The Xiaomi-ASLP Text-to-speech System for Blizzard Challenge 2023

Yuepeng Jiang1, Kun Song1, Fengyu Yang2, Lei Xie1∗, Meng Meng2, Yu Ji2, Yujun Wang2

1Audio, Speech and Language Processing Group (ASLP@NPU)
School of Computer Science, Northwestern Polytechnical University, Xi’an, China
2Xiaomi AI Lab, Beijing, China
jiangyp@mail.nwpu.edu.cn, kunsong.npu.se@gmail.com, lxie@nwpu.edu.cn,
{yangfengyu1 mengmeng jiyul wangyujun}@xiaomi.com

Abstract
This paper describes the Xiaomi-ASLP text-to-speech (TTS) system for the hub task 2023-FH1 of Blizzard Challenge 2023. The goal of the hub task is to build a single-speaker French TTS system trained on (but not limited to) the single-speaker French audiobook corpus released by the Blizzard Challenge 2023 organization. We present a fully end-to-end TTS system based on VITS. In our implementation of the system, we replace the duration alignment module with a length regulator and duration predictor. Additionally, we introduce a style adaptor to model the style and prosody of the generated speech. The style adaptor consists of a fine-grained prosody module and a global style module based on a language model. To further enhance the audio quality of the synthesized output, we leverage the super-resolution capability of the vocoder to upsample the 16kHz synthesized waveform to 48kHz.

Index Terms: Text-to-speech, Blizzard Challenge 2023, VITS, fine-grained prosody, audio super-resolution

1. Introduction
The Blizzard Challenge1, organized since 2005, aims to advance text-to-speech (TTS) technologies by comparing and gaining insights into different approaches. The fundamental objective involves constructing high-quality TTS systems utilizing the speech corpus furnished by the organizers. This year’s challenge2 includes the following two tasks:
• Hub task (FH1): To build a voice from the provided French data, using only publicly available data.
• Spoke task (FS1): To build a voice from the provided French data that is the closest to target speaker as possible.

We focus on completing the hub task. The hub task provides a total of over 50 hours of French audio data from five audiobooks narrated by a female speaker. This task primarily evaluates the intelligibility, speaker similarity, and naturalness of the synthesized results.

Apart from intelligibility, which is a fundamental evaluation criterion for TTS systems, the central focus of this challenge is on speaker similarity and the naturalness of the synthesized speech. To achieve high speaker similarity, it is crucial to ensure that the timbre of the synthesized speech is as close as possible to the target speaker. Building a TTS system with a sufficient amount of data is not a challenging task [1]. Therefore, the main direction for improving speaker similarity lies in exploring modeling ways for style and prosody. The audio quality is a crucial influencing factor in the naturalness of the audio.

In the hub task of this year’s challenge, our TTS system generates audio at a sampling rate of 48kHz for better audio quality performance.

In this paper, we implement a fully end-to-end speech synthesis model based on VITS [2]. The fully end-to-end model, such as VITS, can completely eliminate the audio quality degradation caused by the mismatch problem due to the training separation of the acoustic model and the vocoder. We implement a style adaptor to model style and prosody to varying degrees. We employ a fine-grained prosody module and a global style module based on the language model (LM). The data from audiobooks contains rich expressiveness, but it is challenging to handle it effectively unless the data has been annotated or clustered based on style, emotion, or other attributes [3]. Utilizing a posterior encoder and normalizing flow to implicitly guide the prosody predictor in learning the relationship between textual and acoustic prosody avoids the need for additional reference features and avoids the requirement for prosody annotations. In the past couple of years, the language model has made rapid advancements. The practical application of large language models like ChatGPT in various natural language processing (NLP) tasks has demonstrated powerful semantic understanding capabilities. This provides a shortcut for batch annotation of audiobook text data without style labels, reducing the difficulty of style modeling for audiobooks. Since prosody and style are features strongly correlated with duration, we replaced the random duration predictor and monotonic alignment search used in VITS with the same duration predictor and length regulator as FastSpeech [4]. This modification allows for better learning to predict duration while incorporating external duration information. Our TTS model can only generate synthesized audio at 16kHz. To achieve better audio quality, we upsample synthesized audio to 48kHz by utilizing the super-resolution capability of the vocoder.

