Token merge layer 를 활용한 효율적인 오디오 생성에 관한 연구

정명훈, 김세민, 김남수
서울대학교 전기정보공학부 뉴미디어통신공동연구소 휴먼인터페이스 연구실

{mhjeong, smkim21}@hi.snu.ac.kr, nkim@snu.ac.kr

A Study on the Efficient Parallel Audio Generation using Token Merge Layer

Myeonghun Jeong, Semin Kim, Nam Soo Kim
Human Interface Laboratory,
Department of Electrical and Computer Engineering and INMC,
Seoul National University

요 약

본 논문은 토큰 기반 오디오 생성 모델의 효율적인 인퍼런스를 도모하였다. 기존의 토큰 기반 오디오 생성 모델은 트랜스포머 언어 모델을 활용하기 때문에 인풋 시퀀스의 길이가 길어짐에 따라서 계산량과 메모리 사용량이 매우 커지게 된다. 본 논문에서는 Token Merge (ToMe) layer 를 활용하여 오디오 품질은 유지하면서 계산량과 메모리 사용량을 개선한다.

Ⅰ. 서 론

최근 Natural Language Processing (NLP) 분야에서 Large Language Model (LLM) 이 좋은 성능을 보이며 큰 주목을 받고 있다. 이러한 연구와 더불어, 오디오 생성 분야에서도 거대언어모델을 활용하고자 하는 시도들이 지속되어 왔다. 하지만, 어쿠스틱 토큰은 자연어 토큰 보다 프레임 레이트가 크기 때문에 일반적으로 시퀀스의 길이가 더 길고, 이는 인퍼런스의 비효율을 야기한다. 특히, 트랜스포머 기반 언어모델의 경우에는 시퀀스의 길이가 길어짐에 따라서 계산량과 메모리 사용량이 매우 커지기 때문에 이를 고려해줄 기술이 필요하다. 따라서, 본 논문에서는 Token Merge (ToMe) [1] 레이어를 활용하여 어쿠스틱 토큰 기반 오디오 생성모델의 성능은 유지하면서 인퍼런스 과정의 비효율을 개선하였다.

Ⅱ. 본론

토큰 기반 오디오 생성 모델의 backbone 으로 SoundStorm [2] 의 모델 구조를 활용한다. SoundStorm 은 발음과 관련된 정보를 갖고 있는 semantic token 에서 음향 정보를 갖고 있는 acoustic token 으로 가는 과정을 뉴럴 네트워크로 모델링한다. SoundStorm 은 양방향 Conformer 구조로 이루어져 있으며, 셀프 어텐션 레이어의 long-term dependency 를 활용하여 프롬프트의 화자 정보를 반영한다. SoundStorm 에서는 residual vector quantization (RVQ) 기반의 acoustic token 을 활용하는 데, RVQ 기반 acoustic token 추출 과정은 그림 1 을 통해 설명된다. 오디오 latent feature 에서 연속적인 vector quantization 과정을 거쳐서 acoustic token $Q \in \mathbb{R}^{T \times N_q}$ 을 얻게 된다. 여기서 T는 시간 프레임, N_q는 RVQ depth 를 의미하며, 두번째 VQ layer 부터는 앞선 feature 의 residual 값을 인코딩 하게 된다. 이렇게 인코딩 된 acoustic token 은 첫번째 양자화 레벨에 coarse 한 정보가 인코딩 되고,

