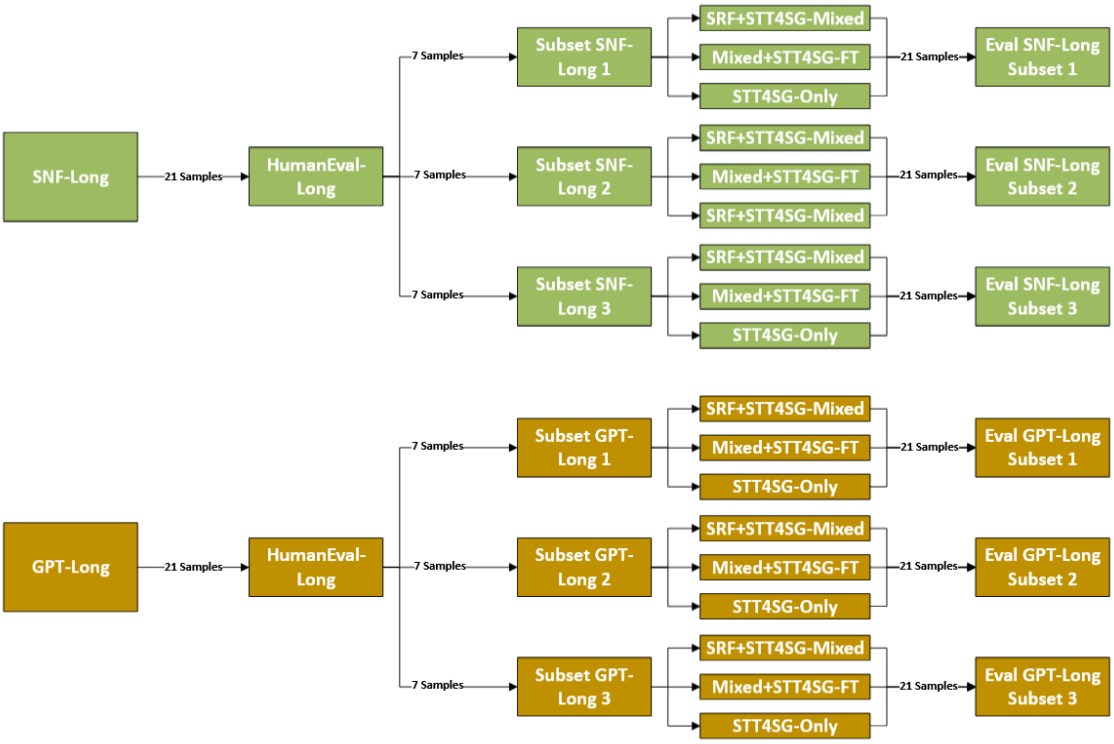

Figure 5.18: HumanEval-Long generation method and subset development.

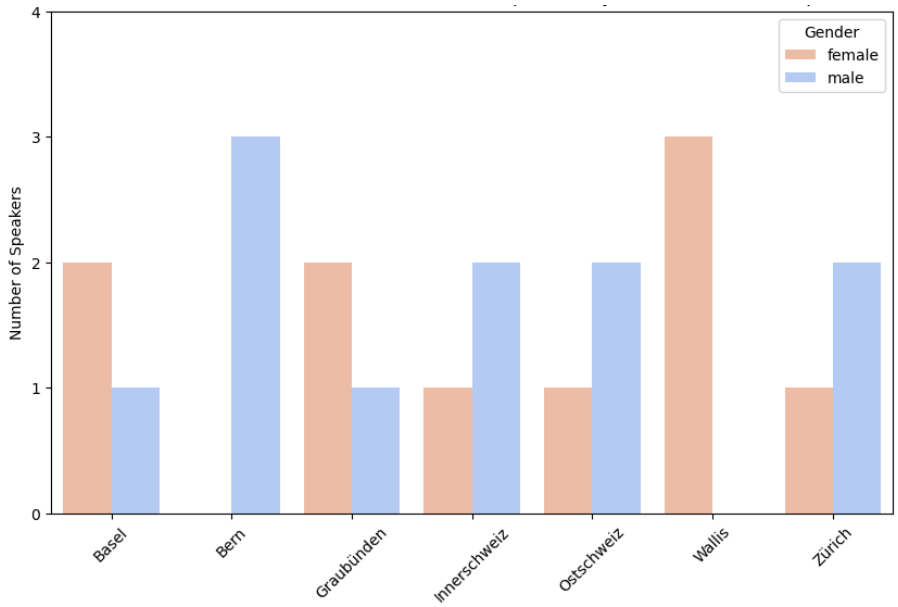

Figure 5.19: HumanEval-Long gender distribution across dialect region.

5.4.4 User Interface for Evaluation

The evaluation User Interface (UI) was developed in gradio [80] by Janick Michot, a fellow master's student at the CAI, and adapted to this thesis's needs. Gradio

automatically creates a webpage on their domain with the content specified in the code. The evaluators were thus able to evaluate the samples from their preferred location.

Each evaluation consisted of a reference sample, a generated sample, and the synthesised text, as shown in Figure 5.20. For HumanEval-Short, the reference had the exact text as the generated sample and could thus be compared directly. No direct reference was available for HumanEval-Long as the sentences are either in part or entirely new. However, the voice adaptation could be evaluated as the speaker was still the same. The evaluators were informed of this difference when they started the HumanEval-Long evaluation.

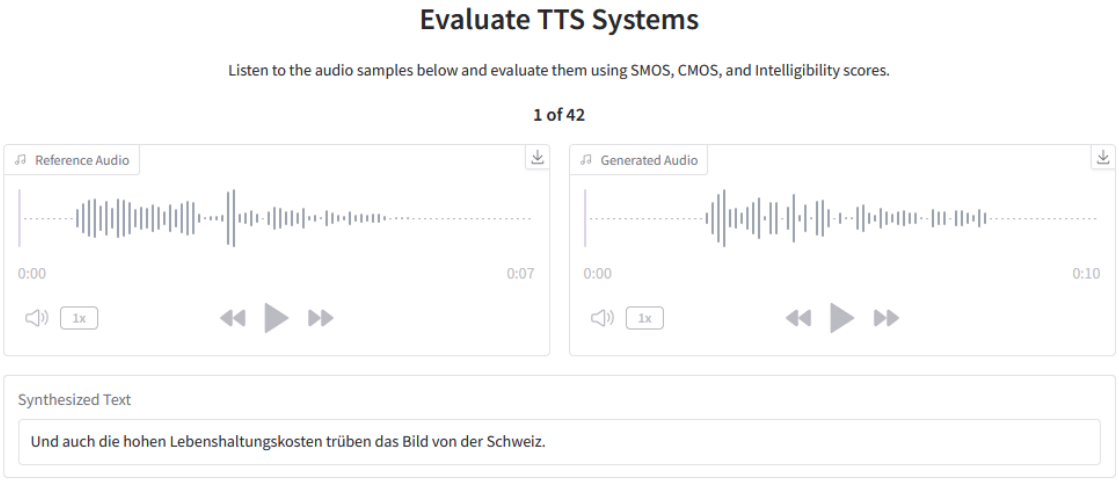

Figure 5.20: Human Evaluation UI hosted on gradio.

5.4.5 Statistical Significance

As with the automated setup, a homogeneity test using Levene [75] and a normality test using Shapiro-Wilk [76] is performed for each metric in each model comparison. If both variance and homogeneity do not fail ($p \leq 0.05$), a statistical significance test is run with One-Way ANOVA [77] and an additional pairwise t-test is executed using Tukey's HSD [78] if any of the values have p value of ≤ 0.05. In cases where tests for homogeneity or normality fail, the Kruskal-Wallis test [79] is applied and followed by the Wilcoxon Test for pairwise comparisons where appropriate ($p \leq 0.0167$ due to Bonferroni). If no specific pairwise comparisons are mentioned in the discussion, it should be assumed that the differences between the models were not statistically significant.

5.4.6 Results

The results will be provided on an evaluation group basis first, with the overall result being shown last.

Evaluation Group 1

The evaluation results of subset 1 in Table 5.18 indicate that Model Mixed+STT4SG-FT performed the strongest on short samples, with SMOS (3.62) and a very high Intelligibility (4.64).SRF+STT4SG-Mixed model, on the contrary, performed well on the long samples, as has been seen multiple times in the automated evaluation, confirming the findings there. The model did not perform as well on Intelligibility as the Mixed+STT4SG-FT model. The Baseline model STT4SG-Only, however, performed the weakest overall, with the lowest SMOS (2.86) and CMOS (-0.93) scores, indicating poorer similarity, quality and naturalness compared to the other models' voice adaptation despite maintaining decent Intelligibility scores.

Eval Type	Model	SMOS	CMOS	Intelligibility	Support
Short	SRF+STT4SG-Mixed	3.20±1.08	-0.61±1.07	3.54±1.53	28
	Mixed+STT4SG-FT	**3.62±0.74**	**-0.29±0.71**	**4.64±0.68**	28
	STT4SG-Only	3.27±0.69	-0.43±0.79	4.43±0.88	28
Long	SRF+STT4SG-Mixed	**4.05±0.58**	**-0.36±0.49**	4.32±0.86	28
	Mixed+STT4SG-FT	3.16±0.90	-0.46±0.64	**4.46±0.84**	28
	STT4SG-Only	2.86±0.81	-0.93±0.77	4.14±0.89	28

Table 5.18: Human evaluation result for evaluation subset 1

When testing the metrics for statistical significance, three were found to be below the threshold of $p = 0.05$. These were the Intelligibility of HumaEval-Short and the SMOS and CMOS metrics of HumanEval-Long, as listed in Table 5.19.

Eval Set	Metric	p-value	Significant ($p < 0.05$)?
Short	SMOS	0.1654	No
Short	CMOS	0.4976	No
Short	Intelligibility	0.0067	Yes → Do pairwise tests
Long	SMOS	0.0000	Yes → Do pairwise tests
Long	CMOS	0.0075	Yes → Do pairwise tests
Long	Intelligibility	0.3091	No

Table 5.19: Kruskal-Wallis Test Results for metrics in subset 1.