The structure of this paper is as follows. Section 2 provides a detailed introduction to the system framework. Section 3 and Section 4 present the analysis of evaluation results and the conclusion.

2. Methods

2.1. Overview
Our system contains three components as shown in Figure 3. The dataset for the hub task comes from audiobooks available on librivox3. In the data preparation stage, we filter the training data, using only a subset of suitable data for training. For the text used in inference, we employ a text front-end to convert it into a phoneme sequence, which serves as the input to

* Lei Xie is the corresponding author.

1https://www.synsig.org/index.php/Blizzard_Challenge/
2https://www.synsig.org/index.php/Blizzard_Challenge_2023/
3https://librivox.org/
The TTS model. The end-to-end TTS model we implemented takes phonemes as input and generates audio waveforms at a sampling rate of 16kHz. Due to the audiobook data having a sampling rate of only 22kHz, we are unable to directly synthesize audio at a higher sampling rate. Furthermore, lower sampling rate audio tends to lack high-frequency components, which can result in relatively poorer quality and a blurry, distorted, or less natural sound. To address this issue, we leverage the super-resolution capability of the vocoder to upsample the synthesized audio to 48kHz, improving the audio quality and clarity of the synthesized speech.

The total parameter count of our final submitted model is 66.7 million. We utilize eight NVIDIA A100 GPUs with a batch size of 512 for approximately 800,000 training steps, based on our previous experience training the VITS model.

![Figure 1: The structure of proposed VITS-based TTS Model.](image)

**Figure 1:** The structure of proposed VITS-based TTS Model.

2.2. Data Preparation

The data for the hub task of the Blizzard Challenge 2023 consists of 51 hours of audio clips from 5 French audiobooks read by a native French speaker. The audio files are in WAV format and were recorded indoors with slight background noise. The audio data in this corpus has a sampling rate of 22050 Hz, a bit depth of 16 bits, and single-channel audio. Upon inspection of the audio slices obtained using the segmentation information provided by the corpus, it’s found that the slices are not affected by the background noise. Therefore, no noise reduction processing is applied to the audio. However, during the inspection process, we notice that the same speaker has slight variations in voice in different audiobooks, and there are also slight differences in the recording environment, resulting in noticeable differences in audio quality between different slices. To ensure that the model learns an average voice sufficiently close to the target speaker, speaker clustering is performed on the data to remove audio with noticeable variations. In addition, the data quality is assessed using the PESQ [5] method, and slices with significantly low scores are filtered out. After two rounds of filtering, approximately 42000 samples with duration annotations are used for the experiments. From this dataset, we randomly select 30 samples to form the test set, while the remaining data is utilized as the training set.

Although the dataset provided for the challenge contains phoneme sequences corresponding to the text, we still need to prepare a text front-end to convert the text into phoneme sequences. Since the evaluation set only contains the text to be synthesized, we use the open source eSpeak as our text front-end. The phoneme set includes all 36 phonemes and an additional long silence symbol in the dataset. For English words in the text, we map their phonemes to the phoneme set used in the system based on their actual pronunciation.

2.3. VITS-based TTS Model

As shown in Figure 1 (a), our proposed system is based on VITS. It takes phoneme sequences as input and generates synthesized audio waveforms as output. We preserve the original text encoder and its corresponding flow in VITS to allow converting textual features into prior distributions. To make full use of the duration data provided in the dataset and to increase the stability of the VITS model, the duration matching scheme is replaced by a length regulator similar to that in FastSpeech. In addition, a matching duration predictor is introduced to predict the corresponding duration of the text from the input. After being extended by the length regulator, the features are passed through the distribution decoder to predict their mean and variance for prior distribution.

Audiobooks typically contain voice performances that give different characters unique voices and personalities. In addition, the rhythm is controlled based on the content to ensure smoothness and appeal [3]. As a result, audiobook data has a rich and varied style and prosody based on the text content. We implement a style adaptor to model the style and prosody of audiobook content. As shown in Figure 1 (b), The style adaptor consists of a fine-grained prosody module and an LM-based global style modeling module. The fine-grained prosody module utilizes a prosody encoder to model the scene-specific prosody patterns. Furthermore, we incorporate an additional normalizing flow and a prosody predictor to learn the relationship between text and prosody distribution. The obtained prosody feature is also fed into the duration predictor to improve prediction accuracy. We employ a large language model for the LM-based global style modeling to annotate the text data, enabling us to model the style based on these embeddings.

