

AUDIO PROCESSING METHOD, MODEL TRAINING METHOD AND APPARATUSES

TECHNICAL FIELD

[0001] The present disclosure relates to the field of audio processing technologies, and in particular,
5 to audio processing methods, model training methods and apparatuses, devices, and a storage medium.

BACKGROUND

[0002] Speech Separation is a task of obtaining audio of separated speakers from a mixture of speakers. It is also often known as a cocktail party problem, which was the observation that humans are able to efficiently pay attention to audio sources in a noisy environment, but this task is difficult for computers.
10 Speech Separation is trained on waveform comparison loss against ground truth clean speech, typically using the Scale invariant Signal-to-Distortion Ratio (SI-SDR) loss function. Due to this stringent loss, it has been found that performance typically suffers as compression of the input audio increases. As a result, recent models have been growing in size and computational cost, especially with models that combine both time and frequency domain features.

15 [0003] This background information is provided as it may be relevant to the present disclosure. No admission is necessarily intended, nor should be construed, that any of the preceding information constitutes prior art against the present disclosure.

SUMMARY

[0004] In a first aspect, an embodiment of the present disclosure provides an audio processing method,
20 including:

inputting mixed audio data into an encoder of a pre-trained neural audio codec (NAC) which is trained with a set of sample mixed audio data for audio separation to obtain a first embedding of the

mixed audio data, where the mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of audio data with a decoder of the pre-trained NAC.

10 [0005] In a second aspect, an embodiment of the present disclosure provides an audio processing method, including:

inputting mixed speech data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple speech signals from multiple participants of a meeting, and the first embedding is a representation of an audio feature of the mixture; the pre-trained NAC is trained with a set of sample mixed speech data for speech separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple speech signals one-by-one, and each of the at least one second embedding is a representation of an audio feature of a corresponding speech signal;

transmitting the at least one second embedding to a cloud server, to enable the cloud server to reconstruct the at least one of the multiple speech signals with a decoder of the NAC.

[0006] In a third aspect, an embodiment of the present disclosure provides an audio processing method, including:

25 inputting mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of music, and the first embedding is a representation of an audio feature of the mixture, where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at

least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of music one-by-one, each of the at least one second embedding is a representation of an audio feature of a corresponding piece of music, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of music with a decoder of the pre-trained NAC.

5 **[0007]** In a fourth aspect, an embodiment of the present disclosure provides a model training method, including:

 acquiring a training sample set, where the training sample set includes plural pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;

10 for a piece of sample mixed audio data in the training sample set,

 inputting the sample mixed audio data into an encoder of a neural audio codec (NAC) to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture;

 inputting the first sample embedding into a pre-trained separator to obtain at least one second sample embedding, where the at least one second sample embedding corresponds to at least one of the plural pieces of audio data one-by-one, and each of the at least one second sample embedding is a representation of an audio feature of corresponding audio data;

15 inputting the at least one second sample embedding into a decoder of the NAC to obtain reconstructed audio data for at least one corresponding sample acoustic source;

20 updating at least one parameter of the NAC based on the reconstructed audio data and the at least one of the plural pieces of sample audio data.

[0008] In a fifth aspect, an embodiment of the present disclosure provides a model training method, including:

25 acquiring a training sample set, where the training sample set includes plural pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;

 for a piece of sample mixed audio data in the training sample set,

 inputting the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture;

inputting the first sample embedding into a decoder of the NAC to reconstruct the sample mixed audio data;

updating at least one parameter of the NAC based on a reconstruction result and the sample mixed audio data.

5 [0009] In a sixth aspect, an embodiment of the present disclosure provides an audio processing apparatus, including a processing module, configured to: input mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is
10 trained with a set of sample mixed audio data for audio separation; perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding piece of audio data, and the at least one second embedding is used for reconstruction of at least one of the multiple
15 pieces of audio data with a decoder of the pre-trained NAC.

[0010] In a seventh aspect, an embodiment of the present disclosure provides an audio processing apparatus, including a processing module, configured to: input mixed speech data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple speech signals from multiple participants of a meeting, and the
20 first embedding is a representation of an audio feature of the mixture; perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple speech signals one-by-one, and each of the at least one second embedding is a representation of an audio feature of a corresponding speech signal; and an outputting module, configured to transmit the at least one second embedding to a cloud
25 server, to enable the cloud server to reconstruct the at least one of the multiple speech signals with a decoder of the NAC.

[0011] In an eighth aspect, an embodiment of the present disclosure provides an audio processing apparatus, including:

a processing module, configured to:

input mixed audio data into an encoder of an neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of music, and the first embedding is a representation of an audio feature of the mixture;

perform audio separation on the first embedding with a separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of music one-by-one, each of the at least one second embedding is a representation of an audio feature of a corresponding piece of music, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of music with a decoder of the pre-trained NAC.

[0012] In a ninth aspect, an embodiment of the present disclosure provides a model training apparatus, including:

an inputting module, configured to:

acquire a training sample set, where the training sample set includes multiple pieces of sample mixed audio data, and each of the multiple pieces of sample mixed audio data is a mixture of multiple pieces of sample audio data from multiple sample acoustic sources;

for a piece of sample mixed audio data in the training sample set,

input the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture;

input the first sample embedding into a pre-trained separator to obtain at least one second sample embedding, where the at least one second sample embedding corresponds to at least one of the multiple pieces of audio data one-by-one, and each of the at least one second sample embedding is a representation of an audio feature of corresponding audio data;

input the at least one second sample embedding into a decoder of the NAC to obtain reconstructed audio data for at least one corresponding sample acoustic source;

an updating module, configured to update at least one parameter of the NAC based on the reconstructed audio data and the at least one of the multiple pieces of sample mixed audio data.

[0013] In a tenth aspect, an embodiment of the present disclosure provides a model training apparatus, including:

an inputting module, configured to:

acquiring a training sample set, where the training sample set includes plural pieces of sample

mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;

for a piece of sample mixed audio data in the training sample set,

input the sample mixed audio data into an encoder of an NAC to obtain a first sample
5 embedding, where the first sample embedding is a representation of an audio feature of the mixture;

input the first sample embedding into a decoder of the NAC to reconstruct the sample mixed
audio data;

an updating module, configured to update at least one parameter of the NAC based on a
reconstruction result and the sample mixed audio data.

10 [0014] In an eleventh aspect, an embodiment of the present disclosure provides an electronic device,
including at least one processor coupled to a memory storing a set of instructions; where the at least one
processor is configured to read the set of instructions in the memory and execute the method according
to any one of the first aspect to the fifth aspect.

[0015] In a twelfth aspect, an embodiment of the present disclosure provides a computer-readable
15 medium storing computer execution instructions which, when executed by a processor, causes the
processor to execute the method according to any one of the first aspect to the fifth aspect.

[0016] In a thirteenth aspect, an embodiment of the present disclosure provides a computer program
product including computer execution instructions which, when executed by a processor, causes the
processor to execute the method according to any one of the first aspect to the fifth aspect.

20 [0017] In a fourteenth aspect, an embodiment of the present disclosure provides a non-transitory
computer-readable medium storing computer execution instructions which, when executed by a
processor, causes the processor to execute the method according to any one of the first aspect to the fifth
aspect.

BRIEF DESCRIPTION OF DRAWINGS

25 [0018] Reference will now be made, by way of example, to the accompanying drawings which show
example embodiments of the present disclosure.

[0019] FIG. 1 shows a schematic illustration of a speech separation method.

[0020] FIG. 2A shows an exemplary scenario where an audio processing method according to one or

more embodiments of the present disclosure may be applied.

[0021] FIG. 2B shows an exemplary scenario where an audio processing method according to one or more embodiments of the present disclosure may be applied.

5 [0022] FIG. 3 shows a schematic flowchart of an audio processing method according to one or more example embodiments of the present disclosure.

[0023] FIG. 4A shows an exemplary structure of a separator according to one or more example embodiments of the present disclosure.

[0024] FIG. 4B shows another exemplary structure of a separator according to one or more example embodiments of the present disclosure.

10 [0025] FIG. 5 shows a schematic flowchart of an audio processing method in a meeting system according to one or more example embodiments of the present disclosure.

[0026] FIG. 6 shows a schematic flowchart of an audio processing method in a music processing system according to one or more example embodiments of the present disclosure.

15 [0027] FIG. 7A shows a schematic flowchart of a model training method according to one or more example embodiments of the present disclosure.

[0028] FIG. 7B shows a schematic flowchart of another model training method according to one or more example embodiments of the present disclosure.

[0029] FIG. 8A shows a schematic illustration of an audio processing method according to one or more example embodiments of the present disclosure.

20 [0030] FIG. 8B shows a schematic illustration of another audio processing method according to one or more example embodiments of the present disclosure.

[0031] FIG. 9 shows a schematic structural diagram of an audio processing apparatus according to one or more embodiments of the present disclosure.

25 [0032] FIG. 10 is a structural diagram of an electronic device according to one or more embodiments of the present disclosure.

DESCRIPTION OF EMBODIMENTS

[0033] In the following description, reference is made to the accompanying figures, which form part of the present disclosure, and which show, by way of illustration, specific aspects of embodiments of the

present disclosure or specific aspects in which embodiments of the present disclosure may be used. It is understood that embodiments of the present disclosure may be used in other aspects and include structural or logical changes not depicted in the figures. The following detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present disclosure is defined by the appended claims.