The pairwise statistical tests revealed that four pairs had significant differences, listed in Table 5.20. For the HumanEval-Short Intelligibility, the comparison between STT4SG-Only and Mixed+STT4SG-FT was significant with a p-value of 0.009581, indicating a notable difference in Intelligibility between these two models. In Long SMOS, both SRF+STT4SG-Mixed vs Mixed+STT4SG-FT (p = 0.001503) and SRF+STT4SG-Mixed vs STT4SG-Only (p = 0.000031) were significant. For Long CMOS, only the comparison between SRF+STT4SG-Mixed and STT4SG-Only was significant (p = 0.010620). Other comparisons showed no significant differences.

Eval-Set	Metric	Model A	Model B	p-value	Significant?
Short	Intelligibility	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.024028	No
Short	Intelligibility	SRF+STT4SG-Mixed	STT4SG-Only	1.0000	No
Short	Intelligibility	STT4SG-Only	Mixed+STT4SG-FT	0.009581	Yes
Long	SMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.001503	Yes
Long	SMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.000031	Yes
Long	SMOS	STT4SG-Only	Mixed+STT4SG-FT	0.236911	No
Long	CMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	1.000	No
Long	CMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.010620	Yes
Long	CMOS	STT4SG-Only	Mixed+STT4SG-FT	0.055831	No

Table 5.20: Pairwise Comparisons for Subset 1 Test Results for each metric. Bonferroni p-value of $\leq$0.0167 was used to determine significance.

Evaluation Group 2

In evaluation subgroup 2, the evaluation results in Table 5.21 indicate that Model SRF+STT4SG-Mixed consistently outperformed the other models in both short and long samples. For short samples, it achieved a SMOS score of 3.59 and a CMOS of -0.46, indicating average similarity to the original speaker and a good Intelligibility score of 4.46. Model Mixed+STT4SG-FT performed better in Intelligibility (4.64) than the SRF+STT4SG-Mixed model but had a lower SMOS of 3.36. Model 9-2, while still intelligible (4.04), considerably lagged in both SMOS (3.34) and CMOS (-1.11), suggesting it was significantly less natural and similar compared to the original voice.

Model SRF+STT4SG-Mixed remained strong in the long samples with a SMOS of 4.23, a CMOS of -0.14, and an Intelligibility score of 4.21. Especially the CMOS score of -0.14 can be considered very good, as the model produced very human-like speech, even when compared to the reference sample. Model 7-6 SNF performed similarly with a SMOS of 3.75 and good Intelligibility (4.25), but Baseline Model STT4SG-Only again showed the weakest performance, especially in Intelligibility (3.89 $\pm$ 0.79), along with lower SMOS and CMOS scores.

Overall, Model SRF+STT4SG-Mixed performed best across short and long samples, while Model 9-2 had relatively lower similarity and naturalness.

Eval Type	Model	SMOS	CMOS	Intelligibility	Support
Short	SRF+STT4SG-Mixed	**3.59±1.11**	**-0.46±0.79**	4.46±0.74	28
	Mixed+STT4SG-FT	3.36±0.68	-0.61±0.63	**4.64±0.56**	28
	STT4SG-Only	3.34±0.68	-1.11±0.92	4.04±1.00	28
Long	SRF+STT4SG-Mixed	**4.23±0.50**	**-0.14±0.80**	4.21±0.83	28
	Mixed+STT4SG-FT	3.75±0.62	-0.61±0.69	**4.25±0.59**	28
	STT4SG-Only	3.62±0.66	-0.71±0.66	3.89±0.79	28

Table 5.21: Human evaluation result for evaluation subset 2

The Kruskal-Wallis test results, given in Table 5.22 show significant differences in Short CMOS ($p = 0.0079$), Short Intelligibility ($p = 0.0370$), Long SMOS ($p = 0.0001$), and Long CMOS ($p = 0.0076$), indicating that pairwise comparisons should be conducted for these metrics.

Eval Set	**Metric**	**p-value**	**Significant ($p < 0.05$)?**
Short	SMOS	0.1108	No
Short	CMOS	0.0079	Yes → Do pairwise tests
Short	Intelligibility	0.0370	Yes → Do pairwise tests
Long	SMOS	0.0001	Yes → Do pairwise tests
Long	CMOS	0.0076	Yes → Do pairwise tests
Long	Intelligibility	0.1478	No

Table 5.22: Kruskal-Wallis Test Results for metrics in subset 2.

The Wilcoxon test results, listed in Table 5.23, revealed significant results for Long SMOS, where the model SRF+STT4SG-Mixed showed significant differences when compared to both Mixed+STT4SG-FT ($p = 0.010837$) and the Baseline STT4SG-Only ($p = 0.001534$). However, no significant differences were found for the other comparisons. Specifically, in Short CMOS and Short Intelligibility, no comparisons showed significance after applying the Bonferroni correction, as all p-values exceeded the threshold of 0.0167. Similarly, Long CMOS comparisons did not yield significant results, with all p-values above the threshold.

Eval-Set	**Metric**	**Model A**	**Model B**	**p-value**	**Significant?**
Short	CMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	1.0000	No
Short	CMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.044229	No
Short	CMOS	STT4SG-Only	Mixed+STT4SG-FT	0.097771	No
Short	Intelligibility	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	1.0000	No
Short	Intelligibility	SRF+STT4SG-Mixed	STT4SG-Only	0.490733	No
Short	Intelligibility	STT4SG-Only	Mixed+STT4SG-FT	0.048633	No
Long	SMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.010837	Yes
Long	SMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.001534	Yes
Long	SMOS	STT4SG-Only	Mixed+STT4SG-FT	0.731347	No
Long	CMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.114320	No
Long	CMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.026137	No
Long	CMOS	STT4SG-Only	Mixed+STT4SG-FT	1.0000	No

Table 5.23: Pairwise Comparisons for Subset 2 Test Results for each metric. Bonferroni p-value of ≤ 0.0167 was used to determine significance.

Evaluation Group 3

The Table 5.24 given evaluation results for subset 3 indicate that SRF+STT4SG-Mixed performed the best overall. It had a relatively strong CMOS score of -0.57 in Short evaluation and -0.36 in Long evaluation, suggesting it was pretty natural and close to the reference. In comparison, Mixed+STT4SG-FT and STT4SG-Only

had lower scores in both SMOS and CMOS, with STT4SG-Only showing the weakest performance across these metrics. For the long samples, SRF+STT4SG-Mixed performed best in SMOS and CMOS, but all three models showed a further decline in naturalness and similarity. However, contrary to the other models, the SRF+STT4SG-Mixed seems to have had considerable Intelligibility issues. This may be due to stuttering in speech or incorrect pronunciation of the model. It has to be noted that this subset had significantly worse scores than the other two subsets across all metrics, which may be due to specific voice-adapting speaker and sentence setups.

Eval Type	Model	SMOS	CMOS	Intelligibility	Support
Short	SRF+STT4SG-Mixed	**3.39±0.95**	**-0.57±0.69**	3.54±1.29	28
	Mixed+STT4SG-FT	2.75±1.02	-0.68±1.09	**4.25±0.75**	28
	STT4SG-Only	2.70±0.97	-0.86±1.11	4.04±0.92	28
Long	SRF+STT4SG-Mixed	**3.14±0.99**	**-0.36±0.87**	3.79±0.92	28
	Mixed+STT4SG-FT	2.55±0.82	-1.00±0.82	**4.14±0.65**	28
	STT4SG-Only	2.45±0.89	-1.18±0.90	3.86±0.89	28

Table 5.24: Human evaluation result for evaluation subset 3

Table 5.25 lists the Kruskal-Wallis results indicate no significant differences for Short SMOS, Short CMOS, and Short Intelligibility, as their p-values were above the 0.05 threshold. In contrast, Long SMOS ($p = 0.0109$) and Long CMOS ($p = 0.0033$) showed significant differences, warranting further pairwise comparisons. No significant difference was found for Long Intelligibility ($p = 0.2341$).

Eval Set	Metric	p-value	Significant ($p < 0.05$)?
Short	SMOS	0.0154	No
Short	CMOS	0.4524	No
Short	Intelligibility	0.1002	No
Long	SMOS	0.0109	Yes → Do pairwise tests
Long	CMOS	0.0033	Yes → Do pairwise tests
Long	Intelligibility	0.2341	No

Table 5.25: Kruskal-Wallis Test Results for metrics in subset 3.

The Wilcoxon test results revealed a significant difference in Long CMOS between SRF+STT4SG-Mixed and STT4SG-Only ($p = 0.007207$), indicating that these two models differ in naturalness. All other comparisons, including those for Long SMOS and the remaining Long CMOS comparisons, were insignificant, with p-values above the threshold of 0.0167.

Overall

The overall performance of the models is now discussed. Table 5.27 shows the breakdown of scores on the short and long samples and their combined results. This

Eval-Set	Metric	Model A	Model B	p-value	Significant?
Long	SMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.025732	No
Long	SMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.019148	No
Long	SMOS	STT4SG-Only	Mixed+STT4SG-FT	1.000000	No
Long	CMOS	SRF+STT4SG-Mixed	Mixed+STT4SG-FT	0.040378	No
Long	CMOS	SRF+STT4SG-Mixed	STT4SG-Only	0.007207	Yes
Long	CMOS	STT4SG-Only	Mixed+STT4SG-FT	0.960838	No

Table 5.26: Pairwise Comparisons for Subset 3 Test Results for each metric. Bonferroni p-value of $\leq$0.0167 was used to determine significance.

section will only discuss the total scores, with the evaluation types provided as additional insight.