4https://espeak.sourceforge.net/
2.4. Style Adaptor

2.4.1. Fine-grained Prosody Module

As shown in Figure 1 (c), the fine-grained prosody module comprises a pair of prior predictors and posterior encoders for extracting prosodic features and a normalizing flow for transforming the prior and posterior distributions. We employ a posterior prosody encoder to model the rich prosody information found in audiobooks implicitly. This modeling approach provides prosodic information for duration prediction, resulting in better stability of the model’s predictions. We add another normalizing flow and a prosody predictor to learn the relationship between textual and acoustic prosody distribution. The prosody encoder employs the same structure as the posterior encoder, which includes one-dimensional convolutional layers before and after, as well as non-causal WaveNet residual blocks (WN) used in WaveGlow [6] and Glow-TTS [7]. A WaveNet residual block consists of layers of dilated convolutions with a gated activation unit and skip connection. The linear projection layer above the blocks produces the mean and variance of the normal posterior distribution.

However, unlike the posterior encoder, the prosody encoder needs to generate phoneme-level prosody features instead of frame-level ones. Therefore, it is necessary to convert the input frame-level features into phoneme-level features. We generate a masking matrix using external duration features, which allows the averaging of several frames within a phoneme to form one frame.

Similar to the posterior encoder, which requires a normalizing flow to transform the posterior distribution of the spectrogram into a prior distribution for learning the mapping between text input and spectrogram distribution, the prosody encoder, serving as a posterior encoder, also necessitates a prosody normalizing flow to convert the posterior distribution of the prosody into a prior distribution. Furthermore, a linear projection layer is employed to generate the mean and variance of the prosody prior distribution.

2.4.2. LM-based Global Style Embedding

As mentioned in the overview, audiobook narrators adjust their reading style based on the content of the text. For instance, phrases such as “Today is such a beautiful day” are typically read with a relaxed and pleasant tone. Consequently, our goal is to model the overall style of the text input to synthesize speech with more natural intonation and rhythm. Nonetheless, annotating the entire dataset with style information manually poses with more natural intonation and rhythm. Nonetheless, annotating the entire dataset with style information manually poses a significant challenge, particularly for those who are not native French speakers. Inspired by the language model of natural language processing [8, 9], we explore the possibility of utilizing GPT-3’s remarkable semantic understanding capabilities for data annotation. Guiding sentences are crafted to prompt GPT-3, pretrained with OpenAI with the codename text-davinci-003, to evaluate the style of text input and classify it into eight categories: neutral, angry, disappointed, disgusted, happy, surprised, fearful, and sad. Subsequently, we tag the style to the pronunciation of the homograph in the synthesis, regarding the correctness of the pronunciation. The synthesized and natural audios are carefully rated by three types of listeners who are involved paid listeners, volunteers, and speech experts.

3. Results

The evaluation of the Blizzard Challenge 2023 hub task includes five aspects: including naturalness test with MOS (mean opinion score), similarity test with SMOS (similarity mean opinion score), naturalness test with MUSHRA, SUS intelligibility test with WER (word error rate) and HOMOS intelligibility test with WER. In the MOS part, participants listen to one sample and choose a score representing how natural or unnatural the sentence sounded on a scale of 1 to 5. For SMOS, they choose a score that represented how similar the synthetic voice sounded to the voice in the reference samples on a scale from 1 to 5. For MUSHRA, listeners listen to one explicit reference to the original speaker and 5 or 6 non-identified audio samples, among which there is one hidden reference. They are asked to rate the non-identified audio samples on a continuous scale from 0 to 100. In the SUS intelligibility test, listeners hear one utterance in each part and type in what they hear. Listeners are allowed to listen to each sentence only once. For the HOMOS intelligibility test, listeners listen to 3 audio samples. One audio sample is the synthesis of an utterance that contains a homograph. The text content of the sentence is displayed on the screen, and the homograph is written in capital letters. The two other audio samples are the two versions of the homograph as an isolated word uttered by a reference speaker. Listeners are asked to select the reference audio that corresponds the best to the pronunciation of the homograph in the synthesis, regardless of the correctness of the pronunciation. The synthesized and natural audios are carefully rated by three types of listeners who are involved paid listeners, volunteers, and speech experts.