5 **[0034]** In related art, since high computational demands of separation models, which are typically too intensive for on-device processing, audio will have to be sent to the cloud for processing, which may have to be transmitted through a codec anyway, which introduces latency and requires reliable internet connectivity. Before sending the audio to the cloud, mixed audio data often has to be transmitted through a codec to compress the audio data for efficient transmission over networks. As shown in FIG. 1, which
10 is a schematic illustration of a speech separation method, the speech separation is normally done at the cloud side, however, the above compression step can introduce distortions that may affect the quality of the audio and, consequently, the performance of the speech separation is poor. In addition, a full audio sequence is used by the models (for example, a Sepformer shown in FIG. 1) to generate the speech separation output. The full audio sequence can be quite large, leading to high memory usage, which may
15 not be feasible on devices with limited memory capabilities. The need to process the entire sequence can result in longer processing times, which is undesirable for applications that require real-time or near-real-time responses. To manage long sequences, models like Sepformer use complex architectures with chunking methods and dual-path mechanisms, which increases the model's complexity and computational requirements.

20 **[0035]** When seeking a more suitable solution to overcome the above problems, the inventor's attention was paid to the way of Neural Audio Codec (NAC), NAC seeks to find a compressed representation of audio that can be reconstructed with good fidelity compared to the original audio. The separation model based on NAC may be trained using generative adversarial networks, where a discriminator model seeks to determine if any given audio sequence is generated or not. In addition, a
25 number of waveform level comparison losses like SI-SDR are also used to measure the model output against the original audio. Due to the high compression achieved by the discrete audio representations generated by the encoder of the NAC, it would be beneficial for applications such as Speaker Verification, Diarization and Speech Recognition. The NAC has also been used as an audio-encoder for multi-modal large language models as an alternative to Whisper-based continuous embedding approaches or

HuBERT-based approaches. Despite a high level of compression, a significant portion of information and a wide variety of features included within the audio can be captured by the neural codecs. This makes NACs an attractive approach for performing speech separation in edge computing situations.

[0036] However, if we simply apply the NAC in the way shown in FIG. 1, as described above, the performance of the speech separation would still be poor. Therefore, in order to improve the separation performance, the present disclosure further proposes performing speech separation using codecs generated by the encoder of the NAC, that is, the speech separation would be done in the codec space (or referred to as compressed embedding space of the NAC). In the process of realizing such separation in the compressed embedding space of the NAC, since audio of speaker mixtures is typically not part of the training set, the codecs may not be able to represent mixtures of voices in its compressed embedding space. Besides, the performance of the speech separation is typically measured through waveform matching which would be very sensitive to the distortions created by the compression.

[0037] As described before, existing speech separation models are difficult to be deployed in edge computing use cases because they are large and have high computational requirements, and audio will have to be sent to the cloud for processing, which may have to be transmitted through a codec anyway, so we can take advantage of such codec transmission to implement the speech separation in the codec space, and consequently benefit from operating in the compressed embedding space of NACs. Additionally, using audio codecs pre-trained on massive amounts of data has the potential to benefit the speech separation task during fine-tuning.

[0038] In view of the above, the present disclosure provides an audio processing method, including: inputting mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, the mixed audio data here is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture, the pre-trained NAC is trained with a set of sample mixed audio data for audio separation; performing audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of audio data with a decoder of the pre-trained NAC. According to

the embodiments of the present disclosure, by virtue of the NAC which has been trained for audio separation target, the audio separation can be performed based on the first embedding within the compressed embedding space of the NAC, which can reduce the computational load, and can allow for faster audio separation. In addition, by using the NAC, significant time compression can be realized at the same time, thus reducing sequence lengths significantly. This allows the separator to accept the full sequence into its transformer layers without the need for any chunking, which significantly reduces the memory and computational requirements of training the separator.

[0039] Embodiments of the present application can be used to implement audio processing technologies, and can be used in various application scenarios. FIG. 2A shows an exemplary scenario where an audio processing method according to one or more embodiments of the present disclosure may be applied. As shown in FIG. 2A, mixed audio data is encoded locally using a NAC's encoder to obtain an embedding. The mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources. A pre-trained separator provided in the embodiments of the present disclosure, which is shown as a Separator, performs separation on the embedding to generate embeddings or masks corresponding to the multiple pieces of audio data one-by-one. The separated embeddings or masks can be transmitted to the cloud, where the NAC's decoder reconstructs the separated audio signals. This approach is beneficial, for example, if the decoding process requires more computational resources than those available locally or if the cloud provides additional services or analytics. In addition, this approach can also be useful in scenarios where the cloud serves as a collection point for audio data from various local sources. In addition, the audio processing method of the present application can also be used in target speaker extraction (TSE). It should be noted that in the embodiments of the present disclosure, the term audio separation may include both cases, technical schemes described with respect to separation of a mixture into multiple pieces of audio data would also be applicable for TSE, similar principle applies and details will be omitted for brevity.

[0040] FIG. 2B shows another possible scenario, in an online meeting scenario or a real-time communication scenario, mixed speech signals of different participants (participants A to C or called users A to C) can be collected through a speech input device, e.g., a microphone, and the mixed speech signals can be processed by the processing device locally to separate their voice from background noise or other speakers using the NAC encoder and Separator (which can also be called Codecformer in the

present disclosure), and then as one choice, the separated embeddings may be sent to a cloud server. By encoding the audio locally and simply sending the compressed embeddings to the cloud side, the bandwidth requirements for the meeting can be reduced, which is especially useful for participants with limited internet connections. In a meeting scenario, the cloud server can act as a central hub for audio processing, allowing for more consistent audio quality and easier management of the meeting's audio feed. It should be noted that although in FIG. 2B, the speech input device and the processing device are shown separately, but they may also be integrated together. For example, the speech input device and the processing device may be integrated in an all-in-one conference system, or via a terminal such as a smartphone, a computer, a tablet device, etc. The speech input device can collect all audio data include:
5 human voices, animal sounds, voice signals from surrounding electronic devices, and environmental audio signals and other sounds around the communication system. The method according to one or more embodiments of the present disclosure can be implemented by the terminal.

[0041] As another choice, the separated embeddings may be further processed by the processing device using the NAC decoder to obtain the speech signals of different participants, that is, the entire audio processing, from encoding to decoding, can also take place locally. After the NAC's encoder compresses the mixed audio data, the Codecformer can perform audio separation to obtain the embeddings. The NAC's decoder, which is also local, can reconstruct the separated audio signals without the need to transmit any data to the cloud. This local end-to-end processing can be advantageous for applications that prioritize privacy, low latency, or operate in environments with limited or no internet connectivity. Besides, this local end-to-end processing is not constrained by the above described chunking restriction.
15

[0042] FIG. 3 shows a schematic flowchart of an audio processing method according to one or more example embodiments of the present disclosure. The method may be applied in the real-time communication scenario above, and may also be applied in other scenarios where a specific piece of audio or music or other kinds of audio needs to be extracted from a mixture of audio data. Illustratively, the audio processing method provided by the present disclosure can be executed by an audio processing device provided by the present disclosure, and the audio processing device can be implemented by any software and/or hardware. For example, the audio processing device can be: a tablet computer, a mobile phone (such as a folding screen mobile phone, a large screen mobile phone, etc.), a wearable device, a
25

vehicle-mounted device, an augmented reality (AR)/virtual reality (VR) equipment, a laptop computer, a ultra-mobile personal computer (UMPC), a netbook, a personal digital assistant (PDA), a smart TV, a smart screen, a HD TV, a 4K TV, a smart speaker, a smart projector and other Internet of Things (IOT) devices. The specific types of electronic devices are not limited in the embodiments of the present disclosure. As shown in FIG. 3, the method can include the following steps.

5 [0043] S301, input mixed audio data into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation.

10 [0044] In the embodiment, mixed audio data is input into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data. The mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture. The mixed audio data can be obtained through an audio input device, e.g., a microphone, or a personal device like a smartphone or tablet, or any other audio recording setup. The multiple acoustic sources may include at least two of: a human voice, an animal sound, a musical instrument, ambient noise, an electronic device, a vehicle, a nature sound, industry and machinery, etc. The audio feature may include at least one of: a frequency spectrum (including pitch, tone, and harmonics present in sounds), a temporal feature (including onset and decay of sounds, rhythm, temporal structure, etc.), intensity, loudness, etc.

20 [0045] The pre-trained NAC is an NAC model trained in advance for audio separation target. Different from existing NACs which are simply responsible for audio compression and decompression, the NAC used in the embodiments of the present disclosure is trained with a set of sample mixed audio data, and for audio separation purpose. In this way, the embeddings obtained by the encoder of the pre-trained NAC can be able to represent the mixture of different audio data in the embedding space, then the embeddings corresponding to the different audio data can be well reconstructed into audio data by the decoder of the NAC, thereby facilitating the audio separation based on the embeddings. For example, based on a NAC which has been trained for audio compression and decompression, self-supervised learning (SSL) could be used for training such NAC with sample mixed audio data, so as to obtain the pre-trained NAC which could function well in representing the sample mixed audio data in the

compressed embedding space.

[0046] In a possible implementation, the NAC may be a Descript NAC (DAC). Using a DAC as the NAC, high levels of compression can be provided, which can improve the efficiency of transmitting or storing audio data. The DAC model has strong performance, and its training dataset includes a diverse
5 range of data, including speech, audio and music.

[0047] S302, perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding corresponding to at least one of the multiple pieces of audio data one-by-one, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of
10 corresponding audio data, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of audio data with a decoder of the pre-trained NAC.