For Similarity Mean Opinion Score, the SRF+STT4SG-Mixed model has the highest score of 3.60, indicating that, on average, this model had better voice adaptation according to the evaluators, though there is moderate variability. The Mixed+STT4SG-FT model follows with a score of 3.20, showing a slightly lower performance, while the Baseline STT4SG-Only model ranks the lowest with a score of 3.04. Regarding Comparative Mean Opinion Score, the SRF+STT4SG-Mixed model again leads with a score of -0.42, indicating a relatively mild degradation compared to the reference samples. The Mixed+STT4SG-FT and STT4SG-Only models show lower scores, -0.61 and -0.87, respectively, illustrating greater perceived quality degradation than the SRF+STT4SG-Mixed model.

For Intelligibility, the Mixed+STT4SG-FT model outperforms the others with a score of 4.40, higher than SRF+STT4SG-Mixed (3.98) and STT4SG-Only (4.07). This suggests that, while SRF+STT4SG-Mixed might have a higher overall SMOS and CMOS rating, the Mixed+STT4SG-FT model is perceived as clearer and more intelligible. This directly correlates with the automated evaluation in Section 5.3, where Mixed+STT4SG-FT was the best-performing model overall.

In summary, the SRF+STT4SG-Mixed model shows the highest SMOS and performs reasonably well regarding Intelligibility and CMOS, but the Mixed+STT4SG-FT model outperforms it in terms of Intelligibility. The Baseline STT4SG-Only model appears to lag behind in all evaluated metrics. The consistent support of 168 across all models helps to ensure the robustness of these results.

Eval Type	Model	SMOS	CMOS	Intelligibility	Support
Short	SRF+STT4SG-Mixed	**3.39±1.05**	-0.55±0.86	3.85±1.29	84
	Mixed+STT4SG-FT	3.24±0.90	**-0.52±0.84**	**4.51±0.69**	84
	STT4SG-Only	3.10±0.83	-0.80±0.98	4.17±0.94	84
Long	SRF+STT4SG-Mixed	**3.81±0.86**	**-0.29±0.74**	4.11±0.89	84
	Mixed+STT4SG-FT	3.15±0.92	-0.69±0.74	**4.29±0.70**	84
	STT4SG-Only	2.98±0.92	-0.94±0.80	3.96±0.86	84
Total	SRF+STT4SG-Mixed	**3.60±0.98**	**-0.42±0.81**	3.98±1.12	168
	Mixed+STT4SG-FT	3.20±0.91	-0.61±0.80	**4.40±0.70**	168
	STT4SG-Only	3.04±0.88	-0.87±0.89	4.07±0.90	168

Table 5.27: Human evaluation result for overall performance of the models

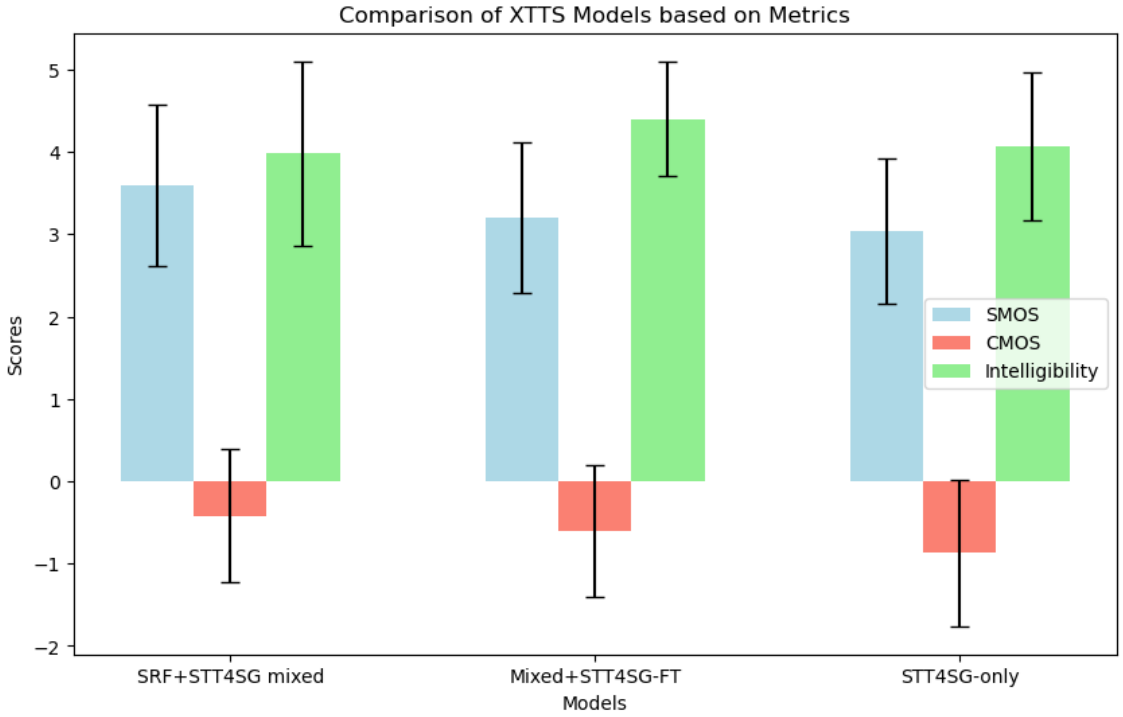

Figure 5.21: Score distribution across models and evaluation subsets and types

Dialect Performance

Inspection of the dialect regions over shows the continued supremacy of the Mixed+STT4SG-FT model over SMOS, seen in Figure 5.22, and CMOS, illustrated in Figure 5.23. Only in the Wallis region was the model outperformed on the CMOS by the Mixed+STT4SG-FT. Intelligibility was, as expected, dominated by the Mixed+STT4SG-FT model on all seven regions, visualized in Figure 5.24. This reflects the overall strength of the model in producing intelligible speech.

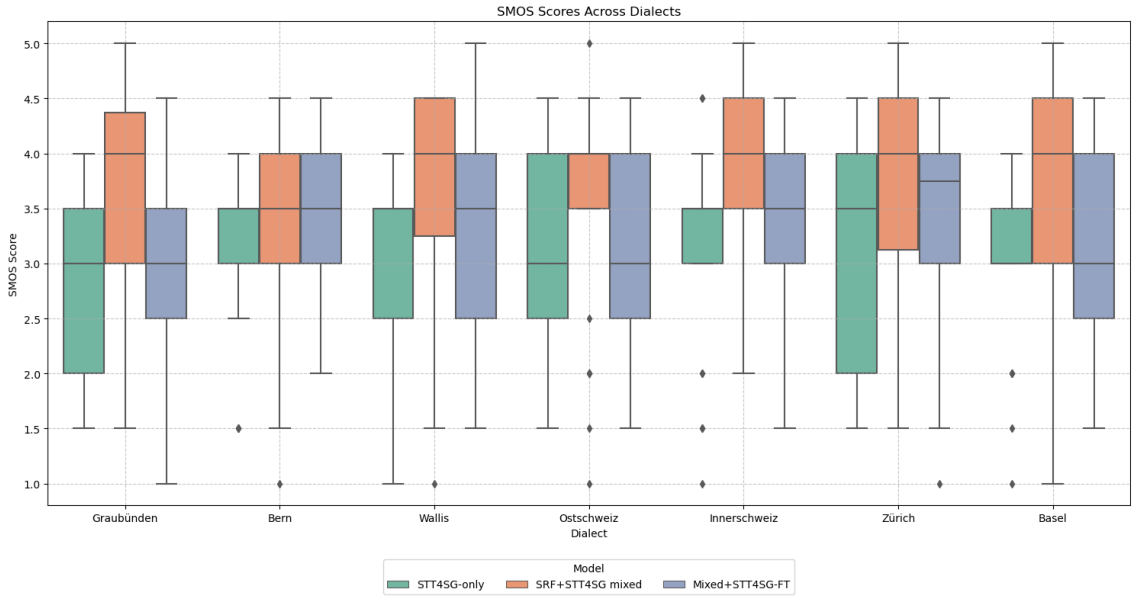

Figure 5.22: SMOS score distribution across dialect regions and models.

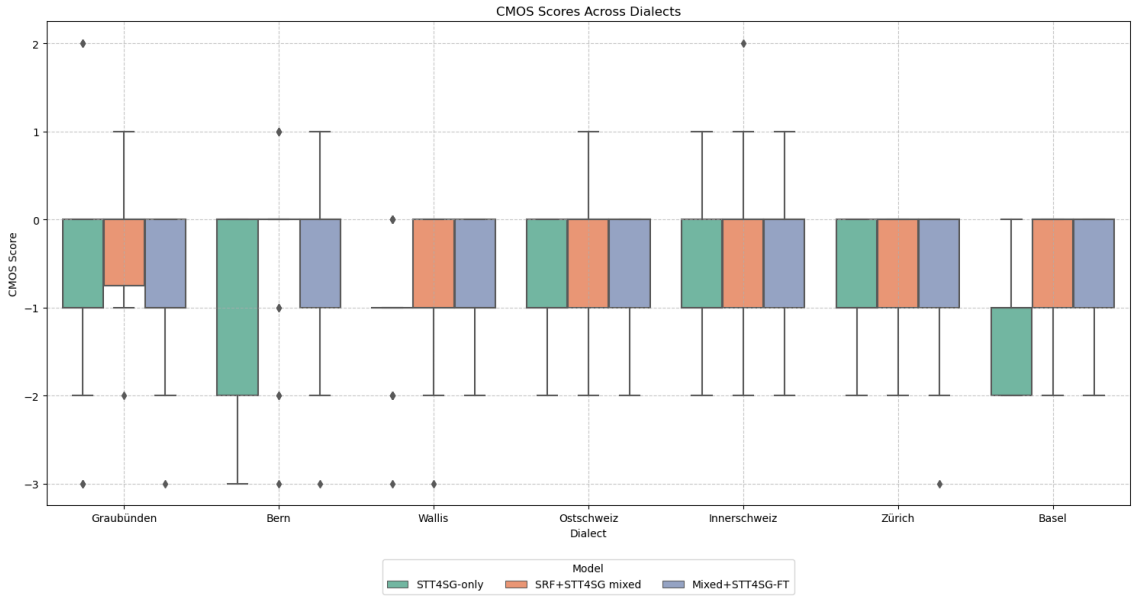

Figure 5.23: CMOS score distribution across dialect regions and models.