Our system is denoted as System D, and natural speech recorded by the original speaker is marked as System A.

3.1. Naturalness Test

The listening results for the naturalness tests of all systems are shown in Figure 3. Our system achieve an average score (mean score 3.4) that can be considered acceptable but not satisfactory. However, upon closer comparison and listening analysis, we identify a significant issue with the synthesis of pauses due to inadequate handling in the text front-end processing. This results in the omission and misplacement of pauses in the synthesized speech. In addition, the evaluation also provided results from both native and non-native listeners. Surprisingly, the scores from non-native listeners (mean score 3.84) are significantly higher than those from native listeners (mean score 3.3). Partly, this suggests that certain small imperfections pre-
vent us from achieving a higher naturalness score, and improper text front-end processing may plays a crucial role in the occurrence of pauses. However, it is important to note that the number of non-native listeners is much smaller than that of native listeners, which raises some doubts about the reliability of this conclusion.

Figure 3: Naturalness MOS scores in the hub task.

3.2. Similarity Test

The box plot in Figure 4 displays the speaker similarity MOS scores for all systems. Unfortunately, our system achieved scores significantly lower than expected (mean score 2.3). As previously mentioned, the dataset used in the hub task consists of audio excerpts from five distinct audiobooks, and we observe variations in the timbre of the same speaker across these audiobooks. To enable the model to learn a more stable and accurate speaker timbre, we perform speaker clustering on the data, excluding instances where the speaker’s voice deviates significantly from others.

While this clustering approach ensures a closer approximation of the average target speaker’s timbre, it also introduces a slight bias compared to training on the entire dataset. This discrepancy likely accounts for the substantial deviation from the expected speaker similarity MOS scores we obtain. Furthermore, we note significant fluctuations in the similarity scores for the results associated with code A, which corresponds to the original recordings of the origin speaker. This observation strongly suggests that there are indeed specific differences in timbre for the same speaker across different audiobooks.

Figure 4: Similarity MOS scores in the hub task.

3.3. Intelligibility of semantically unpredictable sentences

By comparing the data presented in Figure 5, it is evident that our system exhibits a relatively high word error rate (mean word error rate 23%). Given the substantial amount of pause omissions and errors resulting from inadequate text front-end processing, this outcome is understandable. Additionally, artificial phoneme mapping based on pronunciation may not fully capture the actual pronunciation of certain phonemes. This could be one of the reasons for the higher word error rate.

Figure 5: Intelligibility on semantically unpredictable sentences.

3.4. Intelligibility of homographs

The intelligibility test results for homographs are presented in Figure 6. Despite achieving a relatively high rank, the pronunciation accuracy is only 67%. It is worth noting that for most systems, including ours, intelligibility performance on semantically unpredictable sentences and homographs is not entirely consistent. In some cases, the same system may perform differently on these two aspects, and it may even show completely opposite results between intelligibility and homographs. This suggests that in order to improve pronunciation accuracy, it may be necessary to focus on phoneme sets and the text front-end when dealing with homographs.

Figure 6: Intelligibility on homographs.

4. Conclusions

In this paper, we present the Xiaomi-ASLP text-to-speech (TTS) system for the hub task 2023-FH1 of Blizzard Challenge 2023. Based on the characteristics of the competition data, we employed a fine-grained prosody modeling module for implicit modeling of style and prosody and utilized a large language model to extract style tags from the text. By doing so, we can control the global style. Additionally, we applied DSPGAN for further upsampling of synthesized audio, which significantly improved the quality of the synthesis results. However, due to we are not good at French front-end processing and limited time devoted to text front-end processing, there may be several issues in our front-end results, especially concerning pauses and other aspects. This could be a direct reason for our suboptimal performance in terms of naturalness and intelligibility. While data
filtering may have contributed to lower scores in speaker similarity, it is unlikely to be the key factor. In future work, we aim to address the shortcomings in our French front-end processing and explore the impact of front-end processing on the accurate pronunciation of homographs. Additionally, we will continue investigating the key factors leading to our lower speaker similarity scores.

5. References