[0048] In the embodiment, the processing device performs audio separation on the first embedding with the pre-trained separator to obtain one or more second embeddings, for the TSE, the pre-trained separator would be able to obtain one second embedding corresponding to audio data from a target
15 acoustic source, and for the case of separating a mixture into multiple pieces of audio data, multiple second embeddings which correspond to the multiple pieces of audio data one-by-one would be obtained by using the pre-trained separator. The separator may be a component or model which performs audio separation based on an embedding produced by the pre-trained NAC. Each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second
20 embedding is used for reconstruction of the at least one of multiple pieces of audio data with a decoder of the pre-trained NAC. The second embedding can be in a form of vector which is understandable by the decoder of the pre-trained NAC. The separator may be any separation model which could be used for separating audio data from different acoustic sources, which is not limited in the embodiments of the present disclosure.

[0049] The reconstructed multiple pieces of audio data would be multiple pieces of separated audio data, each corresponding to one acoustic source. In a possible implementation, the processing device may input the at least one second embedding into a decoder of the NAC to obtain at least one of the multiple pieces of audio data. The decoder of the NAC can be used to obtain the multiple pieces of audio data based on the input second embedding. The decoder may separate the mixed audio data into constituent
25

audio streams. For example, the decoder may transform the at least one second embedding back into a time-domain waveform, which may involve an inverse short-time Fourier transform (ISTFT) or other suitable inverse techniques that can reconstruct the audio signal from the frequency-domain representation. Each stream may correspond to a different speaker or acoustic source whose audio
5 contributes to part of the mixed audio data. The use of the NAC's decoder can improve the codec's efficiency in audio reconstruction, which is designed to produce high-fidelity audio from compressed representations. In addition, using the NAC's decoder to obtain at least one of the multiple pieces of audio data can reduce artifacts and preserve audio quality during reconstruction.

[0050] According to the embodiments of the present disclosure, by virtue of the NAC which has been
10 trained for audio separation target, the audio separation can be performed based on the first embedding within the compressed embedding space of the NAC, which can reduce the computational load, and can allow for faster audio separation. In addition, by using the NAC, significant time compression can be realized at the same time, thus reducing sequence lengths significantly. This allows the separator to accept the full sequence into its transformer layers without the need for any chunking, which significantly
15 reduces the memory and computational requirements of training the separator.

[0051] With regard to the pre-trained separator, as shown in FIG. 4A, the separator may include a normalizing block, a transforming block and a post-processing block. Specifically, the processing device may input the first embedding into a normalizing block of the separator to obtain a normalized first embedding, may input the normalized first embedding into a transforming block of the separator to obtain
20 a transformed representation, where the transformed representation is a mixture of multiple internal representations, and may input the transformed representation into a post-processing block of the pre-trained separator to obtain the at least one second embedding. The normalization may include various operations such as min-max scaling, z-score normalization (subtracting the mean and dividing by the standard deviation), or more sophisticated approaches like batch normalization, which normalizes the
25 input within each mini-batch during training. The normalized first embedding may be then input into a transforming block of the separator. The transforming block may include a series of transformer layers or similar mechanisms designed to manipulate and learn from the embedded data. The output of the transforming block may be a mixture of multiple internal representations, where each internal representation may represent an audio feature of the corresponding audio data from the original mixture

(mixed audio data). The transformed representation may be input into a post-processing block of the separator, and the post-processing block can output the at least one second embedding, each second embedding corresponding to a piece of audio data in the mixed audio data. The post-processing block can be used to generate the at least one second embedding that is understandable by the decoder of the pre-trained NAC and allow for the separation of the mixed audio data into individual audio data for each acoustic source. The post-processing may include, e.g., filtering, reshaping, etc. The normalization can ensure that the subsequent processing steps work with a consistent scale, which can improve the quality of the separation and can make the separator more robust to variations in the input audio. The use of the post-processing block can enable the transformed representation to be processed into the form of the second embedding so as to facilitate the subsequent reconstruction by virtue of the decoder of the pre-trained NAC. The use of the transforming block and the post-processing block can allow for a flexible network architecture that can be adapted to the specific requirements of the audio separation task, including but not limited to the number of sources to be separated and the complexity of the audio mixture, and can improve computational efficiency.

[0052] In a possible implementation, the post-processing block may include a first convolutional network and an adjusting block, where the first convolutional network may be used for mapping the transformed representation into the multiple internal representations, and the adjusting block may be used for generating the at least one second embedding based on the multiple internal representations. For example, for the case of separating a mixture into multiple pieces of audio data, the first convolutional network may be applied to the transformed representation to map the transformed representation, which is a mixture of internal representations corresponding to audio data from different audio sources, into a set of separated second embeddings, each corresponding to an individual acoustic source. The number of output embeddings from the first convolutional network can be determined by the number of acoustic sources to be separated. The first convolutional network may include one or more convolutional layers that may apply a series of learnable filters (or kernels) to the transformed representation. Each filter may be designed to detect specific features in the audio data, such as certain frequencies or temporal patterns associated with different instruments or voices. The filter may generate a set of feature maps that capture the spatial (or temporal) dependencies and features present in the audio data. The multiple internal representations may be refined to generate the one or more second embeddings by using the adjusting

block. The second embedding, which is the output of the adjusting block, may be used to represent corresponding audio data from the original mixture. For example, for the case of separating a mixture into multiple pieces of audio data, the second embeddings may capture necessary features of the audio sources and are used for the reconstruction of the separated audio signals. By using the first convolutional
5 network and the adjusting block, the efficiency of the separating process can be improved.

[0053] In a possible implementation, at least one mask corresponding to at least one of the multiple pieces of audio data one-by-one may be determined and the at least one second embedding can be obtained based on the at least one mask and the first embedding. The mask may be a set of coefficients or a matrix that, when applied to the mixed audio signal, allows for the extraction or separation of audio
10 data from individual audio sources. The masks may be element-wise and derived from the embedded representation of the audio data after being processed by the separator. The at least one mask may have a one-by-one corresponding relationship with at least one of the multiple pieces of audio data. As shown in FIG. 4B, the separator may include a normalizing block, a transforming block, and a post-processing block. Specifically, the processing device may input the first embedding into a normalizing block of the
15 separator to obtain a normalized first embedding, may input the normalized first embedding into a transforming block of the separator to obtain transformed representation, where the transformed representation is a mixture of multiple internal representations corresponding to the multiple pieces of audio data one-by-one and may input the transformed representation into a post-processing block of the separator to obtain the at least one mask. The normalization may include various operations such as min-
20 max scaling, z-score normalization (subtracting the mean and dividing by the standard deviation), or more sophisticated approaches like batch normalization, which normalizes the input within each mini-batch during training. The normalized first embedding may be then input into a transforming block of the separator. The transforming block may include a series of transformer layers or similar mechanisms designed to manipulate and learn from the embedded data. The output of the transforming block may be
25 a set of internal representations. The internal representations may be input into a post-processing block and the post-processing block can output masks, each mask corresponding to a respective piece of audio data in the mixed audio data. The post-processing block may be used to generate at least one mask that, when applied to the first embeddings or the original audio representation, allow for the separation of the mixed audio data into individual audio data for each acoustic source. For example, for the case of

separating a mixture into multiple pieces of audio data, the masks may be multiplied with the first embedding to obtain the multiple second embeddings. By multiplying the masks with the first embedding, the model may separate the audio features corresponding to a specific acoustic source from the features of the mixed audio data. For example, if the mixed audio data is a mixture of multiple pieces of audio data from N acoustic sources, there will be N masks and N second embeddings. This approach can reduce the computational load and memory usage, making the audio processing more efficient.

[0054] In a possible implementation, an activation function used by the post-processing block may be a Snake activation function. The Snake activation function may be a type of non-linear activation function. In a possible implementation, the Snake activation function may be represented by the formula: $\text{Snake}(x) = x + \sin^2(x)$, where x is an input value to the activation function. The Snake activation function can induce a periodic bias. The use of the Snake activation function in the post-processing block can effectively produce the separation masks or second embeddings that isolate individual audio sources from the mixed audio data, even when operating within the compressed embeddings provided by the NAC. Adopting the Snake activation function in the separator, the separator can better leverage the periodic nature of audio signals and the compressed embeddings to perform high-quality speech separation with reduced computational resources. In addition, since the separator adopts the Snake activation function, outputs of the separator can be mapped to a similar range of values, this would be especially beneficial when the NAC is a descript audio codec (DAC).

[0055] In a possible implementation, the transforming block may include a transformer and a second convolutional network which is used as an adaptation layer. In this case, the normalized first embedding may be input into the transformer and the second convolutional network may be applied to an output of the transformer to obtain the transformed representation, so the output of the transformer may be adapted, at the second convolutional network, to processing to be performed by the post-processing block. The transformer may be a type of neural network architecture, and can be used to capture long-range dependencies and in tasks involving time-series data, such as natural language processing and speech recognition. The output of the transformer is a candidate transformed representation, which may be a representation that has been processed to capture interrelationships between the different acoustic sources in the mixed audio. The second convolutional network which serves as an adaptation layer may adapt the candidate transformed representation extracted by the transformer to be more suitable for the subsequent

separation process. The adaptation may involve learning a mapping that emphasizes most relevant aspects of audio data for source separation. The use of the transforming block and the post-processing block can allow for a flexible network architecture that can be adapted to the specific requirements of the audio separation task, including the number of sources to be separated and the complexity of the audio mixture, and can improve computational efficiency. In some cases, the adaptation layer can also be omitted, we may simply use the transformer as the transforming block.

[0056] In a possible implementation, the normalizing block may include at least one third convolutional network. The normalizing block may include one or more third convolutional networks, which can provide deeper feature extraction capabilities, and can lead to better separation performance by providing more refined embeddings to the subsequent transforming and masking blocks.