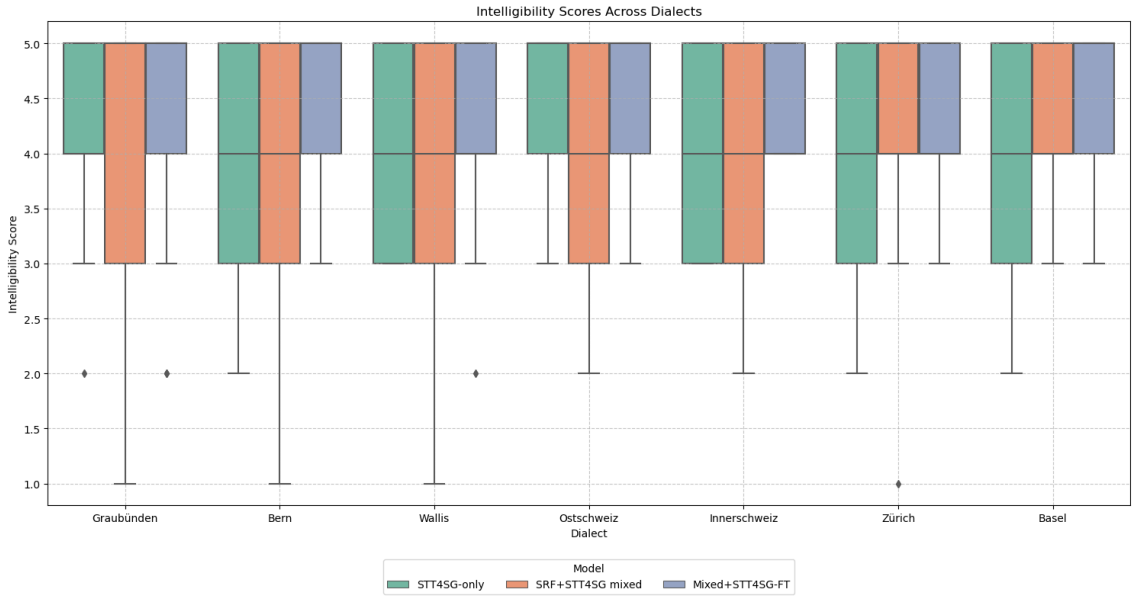

Figure 5.24: Intelligibility score distribution across dialect regions and models.

Speaker Performance

It is also important to examine the performance of the individual synthesised speakers used in this evaluation. All speakers from the STT4SG-350 [10] that were used in the human evaluation are listed in Table 5.28 with them ordered by the SMOS column. The scores are averages over all three models' complete human evaluation output. As the evaluation is not trivial due to various potential impacts from synthesised sentences, audio quality, general model performance, training data, and more, much time must be invested into analysing this. Due to time constraints, no specific analysis could be made on a deeper level. Future work may investigate this further. However, a few inspections were made to provide some general insight into the Voice Adaption performance of the models. First, in terms of the distribution of speakers across dialects and their performance, no apparent underperforming region could be found apart from Intelligibility in the Wallis region, seen in the Figures 5.25, 5.26, and 5.27 respectively.

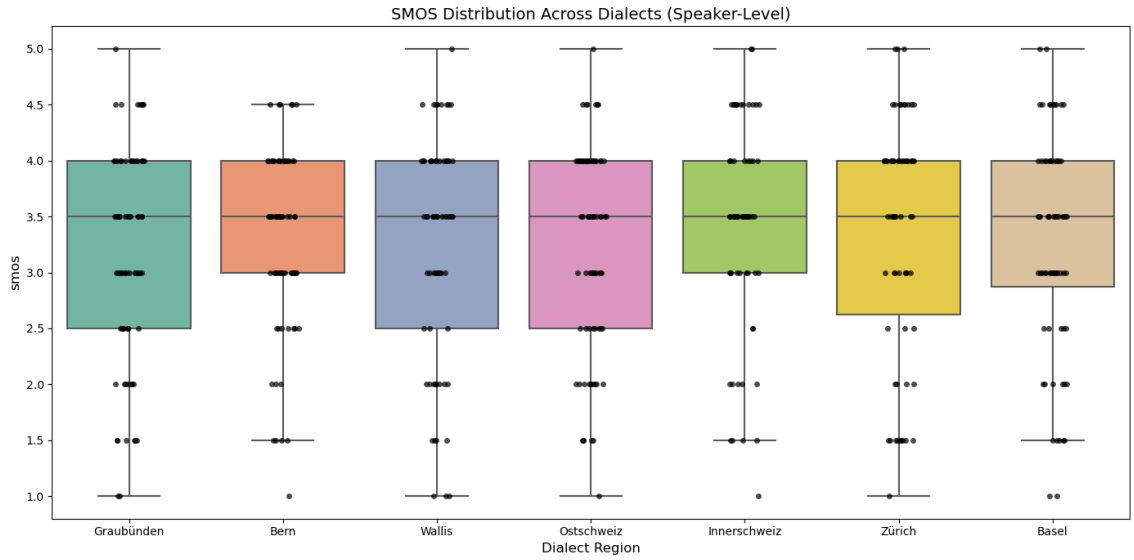

Figure 5.25: SMOS score distribution across dialect regions and models, showcasing specific speaker valuations

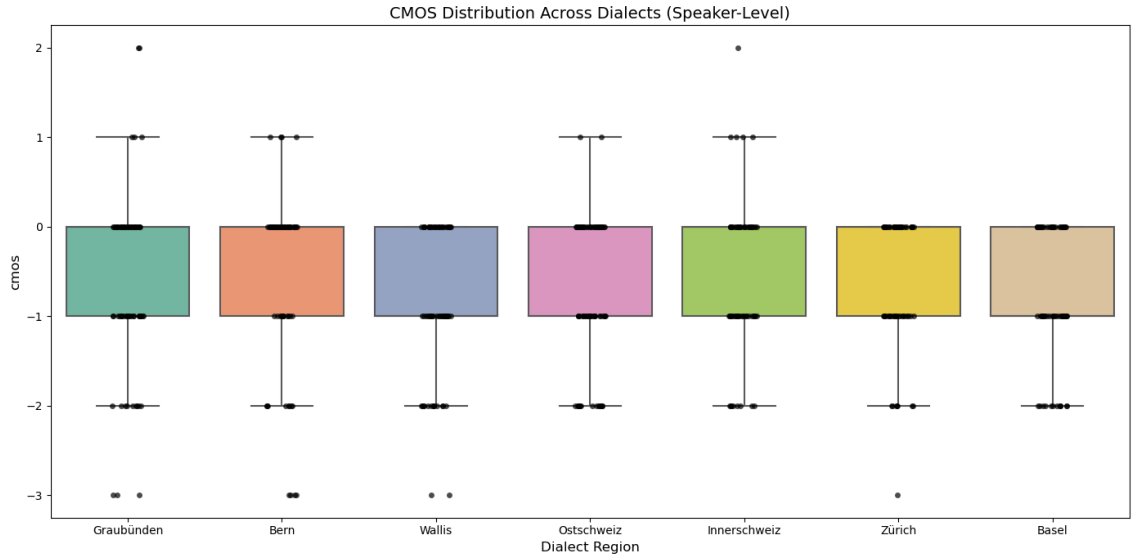

Figure 5.26: CMOS score distribution across dialect regions and models, showcasing specific speaker valuations

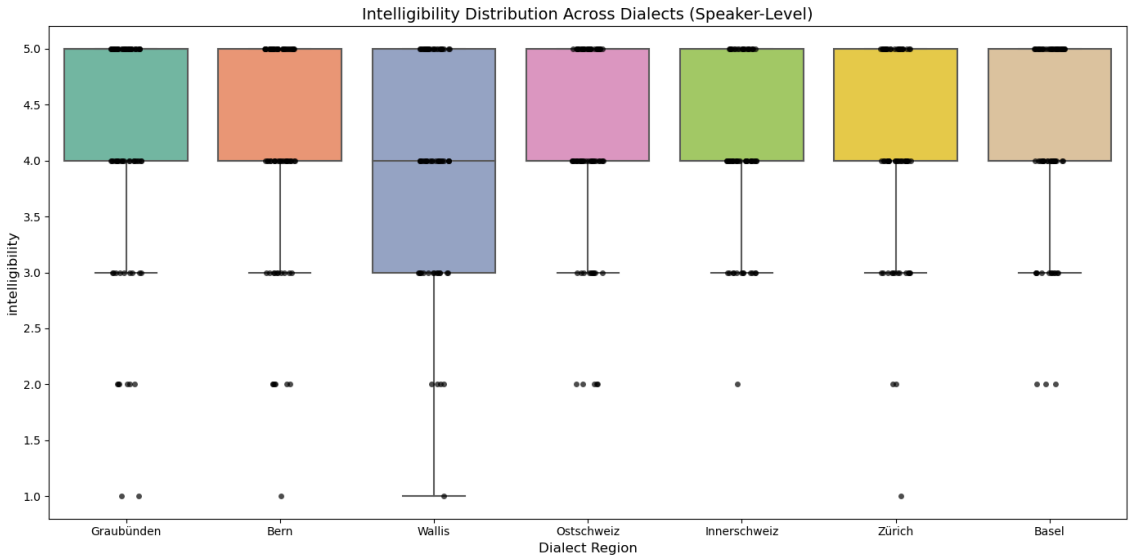

Figure 5.27: Intelligibility score distribution across dialect regions, showcasing specific speaker valuations

Next, to gain insight into the considerable variation in the speaker level, three speakers are investigated and cross-checked on the output of dataspeech [70], which was created during the voice adaptation selection in Section 5.1. First, the speaker 6d34bfbb-2de2-48d9-891d-ae216f9b347b from dialect region Zurich possesses the second highest SMOS score (4.11), a perfect Intelligibility score of 5.0 and a good CMOS score of -0.22. Dataspeech did not provide clear answers to the apparent good performance of this specific speaker. Their condition samples were categorized as good (2) or average (3). However, the samples in the test split of the STT4SG-350 had 52 out of 368 samples labelled as of bad quality. Normally, such a large bad sample size hinted towards lower-quality audio.