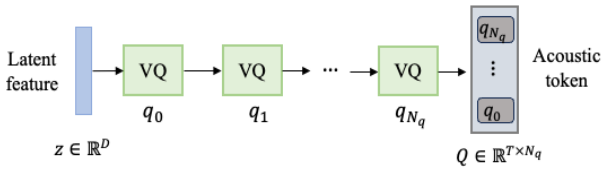

그림 1. RVQ 기반 acoustic token 추출 과정

나머지 레벨은 fine 한 정보가 인코딩 된다. SoundStorm 에서는 이러한 acoustic token 을 효율적으로 예측하기 위해서 인퍼런스 과정에서 confidence-based parallel decoding 방식을 활용한다. 이 방식은 정보량이 가장 많은 첫번 째 양자화 레벨에서 confidence 기반의 반복적인 샘플링을 하게 되고, 생성된 coarse acoustic token 을 활용하여 coarse 에서 fine 방향으로 순차적으로 토큰 시퀀스를 생성하게 된다. 각 과정을 parallel 하게 생성하기 때문에 기존의 autoregressive (AR) 모델에 비해서 인퍼런스 속도를 개선하였지만, 반복적인 샘플링 과정과 토큰 시퀀스의 길이에 크게 영향 받는 트랜스포머 구조는 SoundStorm 인퍼런스의 비효율성을 야기한다.

따라서, 본 논문에서는 ToMe [1] 레이어를 활용하여 SoundStorm 인퍼런스 과정의 비효율을 개선하였다. ToMe [2] 에서는 visual 토큰의 유사성을 기반으로 토큰 시퀀스를 merge 하고, 블럭을 지날 때마다 시퀀스의 길이는 점점 줄어들어 계산 과정에서 효율성을 추구하였다. 우리는 이러한 ToMe 레이어를 acoustic token 생성 과정에 활용한다. 활용 방법은 다음과 같다. 첫번째로, 어텐션 레이어에 들어가기 직전의 프롬프트의 key 임베딩에 대해서 시간 축으로 같은 시퀀스 길이의 두 개의 집합으로 분할한다. 그런 다음 두 집합의 임베딩들에 대해서 코사인 유사성을 구하여 각 집합의 임베딩에 대해서 가장 유사한 토큰 쌍을 구한다. 그렇게 구한 유사성을 기반으로 가장 유사한 r 개의 토큰을 merge 하여 토큰 시퀀스의 길이를 줄이게 된다. 이는 ToMe 논문의 bipartite soft matching 방식을 응용한 것으로 ToMe [2] 논문을 참고하기 바란다. 이렇게

merge 된 프롬프트 임베딩은 attention layer 를 통과한 이후 unmerge 하게 된다. 이 과정을 도식화하여 그림 2 에 나타내었다. c 채널의 두개의 토큰 임베딩 x_1, $x_2 \in \mathbb{R}^c$ 가 주어져 있을 때, Merge 과정은 평균을 취하여 다음과 같이 수행하였다.

$$x_{1,2}^{*} = \frac{x_1 + x_2}{2}$$

또한, unmerge 는 아래와 같이 수행하였다.

$$x_1' = x_{1,2}^{*} \quad x_2' = x_{1,2}^{*}$$

토큰 시퀀스의 길이에 따라 계산량과 메모리 사용량에 큰 영향을 받는 attention layer 의 앞단에서 merge 를 하기 때문에 긴 프롬프트 시퀀스를 입력 받더라도 효율적으로 어쿠스틱 토큰 시퀀스를 생성할 수 있다.