[0057] In a possible implementation, the encoder and the separator may be at a local side, and the decoder of the NAC may be at a cloud side. The processing device may transmit the at least one mask from a local side to the cloud side for the reconstruction of the multiple pieces of audio data. For example, for the case of separating a mixture into multiple pieces of audio data, when the encoder and the separator are at a local side, and the decoder of the pre-trained NAC is at a cloud side, the output second embeddings of the separator can be transmitted from a local side to the cloud side for the reconstruction. Utilizing the cloud for decoding can allow for the use of more sophisticated models and larger computational resources that might not be available locally. Since the most time-sensitive processes (encoding and separation) can be separated from the cloud-based decoding, latency can be reduced, which may be critical for real-time applications. In addition, the cloud-based decoding may be scalable and can handle a variable number of users or streams without the need for each local device to have sufficient decoding capabilities, and can allow for the local encoder and separator to be updated or changed without affecting the cloud decoding process, providing flexibility in evolving the system. The deployment of the encoder, the separator and the decoder can be particularly well-suited for applications where real-time processing and low latency are crucial, such as in teleconferencing, live streaming, or any scenario where audio needs to be separated and transmitted with minimal delay.

[0058] In a possible implementation, the separator may be trained with a preset loss function, and the preset loss function may be based on comparison of the multiple pieces of audio data against reference audio data, and the reference audio data may be obtained after processing clean audio data by the encoder

and the pre-trained separator at the local side, transmitting a result of the processing of the clean audio data from the local side to the cloud side, and processing the result by the decoder at the cloud side. In this training process, the clean audio data experiences the same processing as the mixed audio data to obtain the reference audio data, so by focusing on minimizing such preset loss function which is directly related to the difference between the separated audio and the reference (both undergo same processing), a separator that is more accurate in distinguishing and separating different audio sources can be obtained.

[0059] In related art, a loss function, for example Scale-Invariant Signal-to-Distortion Ratio (SI-SDR) loss function or a convolutive transfer function invariant signal-to-distortion ratio (CI-SDR) loss function, which is obtained by comparing an original ground truth with an estimated signal after scaling them by a factor. However, this approach may not provide a fair comparison because in an actual deployment context the cloud would also be receiving a transmission of the original ground truth, after some distortion. In the embodiments of the present disclosure, the preset loss function may be based on comparison of the multiple pieces of audio data against reference audio data. For example, a preset loss function, which can be called “Codec SISDR” (cSI-SDR), can be represented by the following formula:

$$t = \text{Codec}(s)$$

$$\text{cSI-SDR} = 10\log_{10} \frac{||\alpha t||^2}{||\alpha t - \hat{s}||^2}$$

$$\text{where } \alpha = \frac{\hat{s}^T t}{||t||^2}$$

where s represents an original ground truth (reference audio data), $\hat{s}$ represents an estimated signal (separated audio data), t represents a transmitted signal which is obtained by encoding a clean signal/the original ground truth (reference audio data) into codes locally (local side) and decoding the codes on the server (cloud side).

[0060] Instead of the original clean signal, this preset loss function of the present disclosure may penalize the model for deviations from the transmitted signal, which is a more realistic target. In this way, the additional reconstruction errors introduced by the codec can be accounted for, the separator can learn to deal with the specific artifacts and distortions that can occur during audio transmission, making the system more robust and reliable.

[0061] In a possible implementation, the encoder, the separator and the decoder may be at a local side. That is to say, the entire process, from encoding to decoding, can take place locally on the terminal device. Processing audio locally eliminates the delay that would be incurred from transmitting data to the cloud

and waiting for a response, and keeping audio processing local means that sensitive or private conversations may not be transmitted over the internet, which reduces the risk of eavesdropping or data interception.

[0062] In a possible implementation, before performing audio separation on the first embedding with
5 a pre-trained separator to obtain the at least one second embedding, the processing device may input the first embedding into a Residual Vector Quantization (RVQ) block of the pre-trained NAC to obtain a quantized first embedding, and may perform audio separation on the quantized first embedding with the pre-trained separator to obtain the at least one second embedding. RVQ may compress the embedding, effectively reducing the bitrate required to represent the audio data, which is beneficial for storage and
10 transmission. Performing RVQ before separation may lead to a more efficient compression of the overall signal. The quantized embeddings can be transmitted more efficiently over networks, which is particularly useful for applications where bandwidth is limited.

[0063] In a possible implementation, after performing audio separation on the first embedding with a pre-trained separator to obtain the at least one second embedding, the processing device may input the at
15 least one second embedding into a Residual Vector Quantization (RVQ) block of the pre-trained NAC to obtain at least one quantized embedding, and may update the at least one second embedding with the at least one quantized embedding, that is, instead of using the at least one second embedding, we use the at least one quantized embedding, so the content related to the processing of the second embedding mentioned above would be also applicable for the quantized embeddings. By performing audio
20 separation before quantization, the separator can access to a higher-quality, less distorted version of the audio data, which can lead to better separated embeddings. In addition, after separation, RVQ can be applied to each separated stream individually, allowing for more selective compression.

[0064] FIG. 5 shows a schematic flowchart of an audio processing method in a meeting system according to one or more example embodiments of the present disclosure. Illustratively, the audio
25 processing method provided by the present disclosure can be executed by an audio processing device provided by the present disclosure, and the audio processing device can be implemented by any software and/or hardware. For example, the audio processing device can be: a tablet computer, a mobile phone (such as a folding screen mobile phone, a large screen mobile phone, etc.), a wearable device, a vehicle-mounted device, an augmented reality (AR)/virtual reality (VR) equipment, a laptop computer, a ultra-

mobile personal computer (UMPC), a netbook, a personal digital assistant (PDA), a smart TV, a smart screen, a HD TV, a 4K TV, a smart speaker, a smart projector and other Internet of Things (IOT) devices. The specific types of electronic devices are not limited in the embodiments of the present disclosure. As shown in FIG. 5, the method can include the following steps.

5 **[0065]** S501, input mixed speech data into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple speech signals from multiple participants of a meeting, and the first embedding is a representation of an audio feature of the mixture, where the pre-trained NAC is trained with a set of sample mixed speech data for speech separation.

10 **[0066]** S502, perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, and each of the at least one second embedding is a representation of an audio feature of a corresponding speech signal.

15 **[0067]** S503, transmit the at least one second embedding to a cloud server, to enable the cloud server to reconstruct the at least one of the multiple speech signals with a decoder of the NAC.

20 **[0068]** In the embodiment, mixed speech data, which is a mixture of multiple speech signals from various participants in a meeting, is input into an encoder of a pre-trained NAC to obtain a first embedding. The first embedding represents an audio feature of the mixture. A pre-trained separator is utilized to perform speech separation on the first embedding and obtain at least one second embedding corresponding to at least one of the multiple speech signals individually. The separator operates based on the at least one second embedding to realize separation. The obtained second embedding is then transmitted to a cloud server. The cloud server can be enabled to reconstruct the original speech signal using a decoder of the NAC. The decoder takes the second embedding to recreate the separate speech signals.

25 **[0069]** This method is well-suited for cloud recording of meetings as it allows for local processing of audio to separate individual speakers, followed by efficient transmission of the separation data to the cloud, where the audio can be reconstructed, stored, and potentially further processed or analyzed.

30 **[0070]** In this way, in a cloud recording scenario, bandwidth requirements can be reduced due to the transmission of embeddings rather than raw audio, the ability to leverage powerful cloud-based decoders for high-quality audio reconstruction can be improved. In addition, this approach can help maintain

privacy by not transmitting raw speech data over the network, and it allows for the cloud server to handle the computationally intensive task of decoding, which might not be feasible on local recording devices.

[0071] FIG. 6 shows a schematic flowchart of an audio processing method in a music processing system according to one or more example embodiments of the present disclosure. Illustratively, the audio processing method provided by the present disclosure can be executed by an audio processing device provided by the present disclosure, and the audio processing device can be implemented by any software and/or hardware. For example, the audio processing device can be: a tablet computer, a mobile phone (such as a folding screen mobile phone, a large screen mobile phone, etc.), a wearable device, a vehicle-mounted device, an augmented reality (AR)/virtual reality (VR) equipment, a laptop computer, a ultra-mobile personal computer (UMPC), a netbook, a personal digital assistant (PDA), a smart TV, a smart screen, a HD TV, a 4K TV, a smart speaker, a smart projector and other Internet of Things (IOT) devices. The specific types of electronic devices are not limited in the embodiments of the present disclosure. As shown in FIG. 6, the method can include the following steps.

[0072] S601, input mixed audio data into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of music, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation.

[0073] S602, perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of music one-by-one, each of the at least one second embedding is a representation of an audio feature of a corresponding piece of music, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of music with a decoder of the pre-trained NAC.

[0074] In the embodiment, mixed audio data, which includes a blend of multiple pieces of music, is input into an encoder of a pre-trained NAC to obtain a first embedding. This first embedding may be a compressed representation that captures key audio features of the mixed music. A pre-trained separator is applied to the first embedding to perform audio separation and obtain at least one second embedding corresponding to each individual piece of music within the mixture. The separator may be specifically designed to operate on the compressed embeddings and separate the audio based on those embeddings. After the at least one second embedding is obtained, they may be outputted for use in reconstructing the

original music. For example, for multiple second embeddings cases, the second embeddings can be utilized by a decoder to generate separate audio streams for each piece of music.

[0075] The method according to the embodiment of the present disclosure is particularly useful in music processing applications. For example, individual instruments or tracks can be isolated from a mixed recording via the audio processing method. The separated multiple pieces of music can be valuable for various purposes, such as remixing, transcription, or archiving.

[0076] In this way, high-quality separation of different musical elements can be achieved, computational resources can be efficiently used by operating within the compressed embedding space. In addition, this approach could be beneficial for applications such as music editing, music education, transcription services, or creation of karaoke tracks, where isolated musical components are desirable.

