FireRedTTS: The Xiaohongshu Speech Synthesis System for Blizzard Challenge 2023

Kun Xie∗, Yi-Chen Wu∗, Feng-Long Xie

Xiaohongshu Inc., Shanghai, China
{xie.kun, wu.yichen, feng.long.xie}@xiaohongshu.com

Abstract

This paper presents the FireRed (Team Q) speech synthesis system developed for the FSI speaker adaptation task of Blizzard Challenge 2023, which aims to develop a French text-to-speech voice that best resembles a specific speaker. To achieve this, we constructed a well-trained model from the provided 51-hour corpus of speaker (NEB), and then fine-tuned it on the 2-hour corpus of the target speaker (AD).

To synthesize high-quality and high-fidelity speech, we took a three-pronged approach: 1) We designed and built a French TTS front-end with extensive effort to accurately model the nuances of the French language. 2) We utilized a modified Non-Attentive Tacotron based acoustic model, with implicit variation information (utterance-level and phoneme-level prosody) modeling, to generate mel-spectrograms of high naturalness and speaker similarity. 3) We trained a modified version of HiFi++ vocoder to reconstruct high-quality waveform from mel-spectrograms.

For the FSI task, our system achieves a MOS (Quality) of 4.1 and MOS (Similarity) of 4.3, demonstrating the effectiveness of our proposed system.

Index Terms: Blizzard Challenge 2023, Text-to-speech, HiFi++, Non-Attentive Tacotron

1. Introduction

Text-to-speech (TTS), also known as speech synthesis, is a technique used to convert text into natural and intelligible speech. In recent years, the development of deep learning has led to the emergence of highly expressive TTS models, which have gained popularity due to their improved naturalness and speech quality. Blizzard Challenge 2023 is the 18th annually held challenge that intends to compare speech synthesis models and techniques in building systems on official data. There are two tasks in this year’s challenge. Task FH1 involves building a system suitable for highly expressive audiobook reading, which consists of 289 chapters from five audiobooks with a total length of around 51 hours [1]. The other task, task FS1, is to develop a TTS system with 2,515 utterances with a length of around 2 hours. For both tasks, we use the same system architecture, which comprises three main components: the text frontend, acoustic model, and vocoder.

The text frontend module converts input text into tokenized yet rich phonetic representations. Its subcomponents include grapheme-to-phoneme conversion, homograph disambiguation, prosody prediction, and post-lexical rule modifications. The accuracy of frontend is directly linked to the naturalness and intelligibility of the resulting speech.

The acoustic model plays a crucial role in converting linguistic attributes into acoustic features, e.g. mel-spectrograms. One such model, Tacotron [2, 3], a popular acoustic model architecture, utilizes a sequence-to-sequence feature prediction network that maps phoneme embeddings to mel-spectrograms to generate high-quality and expressive voices. However, this model suffers from robustness issues due to the wrong attention alignments between text and speech in the autoregressive generation [4]. To address these issues, FastSpeech [4, 5] was developed, which employs a feed-forward transformer for parallel generation. Additionally, an external aligner is used to prevent alignment mistakes between the encoder and decoder, improving robustness and generation speed. Another work, the Non-Attentive Tacotron (NAT) [6], enhances the naturalness of speech by using Gaussian upsampling instead of repeated upsampling techniques employed in FastSpeech. Researchers have also explored modeling variation information, such as speaker identity, pitch, and energy, to mitigate the one-to-many mapping problem in speech synthesis. StyleTTS [7] and DelightfulTTS [8] propose using a reference encoder to improve expressiveness and fidelity. Inspired by these works, we propose a new acoustic model based on the Non-Attentive Tacotron, with several improvements. Firstly, conformer blocks [9] are employed in the encoder and decoder to enhance stability. Secondly, utterance-level and phoneme-level prosody encoders [8] are introduced to address the one-to-many mapping problem. Several recent studies [7, 10] have demonstrated that adversarial training can effectively bridge the gap between the predicted acoustic feature distribution of an acoustic model and the actual acoustic feature distribution. This approach subsequently enhances the quality of the output audio, even under conditions of limited samples. Therefore, we adopt an additional adversarial training [11] process, following the successful implementation in GANSpeech [10].