Next, ba118975-5963-4495-927a-a78d19dd98c1 from dialect region Bern was investigated. They had the second lowest SMOS score (2.33) and the worst CMOS score (-2.00). Dataspeech did provide correct insight into this, as out of the total 291 sentences in the test-split, 179 were considered average, 111 were considered bad, and one was considered good.

Lastly, the speaker 684dd9cf-2844-407b-9a7f-12e7b559773f from the dialect region Ostschweiz was analysed. They were average in the evaluation with an SMOS of 3.18, a CMOS of -0.55m, and an Intelligibility score of 4.29. All of their conditioning samples were considered good, and of the overall 358, 315 were considered average, three were considered bad, and 40 were good. This distribution was completely average and somewhat explained their overall performance.

While this analysis is not enough to explain the variations, it does hint towards an issue with the general audio quality of the reference speakers.

Speaker	Dialect	SMOS	CMOS	Intelligibility	Support
8800b4de-fa22-4fdc-9f29-457c4010fd57	Basel	4.12±0.53	-0.75±0.62	3.83±1.03	12
6d34bfbb-2de2-48d9-891d-ae216f9b347b	Zürich	4.11±0.65	-0.22±0.44	5.00±0.00	9
005039b8-898f-48d8-b7cc-8a16bd1055c8	Innerschweiz	3.89±0.55	-0.11±0.93	4.22±1.20	9
ce8839ae-3b20-491c-8c33-cef08b4def6e	Zürich	3.88±0.67	-0.19±0.40	4.29±0.72	21
831d404d-b189-42b6-8120-073c1f8af73c	Ostschweiz	3.83±0.71	-0.56±0.73	4.33±1.00	9
031b0a74-5bdd-47e7-b8b7-9bb58d0e8c72	Graubünden	3.83±0.79	0.33±1.12	4.44±0.53	9
acf67674-c912-42c0-ba3c-f85e2db965ac	Innerschweiz	3.70±0.85	-0.37±0.84	4.22±0.75	27
1cd99b41-8298-4180-aec6-bb65039c9ed7	Basel	3.58±0.74	0.00±0.00	5.00±0.00	6
50e4a935-c887-4c38-aff2-6286aab725d1	Wallis	3.58±0.90	-0.69±0.72	4.17±0.88	42
d2dee463-0eb9-47fa-b739-f1dccd8638f9	Bern	3.57±0.55	-0.48±0.70	4.63±0.56	27
fd72db57-c291-43e6-840e-92e06f83ae56	Basel	3.53±0.85	-0.87±0.74	4.13±0.74	15
6516567b-0d9b-4853-880c-d5f0327dd384	Graubünden	3.50±0.74	-0.55±0.75	4.39±0.83	33
0497b106-6644-42ce-b99e-57f9a6c7fc81	Bern	3.33±1.26	-0.67±1.15	4.33±0.58	3
a677eda9-7709-4c9d-8656-e7b665140f3b	Ostschweiz	3.28±1.10	-0.78±0.89	4.07±1.00	27
d9b44aee-da3d-42c8-8ad1-d1029767f05a	Bern	3.22±0.57	-0.44±0.73	3.44±1.13	9
684dd9cf-2844-407b-9a7f-12e7b559773f	Ostschweiz	3.18±0.72	-0.55±0.74	4.29±0.83	42
7ca44480-9007-4c1c-8046-aade3c4e6a87	Bern	3.17±0.90	-0.33±1.02	4.06±1.12	33
aece75d7-5d2b-47a4-9f87-24962dfd2e38	Graubünden	3.08±1.11	-1.33±1.03	4.50±1.22	6
8050767b-0a0e-43db-8754-2a42e896f7dd	Basel	3.06±0.77	-0.67±0.87	4.11±1.05	9
5184dba3-c5e1-4570-ba26-62176e8c8dcc	Innerschweiz	3.03±1.04	-0.89±0.83	4.06±0.64	18
ba826a45-33f8-47ef-9516-a66a201aac29	Zürich	3.02±1.19	-0.67±0.73	3.57±1.03	21
4a8346e7-fa21-49b9-9a6f-c69ef828a68c	Innerschweiz	3.00±1.09	-0.67±0.71	3.89±0.60	9
0b890339-031a-43ef-bc2e-5e8b5ac5e613	Wallis	2.97±0.93	-0.73±0.88	3.87±1.30	15
12fb73be-cf60-4794-befb-381682ccda9a	Basel	2.73±0.96	-0.77±0.77	4.37±0.81	30
c4f6bcdf-fc02-4fe8-9277-0701cfffebab	Graubünden	2.62±0.90	-0.73±0.98	3.77±1.30	30
4ee1811c-9884-4261-99cb-2f7346c8ea6e	Zürich	2.60±0.97	-1.27±0.80	3.93±0.80	15
fb1b67be-8d8f-47bb-b15c-ee1138f0d4ac	Wallis	2.46±0.94	-1.67±0.65	3.75±0.87	12
ba118975-5963-4495-927a-a78d19dd98c1	Bern	2.33±1.04	-2.00±1.73	2.67±0.58	3
72911186-8b0e-4c0b-af3d-4d8765072930	Ostschweiz	2.33±1.44	-1.67±0.58	4.00±0.00	3

Table 5.28: Human evaluation result for overall performance of the models

General comments

Some evaluators left comments after the evaluation process was finished. Specifically, two were surprised by the accurate placements of interjections such as "ähm" and "also" on pauses in sentences such as commas or full stops. Additionally, five evaluators noted the general slowness of many samples in the evaluation, resulting in deeper or entirely different speech patterns. The authors confirmed this, which needs to be investigated in future work.

Chapter 6

Discussion and Outlook

The contributions of this thesis are threefold. Firstly, it introduces a data pipeline to automatically download, diarise, segment, and weakly label Swiss German audio from YouTube and SRF. The pipeline utilises whisper [42] for audio to Standard German transcription, the Wav2Vec2 phoneme model by Xu et al. [57] for phoneme transcription, and an internally developed Naive Bayes classifier by Bolliger et al. [58] to categorise the audio into one of seven dialect regions defined in [10]. Next, a SRF-corpus based on various podcasts and broadcasts by the Schweizer Radio und Fernsehen and a selected few podcasts from YT, comprising in total around 5000 hours of Swiss German dialects and Standard German speech. The corpus is unbalanced on a dialect basis, and future work should investigate ways to improve this. While the corpus can not be released to the broader public due to lacking clearance with the concerned parties, we provide a blueprint on how and where such data can be sourced and utilised for Swiss German ASR-systems. Lastly, two on the STT4SG-350 [10][11] and SRF-corpus fine-tuned Swiss German XTTS [9] models capable of applying ZS-TTS on the seven dialect regions are presented, which were compared to a baseline model fine-tuned solely on the STT4SG corpus. The exploratory analysis of the performance of these models gave valuable insight into the capabilities of Swiss German ZS voice adaptation.

Executing the data pipeline on the SRF-corpus showed that the performance of the various models and enrichment processes on uncontrolled and more chaotic speech than found in professional settings such as [10] or [11] was comparable to said settings, showcasing the strength of these existing models more generally for Swiss German. However, the decision not to segment longer segments into sentence-basis leads to an unusually large concentration of samples with a length of 15 seconds, which has degraded the XTTS model's performance on shorter sentences, as shown in the downstream evaluations. Future work may investigate the impact the SRF-corpus would have if these segments were shortened using libraries applying translation with forced alignments, such as whisper-x [42].

The three models utilized in this thesis had different characteristics; the SRF+STT4SG-Mixed was trained for nine epochs / 238k steps on the mixed corpora of STT4SG-350 and SRF. The Mixed+STT4SG-FT utilized the checkpoint of SRF+STT4SG-Mixed

and fine-tuned for 26 epochs on [10] exclusively. Lastly, the Baseline STT4SG-Only was trained for 25 epochs on the STT4SG-350 corpus. A broad and in-depth evaluation of these three models showcased the strengths and weaknesses of the models and the corpora.

For the automated evaluation of generated sentence accuracy, primarily WER and CER were investigated, with both BERTScore and BLEU metrics being provided for a more rounded understanding of the results. All metrics were evaluated using the lowered text, as case-sensitive spelling was not the focus of this thesis; instead, the system's overall potential in Swiss German dialects was. Multiple sub-evaluations were executed with different characteristics. First, the SNF-Short evaluation primarily targeted short samples of one sentence with between 7 and 14 tokens spoken by 43 never-before-seen speakers. All three models exhibited mediocre performance at best. The Mixed+STT4SG-FT exhibited the best WER and CER with 0.523 and 0.24, respectively. The SRF+STT4SG-Mixed model instead had the best BLEU of 0.288. Particular dialects performed better than others, with the Innerschweiz region performing the best (best model WER: 0.364, CER: 0.166, BERTScore: 0.905, BLEU: 0.401) and Wallis the worst (best model WER: 0.629, CER: 0.300, BERTScore: 0.796, BLEU: 0.133). This setup was hugely influenced by the chosen speakers and the small audio size, with the additional hurdle of whisper potentially performing worse than usual due to large padding sequences in these small audio samples. The second evaluation, SNF-Long, targeted the same speakers with the same but more extended sentence structures of between 15 and 26 tokens. All three models exhibited better performance, improving by up to 0.263 WER and 0.174 CERT in the case of SRF+STT4SG-Mixed to 0.413 WER, 0.206 CER, and BLEU of 0.449. This confirmed suspicions that the models had difficulties generating audio with short segments, irrespective of the speaker conditioning audio quality. The performance on dialects continued to show significant variance, with Graubünden performing best (WER: 0.251, CER: 0.112, BERScore: 0.914, BLEU 0.620) and Wallis still exhibiting difficulties (WER: 0.537, CER: 0.304, BERTScore: 0.833, BLEU: 0.346).