[0077] FIG. 7A a schematic flowchart of a model training method according to one or more example embodiments of the present disclosure. The model training method can be used to train the above-mentioned pre-trained NAC according to one or more embodiments of the present disclosure. The method may be implemented by an apparatus, such as an audio processing apparatus, a (cloud) server or other devices, such as a chip which has similar function. In a possible implementation, the apparatus may be a processing device. As shown in FIG. 7A, the method can include the following steps.

[0078] S7011, acquire a training sample set, where the training sample set includes multiple pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources.

[0079] S7021, for a piece of sample mixed audio data in the training sample set, input the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture.

[0080] S7031, input the first sample embedding into a pre-trained separator to obtain at least one second sample embedding, where the at least one second sample embedding corresponds to at least one of the multiple pieces of audio data one-by-one, and each of the at least one second sample embedding is a representation of an audio feature of a corresponding piece of sample audio data.

[0081] S7041, input the at least one second sample embedding into a decoder of the NAC to obtain reconstructed audio data for at least one corresponding sample acoustic source.

[0082] S7051, update at least one parameter of the NAC based on the reconstructed audio data and the

at least one of the multiple pieces of sample audio data.

5 [0083] In the embodiment, a training sample set including multiple pieces of sample mixed audio data is acquired, and for a piece of sample mixed audio data in the training sample set, the sample mixed audio data is input into an encoder of an NAC to obtain a first sample embedding which is a representation of an audio feature of the sample mixed audio data. Then audio separation is performed on the first sample embedding with a pre-trained separator and at least one second sample embedding which corresponds to at least one of the multiple pieces of sample audio data one-by-one is obtained. Then the at least one second sample embedding is input into the decoder of the NAC to obtain reconstructed audio data, and parameter(s) of the NAC can be updated based on the reconstructed audio data and the at least one of the multiple pieces of sample audio data. The acquiring of the training sample set would be performed once 10 for each training epoch, and in each training iteration of a training epoch, S7021 to S7051 would be performed. For each training iteration, there would one or more pieces of sample mixed audio data used for training, which is not limited herein.

[0084] Here the sample mixed audio data in different training iterations may be different mixtures of sample audio data from the multiple sample acoustic sources, the multiple pieces of sample audio data 15 serve as ground truth or clean audio data in this training process, and the reconstructed audio data for each of the multiple sample acoustic sources is the separation result, so the reconstructed audio data would be compared with the multiple pieces of sample audio data to provide a basis for parameter update of the NAC.

20 [0085] In this way, the NAC with which audio separation can be performed in the compressed embedding space can be obtained. The iterative update of the NAC parameters based on reconstruction quality can ensure that the model learns to separate and reconstruct audio signals more accurately over time. By including the effects of encoding and decoding in the training, the model becomes more robust to the distortions introduced by the codec.

25 [0086] FIG. 7B a schematic flowchart of another model training method according to one or more example embodiments of the present disclosure. The model training method can be used to train the above-mentioned pre-trained NAC according to one or more embodiments of the present disclosure. The method may be implemented by an apparatus, such as an audio processing apparatus, a (cloud) server or other devices, such as a chip which has similar function. In a possible implementation, the apparatus may

be a processing device. As shown in FIG. 7B, the method can include the following steps.

[0087] S7012, acquire a training sample set, where the training sample set includes multiple pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources.

5 [0088] S7022, for a piece of sample mixed audio data in the training sample set, input the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture.

[0089] S7032, input the first sample embedding into a decoder of the NAC to reconstruct the sample mixed audio data.

10 [0090] S7042, update at least one parameter of the NAC based on a reconstruction result and the sample mixed audio data.

[0091] In the embodiment, a training sample set including multiple pieces of sample mixed audio data is acquired, and for a piece of sample mixed audio data in the training sample set, the sample mixed audio data is input into an encoder of an NAC to obtain a first sample embedding which is a representation of an audio feature of the sample mixed audio data. Then the first sample embedding is input into a decoder of the NAC to reconstruct the sample mixed audio data. After that, parameter(s) of the NAC can be updated based on the reconstruction result and the sample mixed audio data. The acquiring of the training sample set would be performed once for each training epoch, and in each training iteration of a training epoch, S7022 to S7042 would be performed. For each training iteration, there would one or more pieces of sample mixed audio data used for training, which is not limited herein.

[0092] The sample mixed audio data in different training iterations may be different mixtures of sample audio data from the multiple sample acoustic sources, and the reconstruction result would be reconstructed mixed audio data of the multiple sample acoustic sources, so the reconstructed mixed audio data would be compared with the sample mixed audio data (serving as ground truth or clean audio data) to provide a basis for parameter update of the NAC.

25 [0093] The capability of representing a mixture in the embedding space would be improved by using the above training process. The NAC can be enabled to adapt to various types of audio data by being trained via the training method, which improves its generalization capabilities. The NAC can be trained to have a good capability in representing mixed audio data, thus facilitating the audio separation in the

embedding space, and the NAC can be trained to operate efficiently, with a focus on reducing computational resources and potentially enabling real-time processing.

[0094] FIG. 8A shows a schematic illustration of an audio processing method according to one or more example embodiments of the present disclosure. As shown in FIG. 8A, a detailed architecture of a separator of the preset disclosure is illustrated.

[0095] As shown in FIG. 8A, mixed audio data is input into an encoder of a NAC (shown as a DAC Encoder in FIG. 8A as an example), a first embedding of the mixed audio data can be obtained and input into a separator (shown as Codecformer in FIG. 8A). The mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture. The Codecformer performs audio separation on the first embedding with a separator to obtain second embeddings corresponding to the multiple pieces of audio data one-by-one. The Codecformer then outputs the second embeddings and transmits the second embeddings to a decoder (shown as a DAC Decoder in FIG. 8A as an example) of the NAC. The DAC Decoder can then reconstruct the multiple pieces of audio data based on obtained second embeddings.

[0096] The right portion of FIG. 8A shows a detailed implementation of Codecformer. The Codecformer includes a normalizing block, a transforming block and a post-processing block. Conv1D in FIG. 8A refers to a convolutional network. The normalizing block may correspond to the Conv1D right after DAC encoder, although only one Conv1D is shown, there may be one or more Conv1Ds to function as the normalizing block. The transforming block may include a transformer used for outputting a transformed representation. In addition, the transforming block may further include a second convolutional network (which is illustrated as a Conv1D immediate following the transformer in FIG. 8A) used as an adaptation layer between the transformer and the post-processing block. The first embedding is input to the normalizing block to obtain a normalized first embedding, the normalized first embedding is input to the transforming block. The transformed representation output by the transforming block is then input into the post-processing block to obtain the second embeddings. The post-processing block may include a first convolutional network (which is illustrated as the second Conv1D immediate following the transformer in FIG. 8A) and an adjusting block (from weight_norm block to the last Snake1D block in FIG. 8A). The first convolutional network is used for mapping the transformed representation into multiple internal representations, and the adjusting block is used for generating the

multiple second embeddings based on the internal representations. The first convolutional network can obtain multiple internal representations, and each of the internal representations is a representation of an audio feature of corresponding audio data. The internal representations output by the first convolutional network are then input into the adjusting block to obtain the second embeddings. The adjusting block
5 applies a Snake activation function on the internal representations. It should be noted that the structure of the Codecformer is illustrative rather than restrictive, the blocks in each of the normalizing block, the transforming block and the post-processing block can vary depending on actual implementations, which is not limited in the embodiments of the present disclosure.

[0097] FIG. 8B shows a schematic illustration of another audio processing method according to one
10 or more example embodiments of the present disclosure. As shown in FIG. 8B, a detailed architecture of a separator of the preset disclosure is illustrated.

[0098] As shown in FIG. 8B, mixed audio data is input into an encoder of a NAC (shown as a DAC Encoder in FIG. 8B as an example), a first embedding of the mixed audio data can be obtained and input into a separator (shown as Codecformer in FIG. 8B). The mixed audio data is a mixture of multiple pieces
15 of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture. The Codecformer performs audio separation on the first embedding with a separator to obtain multiple second embeddings corresponding to the multiple pieces of audio data one-by-one. The Codecformer then outputs the second embeddings and transmits the second embeddings to a decoder (shown as a DAC Decoder in FIG. 8B as an example) of the NAC. The DAC Decoder can then
20 reconstruct the multiple pieces of audio data based on obtained second embeddings.

[0099] The right portion of FIG. 8B shows a detailed implementation of Codecformer. The Codecformer includes a normalizing block, a transforming block and a post-processing block. Conv1D in FIG. 8B refers to a convolutional network. The normalizing block may correspond to the Conv1D right after DAC encoder, although only one Conv1D is shown, there may be one or more Conv1Ds to
25 function as the normalizing block. The transforming block may include a transformer used for outputting a transformed representation. The transformed representation is a mixture of representations of the multiple acoustic sources. In addition, the transforming block may further include a second convolutional network (which is illustrated as a Conv1D immediate following the transformer in FIG. 8B) used as an adaptation layer between the transformer and the post-processing block. The first

embedding is input to the normalizing block to obtain a normalized first embedding, the normalized first embedding is input to the transforming block. The transforming block can obtain a transformed representation. The transformed representation output by the transforming block is then input into the post-processing block. The first embedding is input to the normalizing block to obtain a normalized first embedding, the normalized first embedding is input to the transforming block. The transformed representation is then input into the post-processing block to obtain the second embeddings. The post-processing block may include a first convolutional network (which is illustrated as the second Conv1D immediate following the transformer in FIG. 8B) and an adjusting block (from weight_norm block to the last Snake1D block in FIG. 8B). The first convolutional network is used for mapping the transformed representation into multiple internal representations, and the adjusting block is used for generating the multiple second embeddings based on the internal representations. The first convolutional network can obtain multiple internal representations, and each of the internal representations is a representation of an audio feature of corresponding audio data. The internal representations output by the first convolutional network are then input into the adjusting block to obtain the second embeddings. Multiple masks corresponding to the multiple pieces of audio data one-by-one are determined based on the internal representations, and the multiple second embeddings can be obtained by multiplying the determined masks and the first embeddings. The adjusting block applies a Snake activation function on the internal representations. It should be noted that the structure of the Codecformer is illustrative rather than restrictive, the blocks in each of the normalizing block, the transforming block and the post-processing block can vary depending on actual implementations, which is not limited in the embodiments of the present disclosure.