The vocoder reconstructs waveform using intermediate acoustic representations. In recent years, there have been significant advancements in applying generative adversarial networks (GANs) [11] in vocoders to generate more realistic waveforms. For instance, MelGAN [12] proposed a multi-scale architecture for discriminators that operate on different scales of raw waveforms. HiFiGAN [13], on the other hand, introduced a discriminator that comprises small sub-discriminators, each of which operates on only specific periodic parts of raw waveforms. HiFi++ [14] modified the generator architecture by introducing three additional subnetworks: the SpectralUnet, WaveUnet, and SpectralMaskNet modules to generate more stable speech. We adopt the HiFi++ architecture, which allows for obtaining higher quality waveforms with fewer parameters. We present modifications and improvements to our implementation, resulting in a system that delivers improved results on
both tasks. Our study demonstrates the significant potential of this implementation to enhance contextual text-to-speech synthesis with high speech quality and intelligibility.

The paper is organized into several sections. Section 2 provides detailed description of our system. Section 3 describes the training and testing pipeline. Finally, at the end of the paper, we present our evaluation results and conclusions.

2. Framework

2.1. Data preparation

The construction of an accurate and rich intermediate representation between the frontend and backend modules is important to the overall performance of a TTS system. For this challenge, phoneme inputs, punctuation and boundary levels are our linguistic inputs of choice.

The representation must be accurate. Based on a previous study [15] that showed the superiority of phoneme-based input over grapheme-based one in French Grapheme-to-Phoneme (G2P) and liaison prediction tasks, we select phoneme input, provided in the dataset, as our primary label.

The representation should also be informative. In addition to the provided word boundaries, punctuation breaks, and sentence breaks, we add an additional level of boundary, rhythmic group boundary, to our linguistic feature set to facilitate the generation of annotation in parallel with the phoneme sequence as an input to the acoustic-phonetic notation of the dataset [1]). With rules, which will be discussed in Section 2.2.2, were used to identify them syntactically. To better model sentence-level prosody, adjacent sub-sentence utterances, typically linked by commas, are recombined to create sentence-level utterances.

It should be noted that in French, syllables can span word boundaries, known as *Encalminement*, where the final consonant sound of a word is transferred to the following word’s first syllable, as in “avec elle” (‘avec’ el → ‘ave’ kel) using phonetic notation of the dataset [1]). With rules, we aligned these word boundaries with syllable boundaries at the left of the word-end consonants, as in the example above, to avoid incorrect generation of pauses within any syllable.

2.2. Frontend

Our frontend system for French, as shown in Figure 1, has three key modules: a G2P module that handles lexicon lookup and homograph disambiguation, a prosody prediction module that extracts boundary information, and a sandhi handling module that applies post-lexical rules of the French language.

We build our system on top of the pre-trained French CamemBERT model [16] integrated in the spaCy package [17]. This model facilitates the extraction of several basic word-level attributes, including Part-of-Speech (POS), Dependency Tree (DEP) and text embeddings. Furthermore, we use a word-vector based Named-Entity-Recognition (NER) model, also integrated in the spaCy package, to extract named entities like names and organizations.

2.2.1. G2P

G2P performance can greatly impact the performance and accuracy of a TTS system. We develop our own lexicon combining various resources and use speaker-specific custom dictionaries to reconcile the discrepancies between dataset annotations and lexicon for some high-frequency words, such as “monsieur” ([muʁzjɛʁ] in annotations, but [muʁʒɛʁ] in the default lexicon) and “le” ([lɛ] in annotations, but [lã] in the default lexicon). This reconciliation is done on the lexicon side only because phonetic annotations should align with audio and remain unaltered. For out-of-vocab (OOV) words, a simple seq2seq RNN-based G2P model is trained. Two additional types of marker are added to our lexicon entries for about 300 words in total: *h muetz* as-pire markers and liaison markers. Details will be discussed in Section 2.2.3.