The third experiment, GPT-Random, utilizing only a single speaker originally from Innerschweiz with confirmed high-quality audio, generated audio for 100 sentences with token counts between 7 and 49 in all seven dialect regions. This provided more insight into dialect performance as it was not dependent on Zero-Shot speaker audio quality. Evaluation of the three models showcased performance improvements in all, with the Mixed+STT4SG-FT performing best overall with a WER of 0.238, CER of 0.123, BERTScore of 0.937, and a BLEU of 0.610. Analysis of CER distributions of the models again proved the suspected long sample issue in the SRF-corpus, with the SRF+STT4SG-Mixed exhibiting huge variations on smaller samples but very stable results on long segments. At the same time, the Baseline and Mixed+STT4SG-FT also showed more variation for short samples but were way less significant. This also resulted in all dialect regions performing better. Wallis still performed the worst (WER: 0.386, CER: 0.212, BERTScore: 0.891, BLEU: 0.439), and Innerschweiz by far the best with a surprising 0.150 WER, 0.075 CER,

0.962 BERTScore, and BLEU of 0.738. In the last experiment, GPT-Long reused the previous 43 speakers but increased their sample size to 30 sentences with between 21 and 48 tokens. Mixed+STT4SG-FT solidified its performance there with a WER of 0.233, CER of 0.138, 0.937 BERTScore, and a BLEU of 0.633. The performance over all four evaluation sets also confirmed Mixed+STT4SG-FT as the best performing model, including on an on-dialect-basis, with a WER of 0.259, CER of 0.141, 0.929 BERTScore, and a BLEU of 0.594.

Unexpectedly, in all of the experiments, Zurich often performed average or worse compared to other dialects depending on the setup, even though it had the largest dialect share of all regions in the SRF-corpus, excluding Standard German. Future work should investigate this further.

The Dialect Identification evaluation was split into two categories for short sentences and long sentences, reusing the 43 speakers. The Baseline considerably beat the SRF trained models on the short segments, reaching a macro F1 of 0.4823. Especially the dialect regions of Graubünden, Ostschweiz, and Wallis could be accurately reproduced. The evaluation of DID on longer segments was unbalanced on both dialect and speaker-basis due to the nature of GPT-Long's definition. Future work may look into more specific DID, evaluating single-speaker adaptation to dialects (similar to GPT-Random setup) and a balanced speaker evaluation from each region. Nonetheless, the SRF-models significantly improved compared to the short segments. SRF+STT4SG-Mixed achieved a weighted F1 of 0.7876, and the Mixed+STT4SG-FT 0.7027, while the Baseline improved moderately to 0.5868. The SRF considerably helped the models reproduce the seven dialect regions with longer sentence structures.

As the automated evaluation of human speech is flawed, a human evaluation was also performed using SMOS, CMOS, and Intelligibility for metrics. For short and long text segments, the SRF+STT4SG-Mixed beat the other models in SMOS, reaching an overall 3.60 compared to 3.20 and 3.04 for Mixed+STT4SG-FT and STT4SG-Only, respectively. For CMOS, the SRF+STT4SG-Mixed also performed best, reaching -0.42. For Intelligibility, the Mixed+STT4SG-FT achieved the highest overall score, 4.40. This may mean that the models still have issues accurately reproducing the speaker's characteristics. However, the variations are huge when investigating the 30 speakers judged in total. Specific speakers had excellent scores across all three models and both evaluations, such as 6d34bfbb-2de2-48d9-891d-ae216f9b347b with a SMOS of 4.11, a CMOS of -0.22 and a perfect intelligibility score of 5.0.

The analysis provided here is by no means complete, and the results showcased here are also not perfect; however, as none of the specific parts were optimized and primarily done on an exploratory nature, the future for Swiss German Zero-Shot voice adaptation TTS system seems very promising. Future work may investigate further in all three domains. The data pipeline should be revised, and experiments should be conducted with samples at the sentence level only. The SRF-dataset should incor-

porate particular dialects more, such as Wallis, which is mainly under-represented therein. The training can optimized, either longer or with a more controlled mix of the dialects. Evaluation should use better speakers where audio quality is not a concern. Contrarily, the training and evaluation could investigate ways to offset audio quality issues in condition samples of speakers to develop a robust and future-proof Zero-Shot-TTS model.

Bibliography

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., Jun. 2017.

[2] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 17 022–17 033. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/c5d736809766d46260d816d8dbc9eb44-Paper.pdf

[3] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, vol. 1, 1996, pp. 373–376 vol. 1.

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards End-to-End Speech Synthesis," in *Interspeech 2017*, 2017, pp. 4006–4010.

[5] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," in *9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, 2016, p. 125.

[6] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/6832a7b24bc06775d02b7406880b93fc-Paper.pdf

[7] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "YourTTS: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato,

Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 2709–2720. [Online]. Available: https://proceedings.mlr.press/v162/casanova22a.html

[8] C. Paonessa, Y. Schraner, J. Deriu, M. Hürlimann, M. Vogel, and M. Cieliebak, "Dialect transfer for Swiss German speech translation," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 15 240–15 254. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.1018/

[9] E. Casanova, K. Davis, E. Gölge, G. Göknar, I. Gulea, L. Hart, A. Aljafari, J. Meyer, R. Morais, S. Olayemi, and J. Weber, "XTTS: a Massively Multilingual Zero-Shot Text-to-Speech Model," in *Interspeech 2024*, 2024, pp. 4978–4982.

[10] M. Plüss, J. Deriu, Y. Schraner, C. Paonessa, J. Hartmann, L. Schmidt, C. Scheller, M. Hürlimann, T. Samardžić, M. Vogel, and M. Cieliebak, "STT4SG-350: A Speech Corpus for All Swiss German Dialect Regions," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 1763–1772. [Online]. Available: https://aclanthology.org/2023.acl-short.150/

[11] P. Dogan-Schönberger, J. Mäder, and T. Hofmann, "SwissDial: Parallel Multidialectal Corpus of Spoken Swiss German," *CoRR*, vol. abs/2103.11401, 2021. [Online]. Available: https://arxiv.org/abs/2103.11401

[12] M. Plüss, L. Neukom, and M. Vogel, "Swiss Parliaments Corpus, an Automatically Aligned Swiss German Speech to Standard German Text Corpus," *CoRR*, vol. abs/2010.02810, 2020. [Online]. Available: https://arxiv.org/abs/2010.02810

[13] M. Plüss, M. Hürlimann, M. Cuny, A. Stöckli, N. Kapotis, J. Hartmann, M. A. Ulasik, C. Scheller, Y. Schraner, A. Jain, J. Deriu, M. Cieliebak, and M. Vogel, "SDS-200: A Swiss German Speech to Standard German Text Corpus," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, and S. Piperidis, Eds. Marseille, France: European Language Resources Association, Jun. 2022, pp. 3250–3256. [Online]. Available: https://aclanthology.org/2022.lrec-1.347/

[14] C. Sicard, V. Gillioz, and K. Pyszkowski, "Spaiche: Extending State-of-the-Art ASR Models to Swiss German Dialects," in *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, H. Ghorbel, M. Sokhn, M. Cieliebak, M. Hürlimann, E. de Salis, and J. Guerne, Eds. Neuchatel, Switzerland: Association for Computational Linguistics, Jun. 2023, pp. 76–83. [Online]. Available: https://aclanthology.org/2023.swisstext-1.8/

[15] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust Speech Recognition via Large-Scale Weak Supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356

[16] T. Bollinger, J. Deriu, and M. Vogel, "Text-to-Speech Pipeline for Swiss German – A comparison," 2023. [Online]. Available: https://arxiv.org/abs/2305.19750

[17] J. Kim, J. Kong, and J. Son, "Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech," 2021. [Online]. Available: https://arxiv.org/abs/2106.06103

[18] V. Timmel, C. Paonessa, R. Kakooee, M. Vogel, and D. Perruchoud, "Fine-tuning Whisper on Low-Resource Languages for Real-World Applications," 2024. [Online]. Available: https://arxiv.org/abs/2412.15726

[19] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Neural Codec Language Models are Zero-Shot Text to Speech Synthesizers," 2023. [Online]. Available: https://arxiv.org/abs/2301.02111

[20] S. Chen, S. Liu, L. Zhou, Y. Liu, X. Tan, J. Li, S. Zhao, Y. Qian, and F. Wei, "VALL-E 2: Neural Codec Language Models are Human Parity Zero-Shot Text to Speech Synthesizers," 2024. [Online]. Available: https://arxiv.org/abs/2406.05370

[21] Z. Zhang, L. Zhou, C. Wang, S. Chen, Y. Wu, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Speak Foreign Languages with Your Own Voice: Cross-Lingual Neural Codec Language Modeling," 2023. [Online]. Available: https://arxiv.org/abs/2303.03926

[22] F. Lux, J. Koch, and N. T. Vu, "Low-Resource Multilingual and Zero-Shot Multispeaker TTS," 2022. [Online]. Available: https://arxiv.org/abs/2210.12223

[23] K. Doan, A. Waheed, and M. Abdul-Mageed, "Towards Zero-Shot Text-To-Speech for Arabic Dialects," in *Proceedings of The Second Arabic Natural Language Processing Conference*, N. Habash, H. Bouamor, R. Eskander, N. Tomeh, I. Abu Farha, A. Abdelali, S. Touileb, I. Hamed, Y. Onaizan, B. Alhafni, W. Antoun, S. Khalifa, H. Haddad, I. Zitouni, B. AlKhamissi, R. Almatham, and K. Mrini, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 123–129. [Online]. Available: https://aclanthology.org/2024.arabicnlp-1.11/

[24] H. Mubarak, A. Hussein, S. A. Chowdhury, and A. Ali, "QASR: QCRI Aljazeera Speech Resource A Large Scale Annotated Arabic Speech Corpus," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli,

Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 2274–2285. [Online]. Available: https://aclanthology.org/2021.acl-long.177/

[25] SRG SSR Developer Portal, "SRG SSR Developer Portal," 2025, accessed: 2025-01-19. [Online]. Available: https://developer.srgssr.ch/

[26] PyTube Contributors, "PyTube: A lightweight, dependency-free Python library for downloading YouTube videos." 2025, accessed: 2025-01-19. [Online]. Available: https://github.com/pytube/pytube

[27] JuanBindez, "pytubefix: A fork of PyTube with fixes for compatibility issues." 2025, accessed: 2025-01-19. [Online]. Available: https://github.com/JuanBindez/pytubefix

[28] Podcast Club Switzerland, "Schweizer Podcasts," 2025, accessed: 2025-01-19. [Online]. Available: https://www.podcastclub.ch/schweizer-podcasts/

[29] Podcast Schmiede, "Podcasts," 2025, accessed: 2025-01-19. [Online]. Available: https://www.podcastschmiede.ch/podcasts/

[30] SRF, "SRF Audio & Podcasts," 2025, accessed: 2025-01-19. [Online]. Available: https://www.srf.ch/audio

[31] Andreas Tobler, "Sprachvirtuosin im Porträt, Sie musste sich das raue Schweizer Hochdeutsch erst antrainieren," *Der Bund*, 2023, accessed: 2025-01-26. [Online]. Available: https://www.derbund.ch/sie-loeste-eine-debatte-aus-dabei-ging-es-ihr-um-etwas-ganz-anderes-158966772382

[32] A. Plaquet and H. Bredin, "Powerset multi-class cross entropy loss for neural speaker diarization," in *Proc. INTERSPEECH 2023*, 2023.