[0100] With reference to FIG. 8A and FIG. 8B, the Codecformer can be trained with a preset loss function, and the preset loss function is based on comparison of the multiple pieces of audio data against reference audio data, and the reference audio data is obtained after processing clean audio data by the encoder and the pre-trained separator at the local side, transmitting a result of the processing of the clean audio data from the local side to the cloud side, and processing the result by the decoder at the cloud side.

[0101] In order to manage long sequence lengths, models such as Sepformer also have to resort to chunking methods and a dual-path architecture as the full audio sequence cannot be fed into a vanilla transformer without exceeding memory limitations. According to the method of the present disclosure,

due to the high temporal compression provided by DAC, this limitation is lifted, allowing for the full sequence to be passed into the model without chunking. The model architecture can be simplified by removing the need for Intra and Inter blocks in a typical dual-path separation approach, replacing them with a single stack of transformer layers that operate over the full sequence.

5 [0102] Moreover, the model of the present disclosure may utilize the Snake activation function. In this way, the output of the separation model can be allowed to work well with the DAC decoder. The Snake activation function is designed to be better at modeling periodic functions by inducing a periodic bias. Since DAC may adopt the Snake activation function, it was necessary for the separation model to also adopt this activation function to map the model outputs to a similar range of values.

10 [0103] The performance of the above audio processing method can be tested by several experiments. The experiments and results will be described in details.

[0104] Dataset

[0105] A popular benchmark dataset WSJ0-2mix is used. The dataset consists of mixtures combined using clean speech from the WSJ0 dataset. Both mixtures and the ground truth consist of speech recorded
15 in acoustically treated environments at an 8kHz sampling rate. The WSJ02Mix dataset consists of mixtures drawn from the WSJ0 corpus and consists of 20,000 mixtures (30 hours) from the WSJ0 si tr s set, a validation set of 5000 mixtures (10 hours) from the WSJ0 si dt 05 set and a test set of 3000 mixtures (5 hours) from the si et 05 set.

[0106] **Hyper parameters**

20 [0107] The DAC model uses embedding size 1024, and where applicable the full-bitrate of the DAC model was used. Codecformer was set up with an embedding size of 256 and 16 transformer blocks. The number of transformer blocks were chosen based on the design of Sepformer, which has 8 intra blocks and 8 inter blocks for a total of 16 blocks. Training was performed with an initial learning rate of $1.5e-4$ and the LR scheduler was set to halve the learning rate with a patience of 2 after epoch 5. All models
25 were trained to 200 epochs and the optimal model was chosen based on validation loss.

[0108] **Baselines**

[0109] Three baselines is reported in Table 3 to benchmark the performance of a popular speech separation model, Sepformer, in various usage scenarios. The model is pre-trained and implemented by the Speechbrain framework. Sepformer (Oracle) is the baseline where no codec (no appliance of NAC)

is used. The input to Sepformer is the original mixture and the output of Sepformer is used directly in the performance comparison. This allows us to measure the distortion with respect to the degraded transmissions.

[0110] Sepformer (Local) where the model is given a clean mixture, but the output is transmitted through DAC, resulting in distortions. This mimics a scenario where the separation is performed on-device and its output is sent to the cloud, as shown in FIG. 2.

[0111] Sepformer (Cloud) where the model is given a distorted mixture that has been transmitted through DAC. This implements a target scenario as shown in FIG. 1 where the separation is happening in the cloud.

[0112] In Table 1, the results of Codecformer is compared against the Sepformer (Cloud) baseline. The Sepformer (Cloud) baseline represents the performance of the model if it was necessary to transmit the mixture to the cloud for processing. Meanwhile, Codecformer performs the separation locally. Since we are comparing transmitted audio, we append “c” to the performance metrics SDRi and SI-SDRi to denote that although the same computation is used, the basis of comparison is the ground-truth that has been distorted by the codec. In this comparison, it can be seen that Codecformer outperforms Sepformer (Local) by 0.3dB on both cSI-SDRi and cSDRi, and is competitive on perceptual quality based on PESQ.

Table 1: Comparison of performance between Sepformer (Cloud) and Codecformer trained using cSI-SDRi

Model	cSI-SDRi	cSDRi	PESQ
Sepformer (Cloud)	9.6	10.1	2.75
Codecformer (Ours)	9.9	10.4	2.58

[0113] Improvements to Training Speed and Computation

[0114] The main advantage of performing speech separation within the embedding space of a NAC is the reduced computation required and improved training speed, which is reported in Table 2. Comparing against Sepformer training, Codecformer achieves a 2.7x improvement in training speech in terms of hours per epoch, which can translate to significant cost savings. These numbers are obtained on a machine with a V100 with 16GB of RAM.

[0115] For practical deployment, we compute Multiple and Accumulate operations (MACs) using

PyTorch-OpCounter to measure the inference requirements for the models in a hardware agnostic manner. In this case the test input utterance is 2 seconds of 8kHz audio and the MAC from the audio compression and decompression is not included since the MAC would apply to both separators. From the results in Table 2 we can see that a Codecformer requires significantly lower MACs, constituting a 52x reduction to process the same length of audio. This significant improvement is largely due to the time compression afforded by DAC, which reduces the 8kHz audio to 50Hz. As the shorter sequence lengths reduces computational requirements.

Table 2: Computation and training speech of Codecformer compared against Sepformer

Model	GMACs	Training Time (h/epoch)
Sepformer	77.3	2.7
Codecformer (Ours)	1.5	1.0

[0116] Ablation study

[0117] In Table 3 the results of ablation studies over the different methods of training Codecformer are reported. For the column labeled “Ground Truth”, we use the original clean speech as the basis for comparison. Next, for the column labeled “Transmission”, degraded speech produced by transmitting the original clean speech through DAC is used as the basis for comparison. In both cases, all 4 commonly used performance metrics for speech separation are reported (with “c” denoting the use of transmission as the basis for comparison), in addition to PESQ to measure the perceptual quality of the output. In this study, PESQ is measured against the original ground truth since it would not be principled to use the distorted audio as the comparison.

Table 3: Ablation study of different training targets for Codecformer, compared against different Sepformer baselines

Baselines	SI-SDR	Ground Truth			cSI-SDR	Transmission		cSDR _i	PESQ
		SI-SDR _i	SDR	SDR _i		cSI-SDR _i	cSDR		
Sepformer (Oracle)	22.4	22.4	22.8	22.6	-20.6	-19.5	12.3	13.0	4.00
Sepformer (Local)	-20.6	-20.6	0.2	0.1	11.6	12.7	12.3	13.0	3.84
Sepformer (Cloud)	-20.7	-20.7	0.0	-0.2	8.6	9.6	9.3	10.1	2.75

Training Target (Loss)	RV Q									
Transmission (cSI-SDR)	✓	-20.6	-20.6	-0.6	-0.8	5.9	7.0	6.9	7.6	2.07
	×	-20.5	-20.5	0.1	-0.1	8.8	9.9	9.7	10.4	2.58
Ground Truth (SI-SDR)	✓	5.0	5.0	6.4	6.2	-20.5	-19.4	6.6	7.3	2.09
	×	8.7	8.7	9.8	9.7	-20.6	-19.6	9.4	10.1	2.52

[0118] Effects of Training Target

[0119] Comparing the results of Codecformer trained on the transmission target against those trained on ground truth target, we see that training on the ground truth benefits the model more broadly, delivering performance under both ground truth and transmission comparisons. Training on a transmission target only benefits the model on the transmission comparison and causes poor performance when compared against the ground truth. We believe that this is because using the ground truth signal provides a cleaner loss gradient to the model, whereas when training using the transmission target the model may occasionally be penalized for the inherent distortions in the transmission.

[0120] Effects Transmission Distortions

[0121] Typically, transmitting audio through a codec results in distortions. This can result in poor performance on the objective metrics but the audio can still perform well on the perceptual metric. In the Sepformer (Oracle) baseline, we see that while the model’s output achieves good performance against the original ground truth, but when the basis of comparison is the transmission, the performance drops significantly although perceptual quality remains high. Conversely, in the Sepformer (Local) baseline, we see that PESQ suffers only a small drop from 4.00 to 3.84 even through the comparison against the ground truth results in poor performance. Additionally, we note that the scale-invariant metrics suffer significantly whenever there is a comparison between a degraded and original waveform. This is likely a scaling issue from the distortions introduced by DAC.

[0122] Effects of Training Target

[0123] Comparing the results of Codecformer trained on the transmission target against those trained on ground truth target, it can be seen that training on the ground truth benefits the model more broadly, delivering performance under both ground truth and transmission comparisons. Training on a transmission target only benefits the model on the transmission comparison and causes poor performance

when compared against the ground truth. We believe that this is because using the ground truth signal provides a cleaner loss gradient to the model, whereas when training using the transmission target the model may occasionally be penalized for the inherent distortions in the transmission.