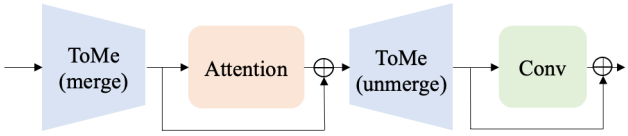

그림 2 ToMe 를 활용한 conformer block

본 논문에는 실험과정에서 Wav2Vec 2.0 [4] 임베딩을 k-means 클러스터링 하여 semantic token 으로 활용하였고, SoundStream [5] 코덱을 활용해서 acoustic token 을 추출하였다. SoundStorm 의 N_q 는 12, 샘플링 iteration 은 (16, 1, 1, … ,1) 로 총 27 회 iteartions 으로 설정했고, 이외에도 [2]에서의 configuration 을 따라서 실험을 진행하였다. 또한, 본 논문에서 r 은 prompt 길이의 절반으로 설정하였다. Training dataset 으로는 Libri-TTS train clean, train-other 를 모두 활용하였고, evaluation dataset 으로는 Libri-TTS clean dataset 을 활용하였다. 먼저, SoundStorm 모델과, ToMe 를 적용한 제안 모델의 inference time 과 peak memory 를 비교 해 보았다. 이 실험에서 target semantic token 은 10 초로 고정하고, 프롬프트의 길이는 short 은 5 초, long 은 20 초로 설정하였다. 표 1 에서 알 수 있듯이, inference time 과 peak memory 가 전체적으로 제안 모델이 SoundStorm 보다 더 작게 나타났다. 특히, long 프롬프트가 주어졌을 때, 제안모델과의 성능 차이가 더 커지는 것을 확인할 수 있다. 이는 제안 모델의 인퍼런스 과정이 계산량과 메모리 측면에서 더 효율적이다는 것을 나타내는 결과이다. 추가로, 우리는 ToMe 를 적용했을 때 제안 모델의 성능이 떨어지는지 확인하기 위해 NISQA-Mean Opinion Score (NISQA-MOS) 와 Speaker Encoder Cosine Similarity (SECS) 를 측정해 보았다. SECS 측정을 위한 스피커 인코더는 WavLM-TDNN [7] 을 활용하였다. 두 항목 모두에서 SoundStorm 모델과 제안모델의 성능 차이가 크지 않기 때문에, 제안 모델은 오디오 음질과 프롬프트 화자 반영도는 유지하면서 인퍼런스 과정이 효율적이라는 결론을 낼 수 있었다.

Model	Inference Time (s)		Peak Memory (MB)	
	short	long	short	long
SoundStorm	1.0868	2.4683	2,279	3,756
Proposed	**1.0559**	**1.9979**	**2,173**	**2,573**

표 1. Inference Time 과 Peak Memory 비교

Model	NISQA-MOS	SECS
Ground Truth	4.122±0.047	0.7926
SoundStorm	3.878±0.033	0.4192
Proposed	3.849±0.033	0.4058

표 2. MOS 와 SECS 비교

Ⅲ. 결론

본 논문에서는 ToMe layer 를 활용함으로써 효율적인 오디오 생성모델을 제안하였다. 토큰 시퀀스의 길이가 길어짐에 따라 인퍼런스 과정이 비효율적이라는 문제의식에 기반하여, 시퀀스의 길이가 길어지더라도 계산량과 메모리의 비용을 최소화 하였다. 이 과정에서 오디오 품질과 프롬프트 화자 반영도가 유지되는 것을 실험적으로 보였다. 이를 통해 토큰 기반 오디오 생성모델도 속도와 메모리에 제약 없이 양질의 샘플을 얻을 수 있게 되었다는 데 의의가 있다.

ACKNOWLEDGMENT

이 논문은 2023 년도 BK21 FOUR 정보기술 미래인재 교육연구단에 의하여 지원되었음.

참 고 문 헌

[1] Bolya, Daniel, et al. "Token merging: Your vit but faster." arXiv preprint arXiv:2210.09461 (2022).

[2] Borsos, Zalán, et al. "SoundStorm: Efficient Parallel Audio Generation." arXiv preprint arXiv:2305.09636 (2023).

[3] Gulati, Anmol, et al. "Conformer: Convolution-augmented Transformer for Speech Recognition." Interspeech 2020 (2020).

[4] Baevski, Alexei, et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations." Advances in neural information processing systems 33 (2020): 12449-12460.

[5] Zeghidour, Neil, et al. "Soundstream: An end-to-end neural audio codec." IEEE/ACM Transactions on Audio, Speech, and Language Processing 30 (2021): 495-507.

[6] Mittag, Gabriel, et al. "Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets." arXiv preprint arXiv:2104.09494 (2021).

[7] Chen, Sanyuan, et al. "Wavlm: Large-scale self-supervised pre-training for full stack speech processing." IEEE Journal of Selected Topics in Signal Processing 16.6 (2022): 1505-1518.