[33] H. Bredin, "pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe," in *Proc. INTERSPEECH 2023*, 2023.

[34] Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng, "AIShell-1: An Open-Source Mandarin Speech Corpus and A Speech Recognition Baseline," in *Oriental COCOSDA 2017*, 2017, p. Submitted.

[35] F. Yu, S. Zhang, Y. Fu, L. Xie, S. Zheng, Z. Du, W. Huang, P. Guo, Z. Yan, B. Ma, X. Xu, and H. Bu, "M2MeT: The ICASSP 2022 Multi-Channel Multi-Party Meeting Transcription Challenge," in *Proc. ICASSP*. IEEE, 2022.

[36] I. Mccowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska Masson, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus," *Int'l. Conf. on Methods and Techniques in Behavioral Research*, 01 2005.

[37] NVIDIA, "NVIDIA NeMo Framework User Guide," 2025, accessed: 2025-01-20. [Online]. Available: https://docs.nvidia.com/nemo-framework/user-guide/latest/nemotoolkit/asr/speaker_diarization/intro.html

[38] "ELAN (Version 6.8) [Computer software]," Nijmegen, The Netherlands, 2024. [Online]. Available: https://archive.mpi.nl/tla/elan

[39] Lubbers, Mart and Torreira, Francisco, "pympi-ling: a Python module for processing ELANs EAF and Praats TextGrid annotation files." https://pypi.python.org/pypi/pympi-ling, 2013-2021, version 1.70.

[40] H. Bredin, "pyannote.metrics: a toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association*, Stockholm, Sweden, August 2017. [Online]. Available: http://pyannote.github.io/pyannote-metrics

[41] h5py Contributors, "h5py: Python interface to the HDF5 binary data format," https://github.com/h5py/h5py, 2023, accessed: 2025-01-22.

[42] M. Bain, J. Huh, T. Han, and A. Zisserman, "WhisperX: Time-Accurate Speech Transcription of Long-Form Audio," *INTERSPEECH 2023*, 2023.

[43] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength Natural Language Processing in Python," *Zenodo*, 2020.

[44] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[45] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, Jul. 2002, pp. 311–318.

[46] C. Callison-Burch, M. Osborne, and P. Koehn, "Re-evaluating the Role of Bleu in Machine Translation Research," in *11th Conference of the European Chapter of the Association for Computational Linguistics*, D. McCarthy and S. Wintner, Eds. Trento, Italy: Association for Computational Linguistics, Apr. 2006, pp. 249–256. [Online]. Available: https://aclanthology.org/E06-1032/

[47] T. Kocmi, C. Federmann, R. Grundkiewicz, M. Junczys-Dowmunt, H. Matsushita, and A. Menezes, "To Ship or Not to Ship: An Extensive Evaluation of Automatic Metrics for Machine Translation," in *Proceedings of the Sixth Conference on Machine Translation*, L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussa, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. J. Yepes, P. Koehn, T. Kocmi, A. Martins, M. Morishita, and C. Monz, Eds. Online: Association for Computational Linguistics, Nov. 2021, pp. 478–494. [Online]. Available: https://aclanthology.org/2021.wmt-1.57/

[48] T. Zhang*, V. Kishore*, F. Wu*, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," in *International*

Conference on Learning Representations, 2020. [Online]. Available: https://openreview.net/forum?id=SkeHuCVFDr

[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[50] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1013

[51] Jitsi, "JiWER: Similarity measures for automatic speech recognition evaluation," https://github.com/jitsi/jiwer, 2023, accessed: 2025-01-22.

[52] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[53] Tiiiger, "BERTScore: Evaluating Text Generation with BERT," https://github.com/Tiiiger/bert_score, 2023, accessed: 2025-01-22.

[54] E. Dolev, C. Lutz, and N. Aepli, "Does Whisper Understand Swiss German? An Automatic, Qualitative, and Human Evaluation," in *Proceedings of the Eleventh Workshop on NLP for Similar Languages, Varieties, and Dialects (VarDial 2024)*, Y. Scherrer, T. Jauhiainen, N. Ljubešić, M. Zampieri, P. Nakov, and J. Tiedemann, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 28–40. [Online]. Available: https://aclanthology.org/2024.vardial-1.3/

[55] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 12 449–12 460. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf

[56] E. Britannica, "Phoneme," 2025, accessed: 2025-01-22. [Online]. Available: https://www.britannica.com/topic/phoneme

[57] Q. Xu, A. Baevski, and M. Auli, "Simple and Effective Zero-shot Cross-lingual Phoneme Recognition," in *Interspeech 2022*, 2022, pp. 2113–2117.

[58] L. Bolliger and S. Waldburger, "Automatische Erkennung schweizerdeutscher Dialekte anhand von Audiodaten via Phonemtranskriptionen," Bachelor Thesis, ZHAW, June 2024. [Online]. Available: https://digitalcollection.zhaw.ch/items/d3fc95ab-21a8-42b4-8c88-14ace0313fe6

[59] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common Voice: A Massively-Multilingual Speech Corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, Eds. Marseille, France: European Language Resources Association, May 2020, pp. 4218–4222. [Online]. Available: https://aclanthology.org/2020.lrec-1.520/

[60] B. McFee, M. McVicar, D. Faronbi, I. Roman, M. Gover, S. Balke, S. Seyfarth, A. Malek, C. Raffel, V. Lostanlen, B. van Niekirk, D. Lee, F. Cwitkowitz, F. Zalkow, O. Nieto, D. Ellis, J. Mason, K. Lee, B. Steers, E. Halvachs, C. Thomé, F. Robert-Stöter, R. Bittner, Z. Wei, A. Weiss, E. Battenberg, K. Choi, R. Yamamoto, C. Carr, A. Metsai, S. Sullivan, P. Friesch, A. Krishnakumar, S. Hidaka, S. Kowalik, F. Keller, D. Mazur, A. Chabot-Leclerc, C. Hawthorne, C. Ramaprasad, M. Keum, J. Gomez, W. Monroe, V. A. Morozov, K. Eliasi, nullmightybofo, P. Biberstein, N. D. Sergin, R. Hennequin, R. Naktinis, beantowel, T. Kim, J. P. Åsen, J. Lim, A. Malins, D. Hereñú, S. van der Struijk, L. Nickel, J. Wu, Z. Wang, T. Gates, M. Vollrath, A. Sarroff, Xiao-Ming, A. Porter, S. Kranzler, Voodoohop, M. D. Gangi, H. Jinoz, C. Guerrero, A. Mazhar, toddrme2178, Z. Baratz, A. Kostin, X. Zhuang, C. T. Lo, P. Campr, E. Semeniuc, M. Biswal, S. Moura, P. Brossier, H. Lee, and W. Pimenta, "librosa/librosa: 0.10.2.post1," May 2024. [Online]. Available: https://doi.org/10.5281/zenodo.11192913

[61] Joblib Development Team, "Joblib: running Python functions as pipeline jobs," https://github.com/joblib/joblib, 2024, accessed: 2025-01-22.

[62] A. Vidhya, "Understanding the Mel Spectrogram," 2020, accessed: 2025-01-23. [Online]. Available: https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

[63] J. Betker, "Better speech synthesis through scaling," 2023. [Online]. Available: https://arxiv.org/abs/2305.07243

[64] P. Gage, "A new algorithm for data compression," *C Users J.*, vol. 12, no. 2, p. 23–38, Feb. 1994.

[65] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millicah, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan, "Flamingo: a visual language model for few-shot learning," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, ser. NIPS '22. Red Hook, NY, USA: Curran Associates Inc., 2022.

[66] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova Baseline System for the VoxCeleb Speaker Recognition Challenge 2020," 2020. [Online]. Available: https://arxiv.org/abs/2009.14153

[67] L. Biewald, "Experiment Tracking with Weights and Biases," 2020, Software available from wandb.com. [Online]. Available: https://www.wandb.com/

[68] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, G. Chauhan, A. Chourdia, W. Constable, A. Desmaison, Z. DeVito, E. Ellison, W. Feng, J. Gong, M. Gschwind, B. Hirsh, S. Huang, K. Kalambarkar, L. Kirsch, M. Lazos, M. Lezcano, Y. Liang, J. Liang, Y. Lu, C. Luk, B. Maher, Y. Pan, C. Puhrsch, M. Reso, M. Saroufim, M. Y. Siraichi, H. Suk, M. Suo, P. Tillet, E. Wang, X. Wang, W. Wen, S. Zhang, X. Zhao, K. Zhou, R. Zou, A. Mathews, G. Chanan, P. Wu, and S. Chintala, "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation," in *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, Apr. 2024. [Online]. Available: https://pytorch.org/assets/pytorch2-2.pdf

[69] J. Hwang, M. Hira, C. Chen, X. Zhang, Z. Ni, G. Sun, P. Ma, R. Huang, V. Pratap, Y. Zhang, A. Kumar, C.-Y. Yu, C. Zhu, C. Liu, J. Kahn, M. Ravanelli, P. Sun, S. Watanabe, Y. Shi, Y. Tao, R. Scheibler, S. Cornell, S. Kim, and S. Petridis, "TorchAudio 2.1: Advancing speech recognition, self-supervised learning, and audio processing components for PyTorch," 2023.