[0124] Effects of RVQ

5 **[0125]** A common feature of neural audio compression models is the RVQ layer which compresses model embeddings by learning a quantization codebook to represent the embeddings with only a small number of codes. The first layer of the RVQ module aims to calculate an estimate of the original embedding and each subsequent layer estimates the residual between the sum of the previous layers and the original embedding. Each subsequent code is thus of decreasing importance to reconstruction,
10 allowing the transmission bitrate to be easily scalable. This design, although efficient, means it is inevitable that the RVQ module results in some degradation of the signal.

[0126] Thus, in addition to experimenting with different targets and losses for training, for each target we perform an ablation study on the use of RVQ in the model. Experiments where RVQ are used show the performance of the separation if the model was run in the cloud. On the other hand, our experiments
15 without the use of RVQ, which show the potential maximum separation performance in the limit of an RVQ block with an infinite number of layers.

[0127] In this work, Codecformer achieves a reduction in the MAC required during inference by 52x compared to the baseline Sepformer architecture and trains 2.7x faster while producing competitive performance on both objective and perceptual metrics. Thus, the results have shown that the proposed
20 task of performing speech separation within the embedding space of a NAC is indeed possible, allowing to gain inference and memory advantages over traditional speech separation approaches.

[0128] FIG. 9 shows a schematic structural diagram of an audio processing apparatus according to one or more embodiments of the present disclosure. As shown in FIG. 9, an audio processing apparatus 900 may include:

25 a processing module 901, configured to:

input mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of audio data from multiple acoustic sources, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio

separation;

perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of audio data with a decoder of the pre-trained NAC.

[0129] In a possible implementation, the pre-trained separator includes a normalizing block, a transforming block and a post-processing block, and the processing module 901 is configured to:

input the first embedding into the normalizing block of the pre-trained separator to obtain a normalized first embedding;

input the normalized first embedding into the transforming block of the pre-trained separator to obtain a transformed representation, where the transformed representation is a mixture of multiple internal representations corresponding to the multiple pieces of audio data one-by-one;

input the transformed representation into a post-processing block of the pre-trained separator to obtain the at least one second embedding.

[0130] In a possible implementation, the post-processing block includes a first convolutional network and an adjusting block;

the processing module 901 is configured to:

map the transformed representation into the multiple internal representations by using the first convolutional network;

generate the at least one second embedding based on the multiple internal representations by using the adjusting block.

[0131] In a possible implementation, the processing module 901 is configured to determine at least one mask and obtain the at least one second embedding based on the at least one mask and the first embedding, where the at least one mask corresponds to the at least one second embedding one-by-one.

[0132] In a possible implementation, an activation function used by the post-processing block is a Snake activation function.

[0133] In a possible implementation, the transforming block includes a transformer and a second

convolutional network;

the processing module 901 is configured to:

input the normalized first embedding into the transformer and apply the second convolutional network to an output of the transformer to obtain the transformed representation, where the output of the transformer is adapted at the second convolutional network to processing to be performed by the post-processing block.

[0134] In a possible implementation, the normalizing block includes at least one third convolutional network.

[0135] In a possible implementation, the processing module 901 is configured to:

input the at least one second embedding into the decoder of the pre-trained NAC to obtain the at least one of the multiple pieces of audio data.

[0136] In a possible implementation, the encoder and the pre-trained separator are at a local side, and the decoder of the pre-trained NAC is at a cloud side;

where the audio processing apparatus 900 further include an outputting module 902, configured to:

transmit the at least one second embedding from the local side to the cloud side for the reconstruction of the at least one of the multiple pieces of audio data.

[0137] In a possible implementation, the pre-trained separator is trained with a preset loss function, and the preset loss function is based on comparison of comparing the multiple pieces of audio data against reference audio data, and the reference audio data is obtained after processing clean audio data by the encoder and the pre-trained separator at the local side, transmitting a result of the processing of the clean audio data from the local side to the cloud side, and processing the result by the decoder at the cloud side.

[0138] In a possible implementation of the first aspect, the preset loss function is a scale-invariant signal-to-distortion ratio (SI-SDR) loss function or a convolutive transfer function invariant signal-to-distortion ratio (CI-SDR) loss function.

[0139] In a possible implementation, the encoder, the pre-trained separator and the decoder are at a local side.

[0140] In a possible implementation, the processing module 901 is configured to:

input the first embedding into a Residual Vector Quantization (RVQ) block of the pre-trained

NAC to obtain a quantized first embedding;

perform audio separation on the quantized first embedding with the pre-trained separator to obtain the at least one second embedding.

[0141] In a possible implementation of the first aspect, the processing module 901 is configured to:

5 input the at least one second embedding into a Residual Vector Quantization (RVQ) block of the pre-trained NAC to obtain at least one quantized embedding, and update the at least one second embedding with the at least one quantized embedding.

[0142] In a possible implementation, the NAC is a Descript NAC (DAC).

[0143] An embodiment of the present disclosure provides an audio processing apparatus, including:

10 a processing module, configured to:

input mixed speech data into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple speech signals from multiple participants of a meeting, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed speech data for speech separation;

15 perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the multiple speech signals one-by-one, and each of the at least one second embedding is a representation of an audio feature of a corresponding speech signal;

an outputting module, configured to transmit the at least one second embedding to a cloud
20 server, to enable the cloud server to reconstruct the at least one of the multiple speech signals with a decoder of the NAC.

[0144] An embodiment of the present disclosure provides an audio processing apparatus, including:

a processing module, configured to:

25 input mixed audio data into an encoder of a pre-trained NAC to obtain a first embedding of the mixed audio data, where the mixed audio data is a mixture of multiple pieces of music, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation;

perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, where the at least one second embedding corresponds to at least one of the

multiple pieces of music one-by-one, each of the at least one second embedding is a representation of an audio feature of a corresponding piece of music, and the at least one second embedding is used for reconstruction of the at least one of the multiple pieces of music with a decoder of the pre-trained NAC.

[0145] An embodiment of the present disclosure provides a model training apparatus, including:

5 an inputting module, configured to:

 acquire a training sample set, where the training sample set includes plural pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;

 for a piece of sample mixed audio data in the training sample set,

10 input the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture;

 input the first sample embedding into a pre-trained separator to obtain at least one second sample embedding, where the second sample embedding corresponds to the multiple pieces of audio data one-by-one, and each of the at least one second sample embedding is a representation of an audio feature

15 of a corresponding piece of sample audio data;

 input the at least one second sample embedding into a decoder of the NAC to obtain reconstructed audio data for at least one corresponding sample acoustic source;

 an updating module, configured to update at least one parameter of the NAC based on the reconstructed audio data and the at least one of the plural pieces of sample audio data.

20 **[0146]** An embodiment of the present disclosure provides a model training apparatus, including:

 an inputting module, configured to:

 acquire a training sample set, where the training sample set includes plural pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;

25 for a piece of sample mixed audio data in the training sample set,

 input the sample mixed audio data into an encoder of an NAC to obtain a first sample embedding, where the first sample embedding is a representation of an audio feature of the mixture;

 input the first sample embedding into a decoder of the NAC to reconstruct the sample mixed audio data;

an updating module, configured to update at least one parameter of the NAC based on a reconstruction result and the sample mixed audio data.

[0147] It should be understood by a person skilled in the art that, the relevant description of the above modules in the possible implementations of the present disclosure may be understood with reference to the relevant description of the speech separation method in the possible implementations of the present disclosure. The technical effects achieved by the above apparatuses are similar as those achieved by the above corresponding method embodiments, which is not repeated herein.

[0148] FIG. 10 is a structural diagram of an electronic device according to one or more embodiments of the present disclosure. As shown in FIG. 10, the electronic device 1000 may include: a processor 1001 coupled to a memory 1002 in a communicative way via an interface 1003; where the memory 1002 stores a computer executable instruction; the processor 1001 executes the computer executable instruction stored in the memory 1002 for executing any of the above speech separation methods. It should be noted that, the memory 1002 may be included or excluded from the electronic device, depending on actual needs.

[0149] An embodiment of the present disclosure provides a computer-readable medium storing computer execution instructions which, when executed by a processor, causes the processor to execute any of the above methods.

[0150] An embodiment of the present disclosure provides a computer program product including computer execution instructions which, when executed by a processor, causes the processor to execute any of the above methods.

[0151] An embodiment of the present disclosure provides a computer program which, when executed by a processor, causes the processor to execute any of the above methods.

[0152] The present disclosure encompasses various embodiments, including not only method embodiments, but also other embodiments such as apparatus embodiments and embodiments related to non-transitory computer readable storage media. Embodiments may incorporate, individually or in combinations, the features disclosed herein.

[0153] Although this disclosure refers to illustrative embodiments, this is not intended to be construed in a limiting sense. Various modifications and combinations of the illustrative embodiments, as well as other embodiments of the disclosure, will be apparent to persons skilled in the art upon reference to the

description.

[0154] Features disclosed herein in the context of any particular embodiments may also or instead be implemented in other embodiments. Method embodiments, for example, may also or instead be implemented in apparatus, system, and/or computer program product embodiments. In addition, although
5 embodiments are described primarily in the context of methods and apparatus, other implementations are also contemplated, as instructions stored on one or more non-transitory computer-readable media, for example. Such media could store programming or instructions to perform any of various methods consistent with the present disclosure.

[0155] Although the present disclosure describes methods and processes with steps in a certain order,
10 one or more steps of the methods and processes may be omitted or altered as appropriate. One or more steps may take place in an order other than that in which they are described, as appropriate.

[0156] Note that the expression “at least one of A or B”, as used herein, is interchangeable with the expression “A and/or B”. It refers to a list in which you may select A or B or both A and B. Similarly, “at least one of A, B, or C”, as used herein, is interchangeable with “A and/or B and/or C” or “A, B, and/or
15 C”. It refers to a list in which you may select: A or B or C, or both A and B, or both A and C, or both B and C, or all of A, B and C. The same principle applies for longer lists having a same format.