Homographs, which are words spelled the same but pronounced differently, can pose a challenge to the G2P module. While most homograph pairs have different POS attributes and are easily discernible, others differ only in their semantic meaning (e.g., convient). Using POS features, we manually craft rules to differentiate between the easy pairs. For more challenging homographs that are syntactically or semantically context dependent, we train a small classifier (2-layer conformer, and one linear) using a fixed window of CamemBERT context embeddings as the input feature. Leveraging previous work by Hajj [18] that listed out which homographs can be predicted with POS and which cannot, we selected a list of 60 homographs as our words of interest. A single model is trained for all the homographs, and the 120 possible pronunciation labels are masked at inference time. And the training data was sourced from either handcrafted or internet-crawled material (including the text provided in the Blizzard dataset) for all 120 possible pronunciations, resulting in a small corpus of 1800 homograph-annotated sentences.

2.2.2. Prosody

French is often described as a syllable-timed language with a fixed final accent on the last syllable of the word or group of words, coinciding with intonational boundaries [19], or what we call rhythmic group boundaries. For example, at the end of each rhythmic group of the phrase *les meilleurs amis du président* (‘the best friends of the president’), the last syllable (capitalized) is where the phrase accent favors. Rhythmic group boundaries have a much stronger association with text syntax than with phoneme sequence, which justifies the need for explicit rhythmic group features predicted by the frontend. This boundary level sequence (no boundary, word boundary, rhythmic group boundary, or punctuational boundary) is used in parallel with the phoneme sequence as an input to the acous-

![Figure 1: Flowchart of our frontend framework.](image-url)
tic model.

With insufficient annotated prosody data, we cannot predict rhythmic group boundaries using a data-driven approach; instead, we must rely on rule-based methods. POS, NER, and dependency tree features are used in rule writing to approximate rhythmic groups boundaries in both training and test data. About 40 rules are written, and an example rule in pseudo format looks like:

“If the word start with s’, and the previous word is not a noun or pronoun, then there is a rhythmic boundary before this word.”

Finally, to further improve precision of rhythmic boundary predictions, as shown in Figure 1, we extract named entities and fixed expressions to reject incorrect segmentations.

However, this rule-based prosody prediction method should only be used as a last option when no prosody annotations are present, even with all the techniques applied.

2.2.3. Enchaînement and Liaison

Sandhi refers to the change of a word as a result of its position in an utterance. In French, there are two major cases of sandhi known as Enchaînement and Liaison. As explained in Section 2.1, enchaînement refers to the process of syllabifying a word-final consonant with the following word’s initial vowel. In cases where the final consonant of the word is otherwise absent, the pronunciation of the added consonant is referred to as a liaison. For example, /z/ in “chez elle” {s’, e} {zel} does not originally exist in the singleton word “chez” {s’, e}.

Rules for the application of liaisons (whether mandatory, impossible, or optional) are determined by linguistic features such as POS, semantic role, and rhythmic boundaries.

To apply these two sandhi accurately, h muet/h aspiré markers and liaison marker have been introduced to the lexicon for words that apply. For words beginning with the letter “h”, it is important that we differentiate between those that are aspirated (h aspiré) and those that are mute (h muet). Enchaînement and liaison apply to h muet but not to h aspiré. For liaisons identification, we tagged around 200 dictionary entries that could potentially apply liaisons with their extra word-end consonants to add (e.g. les /le/ → le[zl]/).

Operating on the lexical phoneme sequence (with the two types of markers), rhythmic boundaries, and other previously extracted features (POS, DEP, NER), around 30 liaison rules are handwritten based on guides available in [20].