[70] Y. Lacombe, V. Srivastav, and S. Gandhi, "Data-Speech," https://github.com/ylacombe/dataspeech, 2024.

[71] Dan Lyth and Simon King, "Natural language guidance of high-fidelity text-to-speech with synthetic annotations," 2024.

[72] Y. Lacombe, V. Srivastav, and S. Gandhi, "Parler-TTS," https://github.com/huggingface/parler-tts, 2024.

[73] Roux, Jonathan Le and Wisdom, Scott and Erdogan, Hakan and Hershey, John R., "SDR – Half-baked or Well Done?" in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.

[74] OpenAI, "GPT-4 Technical Report," *CoRR*, vol. abs/2303.08774, Mar. 2023. [Online]. Available: https://arxiv.org/abs/2303.08774

[75] H. Levene, "Robust tests for equality of variances," in *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*. Stanford University Press, 1960, pp. 279–292.

[76] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3-4, pp. 591–611, 1965.

[77] R. A. Fisher, *Statistical Methods for Research Workers.* Oliver and Boyd, 1925.

[78] J. W. Tukey, *Comparing individual means in the analysis of variance.* International Biometric Society, 1949, vol. 5, no. 2.

[79] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *Journal of the American Statistical Association*, vol. 47, no. 260, pp. 583–621, 1952.

[80] A. Abid, A. Abdalla, A. Abid, D. Khan, A. Alfozan, and J. Zou, "Gradio: Hassle-free sharing and testing of ML models in the wild," Jun. 2019. [Online]. Available: https://arxiv.org/abs/1906.02569

List of Figures

List of Tables

Acronyms

AI Artificial Intelligence. 5

ASR Automatic Speech Recognition. 20, 83

BLEU BiLingual Evaluation Understudy. 9, 21, 22, 50–53, 55–58, 66, 84, 85

BPE Byte-Pair Encoding. 37

CAI Centre for Artificial Intelligence. 26, 40, 67, 70

CER Character Error Rate. 21, 22, 48–51, 53, 55–58, 84, 85, 97

CMOS Comparative Mean Opinion Score. 67, 72–77, 81, 85

DEER Detection Error Rate. 16

DER Diarization Error Rate. 17, 18

DID Dialect Identification. 10, 12, 23, 24, 27, 42, 48, 61, 85, 98

DNN Deep Neural Networks. 5, 7

MSE Mean Squared Error. 59

NLP Natural Language Processing. 5, 6, 8, 9

PESQ Perceptual Evaluation of Speech Quality. 44

SCL Speaker Consistency Loss. 38

SD Speaker Diarization. 13, 14

SI-SDR Scale-Invariant Signal-to-Distortion Ratio. 44

SMOS Similarity Mean Opinion Score. 66, 72–77, 79, 81, 85

SNF Schweizerischer Nationalfonds. 46, 49, 51, 53, 61, 62, 68, 69, 84

SNR Signal to Noise Ratio. 44

SOTA State-Of-The-Art. 5, 9, 10

SRF Schweizer Radio und Fernsehen. 8, 9, 11–14, 18, 19, 23–27, 29, 32, 38–40, 45, 46, 50, 55, 56, 59, 62, 63, 65, 83–85, 96

TTS text-to-speech. 1, 5–11, 24, 29, 37, 39, 42, 48, 65, 66, 83, 85, 86, 103

UI User Interface. 70

VAD Voice Activity Detection. 14, 16

VQ-VAE Vector Quantised-Variational AutoEncoder. 37, 38, 40

WER Word Error Rate. 9, 10, 20–22, 49–53, 55–60, 66, 84, 85

YT YouTube. 11–13, 83, 96

ZS Zero-Shot. 1, 10, 37, 83–86

Appendix A

General Appendix

Podcast	Source	Language	Source Type	Spoken (h)
BuchZeichen	SRF	mixed	API	359.6811
Echo der Zeit	SRF	de	API	924.1700
Einfach Politik	SRF	mixed	API	40.8858
Espresso	SRF	mixed	API	555.2192
Focus	SRF	ch	API	794.4378
Input	SRF	mixed	API	697.7264
Kontext	SRF	de	API	2765.4917
Krimi	SRF	mixed	API	242.5150
Kultur kompakt	SRF	de	API	1609.8997
News Plus	SRF	de	API	320.6578
Persönlich	SRF	ch	API	762.0569
Perspektiven	SRF	de	API	433.5519
Politikum	SRF	de	API	44.3472
Rehmann	SRF	ch	API	216.4439

Table A.1: Unutilized SRF podcasts with amount of hours of audio per podcast

Appendix B

Code & Manual

The code of the data pipeline of this thesis is available on a GitHub repository at https://github.com/stucksam/swiss-zero-shot-va-tts. The fork of the coqui-ai library for the TTS training and evaluation is available at https://github.com/stucksam/coqui-tts.

Appendix C

Evaluation

Client ID	Dialect	Age	Gender	Sentences
005039b8-898f-48d8-b7cc-8a16bd1055c8	Innerschweiz	fourties	female	5
031b0a74-5bdd-47e7-b8b7-9bb58d0e8c72	Graubünden	twenties	female	3
0497b106-6644-42ce-b99e-57f9a6c7fc81	Bern	sixties	male	2
0b890339-031a-43ef-bc2e-5e8b5ac5e613	Wallis	thirties	female	3
12fb73be-cf60-4794-befb-381682ccda9a	Basel	twenties	female	2
1cd99b41-8298-4180-aec6-bb65039c9ed7	Basel	twenties	male	3
24f85b05-10f3-49f3-bfd0-273120e750cb	Bern	fourties	male	1
2a4acb33-759e-411f-bbd0-2470080758fb	Bern	twenties	male	1
3e7e3ad6-938d-40fa-9787-2e9f8c529a66	Zürich	sixties	male	1
4a8346e7-fa21-49b9-9a6f-c69ef828a68c	Innerschweiz	fifties	male	1
4ee1811c-9884-4261-99cb-2f7346c8ea6e	Zürich	twenties	male	2
50e4a935-c887-4c38-aff2-6286aab725d1	Wallis	twenties	female	5
5184dba3-c5e1-4570-ba26-62176e8c8dcc	Innerschweiz	twenties	female	2
6516567b-0d9b-4853-880c-d5f0327dd384	Graubünden	fourties	male	5
684dd9cf-2844-407b-9a7f-12e7b559773f	Ostschweiz	fourties	female	6
6d34bfbb-2de2-48d9-891d-ae216f9b347b	Zürich	twenties	male	5
72725555-9e5d-436a-b311-25048bc0c594	Innerschweiz	fourties	female	1
72911186-8b0e-4c0b-af3d-4d8765072930	Ostschweiz	twenties	male	1
7ca44480-9007-4c1c-8046-aade3c4e6a87	Bern	twenties	male	2
7f4b1ca8-c652-4056-96fb-f02e82182574	Ostschweiz	thirties	female	1
8050767b-0a0e-43db-8754-2a42e896f7dd	Basel	thirties	female	1
831d404d-b189-42b6-8120-073c1f8af73c	Ostschweiz	sixties	male	5
8800b4de-fa22-4fdc-9f29-457c4010fd57	Basel	fourties	female	1
887b50f8-215b-4a1d-8f32-13516da6506f	Bern	sixties	female	1
8d167b1c-7f89-4487-a0a9-2bbe08bf0b41	Ostschweiz	fourties	male	1
8f16829d-1a64-4145-927c-d6738a07ea24	Basel	thirties	male	1
9a9554dd-3ee4-4997-82e6-1d78e0ca3d7f	Zürich	fourties	male	1
a677eda9-7709-4c9d-8656-e7b665140f3b	Ostschweiz	thirties	male	1
aa7d9d2c-90d3-41f9-a92e-f07b93765f8b	Innerschweiz	twenties	female	1
acf67674-c912-42c0-ba3c-f85e2db965ac	Innerschweiz	thirties	male	5
aece75d7-5d2b-47a4-9f87-24962dfd2e38	Graubünden	fourties	female	2
ba118975-5963-4495-927a-a78d19dd98c1	Bern	fourties	male	3
ba826a45-33f8-47ef-9516-a66a201aac29	Zürich	fifties	female	2
c4f6bcdf-fc02-4fe8-9277-0701cfffebab	Graubünden	teens	female	2
ce50fa32-dbc4-4f72-bd5f-5128784a5abc	Graubünden	twenties	female	1
ce8839ae-3b20-491c-8c33-cef08b4def6e	Zürich	sixties	male	4
d2dee463-0eb9-47fa-b739-f1dccd8638f9	Bern	twenties	male	4
d2f85a8f-7c45-48e3-a804-0b192f7f8ad6	Graubünden	twenties	male	1
d9b44aee-da3d-42c8-8ad1-d1029767f05a	Bern	fourties	male	1
e903f18d-7bbb-4fb6-aaa3-0f7d965027fb	Basel	twenties	male	2
f2feaf58-1285-4fc1-831d-5c1b013b7e0b	Graubünden	fourties	female	1
fb1b67be-8d8f-47bb-b15c-ee1138f0d4ac	Wallis	twenties	female	7
fd72db57-c291-43e6-840e-92e06f83ae56	Basel	twenties	male	5

Table C.1: Aggregated test speaker data by dialect region, age, gender, and sentence count. Taken from STTSG-350 [10] test split