[0157] Although the present disclosure is described, at least in part, in terms of methods, a person of ordinary skill in the art will understand that the present disclosure is also directed to the various components for performing at least some of the aspects and features of the described methods, be it by
20 way of hardware components, software or any combination of the two. Accordingly, the technical solution of the present disclosure may be embodied in the form of a software product. A suitable software product may be stored in a pre-recorded storage device or other similar non-volatile or non-transitory computer readable medium, including DVDs, CD-ROMs, USB flash disk, a removable hard disk, or other storage media, for example. The software product includes instructions tangibly stored thereon that
25 enable a processing device (e.g., a personal computer, a server, or a network device) to execute examples of the methods disclosed herein. The machine-executable instructions may be in the form of code sequences, configuration information, or other data, which, when executed, cause a machine (e.g., a processor or other processing device) to perform steps in a method according to examples of the present disclosure.

[0158] The present disclosure may be embodied in other specific forms without departing from the subject matter of the claims. The described example embodiments are to be considered in all respects as being only illustrative and not restrictive. Selected features from one or more of the above-described embodiments may be combined to create alternative embodiments not explicitly described, features
5 suitable for such combinations being understood within the scope of this disclosure.

[0159] All values and sub-ranges within disclosed ranges are also disclosed. Also, although the systems, devices and processes disclosed and shown herein may include a specific number of elements/components, the systems, devices and assemblies could be modified to include additional or fewer of such elements/components. For example, although any of the elements/components disclosed
10 may be referenced as being singular, the embodiments disclosed herein could be modified to include a plurality of such elements/components. The subject matter described herein intends to cover and embrace all suitable changes in technology.

[0160] Although embodiments have been described above with reference to the accompanying drawings, those of skill in the art will appreciate that variations and modifications may be made without
15 departing from the scope thereof as defined by the appended claims.

CLAIMS

1. An audio processing method, comprising:

inputting mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, wherein the mixed audio data is a mixture of plural pieces of audio data from a plurality of acoustic sources, and the first embedding is a representation of an audio feature of the mixture; the pre-trained NAC is trained with a set of sample mixed audio data for audio separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, wherein the at least one second embedding corresponds to at least one of the plural pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second embedding is used for reconstruction of the at least one of the plural pieces of audio data with a decoder of the pre-trained NAC.

2. The method according to claim 1, wherein the pre-trained separator comprises a normalizing block, a transforming block and a post-processing block, and the performing audio separation on the first embedding with the pre-trained separator to obtain the at least one second embedding comprises:

inputting the first embedding into the normalizing block of the pre-trained separator to obtain a normalized first embedding;

inputting the normalized first embedding into the transforming block of the pre-trained separator to obtain a transformed representation, wherein the transformed representation is a mixture of a plurality of internal representations corresponding to the plural pieces of audio data one-by-one;

inputting the transformed representation into the post-processing block of the pre-trained separator to obtain the at least one second embedding.

3. The method according to claim 2, wherein the post-processing block comprises a first convolutional network and an adjusting block;

wherein the obtaining of the at least one second embedding comprises:

mapping the transformed representation into the plurality of internal representations by using the first convolutional network;

generating the at least one second embedding based on the plurality of internal representations by

using the adjusting block.

4. The method according to claim 3, wherein the generation of the at least one second embedding comprises:

5 determining at least one mask and obtaining the at least one second embedding based on the at least one mask and the first embedding, wherein the at least one mask corresponds to the at least one second embedding one-by-one.

5. The method according to any one of claims 2 to 4, wherein an activation function used by the post-processing block is a Snake activation function.

6. The method according to any one of claims 2 to 5, wherein the transforming block comprises a
10 transformer and a second convolutional network;

wherein the inputting the normalized first embedding into the transforming block of the pre-trained separator to obtain the transformed representation comprises:

inputting the normalized first embedding into the transformer and applying the second convolutional network to an output of the transformer to obtain the transformed representation, wherein the output of
15 the transformer is adapted at the second convolutional network to processing to be performed by the post-processing block.

7. The method according to any one of claims 2 to 6, wherein the normalizing block comprises at least one third convolutional network.

8. The method according to any one of claims 1 to 7, wherein the reconstruction of the at least one
20 of the plural pieces of audio data comprises:

inputting the at least one second embedding into the decoder of the pre-trained NAC to obtain the at least one of the plural pieces of audio data.

9. The method according to claim 8, wherein the encoder and the pre-trained separator are at a local side, and the decoder of the pre-trained NAC is at a cloud side;

25 wherein the method further comprises:

transmitting the at least one second embedding from the local side to the cloud side for the reconstruction of the at least one of the plural pieces of audio data.

10. The method according to claim 9, wherein the pre-trained separator is trained with a preset loss function, and the preset loss function is based on comparison of the plural pieces of audio data against

reference audio data, and the reference audio data is obtained after processing clean audio data by the encoder and the pre-trained separator at the local side, transmitting a result of the processing of the clean audio data from the local side to the cloud side, and processing the result by the decoder at the cloud side.

11. The method according to claim 8, wherein the encoder, the pre-trained separator and the decoder
5 are at a local side.

12. The method according to any one of claims 1 to 11, before performing audio separation on the first embedding with the pre-trained separator to obtain the at least one second embedding, further comprising:

inputting the first embedding into a Residual Vector Quantization (RVQ) block of the pre-trained
10 NAC to obtain a quantized first embedding;

wherein performing audio separation on the first embedding with the pre-trained separator to obtain the at least one second embedding comprises:

performing audio separation on the quantized first embedding with the pre-trained separator to obtain the at least one second embedding.

13. The method according to any one of claims 1 to 11, after performing audio separation on the first embedding with the pre-trained separator to obtain the at least one second embedding, further comprising:

inputting the at least one second embedding into a Residual Vector Quantization (RVQ) block of the pre-trained NAC to obtain at least one quantized embedding, and updating the at least one second
20 embedding with the at least one quantized embedding.

14. The method according to any one of claims 1 to 13, wherein the pre-trained NAC is a Descript Neural audio codec (DAC).

15. An audio processing method, comprising:

inputting mixed speech data into an encoder of a pre-trained neural audio codec (NAC) to obtain a
25 first embedding of the mixed audio data, wherein the mixed audio data is a mixture of a plurality of speech signals from a plurality of participants of a meeting, and the first embedding is a representation of an audio feature of the mixture; wherein the pre-trained NAC is trained with a set of sample mixed speech data for speech separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at least

one second embedding, wherein the at least one second embedding corresponds to at least one of the plurality of speech signals one-by-one, and each of the at least one second embedding is a representation of an audio feature of a corresponding speech signal;

transmitting the at least one second embedding to a cloud server, to enable the cloud server to
5 reconstruct the at least one of the plurality of speech signals with a decoder of the NAC.

16. An audio processing method, comprising:

inputting mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, wherein the mixed audio data is a mixture of plural pieces of music, and the first embedding is a representation of an audio feature of the mixture; wherein the pre-
10 trained NAC is trained with a set of sample mixed audio data for audio separation;

performing audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, wherein the at least one second embedding corresponds to at least one of the plural pieces of music one-by-one, each of the at least one second embedding is a representation of an audio feature of a corresponding piece of music, and the at least one second embedding is used for
15 reconstruction of the at least one of the plural pieces of music with a decoder of the pre-trained NAC.

17. A model training method, comprising:

acquiring a training sample set, wherein the training sample set comprises plural pieces of sample mixed audio data, and each of the plural pieces of sample mixed audio data is a mixture of plural pieces of sample audio data from a plurality of sample acoustic sources;
20 for a piece of sample mixed audio data in the training sample set,

inputting the sample mixed audio data into an encoder of a neural audio codec (NAC) to obtain a first sample embedding, wherein the first sample embedding is a representation of an audio feature of the mixture;

inputting the first sample embedding into a pre-trained separator to obtain at least one second sample embedding, wherein the at least one second sample embedding corresponds to at least one of the plural pieces of audio data one-by-one, and each of the at least one second sample embedding is a representation of an audio feature of corresponding audio data;
25

inputting the at least one second sample embedding into a decoder of the NAC to obtain reconstructed audio data for at least one corresponding sample acoustic source;

updating at least one parameter of the NAC based on the reconstructed audio data and the at least one of the plural pieces of sample audio data.

18. An audio processing apparatus, comprising:

a processing module, configured to:

5 input mixed audio data into an encoder of a pre-trained neural audio codec (NAC) to obtain a first embedding of the mixed audio data, wherein the mixed audio data is a mixture of plural pieces of audio data from a plurality of acoustic sources, and the first embedding is a representation of an audio feature of the mixture; where the pre-trained NAC is trained with a set of sample mixed audio data for audio separation;

10 perform audio separation on the first embedding with a pre-trained separator to obtain at least one second embedding, wherein the at least one second embedding corresponds to at least one of the plural pieces of audio data one-by-one, each of the at least one second embedding is a representation of an audio feature of corresponding audio data, and the at least one second embedding used for reconstruction of the at least one of the plural pieces of audio data with a decoder of the pre-trained NAC.

15 19. An electronic device, comprising at least one processor coupled to a memory storing a set of instructions;

wherein the at least one processor is configured to read the set of instructions in the memory and execute the audio processing method according to any one of claims 1 to 16 or the model training method according to claim 17.

20 20. A computer-readable medium storing computer execution instructions which, when executed by a processor, cause the processor to execute the audio processing method according to any one of claims 1 to 16 or the model training method according to claim 17.