2.3. Acoustic models

Acoustic models are commonly used to generate mel-spectrograms from linguistic sequences that embody critical phonemic information, speaker identification, punctuation marks, and boundary levels. In addition, we have incorporated two supplementary elements, consisting of book identification and sentence types (either dialogue or narration). Our rationale for including these feature points stems from the discovery of diverse reading styles within the training corpus. These styles are closely associated with particular book titles and punctuation marks, such as quotation marks for dialogue sequences.

Our approach draws inspiration from the Non-Attentive Tacotron [6], while incorporating several refinements to enhance its performance. Specifically, our network consists of an encoder, a decoder, a prosody encoder/predictor, and an upsampler, as illustrated in Figure 2.

The encoder converts linguistic sequences into encoded feature representations. 4 conformer blocks [9] are used to extract pertinent textual information from the input phoneme sequence. We use three distinct embedding tables to store speaker embeddings, book embeddings, and sentence type embeddings, which are integrated into the encoder features to provide a wealth of information regarding various reading styles. Gaussian upsampling [6] is then applied to align the encoder features and acoustic features, resulting in equivalent lengths between the two features. Next, the decoder generates mel-spectrograms from the encoded input sequence. Contrary to the autoregressive structure employed in the original NAT, we use 6 conformer blocks to directly convert aligned features into mel-spectrograms. Our experiments suggest that a non-autoregressive structure yields more natural and higher quality audio. Finally, a post-processing net known as CBHG [2] is used to refine output quality and reduce pronunciation errors.

In addition to the aforementioned elements, we introduce two key improvements to our model. Firstly, we incorporate a prosody encoder [8] module which extracts prosody-related information using real mel-spectrograms during the training phase. A prosody predictor composed of dual layers of 1D convolutions along with a layer of GRU in order to predict prosody information during the testing phase and the input of the prosody predictor is text features, while the anticipated outcome is closely comparable to that of the prosody encoder. Secondly, the GANSpeech [10] methodology is implemented via the addition of an additional discriminator for adversarial training. Additional adversarial training was performed to make the distribution of predicted mel-spectrograms more closely resemble the real distribution.
2.4. Neural vocoder

In recent years, GAN-based models have shown remarkable performance in neural vocoding, outperforming even the best autoregressive models [21, 22], while exhibiting faster generation speeds. However, further research [23] has revealed that many of these models are prone to unstable fundamental frequencies and pronunciation defects. To address these concerns, the HiFi++ model [14] was introduced, which incorporates three additional modules - SpectralUnet, WaveUNet, and SpectralMaskNet subnetworks - to achieve higher quality audio.

Because of all the advantages, we adopt the HiFi++ model as our vocoder in this challenge. In addition to the original implementation, we add a multi-resolution STFT loss [24] to improve and stabilize GAN training.

Finally, Our system features a total of 356M parameters (100M frontend, 256M backend) and a real-time factor of 0.14 on GPU.

3. System building

3.1. Training phase

In the training phase, the data preparation mentioned in Section 2.1 was first performed. We only used the official data released by the challenge organizer [1], refraining from any external audio data. We randomly selected 100 audio files as the validation set from each provided book, with the remaining files being assigned to the training set.

For the vocoder, all audio data is used in training, with an audio format of 16-bit PCM and a sampling rate of 24 kHz. The model uses 80-band mel-spectrograms as the input condition. The FFT size, window size, and hop size were 1024, 960, and 240 respectively. Setting the transposed convolution stride to [5, 4, 3, 2, 2], we upsampled mel-spectrograms up to 240 times to match the temporal resolution of raw waveforms. The model used AdamW [25] optimizer, with an initial learning rate of 0.0003, $\beta_1 = 0.8$, $\beta_2 = 0.99$, and a learning rate decay of $\lambda = 0.8$. The networks were trained up to 1.0M steps.

For the acoustic model, mel-spectrograms and the linguistic sequences generated by our frontend system were used as inputs. AdamW [25] optimizer were used, with an initial learning rate of 0.0003, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and batch size sets to 48. After 150k training steps, the acoustic model has effectively reached convergence.We then regarded the acoustic model as a generator, represented by $G$, and introduced a discriminator, represented by $D$, to distinguish real and predicted mel-spectrograms. Upon completion of an additional 200k adversarial training steps, the converged model is obtained.

It is important to note that during the acoustic model training process: 1) The Gaussian upsampling module uses annotated ground-truth phoneme durations provided by the challenge organizer. 2) The input for the prosody encoder is the ground-truth mel-spectrogram. 3) The output of the prosody encoder is learned via the prosody predictor, where the input constitutes text features.

3.2. Testing phase

At inference time, the frontend system extracts various linguistic features such as phonemes, sentence types (dialogue or narration), punctuation marks, and boundary levels from the input texts. The book embedding ids, however, do not exist for the test set. We simply selected the style that performs better on the validation set at training time, from the two book styles in the training corpus, as the default. Subsequently, the linguistic features are then processed by the acoustic model to generate corresponding mel-spectrograms. Finally, the mel-spectrograms produced are converted into speech waveforms using our HiFi++ vocoder.

4. Results

The FS1 task featured 17 systems (1 natural voice, 2 baselines, and 14 participants) this year. Among them, System A was a recording of natural speech from the original speaker, System BF and System BT being benchmark systems comprising of FastSpeech2 + HiFi-GAN and Tacotron2 + HiFi-GAN respectively. The remaining systems were identified as C to T, our system being labeled as Q. Subjective evaluations were carried out for naturalness (via MOS quality test), and similarity (via MOS similarity test).

4.1. MOS quality test

MOS quality test reports the naturalness and overall quality of the synthetic speech. Scores are rated by the listeners on a scale of 1 [Very poor] to 5 [Excellent]. As shown in the boxplot in

![Figure 3: Boxplot and distribution of MOS (quality) of each submitted system for all listeners.](image)

training waveform, MOS quality test of the best 4 systems for a refined ranking.

![Figure 4: Boxplot of MUSHRA quality test of the best 4 systems for a refined ranking.](image)
Figure 3, our system scored a mean score of 4.1 and a median score of 4, indicating a good level of naturalness, but there is still a gap between our system and the other top-tier systems. This gap is more apparent in the MUSHRA test where the top systems were evaluated comparatively at once, as shown in Figure 4. We believe that this is due to two primary shortcomings in our system. Firstly, prosody is underrepresented in the phonological form extracted by the frontend. The added rhythmic boundary annotations in Section 2.2.2 are not aligned accurately enough with the audio. Secondly, our current acoustic model lacks explicit pitch features, potentially affecting the expressiveness of the synthesized speech. We believe that tackling these issues could significantly enhance the performance of our system.

4.2. MOS similarity test

Figure 5 presents the MOS boxplot of each submitted system measuring how similar the synthesized speech sounded to the reference recording. Scores are rated on a scale of 1 [Completely different person] to 5 [Exactly the same person]. For the FS1 task, our system achieved the highest mean opinion similarity score of 4.3 and a median score of 5 among all submitted systems, which we attribute mainly to the use of adversarial training in acoustic modeling. This approach enables enhanced learning of real acoustic feature distributions, especially in situations where only limited data is available. Our inclusion of book information and sentence type information also helps mitigate the influence of style divergences in the material.

5. Conclusions

This paper presents our submitted system and its performance in Blizzard Challenge 2023. Our system demonstrated competitive results in FS1 task, particularly in similarity assessments. We pursued two approaches to achieve this. Firstly, we developed a Non-Attentive Tacotron acoustic model that leverages implicit variation modeling and adversarial training. This has proven to be effective in enhancing the similarity of small datasets. Secondly, we employed a HiFi++ vocoder to generate high quality waveform. However, we acknowledge that more work is needed to improve the naturalness of our results. Moving forward, we are exploring the potential benefits of adopting an end-to-end modeling approach, which could potentially improve the naturalness and quality of the audio outputs.

6. References


